Political Analysis of Social Media Data

Supervised Learning 2

Instructor: Gregory Eady
Office: 18.2.10
Office hours: Fridays 13-15

Today

- Supervised learning 2
  - TF-IDF
  - Measuring supervised learning performance
  - Receiver Operator Characteristic (ROC) Curves
  - Area under the curve
  - The research frontier
- Video lectures & exercises

Supervised learning with text data

- In, for example, topic models, we typically use raw counts in a document feature matrix (DFM)
- With supervised learning models, we might want to transform our DFM to emphasize the importance of each word/token to each specific document
- But how can we account for word/token importance?
- The idea: A word/token that is used in *many* documents is not good at differentiating *between* those documents
- Enter: TF-IDF

How relevant is each word to each document?

- **Term Frequency-Inverse Document Frequency**
- Important in information retrieval & machine learning
- Each word/token is scored by how frequent it appears in a specific document (just like our normal Document Frequency Matrix)
- But is weighted by how frequent that word/token appears in all documents
- Term frequency is how often a term occurs in a specific document
- Inverse document frequency is how often a term occurs across documents
- TF-IDF is calculated as: term frequency × inverse document frequency

TF-IDF calculation

- **Term frequency**
  - Count of a term (in a given document)
  - Count of a term as a proportion of all terms in a document
    - The most commonly used
  - Binary of whether a term is in a document or not
  - log(1 + count of a term)

TF-IDF calculation

- **Inverse document frequency:**
  - $log(\frac{\text{Total number of documents}}{\text{Number of documents containing term}})$
  - Other possibilities, but this is the most common
  - log() has the practical implication of assuming importance of a term is not weighted linearly

TF-IDF example

	i	am	a	republican	democrat
Document A	4	3	8	2	0
Document B	4	3	8	0	2

TF-IDF of “i” in Document A

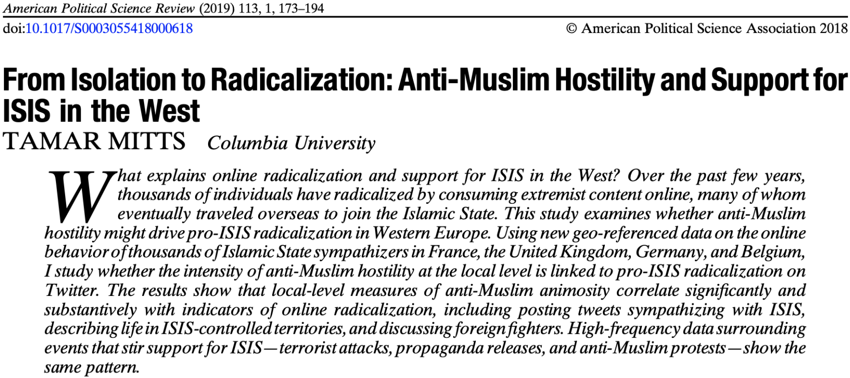
tf: $4 \ / \ (4 + 3 + 8 + 2 + 0) = 0.24$
idf: $\log(2 \ / \ 2) = 0$
tfidf: 0.24 * $0 = 0$

TF-IDF of “republican” in Document A

tf: $2 \ / \ (4 + 3 + 8 + 2 + 0) = 0.12$
idf: $\log(2 \ / \ 1) = 0.30$
tfidf: 0.12 * $0.3 = 0.04$

Use TF-IDF DFM to fit machine learning models

- Some machine learning models benefit from using the TF-IDF version of the Document Feature Matrix (DFM)
- You can use it in your lasso/ridge/elastic net and tree-based (random forests) methods, for example
- In the lab, you will use both the raw DFM count and the TF-IDF transformed DFM in a supervised learning context

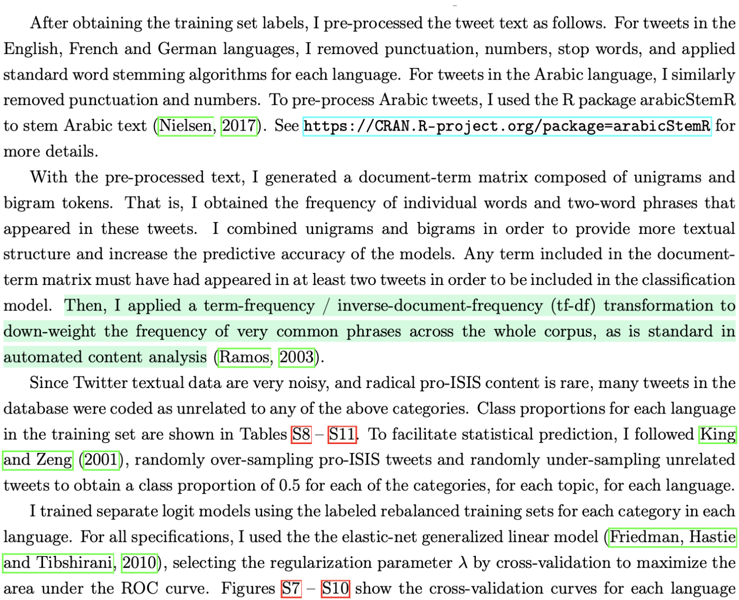


American Political Science Review (2019) 113, 1, 173–194
doi:10.1017/S0003055418000618

From Isolation to Radicalization: Anti-Muslim Hostility and Support for ISIS in the West

TAMAR MITTS *Columbia University*

What explains online radicalization and support for ISIS in the West? Over the past few years, thousands of individuals have radicalized by consuming extremist content online, many of whom eventually traveled overseas to join the Islamic State. This study examines whether anti-Muslim hostility might drive pro-ISIS radicalization in Western Europe. Using new geo-referenced data on the online behavior of thousands of Islamic State sympathizers in France, the United Kingdom, Germany, and Belgium, I study whether the intensity of anti-Muslim hostility at the local level is linked to pro-ISIS radicalization on Twitter. The results show that local-level measures of anti-Muslim animosity correlate significantly and substantively with indicators of online radicalization, including posting tweets sympathizing with ISIS, describing life in ISIS-controlled territories, and discussing foreign fighters. High-frequency data surrounding events that stir support for ISIS—terrorist attacks, propaganda releases, and anti-Muslim protests—show the same pattern.



After obtaining the training set labels, I pre-processed the tweet text as follows. For tweets in the English, French and German languages, I removed punctuation, numbers, stop words, and applied standard word stemming algorithms for each language. For tweets in the Arabic language, I similarly removed punctuation and numbers. To pre-process Arabic tweets, I used the R package arabicStemR to stem Arabic text (Nielsen, 2017). See `https://CRAN.R-project.org/package=arabicStemR` for more details.

With the pre-processed text, I generated a document-term matrix composed of unigrams and bigram tokens. That is, I obtained the frequency of individual words and two-word phrases that appeared in these tweets. I combined unigrams and bigrams in order to provide more textual structure and increase the predictive accuracy of the models. Any term included in the document-term matrix must have had appeared in at least two tweets in order to be included in the classification model. Then, I applied a term-frequency / inverse-document-frequency (tf-df) transformation to down-weight the frequency of very common phrases across the whole corpus, as is standard in automated content analysis (Ramos, 2003).

Since Twitter textual data are very noisy, and radical pro-ISIS content is rare, many tweets in the database were coded as unrelated to any of the above categories. Class proportions for each language in the training set are shown in Tables S8 – S11. To facilitate statistical prediction, I followed King and Zeng (2001), randomly over-sampling pro-ISIS tweets and randomly under-sampling unrelated tweets to obtain a class proportion of 0.5 for each of the categories, for each topic, for each language.

I trained separate logit models using the labeled rebalanced training sets for each category in each language. For all specifications, I used the the elastic-net generalized linear model (Friedman, Hastie and Tibshirani, 2010), selecting the regularization parameter λ by cross-validation to maximize the area under the ROC curve. Figures S7 – S10 show the cross-validation curves for each language

Measuring performance

- If we test multiple supervised learning models, or try multiple hyperparameters for a single model, how do we know which one is “best”?
    - Accuracy
    - Precision
    - Recall
    - F1 score
    - **Receiver Operator Characteristic (ROC) Curve**
    - **Area Under the Curve (AUC)**

Confusion matrix

		Actual value	
		0	**1**
Predicted	**0**	True negatives	False negatives
	1	False positives	True positives

- **Accuracy**: correct predictions / total predictions
- **Precision**: true positives / (true positives + false positives)
- **Recall**: true positives / (true positives + false negatives)

Confusion matrix example

		Actual value	
		0	**1**
Predicted	**0**	4000	100
	1	300	200

- **Accuracy**:
- **Precision**:
- **Recall**:

Confusion matrix example

		Actual value	
		0	**1**
Predicted	**0**	4000	100
	1	300	200

- **Accuracy**: $4200 / 4600 = 0.91$
  - i.e. Proportion of correct predictions
- **Precision**:
- **Recall**:

Confusion matrix example

		Actual value	
		0	**1**
Predicted	**0**	4000	100
	1	300	200

- **Accuracy**: $4200\ /\ 4600 = 0.91$
  - i.e. Proportion of correct predictions
- **Precision**: $200\ /\ 500 = 0.40$
  - i.e. Proportion of correct positive predictions
- **Recall**:

Confusion matrix example

		Actual value	
		0	**1**
Predicted	**0**	4000	100
	1	300	200

- **Accuracy**: $4200 / 4600 = 0.91$
  - i.e. Proportion of correct predictions
- **Precision**: $200 / 500 = 0.40$
  - i.e. Proportion of correct positive predictions
- **Recall**: $200 / 300 = 0.67$
  - i.e. Proportion of positives predicted as positive

The precision versus recall tradeoff

High recall, but **low precision**

		Actual value	
		0	**1**
Predicted	**0**	0	0
	1	4000	100

Accuracy: 100 / (4000 + 100) = 0.02
Precision: 100 / (4000 + 100) = 0.02
Recall: 100 / (100 + 0) = 1

High precision, but **low recall**

		Actual value	
		0	**1**
Predicted	**0**	4000	80
	1	0	20

Accuracy: 4020 / (4000 + 100) = 0.98
Precision: 20 / (0 + 20) = 1
Recall: 20 / (20 + 80) = 0.20

F1 score (or F score)

- A balance between recall and precision
- Punishes an algorithm as the difference between recall and precision increases

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

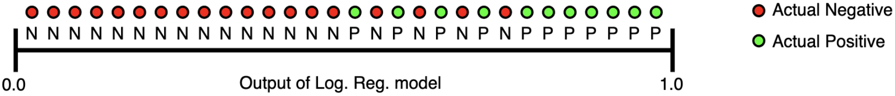
Example of how F1 score punishes differences in precision & recall

Recall	Precision	Average	F1 Score
0.7	0.7	0.7	0.700
0.65	0.75	0.7	0.696
0.6	0.8	0.7	0.686
0.5	0.9	0.7	0.643

Receiver Operator Characteristic (ROC) Curve

- Some machine learning models assign probabilities to classes e.g. logistic regression (lasso/ridge/elastic net)
- This allows us to flexibly examine how well a model predicts an outcome
- Why? Because we have a choice about what cutoff to set to assign a 0 to or a 1 to when we make a prediction (a 1 needn't be only for $\hat{y}_i > 0.5$)
- If we set the cutoff at a specific probability (e.g. 0.3), how many true positives will we classify? How many false positives?
- Useful because decision-making often concerns the tradeoff between true positive and false positives (e.g. COVID)

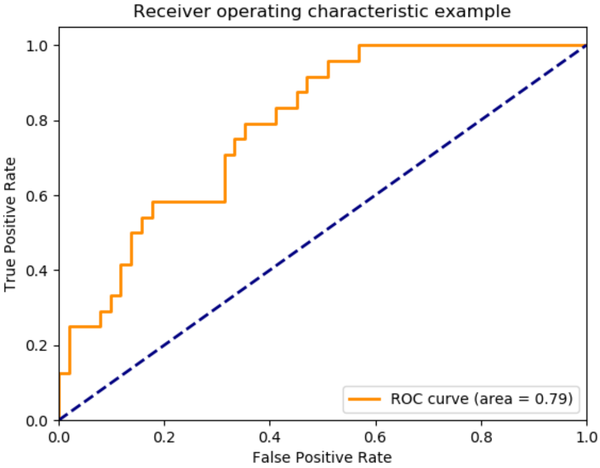
Outcomes of models are often in probabilities


N N N N N N N N N N N N N N N P N P N P N P N P P P P P P P
0.0
Output of Log. Reg. model
1.0
Actual Negative
Actual Positive

Receiver Operator Characteristic (ROC) Curve

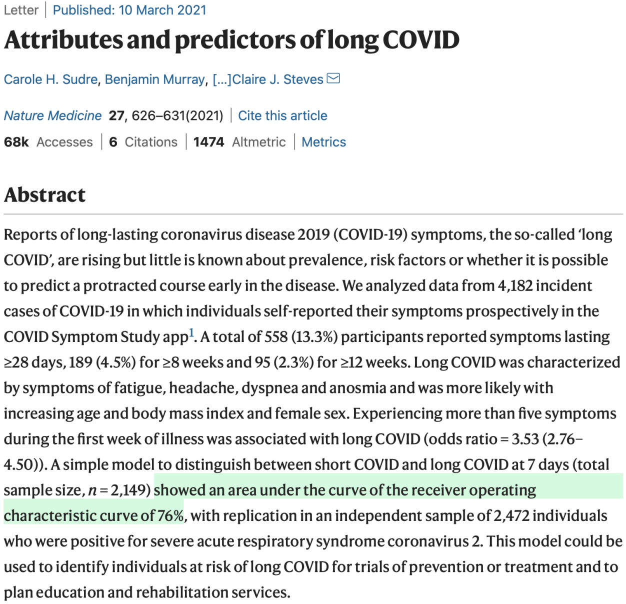
- ROC curves visualize the tradeoff between true positives and false positives
- True Positive Rate: $\frac{\text{True positives}}{\text{True positives}+\text{False negatives}}$
  - i.e. Out of all actual positives, how many do you correctly classify as a positive
- False Positive Rate: $\frac{\text{False positives}}{\text{False positives}+\text{True negatives}}$
  - i.e. Out of all actual negatives, how many do you incorrectly classify as a positive

Receiver Operator Characteristic (ROC) Curve



Area Under the Curve (AUC)

- The area under an ROC curve provides a summary statistic of a classifier's performance across all thresholds
- It also equals the probability that a randomly chosen positive case is ranked higher than a randomly chosen negative case
- Rather than train a machine learning model/algorithm for accuracy, some therefore use the AUC statistic instead



Letter | Published: 10 March 2021

Attributes and predictors of long COVID

Carole H. Sudre, Benjamin Murray, […]Claire J. Steves ✉

Nature Medicine **27**, 626–631(2021) | Cite this article

68k Accesses | **6** Citations | **1474** Altmetric | Metrics

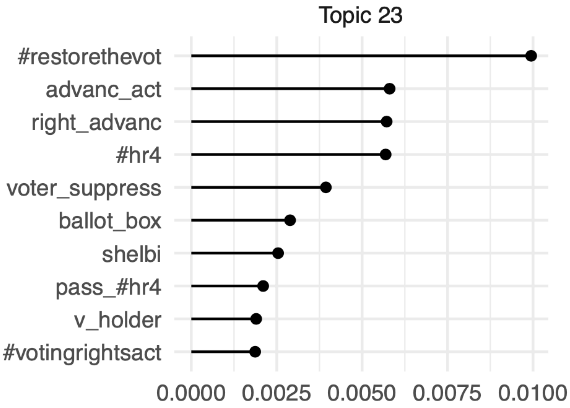
Abstract

Reports of long-lasting coronavirus disease 2019 (COVID-19) symptoms, the so-called 'long COVID', are rising but little is known about prevalence, risk factors or whether it is possible to predict a protracted course early in the disease. We analyzed data from 4,182 incident cases of COVID-19 in which individuals self-reported their symptoms prospectively in the COVID Symptom Study app[1]. A total of 558 (13.3%) participants reported symptoms lasting ≥28 days, 189 (4.5%) for ≥8 weeks and 95 (2.3%) for ≥12 weeks. Long COVID was characterized by symptoms of fatigue, headache, dyspnea and anosmia and was more likely with increasing age and body mass index and female sex. Experiencing more than five symptoms during the first week of illness was associated with long COVID (odds ratio = 3.53 (2.76–4.50)). A simple model to distinguish between short COVID and long COVID at 7 days (total sample size, $n = 2,149$) showed an area under the curve of the receiver operating characteristic curve of 76%, with replication in an independent sample of 2,472 individuals who were positive for severe acute respiratory syndrome coronavirus 2. This model could be used to identify individuals at risk of long COVID for trials of prevention or treatment and to plan education and rehabilitation services.

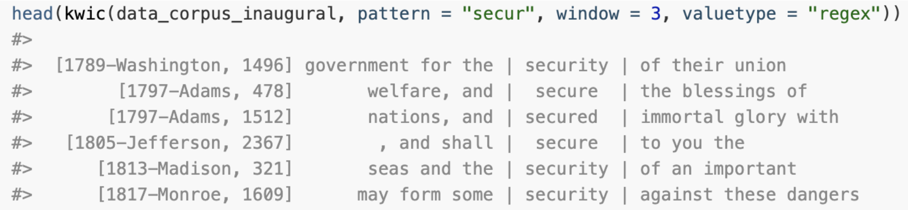
Key Words in Context (KWIC)

- In LDA models, it is often unclear what a specific word or token means
- To get an understanding of specific tokens, it's often useful to look at them in their original context...

Example of the need for token context



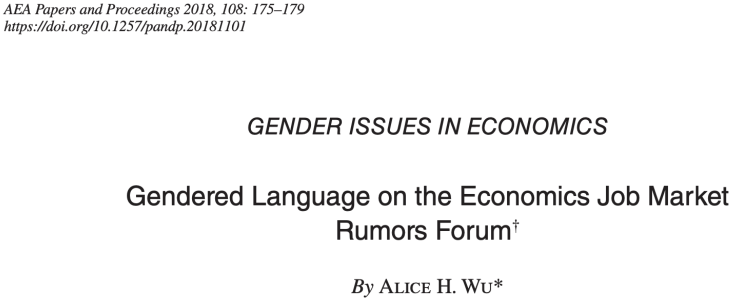
kwic() in quanteda



```
head(kwic(data_corpus_inaugural, pattern = "secur", window = 3, valuetype = "regex"))
#>
#>  [1789-Washington, 1496] government for the | security | of their union
#>        [1797-Adams, 478]       welfare, and |  secure  | the blessings of
#>       [1797-Adams, 1512]       nations, and | secured  | immortal glory with
#>   [1805-Jefferson, 2367]        , and shall |  secure  | to you the
#>      [1813-Madison, 321]       seas and the | security | of an important
#>      [1817-Monroe, 1609]     may form some | security | against these dangers
```

Re-thinking supervised learning

- Standard way to think about supervised learning is as a predictive exercise
- Predictions are a means to classification or, say, forecasting
- The coefficients or predictions in these models nevertheless can be meaningful in themselves
- A couple of examples:
    - Wu (2018)
    - Peterson & Spirling (2018) and Green et al. (2020)



AEA Papers and Proceedings 2018, 108: 175–179
https://doi.org/10.1257/pandp.20181101

GENDER ISSUES IN ECONOMICS

Gendered Language on the Economics Job Market Rumors Forum†

By ALICE H. WU*

Is academic culture in economics unwelcoming to women?

- Women are generally under-represented in STEM
- It has been suggested that part of this is due to an unwelcoming culture toward women in these fields
- But how does one examine this empirically?

Examining culture empirically

- Wu (2018) collects posts from the Economics Job Market Rumors (EJMR) message board
  - An anonymous web forum for discussion among economists, designed originally for talk about the economics job market
- Collects posts from EJMR (Oct 2013 - Oct 2017)
- Use keywords to find posts about women (e.g. “she”, “woman”) and posts about men (e.g. “he”, “man”)

Empirical strategy

- Removes keywords from all posts, and then uses the remaining words/tokens to predict which posts were about a woman, and which were about a man
- Fits a lasso (logistic) regression model to predict posts about women and about men
- Examines the words that are the most predictive of a post being about a woman or a man...

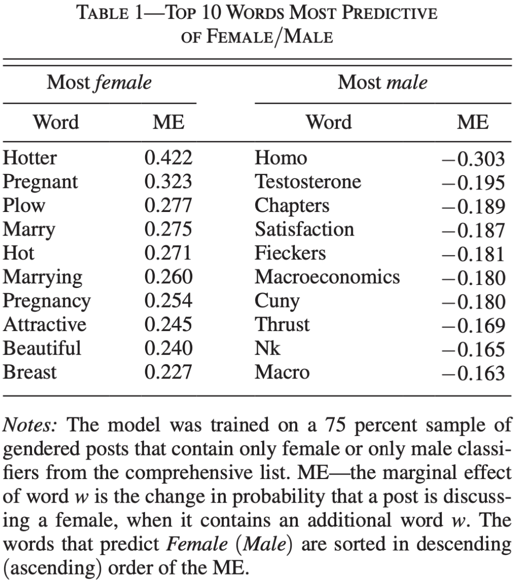


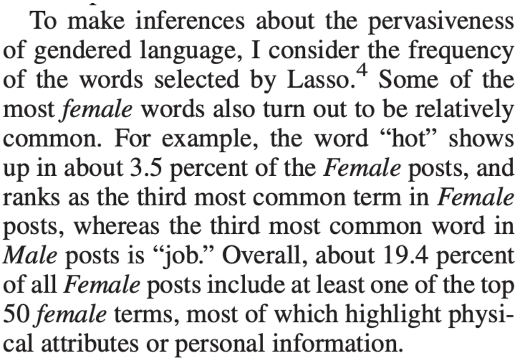
Table 1—Top 10 Words Most Predictive of Female/Male

Most *female*		Most *male*	
Word	ME	Word	ME
Hotter	0.422	Homo	−0.303
Pregnant	0.323	Testosterone	−0.195
Plow	0.277	Chapters	−0.189
Marry	0.275	Satisfaction	−0.187
Hot	0.271	Fieckers	−0.181
Marrying	0.260	Macroeconomics	−0.180
Pregnancy	0.254	Cuny	−0.180
Attractive	0.245	Thrust	−0.169
Beautiful	0.240	Nk	−0.165
Breast	0.227	Macro	−0.163

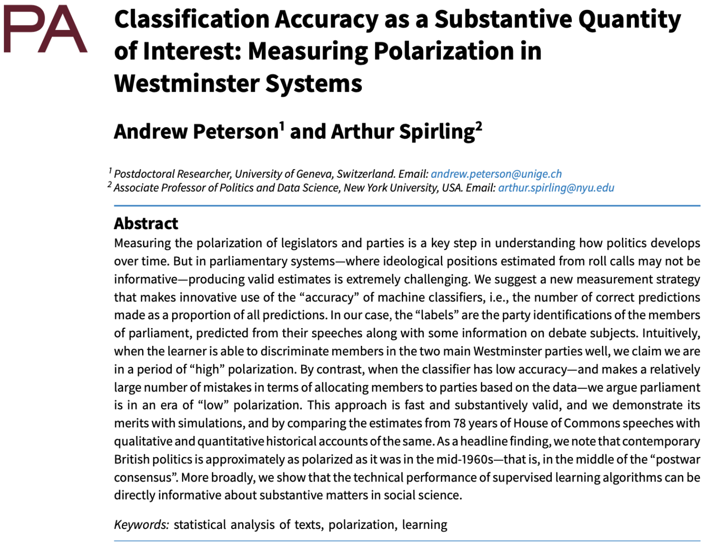
Notes: The model was trained on a 75 percent sample of gendered posts that contain only female or only male classifiers from the comprehensive list. ME—the marginal effect of word *w* is the change in probability that a post is discussing a female, when it contains an additional word *w*. The words that predict *Female* (*Male*) are sorted in descending (ascending) order of the ME.

Always examines the frequency of “female” words

- Some of the most predictive stereotypical words were also the most frequent...



To make inferences about the pervasiveness of gendered language, I consider the frequency of the words selected by Lasso.[4] Some of the most *female* words also turn out to be relatively common. For example, the word “hot” shows up in about 3.5 percent of the *Female* posts, and ranks as the third most common term in *Female* posts, whereas the third most common word in *Male* posts is “job.” Overall, about 19.4 percent of all *Female* posts include at least one of the top 50 *female* terms, most of which highlight physical attributes or personal information.



Classification Accuracy as a Substantive Quantity of Interest: Measuring Polarization in Westminster Systems

Andrew Peterson[1] and Arthur Spirling[2]

[1] *Postdoctoral Researcher, University of Geneva, Switzerland. Email: andrew.peterson@unige.ch*
[2] *Associate Professor of Politics and Data Science, New York University, USA. Email: arthur.spirling@nyu.edu*

Abstract

Measuring the polarization of legislators and parties is a key step in understanding how politics develops over time. But in parliamentary systems—where ideological positions estimated from roll calls may not be informative—producing valid estimates is extremely challenging. We suggest a new measurement strategy that makes innovative use of the "accuracy" of machine classifiers, i.e., the number of correct predictions made as a proportion of all predictions. In our case, the "labels" are the party identifications of the members of parliament, predicted from their speeches along with some information on debate subjects. Intuitively, when the learner is able to discriminate members in the two main Westminster parties well, we claim we are in a period of "high" polarization. By contrast, when the classifier has low accuracy—and makes a relatively large number of mistakes in terms of allocating members to parties based on the data—we argue parliament is in an era of "low" polarization. This approach is fast and substantively valid, and we demonstrate its merits with simulations, and by comparing the estimates from 78 years of House of Commons speeches with qualitative and quantitative historical accounts of the same. As a headline finding, we note that contemporary British politics is approximately as polarized as it was in the mid-1960s—that is, in the middle of the "postwar consensus". More broadly, we show that the technical performance of supervised learning algorithms can be directly informative about substantive matters in social science.

Keywords: statistical analysis of texts, polarization, learning

How can we measure political polarization?

- Measuring polarization in a parliamentary system is challenging because roll call votes are not overly informative
- Authors use the *errors* in a supervised learning model to measure polarization

How can we measure political polarization?

- Measuring polarization in a parliamentary system is challenging because roll call votes are not overly informative
- Authors use the *errors* in a supervised learning model to measure polarization
- Data from *Hansard* record of British parliamentary debates from 1935 to 2013 (3.5 million speeches)

Measuring polarization with supervised learning accuracy

- Machine learning model used to predict whether a speaker is from one party or the other (Conservative or Labour)
- In periods of low polarization, little will differentiate the speech of members of each party, thus *low* predictive accuracy
- In periods of high polarization, much will differentiate the speech of members of each party, thus *high* predictive accuracy

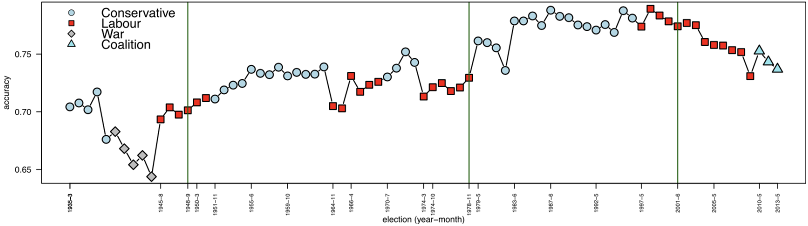
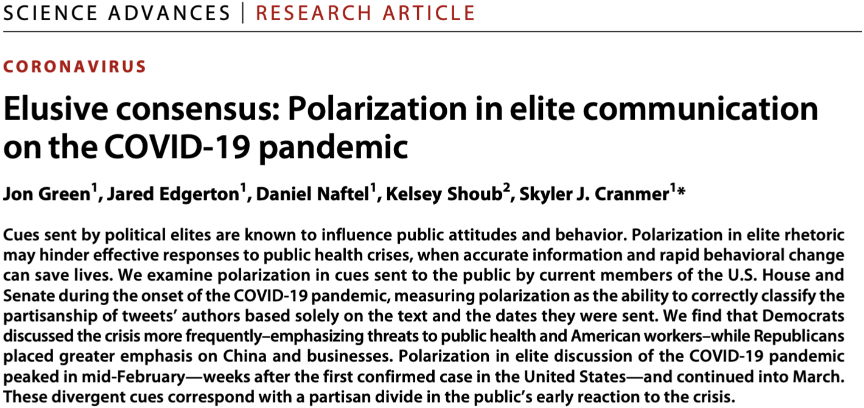


Figure 3. Estimates of parliamentary polarization, by session. Election dates mark *x*-axis. Estimated change points are [green] vertical lines.



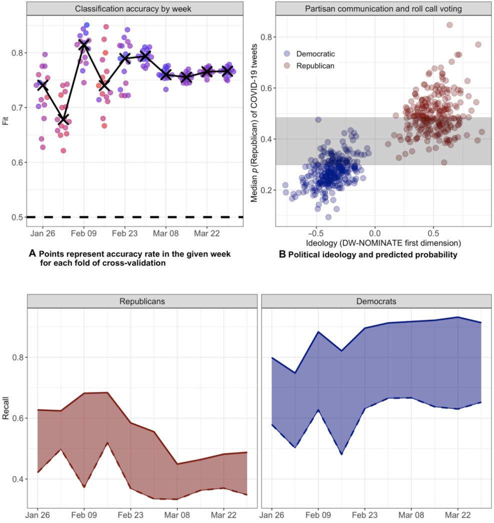
SCIENCE ADVANCES | RESEARCH ARTICLE

CORONAVIRUS

Elusive consensus: Polarization in elite communication on the COVID-19 pandemic

Jon Green[1], Jared Edgerton[1], Daniel Naftel[1], Kelsey Shoub[2], Skyler J. Cranmer[1]*

Cues sent by political elites are known to influence public attitudes and behavior. Polarization in elite rhetoric may hinder effective responses to public health crises, when accurate information and rapid behavioral change can save lives. We examine polarization in cues sent to the public by current members of the U.S. House and Senate during the onset of the COVID-19 pandemic, measuring polarization as the ability to correctly classify the partisanship of tweets' authors based solely on the text and the dates they were sent. We find that Democrats discussed the crisis more frequently–emphasizing threats to public health and American workers–while Republicans placed greater emphasis on China and businesses. Polarization in elite discussion of the COVID-19 pandemic peaked in mid-February—weeks after the first confirmed case in the United States—and continued into March. These divergent cues correspond with a partisan divide in the public's early reaction to the crisis.



A Points represent accuracy rate in the given week for each fold of cross-validation

B Political ideology and predicted probability

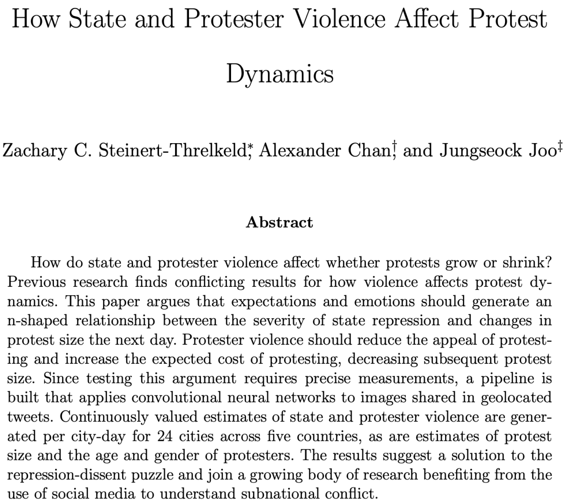
C Recall above no-information rate for Republicans and Democrats

Fig. 2. Classification accuracy, partisan COVID-19 language by roll call voting, and recall above no-information rate. Plot (**A**) *k*-fold prediction out of sample by week. Classification accuracy increases over time. This suggests that Democratic and Republican members of Congress are becoming more polarized over time. Plot (**B**) shows the increases of political ideology of members of Congress by the median predicted probability of their test set tweets being authored by a Republican. Plot (**C**) shows rates of recall (recovery of true cases) by party. The lower bound is the naive probability of correctly classifying a Republican or Democratic member as such based solely on prevalence in the test sets, the upper bound displays the observed rate of recall, and the shaded area represents the increase in recall above the no-information rate.

Other topics you might be interested in

- Image data
- Network analysis and visualization

Image data



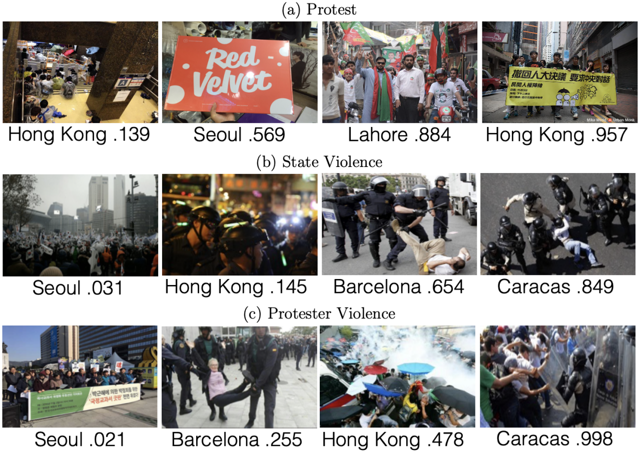
How State and Protester Violence Affect Protest Dynamics

Zachary C. Steinert-Threlkeld[*], Alexander Chan[†], and Jungseock Joo[‡]

Abstract

How do state and protester violence affect whether protests grow or shrink? Previous research finds conflicting results for how violence affects protest dynamics. This paper argues that expectations and emotions should generate an n-shaped relationship between the severity of state repression and changes in protest size the next day. Protester violence should reduce the appeal of protesting and increase the expected cost of protesting, decreasing subsequent protest size. Since testing this argument requires precise measurements, a pipeline is built that applies convolutional neural networks to images shared in geolocated tweets. Continuously valued estimates of state and protester violence are generated per city-day for 24 cities across five countries, as are estimates of protest size and the age and gender of protesters. The results suggest a solution to the repression-dissent puzzle and join a growing body of research benefiting from the use of social media to understand subnational conflict.

Figure 1: Sample Images and Their Classifier Outputs



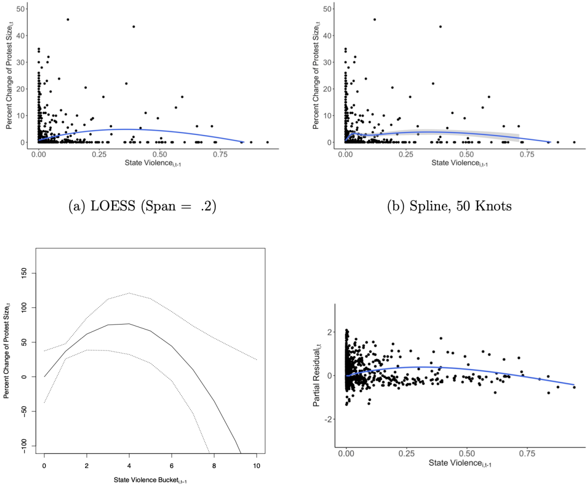
(a) Protest

(b) State Violence

(c) Protester Violence

Note: The top panel shows sample images and the protest classifier's rating of them. The use of hard negatives in the training set ensures that scenes that contain crowds (bottom row, left), individuals walking on streets (top row, third), or a non-protest sign (bottom row, third) are not included in analysis. The middle panel shows protest images with their state violence rating and the bottom shows protest images' protester violence rating. Labels contain each image's city and label probability.

Figure 5: State Violence Results Remain in Flexible Operationalizations



(a) LOESS (Span = .2)

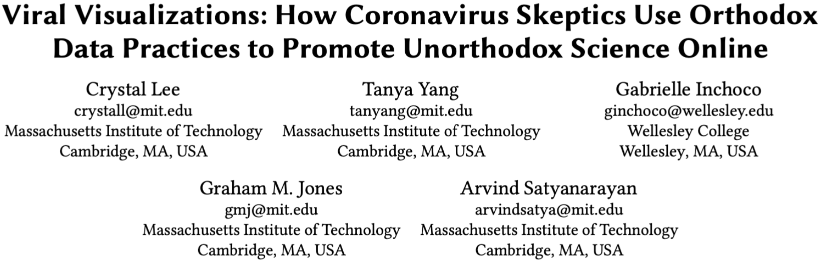
(b) Spline, 50 Knots

(c) Binned Marginal Effects

(d) LOESS on Partial Residuals

Note: The n-shaped relationship holds in a non-parametric relationship (a, b). Generating fixed effects for state violence in bins of width of .1 finds the same relationship (c). Regressing state violence on the partial residuals of Model 3 from Table A5.

Network analysis and visualization / digital ethnography



Viral Visualizations: How Coronavirus Skeptics Use Orthodox Data Practices to Promote Unorthodox Science Online

Crystal Lee
crystall@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, USA

Tanya Yang
tanyang@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, USA

Gabrielle Inchoco
ginchoco@wellesley.edu
Wellesley College
Wellesley, MA, USA

Graham M. Jones
gmj@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, USA

Arvind Satyanarayan
arvindsatya@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, USA

Tweet, retweet, quote tweet, mention, reply network

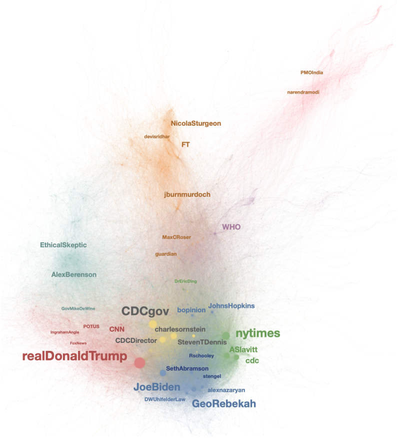
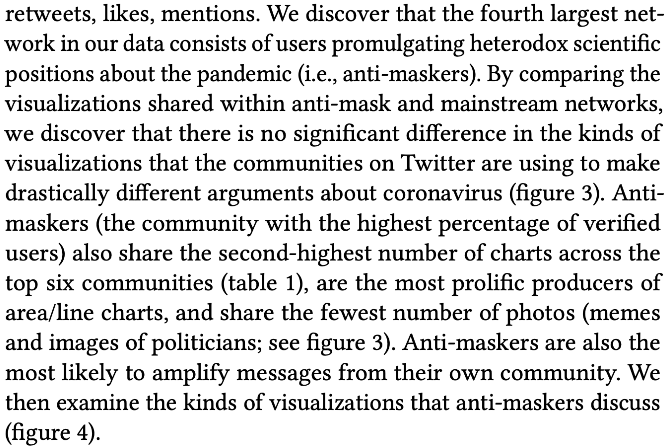


Figure 2: A network visualization of Twitter users appearing in our corpus. Color encodes community as detected by the Louvain method [12], and nodes are sized by their degree of connectedness (i.e., the number of other users they are connected to).

Anti-maskers use data similarly on social media



retweets, likes, mentions. We discover that the fourth largest network in our data consists of users promulgating heterodox scientific positions about the pandemic (i.e., anti-maskers). By comparing the visualizations shared within anti-mask and mainstream networks, we discover that there is no significant difference in the kinds of visualizations that the communities on Twitter are using to make drastically different arguments about coronavirus (figure 3). Anti-maskers (the community with the highest percentage of verified users) also share the second-highest number of charts across the top six communities (table 1), are the most prolific producers of area/line charts, and share the fewest number of photos (memes and images of politicians; see figure 3). Anti-maskers are also the most likely to amplify messages from their own community. We then examine the kinds of visualizations that anti-maskers discuss (figure 4).

Visualization sharing among communities (based on image embeddings to link similar images)

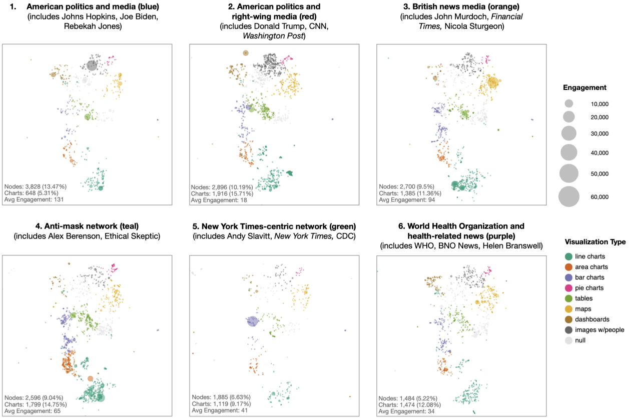


Figure 3: Visualizing the distribution of chart types by network community (with top accounts listed). While every community has produced at least one viral tweet, anti-mask users (group 6) receive higher engagement on average.