The Elephant in the Chamber? Incorporating Tweets about Trump into Congressional Ideal Point Estimates

Gregory Spell¹, Brian Guay², D. Sunshine Hillygus³, and Lawrence Carin⁴

¹, ⁴Department of Electrical and Computer Engineering, Duke University
², ³Department of Political Science, Duke University

Abstract

Political science has traditionally measured legislator preferences using roll-call voting behavior, but we consider the possibility that legislative votes might fail to adequately capture attitudes toward President Donald Trump. We estimate Congressional ideal points using an alternative expression of preferences: tweeting. Sampling from a population of all tweets authored by members of the U.S. House and Senate related to Donald Trump from November 2016 to February 2018, we incorporate the frequency, timing, and content of these tweets into representations of legislator ideology using a neural net approach. We construct a model in which legislator attitudes towards Trump are represented as vector embeddings, and Donald Trump himself is represented as a vector that is constructed using a neural network from the text of his daily tweets. In our model, the interaction between legislator embeddings and Trump embeddings produces predictions of both the number of times legislators tweet about Donald Trump and the sentiment of tweets about Donald Trump. We assess the quality of our learned representations for legislators by comparing to the canonical DW-NOMINATE representations as well as votes on Trump-endorsed legislation.
**Introduction**

Despite the political polarization pervading U.S. politics today, one does not need look far to find examples of high-profile Republicans criticizing Republican president Donald Trump. In 2017, Republican Senator Bob Corker referred to the Trump White House as an “adult day-care center,” John McCain wrote in the *Washington Post* that the President “is often poorly informed and can be impulsive in his speech and conduct,” and former Arizona Senator Jeff Flake once called Trump a “danger to a democracy.” After Trump’s failure to denounce the violent white nationalist rally in Charlottesville—saying “You also had some very fine people on both sides”— dozens of Republican Senators and Representatives rebuked the president’s response to the event. These public criticisms of Trump have come at the same time that many of these same members of Congress have voted with the President on policy legislation. For example, Arizona Senators Jeff Flake and John McCain—Republican Senators with a reputation for criticizing Trump—nonetheless voted with him over 80% of the time while in office.

These examples highlight a core limitation of using roll-call votes to measure the political preferences of members of Congress. In addition to well-documented substantive and methodological limitations (Groseclose and McCarty, 2001; Snyder Jr and Groseclose, 2000; Clinton et al., 2004; Clinton, 2012), roll-call based measures of legislator preferences also risk failing to capture important dimensions of political preferences that may be orthogonal to legislative policy preferences, such as attitudes toward a sitting president. While scholars of mass behavior routinely rely on surveys measures of both ideology and presidential approval in seeking to explain the public’s political attitudes and behaviors, no such measure currently exists for political elites (who are somewhat less inclined to answer political surveys).

In this paper, we estimate Congressional ideal points using legislators’ tweets, which are both more abundant and complex than binary roll-call votes. For the first time in history, all members of the House and Senate had Twitter accounts in 2018, up from just 30% in 2009, resulting in thousands of tweets published every week. Using a population of tweets about President Trump authored by legislators during a 15-month period following election day in 2016, we estimate ideal points for members of the U.S. House and Senate. We incorporate both
the frequency and content of these tweets using a joint count-sentiment model, which assigns each legislator a vector embedding that encodes their attitudes regarding President Trump. Additionally, these legislator embeddings interact with vector representations of Donald Trump himself, constructed from a neural network using the text and timing of his (many) tweets during this same time frame. The predictive capabilities of the model are demonstrated on a set of previously "unseen" (i.e., held-out) tweets, which provides a quantitative means with which to assess the quality of our learned representations. We then compare our model-obtained embeddings to the canonical DW-NOMINATE representations as well as other metrics of support for Trump, such as the proportion of the time legislators vote with Trump and the president’s vote share in members’ districts in 2016.

Representing Legislator Preferences

A mainstay of modern political analysis over the past half-century has been the modeling of the ideal point of a legislator as a standard measure of the political preferences of members of Congress. These ideal points constitute a spatial model for legislative behavior, as both legislators and policies are represented in a low-dimensional Euclidean space (Poole and Rosenthal, 1985, 1991, 1997). Traditionally, the primary data with which to estimate ideal points are the recorded votes legislators cast on bills, known as roll-call votes. These ideal points are interpreted as measures of ideological preferences and have been used to test theories of legislative behavior, polarization, and political representation (Tausanovitch and Warshaw, 2018; Treier, 2010, 2011; Carnes, 2012).

There are, however, several challenges with representing a legislator’s preferences using chamber votes on legislation. First and foremost, roll-call scaling methods rely on the assumption of “sincere proximity voting” (Laver 2014), that legislators with relatively similar voting records will have closer ideal points than those with relatively different voting records. When this assumption is violated, the meaning of measures of ideological preferences using roll-call scaling methods becomes less clear. One way in which this assumption can be violated is when party leadership exerts pressure on caucus members to vote in a specific manner (Snyder Jr, 1991; Snyder Jr and Groseclose, 2000; Groseclose and McCarty, 2001). In such cases,
NOMINATE scores are more of an indication of which party a member belongs to, rather than where the member is ideologically situated within their party (Spirling and McLean, 2006). Strategic voting and abstentions, when legislators intentionally vote against their preferences or refuse to vote entirely, offer a similar opportunity for a legislator’s preferences and voting decisions to diverge (Roberts, 2007; Spirling and McLean, 2007; Clinton, 2012; Cohen and Noll, 1991; Forgette and Sala, 1999). The actual substance of bills themselves can also introduce confounds for estimating preferences when the analysis includes commemorative legislation or bills with unrelated amendments (i.e., christmas tree bills or poison pill legislation). Furthermore, because roll-call measures of ideology require casting a vote, they are unable to measure preferences of political entities such as non-incumbent candidates (but see Bonica (2013)).

Roll-call scaling measures also face several methodological challenges. Typical means to estimate legislator ideal points often rely upon Bayesian techniques trained over the entirety of a vote matrix for legislation (Clinton et al., 2004; Jackman, 2001; Goplerud, 2018). These methods, however, suffer from the extensive computation required to perform such training algorithms as Markov Chain Monte Carlo (MCMC) or Expectation Maximization (EM). Even more importantly, they are incapable of predicting how legislators will vote on a new piece of legislation, as some legislator votes must be observed in order for the model to characterize the policy under consideration.

A variety of solutions have been proposed to address these challenges (see Tausanovitch and Warshaw (2017) for an overview). Recent work has attempted to integrate the text of legislation with roll-call data to improve ideal point estimates and vote predictions. For instance, Gerrish and Blei (2011)’s extension of the ideal point model places legislation into a “political space” based upon the latent topics of the legislation’s text. In addition to allowing vote prediction on new bills, their model — the Ideal Point Topic Model (IPTM) — also aids in political study by producing clusters of terms that may be interpreted as prevalent political issues. Further efforts to incorporate bill text using topic models come from Gerrish and Blei (2012), Nguyen et al. (2015), and Gu et al. (2014). Other efforts have incorporated ancillary data into ideal point estimates such as constituent preferences and party influence (Clinton et al., 2004; Groseclose and McCarty, 2001).
Leveraging Text to Model Legislator Preferences

Among the most promising advances in measuring legislator preferences are those that leverage political text. With the development of expressive vector representations of words (Bengio et al., 2003; Turian et al., 2010; Mikolov et al., 2013; Pennington et al., 2014), the natural next step in legislative text analysis was to represent bills using word embeddings. Kraft et al. (2016) represent legislators using ideal vectors as a multi-dimensional extension to ideal points\(^1\). Kornilova et al. (2018) augment bill text with bill metadata — i.e., bill sponsor information — to improve the predictive capabilities of legislator embeddings. An additional improvement over (Kraft et al., 2016), Kim (2014) model bill text using a convolutional neural network (CNN) rather than the average over bill word embeddings.

In addition to augmenting existing roll-call scaling measures, text data may also be used to measure political preferences and attitudes independently. Especially after the most recent United State presidential election in 2016, the influence of social media on politics has been studied extensively, specifically for the microblogging website Twitter. Recent work measures political attitudes and behavior using “likes” and “retweets” (Wang et al., 2016b,d,a,c), the network of political actors individuals follow (Barbera, 2015), and the text of individual tweets (Lreotiuc-Pietro et al. 2017, Laver, Benoit & Garry 2003, Marwick & Boyd 2011) (Preoiuc-Pietro et al., 2017; Laver et al., 2003; Marwick and boyd, 2011).

Moreover, one of the limitations of much of the existing analyses using Twitter data to model individual preferences is the unknown generalizability of the results. When analyzing a sample of Twitter users it is unclear if the estimated preferences of representative of a broader population of interest. Unlike surveys that have traditionally sampled from a known population, enabling inferences to that population to be drawn, identities on the ‘twittersphere’ cannot be verified and self-selection into using Twitter creates inferential confounds. Therefore, Twitter is best used to measure political preferences when a well-defined population is available to sample from (e.g., Barbera (2015)).

In this paper, we model attitudes toward President Donald Trump using an embedding

---

\(^1\)It is worth noting that some previous methods for estimating ideal points admitted multiple dimensions, but none considered these representations to be embeddings. We recognize that it should still be proper to refer to ideal vectors as ideal points and specify the dimensionality when appropriate.
method on a well-defined Twitter population — members of the U.S. House and Senate. We assembled a dataset comprising all of the tweets by legislators specifically about President Trump during the 15-month period proceeding his election, as well as all tweets authored by President Trump since his inauguration in January 2017. Using training data from a sample of legislator tweets that were labeled with respect to sentiment, we construct a joint model for both the legislator tweet counts and tweet sentiment regarding Trump. We then assess the quality of our learned representations for legislators by comparing to the canonical DW-NOMINATE representation and votes on Trump-endorsed legislation. Our initial analysis not only validates the modeling approach but also highlights that attitudes towards Donald Trump are not being captured by legislative voting behavior.

Model

For legislator tweets about Donald Trump, we seek to jointly model both the number of tweets sent and the sentiment of some of those tweets. For each legislator, our model predicts the tweet counts on each day, and it additionally predicts the tweet sentiment for labeled tweets using information about the legislator who tweeted and on which day. We learn both a set of “day embeddings” (or “Trump embeddings”) that interact with a set of legislator embeddings. For a particular day \( t \), let \( \tau_t \in \mathbb{R}^K \) be a vector that represents Donald Trump on that day. Indexing legislators by \( i \in \{1, 2, \ldots, N\} \), we endow a legislator \( i \) with a vector \( v_i \in \mathbb{R}^K \) as well as a bias term \( b_i \in \mathbb{R} \). While the legislator embeddings are learned as free parameters of the model, the Trump embeddings are constructed using the text of Donald Trump’s tweets.

Tweet Count Model

Let \( x_{it} \) be the number of tweets that legislator \( i \) sends about Donald Trump on day \( t \). A natural choice to model this count variable is as being drawn from a Poisson distribution:

\[
x_{it} \sim \text{Poisson}(\lambda_{it})
\]  

(1)

The rate parameter of this Poisson distribution, \( \lambda_{it} \), is assumed to be a function of the representations for Donald Trump at \( t \) and for legislator \( i \): \( \lambda_{it} = f(\tau_t, v_i) \). Since we seek
to map Trump and legislator embeddings to a scalar, and since we further seek to model the
interaction between these embeddings, we choose the exponent of the dot-product between
these embeddings. Using an exponential ensures the rate parameter is non-negative:

$$\lambda_{it} = f(\mathbf{\tau}, \mathbf{v}_i) = \exp(\mathbf{\tau}_i^\top \mathbf{v}_i)$$  \hspace{1cm} (2)$$

With the count model defined as above, we train the model by minimizing the negative log-
likelihood of the training data. For a single day and single legislator, we define the count-loss as:

$$L_{\text{count}}(x_{it}; \lambda_{it}) = -\log\left(\frac{\lambda_{it}^{x_{it}} e^{-\lambda_{it}}}{x_{it}!}\right)$$  \hspace{1cm} (3)$$

and for a training set $X_{tr}$, the total count-loss is then:

$$L_{\text{count}} = \sum_{x_{it} \in X_{tr}} L_{\text{count}}(x_{it}; \lambda_{it})$$  \hspace{1cm} (4)$$

An alternative distribution with which to model the tweet counts is the Negative Binomial
distribution. Unlike the Poisson distribution, the Negative Binomial is parameterized by two
parameters, which allows it more modeling flexibility. In particular, while the Poisson dis-
tribution is limited in having its mean and variance be the same value, such is not true for
the Negative Binomial. Consequently, the Negative Binomial distribution is more suitable for
overdispersed data — in which the variance is larger than the mean. This characteristic holds
true for the tweet counts, and so we investigate the Negative Binomial model and compare to
the Poisson in Section.

To describe the Negative Binomial using parameters $(p, r)$, the probability mass function,
when $x_{it} \sim \text{Neg.Bin}(p, r)$ is written as:

$$\Pr(x_{it}; p, r) = \binom{x_{it} + r - 1}{x_{it}} p^{x_{it}} (1 - p)^r$$  \hspace{1cm} (5)$$

where $\binom{N}{k}$ is the binomial coefficient. As the parameter $p$ represents the probability of success
in a Bernoulli trial, we choose to learn a mapping to $p$ similarly as we did for $\lambda$ in the Poisson
distribution. Again, modeling the interaction between Trump and legislator embeddings, we
express this map as:

\[ p_{it} = f(\tau_t, v_i) = \sigma(\tau_t^T v_i) \]  

(6)

where \( \sigma(\cdot) \) is the sigmoid function defined by \( \sigma(x) = \frac{1}{1+\exp(-x)} \), which is used to transform the input onto \((0, 1)\) to represent a probability. The remaining parameter \( r \) for the Negative Binomial is learned as a common free parameter for all legislators and days. As with the Poisson formulation of the count model, the parameters are learned by minimizing the negative log-likelihood of the training data, which again allows us to compute the total count-loss for the training set \( \mathcal{X}_{it} \) using equation 4.

**Tweet Sentiment Model**

Let \( y_{it} \) be an ordinal variable that encodes the sentiment legislator \( i \) expresses in a tweet about Donald Trump on day \( t \). We consider an ordinal model to account for the varying possible gradations of approval and disapproval. Assuming \( L \) possible sentiment levels, the ordinal model is parameterized by a set of cutpoints:

\[ C = \{ c_0 < c_1 \leq c_2 \leq \cdots \leq c_{L-1} < c_L \} \]  

(7)

where cutpoints \( c_0 \) and \( c_L \) are defined to be \(-\infty\) and \( \infty \), respectively. The remaining cutpoints are learned during model training.

Let \( z_{it} \in \mathbb{R} \) be a continuous latent variable underlying the ordinal response \( y_{it} \in \{1, 2, \ldots L\} \). Then for a thresholded ordinal model, the predicted tweet sentiment takes value \( l \) for which: \( c_{l-1} < z_{it} < c_l \). Under a cumulative link model (CLM) for ordinal regression, this yields a model in which the predicted probability of a particular sentiment level \( l \) is given by:

\[ p(y_{it} = l | z_{it}; C) = \sigma(c_l - z_{it}) - \sigma(c_{l-1} - z_{it}) \]  

(8)

where again \( \sigma(\cdot) \) is the sigmoid function described in section .

The latent variable \( z_{it} \) is a function of the attributes of legislator \( i \) and of Trump at day \( t \). As with the count model, we seek to employ a map that captures the interaction between the
legislator and Trump embeddings, and thus we employ a weighted inner product. Additionally, we expect that legislators maintain a concrete bias towards Trump, which we include in the term $b_i$ for each legislator. Thus, we obtain the variable $z_{it}$ through the following map:

$$z_{it} = g(v_i, \tau_t, b_i) = \tau_t^T H_g v_i + b_i$$  \hspace{1cm} (9)$$

where $H_g \in \mathbb{R}^{K \times K}$ is a learned weight matrix. As with the count model, the sentiment model is trained by optimizing the negative log-likelihood of the sentiment-labeled tweets in the training set. With the predicted probability of the correct label, $p(y_{it} = l)$, given by equation 8, and the set of all labeled tweets in the training set being $\mathcal{Y}_{tr}$, then the total loss for the sentiment model is given by:

$$L_{sent} = \sum_{y_{it} \in \mathcal{Y}_{tr}} \sum_{l \in \{1, 2, \ldots, L\}} I(y_{it} = l) \log p(y_{it} = l)$$  \hspace{1cm} (10)$$

where $I(\cdot)$ denotes the indicator function, in which $I(\cdot) = 1$ when the argument is true and 0 otherwise.

**Trump Embedding Construction**

In the ideal point/vector models that consider roll-call data, legislator behavior was a response to policies which were described by the text of bills. As we seek to pivot away from voting behavior and toward the consideration of a more publicly demonstrated behavior — tweeting about Donald Trump — we seek to construct embeddings that legislators can respond to. Since Donald Trump is our entity of investigation, and Twitter the medium, we use the text of his tweets to construct such embeddings.

To map Donald Trump’s tweet text to a political embedding representation, we employ a Simple Word-Embedding Model (SWEM), explored by Shen et al. (2018). SWEMs rely upon word embeddings (Bengio et al., 2003) and pooling operations to encode the compositionality of text without the heavy parameterization required of such models as RNNs (Socher et al., 2011) or CNNs (Kalchbrenner et al., 2014; Kim, 2014). Endowing each word-token $u_i$ in a lexicon with an embedding $w_i \in \mathbb{R}^d$, then we may represent a sequence of $n$ words as a matrix of stacked embeddings: $\{w_1, \ldots, w_L\} = W \in \mathbb{R}^{n \times d}$. To extract the most salient features
from every word-embedding dimension, we employ a max-pooling operation, which amounts to a column-wise maximum of matrix $W$. Supposing that $W_t$ contains the embeddings from all Donald Trump tweets on day $t$, then we will denote $\alpha_t \in \mathbb{R}^d$ as the max-pooled vector.

After obtaining a feature vector $\alpha_t$ for the tweet text, the feature vector is mapped to the daily Trump vector via a function $\tau_t = h(\alpha_t)$. The simplest choice for the map $h(\cdot)$ is an affine transformation:

$$\tau_t = M\alpha_t + a$$  \hspace{1cm} (11)

where $M \in \mathbb{R}^{d \times K}$ and $a \in \mathbb{R}^K$ are a weight matrix and bias vector that are shared by all days $t$.

This transform can be made more flexible by introducing a non-linearity, the hallmark of a neural network. Choosing a nonlinear activation function, $\phi(\cdot)$, such as the hyperbolic tangent (Tanh) or rectified linear unit (ReLU), we can obtain more modeling flexibility at the expense of increasing the number of parameters and possibly overfitting the data. This non-linear “hidden” layer is described by:

$$\tau_t = M_2 \phi(M_1 \alpha_t + a_1) + a_2$$  \hspace{1cm} (12)

where an additional weight matrix and bias vector have been appended.

**Model Training**

**Data**

We obtained all publicly-available tweets by members of the U.S. House of Representative and Senate from TweetCongress, a Sunlight Foundation initiative that has compiled tweets from Members of Congress since 2009. We restricted the sample to only those tweets that contained any of a specific set of terms related to Donald Trump (in addition to twitter handle): “Donald Trump”, “Trump”, “realDonaldTrump”, “MAGA” (an acronym for Trump’s campaign slogan “Make America Great Again”), “whitehouse”, “WhiteHouse”, “POTUS” (acronym for “President of the United States”), and “potus”. Of these, we further restricted the tweets to span
in time from November 2016 to February 2018. This culling process yielded 29,411 tweets. The mean number of Trump-related tweets during this time period was 62.5 (median = 34.5). Only 11% of legislators had fewer than 5 tweets related to Trump, and Democrats had over twice as many tweets (mean = 80.7, median = 46) as Republicans (mean = 30.4, median = 20). Post-estimation we restrict our analysis to Members with at least 5 tweets related to Trump.

Of the population of 29,411 Trump-related tweets, a subset of 4,588 tweets were randomly selected to be coded with respect to their sentiment about Trump. Five undergraduate research assistants were trained to categorize each tweet about Trump as either very positive, somewhat positive, neutral, somewhat negative, or very negative. The coders were given the text of each tweet, as well as the author’s name and party affiliation for context, but were instructed not to simply use party as a heuristic to determine sentiment. A random 1% sample of tweets was selected to be coded by each of the five coders in order to assess inter-coder reliability. The mean level of agreement among each pair of coders was only 60.0%, largely due to disagreement over the degree to which each tweet was positive or negative (e.g., somewhat positive vs. very positive, or somewhat negative vs. very negative). However, when collapsing into 3 categories (positive, neutral, negative), the mean level of agreement rose to 91.7%. Therefore, we used each tweet’s collapsed 3-category label to train the model.

We divided the tweets temporally by day into disjoint training, validation, and test sets, such that such that all tweets from each day were randomly assigned to one of the three sets. The training set contains 70% of all days, the validation set contains 10%, and the test set contains 20%. A table outlining how many days and tweets (both labeled and total) are included in each set is provided in Table 1.

<table>
<thead>
<tr>
<th></th>
<th># Days</th>
<th># Labeled</th>
<th># Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>299</td>
<td>3024</td>
<td>19947</td>
</tr>
<tr>
<td>Validation</td>
<td>41</td>
<td>407</td>
<td>2400</td>
</tr>
<tr>
<td>Test</td>
<td>85</td>
<td>1157</td>
<td>7064</td>
</tr>
</tbody>
</table>

Table 1: Breakdown of Splits into Training, Validation, and Test
Parameter Learning

The parameters in the model to be learned include the legislator embeddings/biases, the word embeddings, and the parameters of the maps to count and ordinal variables. We jointly refer to this collection as $\Theta$. The optimization objective is the combination loss of the negative-log likelihood of the count and ordinal models:

$$
\mathcal{L}(\Theta) = \gamma \mathcal{L}_{\text{count}} + (1 - \gamma) \mathcal{L}_{\text{ord}}
$$

where $\mathcal{L}_{\text{count}}$ and $\mathcal{L}_{\text{ord}}$ are given by equations 4 and 10, respectively, and $\gamma$ is a hyperparameter that controls the relative importance of the two component losses. The construction of equation 13 allows the researcher to only admit tweet count information by setting $\gamma = 1$ and only admit tweet sentiment information by setting $\gamma = 0$; a balance may be achieved by choosing $\gamma \in (0, 1)$.

The Adam algorithm (Kingma and Ba, 2015) is used for gradient-based optimization of 13 with a learning rate of $\eta = 10^{-4}$.

Predictive Results

To demonstrate the efficacy of our model for legislator tweeting behavior with respect to President Donald Trump, we first show that the construction of Trump embeddings from the language of his own tweets provides an informational signal for legislators to react to. We train our model using the days for the training set and present the predictive results for held-out (“unseen”) days. Since the model seeks to capture two aspects of legislator tweeting behavior, we evaluate the model using two metrics: the negative-log likelihood of the count model and the mean-absolute-error (MAE) of the sentiment model. Overall model performance is also captured by the total loss of the model, which is the weighted negative-log likelihood of both the count and sentiment models, described in equation 13. MAE is used rather than accuracy to account for the ordinal nature of the sentiment model.
<table>
<thead>
<tr>
<th></th>
<th>Text</th>
<th>No Text</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Loss</td>
<td>MAE</td>
</tr>
<tr>
<td>Poisson</td>
<td>6831</td>
<td>0.60</td>
</tr>
<tr>
<td>Neg. Bin.</td>
<td><strong>6123</strong></td>
<td><strong>1.03</strong></td>
</tr>
</tbody>
</table>

Table 2: Predictive evaluation metrics for our model with $\gamma = 1$. Results achieved on the sequestered test set. Note that because only the count loss is being optimized, MAE does not reflect model performance here. Best model result is bolded.

$\gamma = 1$ (only count model):

When $\gamma = 1$, only the loss from the part of the model that handles tweet counts contributes to the total loss in equation 13. We present the final negative log-likelihood of the count model for both the Poisson and Negative Binomial models described in Section , and for both the case in which the text of Donald Trump’s tweets is used to construct his daily embedding representation and the case in which the Trump embeddings are free parameters of the model. In all cases, the dimension of the model was set to be $K = 2$ and the model was trained for 3000 iterations, except for the Poisson model without text, which was trained for 20,000 iterations. The predictive results are shown in Table 2.

Two important aspects of count model performance are immediately clear from Table 2. The first is that modeling legislator tweet counts using the Negative Binomial distribution achieves superior performance to modeling using the Poisson distribution. The second is that using the text of Donald Trump’s tweets to construct his daily embedding that legislator embeddings interact with provides significantly better results than neglecting the text and allowing the Trump embeddings to be free parameters of the model. This effect is more pronounced for the Poisson distribution, but is present for the Negative Binomial model as well. Indeed, this aspect of the model is to be expected, since the model is evaluated on days of which there are no examples in the training set. Without using the text in this case, there is no way for the model to represent an “unseen” day.

Since the Negative Binomial count model outperforms the Poisson model, we shall henceforth only show results for the Negative Binomial.
Table 3: Predictive evaluation metrics for our model with $\gamma = 0$. Results achieved on the sequestered test set. Best model result with respect to MAE is bolded. Comparison drawn between the sentiment model with and without the legislator bias term as well as with/without Trump tweet text.

<table>
<thead>
<tr>
<th></th>
<th>Text</th>
<th>No Text</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Loss</td>
<td>MAE</td>
</tr>
<tr>
<td>No Bias</td>
<td>219.23</td>
<td>0.167</td>
</tr>
<tr>
<td>Bias</td>
<td>218.71</td>
<td>0.167</td>
</tr>
</tbody>
</table>

$\gamma = 0$ (only sentiment model):

When $\gamma = 0$, only the loss from the part of the model that handles legislator tweet sentiment contributes to the total loss in equation 13. We present the final model loss — which is the negative log-likelihood of the sentiment model — as well as the model MAE when the model is evaluated on the held-out test set consisting of unseen days of tweets. Again, we show results for the case in which Trump’s tweet text is used to construct embeddings and the case in which the text is not used. Consistent with the count model for which results have been shown, the model dimension was set to $K = 2$, but now the number of iterations has been adjusted to 3000 for training with text and 45,000 for training without text. Further note that an additional model setting is being toggled for analysis: the inclusion of the legislator bias term, $b_i$, from equation 9. The results are presented in Table 3.

In addition to the MAE of the ordinal model, we note that the model accuracy — which is more intuitive but less exact than MAE — is 86.7% for the model (including legislator bias) with text and is 67.1% for the model with no text and no legislator bias. Again, there are several aspects of Table 3 that merit discussion. It is interesting to note that when the text of Donald Trump’s tweets is used, the model performs as well with respect to MAE with the inclusion of the legislator bias as without it. Additionally, when the bias term is included, the model is able to achieve comparable results without the use of daily Trump text. With respect to MAE, these results are inferior to those including text, but the model loss is curiously lesser. For the case of no Trump text and no legislator bias, the model is incapable of achieving test MAE better than how it performs upon initialization. We note that while the model does train, performance on the test (and validation) sets never improves in that case.

The results presented above warrant discussion of the legislator bias term in equation 9. Table 3 suggests that the legislator bias (when present) accounts for much of the model’s ability
Table 4: Predictive evaluation metrics for our model with $\gamma = 0.01$. Results achieved on the sequestered test set.

<table>
<thead>
<tr>
<th>Count NLL</th>
<th>MAE</th>
<th>Total Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Text</td>
<td>7310</td>
<td>0.390</td>
</tr>
<tr>
<td>Text</td>
<td>6108</td>
<td>0.160</td>
</tr>
</tbody>
</table>

to predict legislator tweet sentiment, since the model achieves decent results even when no Trump text is used to construct meaningful Trump embeddings to interact with the trained legislator embeddings. Without the bias term, the interaction between Trump and legislator embeddings is the only means toward predicting tweet sentiment, which is why the necessity of text is so critical in that case.

$\gamma = 0.01$ (both counts & sentiment):

For any other value of $\gamma \in (0, 1)$, the total loss in equation 13 will have contributions from both the count and sentiment losses, and thus both aspects of the model are trained jointly. Using the validation set, we determined that setting $\gamma = 0.01$ achieves a good balance between both the count and sentiment parts of the model, obtaining a good MAE without neglecting modeling of the counts. Given the considerations discussed for $\gamma = 0, 1$, we only examine the Negative Binomial count model and the inclusion of the legislator bias term. When the model was trained using the text of Donald Trump’s tweets, it was trained for 4500 iterations, while it was trained for 25,500 iterations when the text was not used. Again, the model dimension was chosen to be $K = 2$. The joint model performance is shown in Table 4, with MAE, total loss, and unweighted count model negative log-likelihood shown.

Legislator Embeddings

Training our joint count-sentiment legislator tweeting behavior model yields a key byproduct: the legislator embeddings. As with previous spatial representations of legislator preferences, our model allows us to visualize the position of legislators in space. In Figure 1 we present the two dimensions of legislator embeddings from the model presented in Table 4. Examining the joint distribution of the two embedding dimensions provides a preliminary assessment of their validity.
Figure 1: Comparison of the Two Dimensions of Legislator Embeddings.

One noticeable characteristic of the embeddings is how they separate legislators across party lines. Although party affiliations were not incorporated into the model learning in any way, Figure 1 illustrates that the model generally separates Democrats and Republicans. Democrats have high values of the first dimension and low values of the second, while Republicans have low values of the first dimension and high values of the second. Thus, based on the count, timing, and affective sentiment of tweets about Trump, we clearly identify whether a legislator is a Democrat or a Republican. This offers initial reassurance of the validity of the model.

Another initial validating characteristic of the embeddings is the clustering of prominent Republican senators who have been publicly critical of Trump, including Jeff Flake, Bob Corker, and John McCain. As an additional validity check, we examine at the spatial positions of Republican senators who were categorized by the *Washington Post* as critical of the President based on analysis of responses to controversial moments in Trump’s presidency, such as Trump’s firing of FBI Director James Comey, and response to the Charlottesville protests, as
well as overall rhetoric used when discussing Trump (Lewis et al., 2017). Of these 8 additional senators, 7 are clustered closely with Flake, Corker, and McCain.

In the figure we also see evidence that embeddings are not an artifact of the number of tweets about Trump authored by the legislator, nor whether the legislator is a member of the House or Senate. Legislators with more extreme values of Twitter sentiment relative to other members of their party appear to be representative of both chambers of Congress and range from having authored less than 100 tweets about Trump to over 500.

To gain a better understanding of the substantive meaning of the embeddings we compare them to DW-NOMINATE and other measures of Trump support. One such measure is the proportion of the time that legislators voted with Trump during the period in which legislator tweets were collected. This metric was calculated using a dataset published by Fivethirtyeight and includes only legislation on which the Trump administration stated a clear position on (fiv, 2019). While this measure is limited by many of the same constraints as DW-NOMINATE, it offers the advantage of being an exogenous metric for bills linked directly to Trump.

The first panel in Figure 1 illustrates the relationship between DW-NOMINATE and the percentage of the time legislators voted against Trump. For Democrats, the relationship is positive and linear: as legislators become more liberal they vote against Trump more frequently. However, the relationship is much less clear in the case of Republican legislators. While Republican legislators are normally distributed across DW-NOMINATE, there is far less variation in the amount of time they vote against Trump. As we might expect, Republicans who are on the extreme ends of ideology are more likely to vote against the president, however less extreme Republicans rarely vote against him at all. Even Republican senators known for being critical of Trump vote against him infrequently.

What happens when we make a similar comparison, but substitute the first dimension of our legislator embeddings for the percentage of the time legislators vote against Trump? The second panel in Figure 2 illustrates the relationship between DW-NOMINATE and the legislator embeddings representing attitudes toward Trump expressed on Twitter. There are several notable differences between the first and second panel. First, we observe far more variation in Twitter-based embedding across DW-NOMINATE than we did in the frequency of voting against Trump. While very few Republican legislators voted against Trump (only
Figure 2: **Panel A** illustrates the relationship between voting against Trump and DW-NOMINATE. For Democrats the relationship is positive and linear, though for Republicans there does not appear to be a strong relationship due to the scarcity of voting against Trump. **Panel B** illustrates the relationship between legislator embeddings from our model and DW-NOMINATE, where there is greater variation in attitudes toward Trump within the Republican party.
4 legislators voted against him more than 25% of the time), there is substantial variation in legislator embeddings among Republicans.

While there is certainly a positive linear relationship between legislator ideology and Trump sentiment, we observe wide variation in Trump sentiment across DW-NOMINATE. This suggests that attitudes toward Trump are not captured perfectly by traditional measures of legislator preferences. Furthermore there is also much less variation between parties with the Twitter ratings than we do with percent of the time voting with Trump. While there is almost no overlap between Democrats and Republicans in how frequently they vote against Trump, 43.2% of Republicans had larger embedding values than the lower 10\textsuperscript{th} percentile of Democrat embedding values. Finally, we can see that the most of the well-known Republican critics of Trump have high embedding values relative to other Republicans, and even some Democrats.
Conclusion

In this paper, we argue that ideal points estimated from roll-call votes miss a critical aspect of political preferences for members of Congress: attitudes towards President Donald Trump. Whereas legislative voting might recover ideological similarities and differences with the president, it does not capture any attitudes toward the president orthogonal to policy preferences, such as criticisms of his rhetoric and tone. To address this shortcoming and obtain legislator representations of legislator’s attitudes toward Trump, we have proposed a model that assigns a vector to each legislator based on the content of their tweets. We similarly represent Donald Trump with a vector for each day he tweets, constructed using the text of his daily tweets. Legislator vectors and Trump vectors interact to produce predictions of both the sentiment of legislator tweets about Donald Trump and the number of tweets produced each day. From this model we obtain representations of legislators that capture their attitudes toward the president.

Our model’s predictive performance is robust to a variety of settings and achieves sentiment predictive performance of 0.16 mean-absolute-error and 87% accuracy, demonstrating its capability to predict legislator tweeting behavior. When visualizing the two dimensions of legislator embeddings we find that the model separates legislators across party lines (despite not being trained on the party of legislators) and groups together Republican senators who are well-known critics of Trump (despite overwhelmingly voting with him on legislation).

While our aims in this paper were to develop a method of modeling attitudes toward Trump beyond legislative policy preferences, this method can be used to test a wide range of hypotheses about modern U.S. politics. For instance, the legislator embeddings can be used to explore how legislators appeal to different audiences—to party leaders with their legislative behavior and constituents with their Tweets. This model could be used to evaluate if members of Congress are punished or rewarded in elections for criticizing (or praising) the president. Because our model does not rely on roll-call votes, it can also be used to model attitudes toward Trump among candidates running for Congress and other offices.

Though our model demonstrates the capability of representing legislators’ attitudes toward Trump and performs well with respect to predicting tweet counts and sentiment based upon Donald Trump’s tweets, our method may be improved in several ways. First, while the Trump
vectors are currently constructed from the text of his daily tweets, they could be enriched by incorporating other sources of text, such as White House press releases and speeches. While we restrict ourselves to Twitter data in this paper to maintain consistency across the sources of data for vectors representing Trump and legislators, the incorporation of auxiliary text data would likely provide more context to legislators’ tweets. Another avenue for improving the model is to allow it to capture legislators’ dynamic attitudes toward Trump over time. While legislator attitudes are currently modeled as static embeddings, allowing each legislator’s embedding to change over time would enable the exploration of temporal dynamics and hypothesis testing about when legislators are more likely to tweet negatively about Trump, what factors contribute to a legislator’s decision to tweet about Trump, and how the Trump’s tweets interact with legislator’s tweets over time.

Acknowledgments

We thank Chris McCroskey, Tweet Congress, and Todd Shields for enabling access to the population of legislator tweets.
References


Yupeng Gu, Yizhou Sun, Ning Jiang, Bingyu Wang, and Ting Chen. 2014. Topic-factorized ideal point estimation model for legislative voting network. KDD 2014.


22


