

### Appendix A: Overview of Manual Content Analysis

**Table A1.** Overview of included TV news shows and the number of manually annotated comments per show.

| Regular news                             | n     |
|--|-------|
| 60 Minutes (CBS)                         | 150   |
| ABC Nightline                            | 172   |
| CBS Evening News                         | 170   |
| Face the Nation (CBS)                    | 155   |
| Meet the Press (NBC)                     | 51    |
| NBC News                                 | 230   |
| PBS NewsHour                             | 30    |
| The 11th Hour (MSNBC)                    | 127   |
| World News Tonight (ABC)                 | 117   |
| $n_{\text{regular-news}} =$              | 1,202 |
| Partisan news                            |       |
| Anderson Cooper 360 (CNN)                | 316   |
| Hannity (FoxNews)                        | 304   |
| Hardball with Chris Matthews (MSNBC)     | 317   |
| The Rachel Maddow Show (MSNBC)           | 37    |
| Tucker Carlson Tonight (FoxNews)         | 371   |
| $n_{\text{partisan-news}} =$             | 1,345 |
| Satirical news                           |       |
| Full Frontal with Samantha Bee (TBS )    | 124   |
| Last Week Tonight (HBO)                  | 136   |
| Late Night with Seth Meyers (NBC)        | 216   |
| Late Show with Colbert (CBS)             | 251   |
| Patriot Act with Hasan Minhaj (Netflix ) | 92    |
| Real Time with Bill Maher (HBO)          | 242   |
| The Daily Show (Comedy Central)          | 254   |
| $n_{\text{satirical-news}} =$            | 1,315 |

For both *Twitter* or *YouTube*, data were collected that contained the text that TV shows' accounts had posted to the respective platform, but also included all available additional details (i.e. metadata), such as whether the post contained a video (for *Twitter*). The latter was used as a requirement (a) to hold the conditions for both platforms comparable

because *YouTube* per definition carries a video and (b) to increase the likelihood that comments were directed at a news item and not to, for example, a schedule announcement.

### **Details of Data Retrieval Procedures**

**YouTube.** The *YouTube* Data API (v3) was employed to access the relevant information of *YouTube* videos. The *YouTube Data API* has a default quota allocation of 10,000 units per day, and each API request incurs at least one unit quota (quota cost is determined by the request type). Due to the quota limit, the collection of data lasted several weeks. Posts were collected that were sent between 2011 and 2019.

Three functions were developed to gather the data. First, video IDs of all the channels and playlists were collected using the `channels.list` and `playlistItems.list` methods with the `part` parameter set to “id”. The maximum number of items per page was set to 50. Second, with the video-IDs fetched from the first step, video information details (i.e., the title, description, datetime, view count, like and dislike count, and comment count) were collected using the `videos.list` method with the `part` parameter set to “id”, “snippet” and “statistics”. In total, the details of 58,252 news videos were stored.

Third, the top-level comments (comments that reply directly to these videos; thus, the *user-comments* that are concentrated on in this study) were collected using `commentThreads.list`-method with the fetched video IDs from the first step. Replies (comments that reply to the top-level comments) were collected using `comments.list` method with the parent comment IDs fetched from the top-level comments. To get the respective username, like-count and reply-count of each comment, the `part` parameter was set to “snippet” for both the top-level comments and replies. The maximum number of items per page was set to 100.

In the end, a while-loop was created, within which all three functions were called. As a result, three separate data frames (`video_ids`, `video_info`, and `video_comments`) were generated and stored in an SQL database. A sleep time of 24 hours was also included at the end of the while loop; so, every time the scraper hit the quota limit, it stopped calling the API for a day and re-fetched the API on the next day. In total, 33,640,673 *YouTube* user comments were stored in the database.

**Twitter.** The *Twitter* API—access granted on academic research grounds—allowed the collection of the most recent 3,200 tweets from a single user and an equal number of corresponding replies to those tweets. The *Python* library “*Twython*” was used to collect tweets posted by the selected news shows via the “`get_user_timeline`”-function with “`tweet_mode`”-parameter set to “`extended`” and “`count`” was set to the maximum of 200 per request. A “`while`”-statement was used with the “`get_user_timeline`”-function set to have a “`max_id`”-value of the last tweet collected to eventually reach the 3,200-tweet-limit. The oldest tweet dated back to 2016.

Tweets by the news shows were collected along with all the variables and their metadata. The data was saved in a data frame and later stored at an external SQL server as advised by Oussalah et al. (2013). In total, 4,895 tweets posted by the news accounts were stored.

The reply-tweets (i.e., the *user comments* that we are interested in for this study) were, subsequently, collected via the “`search`”-function set to search any tweets that were directed at the TV shows’ *Twitter* handle. Further code was added to filter out the majority of replies tweets that were not direct replies to our sample of collected tweets that included a video. A maximum of 3,200 replies were acquired *per post* with a “`while`”-statement with similar parameters to the one described above. The “`tweet_mode`”-parameter was set to “`extended`” and the “`count`”-parameter was set to the per-request maximum of 100. Retweets were not

included in the collected tweets nor the replies to retweets and both datasets were cleared of any duplicates. All the available variables found in the replies' .json-file were collected and stored at an external server before they were uploaded to the SQL database ( $n = 2.950.500$  tweets).

### **Sampling**

First, a stratified random sample was drawn from the large databases of tweets and *YouTube* posts of the TV shows. For this selection of *Twitter* and *YouTube* posts by the TV shows, we then retrieved the earliest comments (up to 20 maximum). The sample was stratified according to the three news genres that were included, so this was evenly distributed among regular news, partisan news, and news satire shows.

To augment the similarity of what the comments are discussing, we used the Latent Dirichlet Allocation Topic Model (LDA) approach to determine the most prominent themes in the scraped *YouTube* videos (see Appendix C of Authors, 2022, for the full procedure). Three themes occurred as most prominent in U.S. TV news: the Mueller/Comey investigation, Economy, and the Middle East. All *YouTube* comments responding to videos about these themes were selected employing a keyword search (Table 5c of Authors, 2022), after which a second stratified sample was taken for each of the three topics to assure that an equal distribution was achieved of comments under videos of regular news, partisan news, and satirical news shows. Approximately 3100 user-comments were selected for YouTube, and another 700 for Twitter, see overview in Table 2). The full sample of 3,862 comments was later manually annotated.

**Table 2.** Overview of where user comments originate from.

| Sample                                       | n     |
|--|-------|
| <i>YouTube</i> comments (general)            | 679   |
| <i>YouTube</i> : Mueller/Comey investigation | 828   |
| <i>YouTube</i> : Economy                     | 800   |
| <i>YouTube</i> : Middle East                 | 825   |
| <i>Twitter</i> replies (general)             | 730   |
| Total ( <i>n</i> )                           | 3,862 |

### Appendix B: Codebook items

**Table B1.** Overview of coded variables, their origin, and intercoder reliability.

| <b>Variable</b>         | <b>Brennan-<br/>Prediger's<br/>Kappa</b> | <b>Krippendorff's<br/>Alpha</b> | <b>Percent<br/>Agreement</b> |
|-------------------------|--|---------------------------------|------------------------------|
| Incivility              | 0.68                                     | 0.67                            | 84%                          |
| Interactivity           | 0.60                                     | 0.56                            | 80%                          |
| Rationality             | 0.41                                     | 0.23                            | 70%                          |
| Diversity: Liberal      | 0.78                                     | 0.69                            | 89%                          |
| Diversity: Conservative | 0.88                                     | 0.76                            | 94%                          |

*Note.* Rationality is only based on the presence of reasoning and background information, since its third constitutive indicator (external evidence) was not present sufficiently in the sample to assess its reliability.

## Appendix C: Details of Model Construction and Selection

### Rule-Based Measurements

Before applying the rule-based measures, we lowercased, tokenized, and stemmed all comments in the corpus using the NLTK-package (i.e., `TreebankWordTokenizer` and `PorterStemmer`) (Bird et al., 2009). All dictionaries were stemmed as well to avoid a mismatch between comments and dictionaries. For each concept (interactivity, incivility, rationality, and diversity), we selected multiple automated measurements that appeared to fit our purpose and could reasonably be selected by a researcher interested in studying the normative standards of public discussion under the model of deliberative democracy.

**Interactivity.** No good text-based dictionary seems to exist to measure interactivity. Similar to existing literature (e.g., Collins & Nerlich, 2015; Gruzd et al., 2011), therefore, the present study attempts to simply capture interactivity by detecting @-mentions in comments. Note that for the *Twitter*-corpus (i.e., replies under a show's tweet), each entry already consisted of at least one mention (i.e., responding to the TV show's original tweet), which was first removed before any further investigation.

**Diversity.** Again, it was difficult to find an appropriate dictionary for diversity. The best dictionaries available to measure the partisan nature of comments focus on ideology, especially moral values (f.e. see Zhou et al., 2024). We selected three versions of the Moral Foundation Dictionary (MFD). The MFD is designed to measure the ideological position of the texts by examining the languages used in them, and are both theoretically and empirically related to the partisan nature of text although the exact nature of that relationship remains disputed (Graham et al., 2009; Haidt & Graham, 2007; Hopp et al., 2021). MFD 2.0 is an updated version with further enhancement of psychometric properties that should improve the normality and predictive validity of the dictionary (Frimer et al., 2019). The extended Moral

Foundations Dictionary (eMFD) is the most recent update, which was developed based on crowd-sourced annotated texts (Hopp et al., 2021). Conservative and liberal are measured, respectively, by calculating the ratio of corresponding words indicative of liberal values (fairness, care) and conservative values (authority, loyalty, purity). Two dummy variables were created to represent the ideology of each version of MFD: If a comment has more conservative words, the conservative variable is coded as 1, and the liberal variable as 0, and vice versa. If the counts of conservative and liberal words are equal, both variables are coded as 0.

**Rationality.** For rationality various formal text metrics from the field of computational linguistics are available which are easy to implement and bear similarities to the concept, although it was more difficult to find a good dictionary-based measure. We selected the Flesch-Kincaid index (FK, see Flesch, 1948) and language formality (Heylighen & Dewaele, 2002) metrics to measure language complexity and formality of comments. In addition to the original calculation, all scores from automated approaches were later transformed to dummy variable as well for further analysis (i.e., to calculate F1 scores, precision and recall). The Flesch-Kincaid index was recoded by calculating the difference from the maximum value, so higher scores indicated more reading difficulty. Since the FK score per se has no boundary between high and low, the mean of all data was used to create a dummy variable.

Another index used to measure rationality is the Integrative Complexity score (IC, Owens & Wedeking, 2011). Dissimilar to the FK score, the IC score attempts to measure the semantic complexity of texts, as reflected using certain words belonging to a prescribed category in the LIWC dictionary. Precisely, the IC score is obtained by subtracting the number of words belonging to the negative category (e.g., exclusiveness, certainty, etc.) from the positive category (e.g., inclusiveness, causation, etc.). As counting the positive category of

*sixl* (i.e., words with 6 or more letters) is very sensitive to text length ( $r = .98$ ), we have taken the percentage of *sixl* rather than the absolute count of the category. The resulting correlation has decreased to  $r = .07$ . The mean of all data was used to create a dummy variable.

Also for language formality the scale is normalized to a range from 0 to 100 (see formula below), where the higher score indicates a stronger level of rationality. Such score reflects the *deeper formality* of language, which is often utilized to achieve mutual understanding by reducing the fuzziness and context-dependent words, and simultaneously reinforcing the objectivity and accuracy. Therefore, formality-score is assumed to be a suitable measure for rationality. Frequency in the formula below denotes the “percentages of the number of words belonging to a particular category with respect to the total number of words in the excerpt” (Heylighen & Dewaele, 2002, p. 309).

$$\text{Formality} = \frac{\text{Freq}_{\text{noun}} + \text{Fr}_{\text{adj}} + \text{Freq}_{\text{prep}} + \text{Fr}_{\text{art}} - \text{Freq}_{\text{pron}} - \text{Freq}_{\text{verb}} - \text{Fr}_{\text{adv}} - \text{Fr}_{\text{intj}} + 100}{2}$$

The dummy variable for formality score was split on theoretical grounds at 65, scores above that indicate a formality level on par with scientific texts (Heylighen & Dewaele, 2002). Similarly, the split for the FK score was set at 60, denoting a 10<sup>th</sup> – 12<sup>th</sup> grade readability difficulty.

**Incivility.** Multiple dictionaries have been developed to measure the construct of incivility. We identified six different dictionaries to be tested for this manuscript. These dictionaries include (1) Ksiazek et al.’s (2015) Hostility dictionary and (2) Ksiazek et al.’s Civility dictionary (*reverse-coded*), (3) the Incivility dictionary developed by Muddiman and Stroud (2017), (4) the LIWC-22 (Boyd et al., 2022)<sup>1</sup>, (5) Google’s What Do You Love

---

<sup>1</sup> We used ‘simple swear’ which lists a comment as uncivil if the LIWC-22 swear score > 0.

Project (WDYL) Censored wordlist (Lewis, n.d.), and (6) the Hatebase wordlist constructed by Hatebase.org (2020). All these dictionaries are recoded into dummy variables (0 or 1), to maximize the comparability with the hand-coded data. We define that a comment shows incivility if at least one uncivil word appears.

The rationales and contexts in which these dictionaries were created differ from each other. For example, the Ksiazek et al.’s dictionaries are built to measure user comments on news platforms and social media, meanwhile *LIWC-22* and *Hatebase* wordlists cover a wide range of texts sourced from online and offline texts. Distinctly, *Google* WDYL censored words are not scientifically validated, meaning that whether a word is considered “bad” is based on *Google*’s judgment.

### **Traditional Supervised Machine-Learning using count and tf-idf features**

We used various kinds of Traditional Supervised Machine Learning using count and tf-idf features (SML) to train and build specific classifiers for each of the debate quality concepts using the manually coded data. In this context, our approach involves using models that leverage bag-of-words representations, which can either be count-based or tf-idf-based.

We used an 80:20 train-test split. Since the sample was highly imbalanced for some concepts (e.g., 20.83% of the sample was coded as 1 in the rationality dimension, and 14.88% were coded as Conservative in the diversity dimension), the *resample* function in the *sklearn* package was used to *undersample* the majority class to generate a more balanced training set.<sup>2</sup> The evaluation metrics are calculated on the untouched, fully random, test set.

---

<sup>2</sup> We also tried *Imblearn* for the re-sampling, but this was eventually dropped—and will not be further discussed in this manuscript—since it performed not better than machine-learning without re-sampling and

When training the classifiers towards the best model performance on the training set, eight models were estimated for each variable: two vectorizers (Count and tf-idf)  $\times$  four classifiers (Multinomial Naïve Bayes, Logistic Regression, SVC with a radial (“rbf”) kernel, and SVC with linear kernel). Each model was further tuned by modifying (1) the number of words considered when tokenizing a sentence (ngram\_range); (2) the range of word frequency (max\_df and min\_df), and (3) the standard for regularization that aims to avoid over-fitting in the classifier (classifier\_C).

We then needed to select the best-performing traditional supervised machine-learning (SML) classifiers for each variable. A 5-fold cross-validation with grid search was then conducted for each tuning. After finding the best parameter with the function GridSearchCV of each model (8 models  $\times$  5 variables), the model was validated with a corresponding validation set and its classification scores were saved. Table C1 shows the performance of the best parameter settings for each model on each variable in the test set in terms of macro F1, thus across classes, to enable easier comparison with the results reported in the main manuscript. Within each variable, the model with the highest F1 score of the label 1 (the positive class: a variable was present) among the 8 models was selected as the best SML model for the results section of the main manuscript. Table C2 lists the models that performed best per concept in terms of F1 score on this positive class along with their performance in this class on the test set.

### **Fine-Tuned Transformer Model**

In addition to training classifiers using the above-mentioned traditional techniques, we explore the potential of using transformer-based models in our classification pipeline using Python’s PyTorch library. Here, we use the uncased version of the English-language BERT

---

worse than resample function in *sklearn*. This indicates that *Imblearn*, given its nature of numeric algorithm, might not be able to fit textual data well.

model (bert-base-uncased) and fine-tune it for our classification tasks. During this fine-tuning process, the model's parameters are updated to better suit the specific task at hand.

To address the issue of strong class imbalance, we have implemented a `WeightedLossTrainer` class during the training phase to account for disparities in class representation. Additionally, and for most concepts, optimize training based on the F1 score of the minority class during training. For incivility and liberal, this strategy proved insufficient. Here, we used a down-sampling strategy for the majority classes in the training dataset and subsequently optimized training on a weighted F1 score. We performed hyperparameter tuning, including exploring a range of learning rates and batch sizes. We assessed the model's performance using macro F1 scores and minority class F1 scores on the training set.

## **Generative AI**

Since running GLLMs locally is computationally heavy, we focused on two specific model families for this project: OpenAI's GPT and Meta's Llama. We chose state-of-the-art variants of these models. For Meta we used the latest large model, Llama3.1, more specifically we use the llama3.1:70b-instruct-q6\_K-variant (hereafter Llama3.1:70b). For comparison, we also used the smaller version llama3.1:8b-instruct-q6\_K (hereafter Llama3.1:8b). From OpenAI we choose the two most recent and advanced models available through the Azure OpenAI API: GPT4o (the one released on 2024-08-06) and GPT4-Turbo (the one released on 2024-04-09).

We compared the effect of using different prompts. Since running classifications on (large) generative AI models is computationally expensive, and for OpenAI also financially costly, we tested the effects of different prompts in Llama3.1:70b. For the instruction, or prompt, given to Llama3.1, we first followed the codebook nearly verbatim, often only adding small label specifications (i.e. "not present (*l*)") to help the model classify the data in the correct

classes. For some items, the wording of the codebook appeared to confuse the model, which resulted in high numbers of missing values. In these cases, we asked OpenAI's GPT4o to reformulate the prompt to make it better interpretable for GLLMs. All prompts were checked manually to contain the same information and examples as the codebook. The only changes were in the structuring and wording of the instructions. This procedure resulted in a large number of long prompts, since incivility and rationality were measured by multiple indicators.

However, long prompts are often not optimal for GLLM performance and running them is computationally costly, since you need to process many runs (one for each prompt-and-comment combination) of many tokens (i.e. many words in each prompt) (Törnberg, 2024b). Therefore we also considered a simpler approach. This simpler approach would be the most likely one a researcher without our codebook would use: short concise single prompts per concept rather than indicator.

To test the effect of model size and family, we ran these simple prompts with GPT4o, GPT4-Turbo and the smaller Llama3.1:8b. At the time of writing GPT4o and GPT4-Turbo are the two most advanced OpenAI models available via the Azure OpenAI service (Microsoft, 2024b). We followed advise by Törnberg (2024a) and used a low temperature setting in combination with a seed to improve the replicability of our results, although the creative nature of GLLMs makes perfect replicability impossible. All Llama classifications were run on a server hosted by the University Trier which utilizes four NVIDIA L40S cards with 192 GB of video memory, has two Intel(R) Xeon(R) Silver 4310 CPU @ 2.10GHz with 48 threads, and 256 GB RAM capacity. We used the default model parameters, except for temperature and seed which were set at 0.1 and 42 respectively (see Morgan & Chiang, n.d.). We used the Microsoft Azure OpenAI service to run the classifications for GPT4o and GPT4-Turbo setting the parameters to default, except for temperature (0) and seed (42). To make sure no data is shared with any third parties our University of Amsterdam opted out of abuse

monitoring and content logging in addition to the guarantees offered by Microsoft that no data is shared with third parties (Microsoft, 2024a).

Appendix D shows that Llama3.1:70b attained the best minority class F1 scores for three (rationality, incivility and interactivity) out of our four concepts on the training set. The differences were at times substantial, like for rationality where Llama3.1:70b had a minority F1 of 0.46 versus only 0.30 (GPT4o) and 0.24 (GPT4-Turbo). Llama3.1:70b also reached higher F1 macro scores for rationality and interactivity than GPT4o and GPT4-Turbo. For the variables where GPT4o or GPT4-Turbo did outperform Llama3.1 differences were marginal. For diversity, GPT4o had a slightly higher minority F1 for both liberal 0.67 and conservative 0.67 versus 0.65 and 0.64 respectively for Llama3.1:70b. Likewise, GPT4o had slightly better F1 macro scores for this concept (liberal: 0.79 vs 0.77; conservative 0.80 vs 0.78), while GPT4-Turbo beat it marginally for F1 macro on incivility (0.78 vs 0.75). Llama3.1:70b thus performed comparable or better than the OpenAI models (see Appendix D for full results of all models on the training set). Based on this performance and the financial and ethical considerations mentioned above, the results for Llama3.1:70b are presented in the main results section.

Note that our main conclusions and recommendations hold regardless of whether we had selected the best model per group on macro F1 or minority class F1. Code and all full prompts are available on GitHub.<sup>3</sup> The wording of the prompts used for the main analysis, i.e. the simple prompts per concept are listed in Table C3.

**Table C1.** Performance of different supervised machine-learning (SML) classifiers for each variable based on the test set in macro average F1, Precision, Recall and Accuracy.

| Variable | Vectorizer | Classifier | F1 score | Precision | Recall | Accuracy |
|----------|------------|------------|----------|-----------|--------|----------|
|----------|------------|------------|----------|-----------|--------|----------|

<sup>3</sup> <https://github.com/sjoerdstolwijk/Automatic-Social-Debate-Quality-Metrics>; [https://github.com/cl-trier/TWON-Exploration/blob/master/27--Classify%20PublicSphere-paper\\_2024.ipynb](https://github.com/cl-trier/TWON-Exploration/blob/master/27--Classify%20PublicSphere-paper_2024.ipynb).

|               |       |                     |       |       |       |       |
|---------------|-------|---------------------|-------|-------|-------|-------|
| Interactivity | Count | Multinomial NB      | 0.514 | 0.611 | 0.620 | 0.515 |
|               |       | Logistic Regression | 0.614 | 0.625 | 0.655 | 0.640 |
|               |       | SVC("rbf")          | 0.607 | 0.619 | 0.648 | 0.631 |
|               |       | SVC("linear")       | 0.564 | 0.589 | 0.611 | 0.583 |
|               | Tfidf | Multinomial NB      | 0.551 | 0.635 | 0.652 | 0.554 |
|               |       | Logistic Regression | 0.623 | 0.651 | 0.688 | 0.636 |
|               |       | SVC("rbf")          | 0.621 | 0.649 | 0.686 | 0.635 |
|               |       | SVC("linear")       | 0.625 | 0.651 | 0.688 | 0.640 |
|               |       |                     |       |       |       |       |
| Liberal       | Count | Multinomial NB      | 0.405 | 0.572 | 0.595 | 0.410 |
|               |       | Logistic Regression | 0.616 | 0.617 | 0.682 | 0.697 |
|               |       | SVC("rbf")          | 0.601 | 0.613 | 0.684 | 0.669 |
|               |       | SVC("linear")       | 0.611 | 0.614 | 0.680 | 0.690 |
|               | Tfidf | Multinomial NB      | 0.395 | 0.576 | 0.597 | 0.398 |
|               |       | Logistic Regression | 0.544 | 0.592 | 0.654 | 0.589 |
|               |       | SVC("rbf")          | 0.532 | 0.589 | 0.650 | 0.572 |
|               |       | SVC("linear")       | 0.544 | 0.589 | 0.650 | 0.591 |
|               |       |                     |       |       |       |       |
| Conservative  | Count | Multinomial NB      | 0.392 | 0.570 | 0.614 | 0.404 |
|               |       | Logistic Regression | 0.557 | 0.571 | 0.631 | 0.665 |
|               |       | SVC("rbf")          | 0.528 | 0.555 | 0.606 | 0.627 |
|               |       | SVC("linear")       | 0.543 | 0.564 | 0.621 | 0.647 |
|               | Tfidf | Multinomial NB      | 0.401 | 0.558 | 0.600 | 0.418 |
|               |       | Logistic Regression | 0.511 | 0.573 | 0.647 | 0.572 |
|               |       | SVC("rbf")          | 0.505 | 0.571 | 0.641 | 0.563 |
|               |       | SVC("linear")       | 0.514 | 0.571 | 0.643 | 0.578 |
|               |       |                     |       |       |       |       |
| Rationality   | Count | Multinomial NB      | 0.316 | 0.599 | 0.573 | 0.318 |
|               |       | Logistic Regression | 0.626 | 0.621 | 0.672 | 0.710 |
|               |       | SVC("rbf")          | 0.670 | 0.657 | 0.718 | 0.750 |
|               |       | SVC("linear")       | 0.661 | 0.649 | 0.700 | 0.750 |
|               | Tfidf | Multinomial NB      | 0.384 | 0.605 | 0.608 | 0.384 |
|               |       | Logistic Regression | 0.570 | 0.609 | 0.675 | 0.612 |
|               |       | SVC("rbf")          | 0.581 | 0.614 | 0.683 | 0.625 |
|               |       | SVC("linear")       | 0.579 | 0.613 | 0.682 | 0.622 |
|               |       |                     |       |       |       |       |
| Incivility    | Count | Multinomial NB      | 0.563 | 0.661 | 0.609 | 0.592 |
|               |       | Logistic Regression | 0.683 | 0.683 | 0.683 | 0.684 |
|               |       | SVC("rbf")          | 0.660 | 0.660 | 0.660 | 0.661 |
|               |       | SVC("linear")       | 0.657 | 0.657 | 0.657 | 0.658 |
|               | Tfidf | Multinomial NB      | 0.561 | 0.656 | 0.606 | 0.590 |
|               |       | Logistic Regression | 0.644 | 0.663 | 0.654 | 0.647 |
|               |       | SVC("rbf")          | 0.657 | 0.674 | 0.666 | 0.660 |
|               |       | SVC("linear")       | 0.646 | 0.662 | 0.655 | 0.648 |
|               |       |                     |       |       |       |       |

**Table C2.** The best traditional supervised machine-learning (SML) classifiers for each variable on the positive class (i.e. variable is present) on the test set and their performance on these metrics.

| Variable      | Original ratio | Vectorizer | Classifier          | F1  | Recall | Precision | Accuracy |
|---------------|----------------|------------|---------------------|-----|--------|-----------|----------|
| Interactivity | .28            | TfIdf      | Logistic Regression | .55 | .80    | .42       | .64      |
| —Liberal      | .18            | Count      | Logistic Regression | .45 | .66    | .33       | .70      |
| —Conservative | .15            | TfIdf      | Logistic Regression | .34 | .75    | .22       | .57      |
| Rationality   | .20            | Count      | SVC(“rbf”)          | .51 | .66    | .41       | .75      |
| Incivility    | .47            | TfIdf      | SVC(“rbf”)          | .69 | .79    | .61       | .66      |

**Table C3.** Prompt wording simple prompts.

| variable      | prompt  | categories  |
|---------------|---|---|
| interactivity | Does this comment acknowledge or respond to another user's comment?<br>Instructions: Code Yes (1) if the comment shows agreement or disagreement with a specific user's statement, often signaled by a username or phrases like ‘Yes,’ ‘No,’ or ‘I agree.’ Code No (0) if it lacks a clear acknowledgment or is only an insult.<br>Respond with only the predicted class (0 or 1) of the request.<br>Text: {text}<br>Class: | "0": "No",<br>"1": "Yes"                                  |
| diversity     | Classify the following message as ideologically liberal (0), ideologically neutral (1), or ideologically conservative (2). Ideology here is defined in the context of the US political system. Messages with no ideological content are classified as neutral.<br>Respond with only the predicted class (0 or 1 or 2) of the request.<br>Text: {text}<br>Class:   | "0": "liberal",<br>"1": "neutral",<br>"2": "conservative" |

|             |   |                                   |
|-------------|---|-----------------------------------|
| rationality | <p>Does this comment provide rational analysis?<br/> Instructions: Code Yes (1) if the comment includes:<br/> Context or background,<br/> Evidence (facts, sources, authorities),<br/> Reasoning or structured argument.<br/> Code No (0) if these are absent.<br/> Respond with only the predicted class (0 or 1) of the request.<br/> Text: {text}<br/> Class:</p>                        | <p>"0": "No",<br/> "1": "Yes"</p> |
| incivility  | <p>Does this comment display incivility?<br/> Instructions: Code Yes (1) if the comment includes name-calling, insults, inflammatory language, sarcasm, shouting (ALL CAPS), vulgarity, discrimination, threats, or restrictions on rights. Code No (0) if none of these are present.<br/> Respond with only the predicted class (0 or 1) of the request.<br/> Text: {text}<br/> Class:</p> | <p>"0": "No",<br/> "1": "Yes"</p> |

**Appendix D. Individual classification results of different Generative AI prompts and models on the training set**

**Table D1.** Precision, Recall and F1 score of **simple, short prompts in Llama3.1:8b** against manually coded comments in the training set.

|                            |               | Precision | Recall | F1 score | N    |
|----------------------------|---------------|-----------|--------|----------|------|
| Diversity:<br>Liberal      | 0 (No)        | .88       | .92    | .90      | 2440 |
|                            | 1 (Yes)       | .64       | .51    | .57      | 649  |
|                            | Accuracy      |           |        | .84      |      |
|                            | Macro average | .76       | .72    | .73      | 3089 |
| Diversity:<br>Conservative | 0 (No)        | .92       | .81    | .86      | 2611 |
|                            | 1 (Yes)       | .37       | .60    | .46      | 478  |
|                            | Accuracy      |           |        | .78      |      |
|                            | Macro average | .64       | .71    | .66      | 3089 |
| Rationality                | 0 (No)        | .84       | .98    | .91      | 2541 |
|                            | 1 (Yes)       | .66       | .15    | .24      | 548  |
|                            | Accuracy      |           |        | .83      |      |
|                            | Macro average | .75       | .56    | .57      | 3089 |
| Incivility                 | 0 (No)        | .65       | .93    | .77      | 1567 |
|                            | 1 (Yes)       | .87       | .50    | .63      | 1522 |
|                            | Accuracy      |           |        | .72      |      |
|                            | Macro average | .76       | .71    | .70      | 3089 |
| Interactivity              | 0 (No)        | .83       | .63    | .71      | 2297 |
|                            | 1 (Yes)       | .36       | .62    | .46      | 792  |
|                            | Accuracy      |           |        | .63      |      |
|                            | Macro average | .60       | .62    | .59      | 3089 |

**Table D2.** Precision, Recall and F1 score of **simple, short prompts in Llama3.1:70b** against manually coded comments in the training set.

|                            |               | Precision | Recall | F1 score | N    |
|----------------------------|---------------|-----------|--------|----------|------|
| Diversity:<br>Liberal      | 0 (No)        | .93       | .85    | .89      | 2440 |
|                            | 1 (Yes)       | .57       | .77    | .65      | 649  |
|                            | Accuracy      |           |        | .83      |      |
|                            | Macro average | .75       | .81    | .77      | 3089 |
| Diversity:<br>Conservative | 0 (No)        | .95       | .90    | .92      | 2611 |
|                            | 1 (Yes)       | .57       | .73    | .64      | 478  |
|                            | Accuracy      |           |        | .87      |      |
|                            | Macro average | .76       | .81    | .78      | 3089 |

|               |               |     |     |     |      |
|---------------|---------------|-----|-----|-----|------|
| Rationality   | 0 (No)        | .87 | .96 | .92 | 2541 |
|               | 1 (Yes)       | .66 | .36 | .46 | 548  |
|               | Accuracy      |     |     | .85 |      |
|               | Macro average | .77 | .66 | .69 | 3089 |
| Incivility    | 0 (No)        | .85 | .62 | .72 | 1567 |
|               | 1 (Yes)       | .69 | .89 | .78 | 1522 |
|               | Accuracy      |     |     | .75 |      |
|               | Macro average | .77 | .75 | .75 | 3089 |
| Interactivity | 0 (No)        | .87 | .71 | .78 | 2297 |
|               | 1 (Yes)       | .45 | .70 | .55 | 792  |
|               | Accuracy      |     |     | .70 |      |
|               | Macro average | .66 | .70 | .66 | 3089 |

**Table D3.** Precision, Recall and F1 score of **near-verbatim codebook-based prompts in Llama3.1:70b** aggregated similarly to the manually coded concepts against manually coded comments in the training set.

|                            |               | Precision | Recall | F1 score | N    |
|----------------------------|---------------|-----------|--------|----------|------|
| Diversity:<br>Liberal      | 0 (No)        | .89       | .90    | .89      | 2440 |
|                            | 1 (Yes)       | .60       | .59    | .60      | 649  |
|                            | Accuracy      |           |        | .83      |      |
|                            | Macro average | .75       | .74    | .75      | 3089 |
| Diversity:<br>Conservative | 0 (No)        | .96       | .77    | .86      | 2611 |
|                            | 1 (Yes)       | .40       | .83    | .54      | 478  |
|                            | Accuracy      |           |        | .78      |      |
|                            | Macro average | .68       | .80    | .70      | 3089 |
| Rationality                | 0 (No)        | .97       | .57    | .72      | 2541 |
|                            | 1 (Yes)       | .32       | .93    | .47      | 548  |
|                            | Accuracy      |           |        | .64      |      |
|                            | Macro average | .75       | .56    | .57      | 3089 |
| Incivility                 | 0 (No)        | .90       | .47    | .62      | 1567 |
|                            | 1 (Yes)       | .63       | .94    | .76      | 1522 |
|                            | Accuracy      |           |        | .70      |      |
|                            | Macro average | .77       | .71    | .69      | 3089 |
| Interactivity              | 0 (No)        | .85       | .71    | .77      | 2297 |
|                            | 1 (Yes)       | .42       | .62    | .50      | 792  |
|                            | Accuracy      |           |        | .69      |      |
|                            | Macro average | .63       | .67    | .64      | 3089 |

**Table D4.** Precision, Recall and F1 score of **simple, short prompts in GPT4o** against manually coded comments in the training set.

|                            |               | Precision | Recall | F1 score | N    |
|----------------------------|---------------|-----------|--------|----------|------|
| Diversity:<br>Liberal      | 0 (No)        | .91       | .93    | .92      | 2440 |
|                            | 1 (Yes)       | .71       | .63    | .67      | 649  |
|                            | Accuracy      |           |        | .89      |      |
|                            | Macro average | .81       | .78    | .79      | 3089 |
| Diversity:<br>Conservative | 0 (No)        | .94       | .93    | .94      | 2611 |
|                            | 1 (Yes)       | .64       | .70    | .67      | 478  |
|                            | Accuracy      |           |        | .88      |      |
|                            | Macro average | .79       | .82    | .80      | 3089 |
| Rationality                | 0 (No)        | .85       | .99    | .91      | 2541 |
|                            | 1 (Yes)       | .80       | .19    | .30      | 548  |
|                            | Accuracy      |           |        | .85      |      |
|                            | Macro average | .82       | .59    | .61      | 3089 |
| Incivility                 | 0 (No)        | .68       | .90    | .78      | 1567 |
|                            | 1 (Yes)       | .85       | .57    | .68      | 1522 |
|                            | Accuracy      |           |        | .74      |      |
|                            | Macro average | .77       | .74    | .73      | 3089 |
| Interactivity              | 0 (No)        | .81       | .77    | .79      | 2297 |
|                            | 1 (Yes)       | .42       | .48    | .44      | 792  |
|                            | Accuracy      |           |        | .69      |      |
|                            | Macro average | .61       | .62    | .62      | 3089 |

**Table D5.** Precision, Recall and F1 score of **simple, short prompts in GPT4-Turbo** against manually coded comments in the training set.

|                            |               | Precision | Recall | F1-score | N    |
|----------------------------|---------------|-----------|--------|----------|------|
| Diversity:<br>Liberal      | 0 (No)        | .92       | .88    | .90      | 2440 |
|                            | 1 (Yes)       | .61       | .71    | .65      | 649  |
|                            | Accuracy      |           |        | .84      |      |
|                            | Macro average | .76       | .79    | .78      | 3089 |
| Diversity:<br>Conservative | 0 (No)        | .94       | .92    | .93      | 2611 |
|                            | 1 (Yes)       | .60       | .66    | .63      | 478  |
|                            | Accuracy      |           |        | .88      |      |

|               |               |     |     |     |      |
|---------------|---------------|-----|-----|-----|------|
|               | Macro average | .77 | .79 | .78 | 3089 |
| Rationality   | 0 (No)        | .84 | .99 | .91 | 2541 |
|               | 1 (Yes)       | .84 | .14 | .24 | 548  |
|               | Accuracy      |     |     | .84 |      |
|               | Macro average | .84 | .57 | .57 | 3089 |
| Incivility    | 0 (No)        | .74 | .87 | .80 | 1567 |
|               | 1 (Yes)       | .83 | .69 | .75 | 1522 |
|               | Accuracy      |     |     | .78 |      |
|               | Macro average | .79 | .78 | .78 | 3089 |
| Interactivity | 0 (No)        | .83 | .79 | .81 | 2297 |
|               | 1 (Yes)       | .46 | .53 | .49 | 792  |
|               | Accuracy      |     |     | .72 |      |
|               | Macro average | .65 | .66 | .65 | 3089 |

**Appendix E: Individual classification results of different rule-based measures****Table E1a.** Precision, Recall and F1 score of diversity measures against manually coded diversity scores (Liberal)

|                      |                 | Precision | Recall | F1 score | N   |
|----------------------|-----------------|-----------|--------|----------|-----|
| MFD 1.0<br>(Liberal) | 0 (Non-liberal) | .82       | .84    | .83      | 633 |
|                      | 1 (Liberal)     | .19       | .17    | .18      | 140 |
|                      | Accuracy        |           |        | .72      |     |
|                      | Macro average   | .50       | .50    | .50      | 773 |
| MFD 2.0<br>(Liberal) | 0 (Non-liberal) | .83       | .71    | .77      | 633 |
|                      | 1 (Liberal)     | .22       | .36    | .27      | 140 |
|                      | Accuracy        |           |        | .65      |     |
|                      | Macro average   | .52       | .53    | .52      | 773 |
| eMFD<br>(Liberal)    | 0 (Non-liberal) | .82       | .46    | .59      | 633 |
|                      | 1 (Liberal)     | .19       | .56    | .28      | 140 |
|                      | Accuracy        |           |        | .48      |     |
|                      | Macro average   | .51       | .51    | .43      | 773 |

**Table E1b.** Precision, Recall and F1 score of diversity measures against manually coded diversity scores (Conservative)

|                           |                      | Precision | Recall | F1 score | N   |
|---------------------------|----------------------|-----------|--------|----------|-----|
| MFD 1.0<br>(Conservative) | 0 (Non-conservative) | .86       | .86    | .86      | 660 |
|                           | 1 (Conservative)     | .19       | .20    | .19      | 113 |
|                           | Accuracy             |           |        | .76      |     |
|                           | Macro average        | .53       | .53    | .53      | 773 |
| MFD 2.0<br>(Conservative) | 0 (Non-conservative) | .88       | .66    | .75      | 660 |
|                           | 1 (Conservative)     | .19       | .49    | .28      | 113 |
|                           | Accuracy             |           |        | .63      |     |
|                           | Macro average        | .54       | .57    | .51      | 773 |
| eMFD<br>(Conservative)    | 0 (Non-conservative) | .84       | .67    | .74      | 660 |
|                           | 1 (Conservative)     | .12       | .26    | .16      | 113 |
|                           | Accuracy             |           |        | .60      |     |
|                           | Macro average        | .48       | .46    | .45      | 773 |

**Table E2.** Precision, Recall and F1 score of rationality measures against manually coded rationality score

|                        |                | Precision | Recall | F1 score | N   |
|------------------------|----------------|-----------|--------|----------|-----|
| FK-score               | 0 (Irrational) | .84       | .59    | .69      | 624 |
|                        | 1 (Rational)   | .24       | .53    | .33      | 149 |
|                        | Accuracy       |           |        | .58      |     |
|                        | Macro average  | .54       | .56    | .51      | 773 |
| Language formality     | 0 (Irrational) | .77       | .68    | .72      | 624 |
|                        | 1 (Rational)   | .11       | .17    | .13      | 149 |
|                        | Accuracy       |           |        | .58      |     |
|                        | Macro average  | .44       | .42    | .43      | 773 |
| Integrative Complexity | 0 (Irrational) | .78       | .57    | .66      | 624 |
|                        | 1 (Rational)   | .16       | .35    | .22      | 149 |
|                        | Accuracy       |           |        | .53      |     |
|                        | Macro average  | .47       | .46    | .44      | 773 |

**Table E3.** Precision, Recall and F1 score of incivility measures against manually coded general incivility.

|  |               | Precision | Recall | F1 score | N   |
|--|---------------|-----------|--------|----------|-----|
| Ksiazek's hostility dictionary               | 0 (Civil)     | .65       | .86    | .74      | 408 |
|  | 1 (Uncivil)   | .76       | .49    | .59      | 365 |
|  | Accuracy      |           |        | .68      |     |
|  | Macro average | .71       | .67    | .67      | 773 |
| Ksiazek's civility dictionary (reverse code) | 0 (Civil)     | .50       | .62    | .56      | 408 |
|  | 1 (Uncivil)   | .43       | .31    | .36      | 365 |
|  | Accuracy      |           |        | .48      |     |
|  | Macro average | .46       | .47    | .46      | 773 |
| Google What Do You Love Offensive Wordlist   | 0 (Civil)     | .57       | .98    | .72      | 408 |
|  | 1 (Uncivil)   | .87       | .17    | .28      | 365 |
|  | Accuracy      |           |        | .60      |     |
|  | Macro average | .72       | .57    | .50      | 773 |
|  | 0 (Civil)     | .56       | .99    | .71      | 408 |

|  |               |     |     |     |     |
|--|---------------|-----|-----|-----|-----|
| Muddiman's<br>incivility<br>dictionary | 1 (Uncivil)   | .94 | .12 | .21 | 365 |
|  | Accuracy      |     |     | .58 |     |
|  | Macro average | .75 | .56 | .46 | 773 |
| <hr/>                                  |               |     |     |     |     |
| LIWC-22<br>'simple<br>swear'           | 0 (Civil)     | .58 | .98 | .73 | 408 |
|  | 1 (Uncivil)   | .89 | .19 | .32 | 365 |
|  | Accuracy      |     |     | .61 |     |
| <hr/>                                  |               |     |     |     |     |
| Hatebase<br>wordlist                   | 0 (Civil)     | .55 | .96 | .70 | 408 |
|  | 1 (Uncivil)   | .72 | .11 | .19 | 365 |
|  | Accuracy      |     |     | .56 |     |
| <hr/>                                  |               |     |     |     |     |
|  | Macro average | .63 | .54 | .45 | 773 |

## References

- Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc.
- Boyd, R. L., Ashokkumar, A., Seraj, S., & Pennebaker, J. W. (2022). *The development and psychometric properties of LIWC-22*. University of Texas at Austin.  
<https://www.liwc.app>
- Flesch, R. (1948). A new readability yardstick. *Journal of Applied Psychology*, 32(3), 221–233. <https://doi.org/10.1037/h0057532>
- Frimer, J., Boghrati, R., Haidt, J., Graham, J., & Dehghani, M. (2019). *Moral Foundations Dictionary for Linguistic Analyses 2.0* [Dataset].  
<https://doi.org/10.17605/OSF.IO/EZN37>
- Graham, J., Haidt, J., & Nosek, B. A. (2009). Liberals and conservatives rely on different sets of moral foundations. *Journal of Personality and Social Psychology*, 96(5), 1029–1046. <https://doi.org/10.1037/a0015141>
- Haidt, J., & Graham, J. (2007). When Morality Opposes Justice: Conservatives Have Moral Intuitions that Liberals may not Recognize. *Social Justice Research*, 20(1), 98–116.  
<https://doi.org/10.1007/s11211-007-0034-z>
- Heylighen, F., & Dewaele, J.-M. (2002). Variation in the Contextuality of Language: An Empirical Measure. *Foundations of Science*, 7, 293–340.
- Hopp, F. R., Fisher, J. T., Cornell, D., Huskey, R., & Weber, R. (2021). The extended Moral Foundations Dictionary (eMFD): Development and applications of a crowd-sourced approach to extracting moral intuitions from text. *Behavior Research Methods*, 53(1), 232–246. <https://doi.org/10.3758/s13428-020-01433-0>

- Ksiazek, T. B., Peer, L., & Zivic, A. (2015). Discussing the News: Civility and hostility in user comments. *Digital Journalism*, 3(6), 850–870.  
<https://doi.org/10.1080/21670811.2014.972079>
- Lewis, R. (n.d.). *Naughty word list, compiled by Google and @jamiew*. Github. Retrieved November 20, 2024, from <https://gist.github.com/ryanlewis/a37739d710ccdb4b406d>
- Microsoft. (2024a, November 19). *Data, privacy, and security for Azure OpenAI Service—Azure AI services*. <https://learn.microsoft.com/en-us/legal/cognitive-services/openai/data-privacy>
- Microsoft. (2024b, December 5). *Azure OpenAI Service models—Azure OpenAI*. <https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/models>
- Morgan, J., & Chiang, M. (2024). *Ollama/docs/modelfile.md at main · ollama/ollama*. GitHub. <https://github.com/ollama/ollama/blob/main/docs/modelfile.md>
- Muddiman, A., & Stroud, N. J. (2017). News Values, Cognitive Biases, and Partisan Incivility in Comment Sections: Uncivil Comments. *Journal of Communication*, 67(4), 586–609. <https://doi.org/10.1111/jcom.12312>
- Oussalah, M., Bhat, F., Challis, K., & Schnier, T. (2013). A software architecture for Twitter collection, search and geolocation services. *Knowledge-Based Systems*, 37, 105–120.
- Owens, R. J., & Wedeking, J. P. (2011). Justices and Legal Clarity: Analyzing the Complexity of U.S. Supreme Court Opinions. *Law & Society Review*, 45(4), 1027–1061. <https://doi.org/10.1111/j.1540-5893.2011.00464.x>
- Quinn, T. (2020). *Introducing Hatebase: The world's largest online database of hate speech*. <https://thesentinelproject.org/2013/03/25/introducing-hatebase-the-worlds-largest-online-database-of-hate-speech/>
- Törnberg, P. (2024a). Best Practices for Text Annotation with Large Language Models. *Sociologica*, 18(2), 67–85. <https://doi.org/10.6092/issn.1971-8853/19461>

Törnberg, P. (2024b). Large Language Models Outperform Expert Coders and Supervised Classifiers at Annotating Political Social Media Messages. *Social Science Computer Review*. <https://doi.org/10.1177/08944393241286471>

Zhou, A., Liu, W., Kim, H. M., Lee, E., Shin, J., Zhang, Y., Huang-Isherwood, K. M., Dong, C., & Yang, A. (2024). Moral Foundations, Ideological Divide, and Public Engagement with U.S. Government Agencies' COVID-19 Vaccine Communication on Social Media. *Mass Communication and Society*, 27(4), 739–764. <https://doi.org/10.1080/15205436.2022.2151919>