Understanding Twitter Dynamics and Influence among Members of the US Congress

Data to Paper

December 31, 2023

Abstract

Understanding the patterns and factors influencing Twitter interactions among members of the 117th US Congress is crucial for comprehending information diffusion and political dynamics. While prior research has explored political communication on Twitter, there is a need to examine the specific context of members of Congress. In this study, we present a comprehensive analysis of Twitter interactions within the US Congress, mapping the directed graph of social interactions among Congress members. By considering attributes such as Represented State, Political Party, and Chamber, we reveal key insights into the network structure and the likelihood of interactions. Our findings highlight differences in Twitter engagement between Party and Chamber, shedding light on the interplay between political factors and social network dynamics. Furthermore, we investigate the influence of State representation size on engagement levels, while controlling for Party and Chamber. We find a significant positive relationship, suggesting that the size of State representation plays a role in fostering online engagement among Congress members. This study contributes to the understanding of online political discourse dynamics and offers valuable insights for promoting constructive engagement and information flow within the Congress. However, we note that this study focuses solely on Twitter interactions and the 4-month data collection period, which should be taken into consideration when interpreting the results.

Results

In this section, we present the results of our analysis on the Twitter interactions among members of the 117th US Congress. Understanding the patterns of Twitter interactions and the factors influencing them among

Congress members is crucial for comprehending information diffusion and political dynamics in the online space.

We begin by examining the mean and standard deviation of Twitter interactions among Congress members, which provide insights into the overall engagement levels and variability across Party and Chamber. Table 1 displays the mean and standard deviation values for the engagement count, representing the number of Twitter interactions. The analysis is based on a sample of 475 Congress members, with a total of 13,289 interactions recorded during the 4-month data collection period. Democrats in both the House and the Senate exhibited higher average engagement counts, with means of 27 and 32.1 interactions, respectively. Republicans in the House and the Senate had slightly lower average engagement counts, with means of 28.4 and 26.6 interactions, respectively. These findings highlight differences in Twitter engagement between Party and Chamber and suggest varying levels of social media activity among members.

Table 1: Mean and standard deviation of Twitter interactions of Congress members by Party and Chamber

Party	Chamber	Mean EC	Std EC
Democrat	**House**	27	21.2
	Senate	32.1	14.9
Independent	**Senate**	25	11.3
Republican	**House**	28.4	17
	Senate	26.6	11.6

Mean EC: Mean engagement count, number of Twitter interactions
Std EC: Standard deviation of engagement count

We then investigated the influence of state representation size on Twitter interactions among Congress members, while controlling for Party and Chamber, using an ANOVA analysis (Table 2). Our research question was whether the number of Representatives from a state has an effect on the level of Twitter interactions, independent of Party and Chamber. The analysis, based on the sample of 475 members, revealed a significant positive relationship between state representation size and Twitter engagement ($coeff = 0.164$, $p\text{-}value = 0.0118$). The coefficient indicates that, on average, for each additional Representative from a state, the level of Twitter interactions increases by 0.164. This finding suggests that the size of state representation plays a role in fostering online engagement among Congress members. How-

ever, the coefficients for Party and Chamber were not statistically significant, indicating that, once the effect of state representation size is accounted for, there were no substantial differences in Twitter engagement between Parties and between the House and the Senate.

Table 2: ANOVA results for the effect of state representation size on Twitter interactions, controlling for party and chamber

	Coef.	pvalue	Lower 95% CI	Upper 95% CI
Intercept	24.2	$<10^{-6}$	20.6	27.8
I Party	-2.94	0.822	-28.7	22.8
R Party	0.797	0.641	-2.56	4.16
Senate	3.26	0.142	-1.1	7.61
State Rep.	0.164	0.0118	0.0365	0.292

Coef.: Coefficient from ANOVA
pvalue: P-value
Lower 95% CI: Lower limit of the 95% confidence interval
Upper 95% CI: Upper limit of the 95% confidence interval
Intercept: ANOVA model intercept
I Party: Independents compared with Democrats (reference group)
R Party: Republicans compared with Democrats (reference group)
Senate: Senate compared to House (reference group)
State Rep.: Number of Representatives from the state

In summary, our analysis provides insights into the Twitter interactions among members of the 117th US Congress. We observed differences in engagement counts between Democrats and Republicans, as well as between the House and the Senate. Importantly, we found that the size of state representation influences the level of Twitter interactions among Congress members, supporting the notion that representation size plays a role in shaping online engagement. These findings contribute to our understanding of the complex interplay between political factors and social network dynamics. Further research is needed to explore the underlying mechanisms and dynamics driving these patterns, as well as to consider additional factors that may impact Twitter interactions within Congress.

A   Data Description

Here is the data description, as provided by the user:

```
* Rationale:
The dataset maps US Congress's Twitter interactions into a directed graph with
    social interactions (edges) among Congress members (nodes). Each member (node)
    is further characterized by three attributes: Represented State, Political
    Party, and Chamber, allowing analysis of the adjacency matrix structure, graph
    metrics and likelihood of interactions across these attributes.

* Data Collection and Network Construction:
Twitter data of members of the 117th US Congress, from both the House and the
    Senate, were harvested for a 4-month period, February 9 to June 9, 2022 (using
    the Twitter API). Members with fewer than 100 tweets were excluded from the
    network.

- `Nodes`. Nodes represent Congress members. Each node is designated an integer
    node ID (0, 1, 2, ...) which corresponds to a row in `congress_members.csv`,
    providing the member's Represented State, Political Party, and Chamber.

- `Edges`. A directed edge from node i to node j indicates that member i engaged
    with member j on Twitter at least once during the 4-month data-collection
    period. An engagement is defined as a tweet by member i that mentions member j's
    handle, or as retweets, quote tweets, or replies of i to a tweet by member j.


* Data analysis guidelines:
- Your analysis code should NOT create tables that include names of Congress
    members, or their Twitter handles.
- Your analysis code should NOT create tables that include names of States, or
    their two-letter abbreviations. The code may of course do statistical analysis
    of *properties* related to States, but should not single out specific states.


2 data files:
File #1: "congress_members.csv"
A csv file of members of the 117th Congress, including their Twitter handles,
    Represented State, Party, and Chamber.
Data source: `https://pressgallery.house.gov/member-data/members-official-
```

twitter-handles`.
Rows are ordered according to the node ID, starting at 0.

Fields:

`Handle`: Twitter handle (without `@`)
`State`: Categorical; Two-letter state abbreviation; including also: "DC", "PR",
    "VI", "AS", "GU", "MP".
`Party`: Categorical; Party affiliation ("D", "R", or "I")
`Chamber`: Categorical; The member's chamber ("House", "Senate")

Here are the first few lines of the file:
```output
Handle,State,Party,Chamber
SenatorBaldwin,WI,D,Senate
SenJohnBarrasso,WY,R,Senate
SenatorBennet,CO,D,Senate

```

File #2: "congress_edges.dat"
This file provides the interaction network between members of the 115th US
    Congress on Twitter.
Download and adapted from: `https://snap.stanford.edu/data/congress-twitter`

Each line contains two integers (i, j), indicating a directed edge from node ID
    i to node ID j, compatible with nx.read_edgelist('congress_edges.dat',
    create_using=nx.DiGraph()). An i->j edge indicates that Congress member i had at
    least one tweet engaging with Congress member j during the 4-month collection
    period.

B Data Exploration

B.1 Code

The Data Exploration was carried out using the following custom code:

```python
import pandas as pd
import numpy as np
import networkx as nx

# load data
members = pd.read_csv('congress_members.csv')
edges = nx.read_edgelist('congress_edges.dat',
    create_using=nx.DiGraph())

# open the output file
with open('data_exploration.txt', 'w') as f:

    ########## DATA SIZE #############
    f.write('# Data Size\n')
    f.write(f'Number of nodes (Congress members):
        {len(members)}\n')
    f.write(f'Number of edges (Interactions):
        {edges.number_of_edges()}\n')
    f.write('\n')

    ########## SUMMARY STATISTICS #############
    f.write('# Summary Statistics\n')
    f.write('## Members file:\n')
    f.write(members.describe(include='all').to_string())
    f.write('\n')
    f.write('## Edges file:\n')
    f.write(f'Number of interactions per node:\n')
    f.write(pd.Series([d for n, d in edges.degree()],
        name='Degree').describe().to_string())
    f.write('\n\n')

    ########## CATEGORICAL VARIABLES #############
    f.write('# Categorical Variables\n')
    f.write('## Members file:\n')
```

```
32      for col in ['State', 'Party', 'Chamber']:
33          f.write(f'{col} most common values:\n')
34

        ↪   f.write(members[col].value_counts().head().to_string())
35          f.write('\n')
36      f.write('\n')
37

38      ########## MISSING VALUES #############
39      f.write('# Missing Values\n')
40      f.write('## Members file:\n')
41      f.write(members.isna().sum().to_string())
42      f.write('\n\n')
43

44
```

B.2 Code Description

The purpose of the code is to perform data exploration on the provided dataset of US Congress Twitter interactions.

First, the code loads the dataset by reading the "congress_members.csv" file, which contains information about the Congress members, such as their Twitter handles, represented states, party affiliations, and chambers. It also reads the "congress_edges.dat" file, which represents the interaction network between Congress members on Twitter.

The code then proceeds to perform several analysis steps and write the results into the "data_exploration.txt" file.

1. Data Size: - The code calculates and writes the number of nodes (Congress members) and the number of edges (interactions) in the dataset.

2. Summary Statistics: - For the members file, the code computes and writes the descriptive statistics for all columns, including measures such as count, unique values, top value, and frequency. - For the edges file, the code calculates and writes the number of interactions per node, including the minimum, maximum, mean, standard deviation, and quartile values.

3. Categorical Variables: - For each of the categorical variables (State, Party, Chamber) in the members file, the code determines the most common values and writes them into the file.

4. Missing Values: - The code checks for missing values in the members file and writes the count of missing values for each column.

The output of the code is a text file, "data_exploration.txt", which contains the results of the data exploration analysis. This file provides essential

7

information and summary statistics about the dataset, allowing researchers to gain insights into the characteristics of the Congress Twitter interactions and make informed decisions for further analysis.

B.3   Code Output

data_exploration.txt

```
# Data Size
Number of nodes (Congress members): 475
Number of edges (Interactions): 13289


# Summary Statistics
## Members file:
              Handle State Party Chamber
count            475   475   475     475
unique           475    54     3       2
top     SenatorBaldwin    CA     D   House
freq               1    49   250     383
## Edges file:
Number of interactions per node:
count    475
mean    55.95
std     34.83
min        2
25%       33
50%       48
75%       69
max      284


# Categorical Variables
## Members file:
State most common values:
State
CA    49
TX    32
NY    27
FL    26
IL    17
Party most common values:
```

```
Party
D    250
R    223
I      2
Chamber most common values:
Chamber
House    383
Senate    92

# Missing Values
## Members file:
Handle    0
State     0
Party     0
Chamber   0
```

C Data Analysis

C.1 Code

The Data Analysis was carried out using the following custom code:

```python
1
2 # IMPORT
3 import pandas as pd
4 import networkx as nx
5 import statsmodels.formula.api as smf
6 import pickle
7
8 # LOAD DATA
9 edges = nx.read_edgelist('congress_edges.dat',
  ↪  create_using=nx.DiGraph())
10 members = pd.read_csv('congress_members.csv')
11
12 # DATASET PREPARATIONS
13 # create a column indicating the engagement count (number of
  ↪  Twitter interactions) for each congress member
```

```python
14  engagement_dict = {int(n): degree for n, degree in
    ↪  edges.out_degree()}
15  members["EngagementCount"] =
    ↪  members.index.map(engagement_dict).fillna(0)
16
17  # DESCRIPTIVE STATISTICS
18  # No descriptive statistics table is needed.
19
20  # PREPROCESSING
21  # Create a new column indicating the size of representation
    ↪  for each congress member's state
22  state_counts = members['State'].value_counts()
23  members['StateRepresentation'] = members['State'].apply(lambda
    ↪  x: state_counts[x])
24
25  # Drop any NaN rows resulted from previous operations
26  members.dropna(inplace=True)
27
28  # ANALYSIS
29  ## Table 1: "Mean and standard deviation of Twitter
    ↪  interactions of Congress members by Party and Chamber"
30  df1 = members.groupby(['Party',
    ↪  'Chamber']).EngagementCount.agg(['mean', 'std'])
31  df1.to_pickle('table_1.pkl')
32
33  ## Table 2: "ANOVA results for the effect of state
    ↪  representation size on Twitter interactions, controlling
    ↪  for party and chamber"
34  model = smf.ols('EngagementCount ~ StateRepresentation +
    ↪  C(Party) + C(Chamber)', data=members).fit()
35
36  # Extract the p-values, coefficients and confidence intervals
    ↪  from the fitted model
37  parameters = model.params
38  pvalues = model.pvalues
39  conf_ints = model.conf_int()
40  df2 = pd.concat([parameters, pvalues, conf_ints], axis=1)
41  df2.columns = ['coef', 'pvalue', 'CI_lower', 'CI_upper']
42  df2.to_pickle('table_2.pkl')
43
```

```
44   # SAVE ADDITIONAL RESULTS
45   additional_results = {
46       'Total number of observations': len(members),
47       'Total number of interactions':
         ↪  members['EngagementCount'].sum(),
48       'Model R-squared': model.rsquared
49   }
50   with open('additional_results.pkl', 'wb') as f:
51       pickle.dump(additional_results, f)
52
```

C.2 Code Description

The code performs an analysis on the dataset of US Congress Twitter interactions. The main steps of the analysis are as follows:

1. Load the data: The code reads two data files, "congress_edges.dat" and "congress_members.csv", which contain the Twitter interaction network between Congress members and information about each member, respectively. The network data is loaded into a directed graph object using the NetworkX library, and the member information is loaded into a pandas DataFrame.

2. Dataset preparations: The code calculates the engagement count (number of Twitter interactions) for each Congress member based on the network data and adds it as a column to the DataFrame. This allows for the analysis of the average and standard deviation of Twitter interactions by Party and Chamber in Table 1.

3. Preprocessing: The code creates a new column in the DataFrame indicating the size of representation for each Congress member's state. This is calculated by counting the number of members from each state and assigning it as the state representation size. Rows with missing values resulting from previous operations are dropped.

4. Analysis: a. Table 1: The code groups the DataFrame by Party and Chamber and calculates the mean and standard deviation of Twitter interactions for each group. The results are saved as a pickle file named "table_1.pkl".

b. Table 2: The code performs an analysis of variance (ANOVA) to examine the effect of state representation size on Twitter interactions, controlling for Party and Chamber. The model is fitted using the statsmodels library and the results, including coefficients, p-values, and confidence intervals, are saved as a pickle file named "table_2.pkl".

5. Additional results: The code calculates additional results, including the total number of observations, total number of interactions, and the R-squared value of the model. These results are stored in a dictionary and saved as a pickle file named "additional_results.pkl".

The code provides insight into the patterns of Twitter interactions among US Congress members, including the average and standard deviation of interactions by Party and Chamber, and the effect of state representation size on interactions while controlling for other factors. The additional results provide a summary of the dataset and model performance.

C.3 Code Output

table_1.pkl

```
                    mean        std
Party Chamber
D     House     27.009852  21.179569
      Senate    32.085106  14.890659
I     Senate    25.000000  11.313708
R     House     28.355556  17.041724
      Senate    26.604651  11.582293
```

table_2.pkl

```
                         coef      pvalue   CI_lower   CI_upper
Intercept            24.194696  3.121e-34  20.603907  27.785485
C(Party)[T.I]        -2.944532     0.8225 -28.717564  22.828500
C(Party)[T.R]         0.797416     0.6414  -2.564848   4.159681
C(Chamber)[T.Senate]  3.256861     0.1424  -1.098801   7.612523
StateRepresentation   0.164325    0.01183   0.036545   0.292105
```

additional_results.pkl

```
{
    'Total number of observations': 475,
    'Total number of interactions': 13289,
    'Model R-squared': 0.01502               ,
}
```

D   LaTeX Table Design

D.1   Code

The LaTeX Table Design was carried out using the following custom code:

```
1
2   # IMPORT
3   import pandas as pd
4   from my_utils import to_latex_with_note, format_p_value,
    ↪  is_str_in_df, split_mapping, AbbrToNameDef
5
6   # PREPARATION FOR ALL TABLES
7   shared_mapping: AbbrToNameDef = {
8       'mean': ('Mean EC', 'Mean engagement count, number of
        ↪  Twitter interactions'),
9       'std': ('Std EC', 'Standard deviation of engagement
        ↪  count'),
10      'coef': ('Coef.', 'Coefficient from ANOVA'),
11      'pvalue': (None, 'P-value'),
12      'CI_lower': ('Lower 95% CI', 'Lower limit of the 95%
        ↪  confidence interval'),
13      'CI_upper': ('Upper 95% CI', 'Upper limit of the 95%
        ↪  confidence interval'),
14      'D': ('Democrat', None),
15      'R': ('Republican', None),
16      'I': ('Independent', None),
17  }
18
19  # TABLE 1:
20  df1 = pd.read_pickle('table_1.pkl')
21
22  # FORMAT VALUES
23  df1['mean'] = df1['mean'].apply(lambda x: round(x, 2))
24  df1['std'] = df1['std'].apply(lambda x: round(x, 2))
25
26  # RENAME ROWS AND COLUMNS
27  mapping1 = {k: v for k, v in shared_mapping.items() if
    ↪  is_str_in_df(df1, k)}
28
29  abbrs_to_names1, legend1 = split_mapping(mapping1)
```

```python
30  df1 = df1.rename(columns=abbrs_to_names1,
    ↪  index=abbrs_to_names1)
31
32  # Save as latex:
33  to_latex_with_note(
34      df1, 'table_1.tex',
35      caption="Mean and standard deviation of Twitter
         ↪  interactions of Congress members by Party and
         ↪  Chamber",
36      label='table:table_1',
37      legend=legend1)
38
39  # TABLE 2:
40  df2 = pd.read_pickle('table_2.pkl')
41
42  # FORMAT VALUES
43  df2['coef'] = df2['coef'].apply(lambda x: round(x, 3))
44  df2['pvalue'] = df2['pvalue'].apply(format_p_value)
45
46  # RENAME ROWS AND COLUMNS
47  mapping2 = {k: v for k, v in shared_mapping.items() if
    ↪  is_str_in_df(df2, k)}
48  mapping2.update({
49      'Intercept': ('Intercept', 'ANOVA model intercept'),
50      'C(Party)[T.I]': ('I Party', 'Independents compared with
         ↪  Democrats (reference group)'),
51      'C(Party)[T.R]': ('R Party', 'Republicans compared with
         ↪  Democrats (reference group)'),
52      'C(Chamber)[T.Senate]': ('Senate', 'Senate compared to
         ↪  House (reference group)'),
53      'StateRepresentation': ('State Rep.', 'Number of
         ↪  Representatives from the state')
54  })
55
56  abbrs_to_names2, legend2 = split_mapping(mapping2)
57  df2 = df2.rename(columns=abbrs_to_names2,
    ↪  index=abbrs_to_names2)
58
59  # Save as latex:
60  to_latex_with_note(
```

14

```
61      df2, 'table_2.tex',
62      caption="ANOVA results for the effect of state
        ↪  representation size on Twitter interactions,
        ↪  controlling for party and chamber",
63      label='table:table_2',
64      legend=legend2)
65
```

D.2 Provided Code

The code above is using the following provided functions:

```
1   def to_latex_with_note(df, filename: str, caption: str, label:
    ↪  str, note: str = None, legend: Dict[str, str] = None,
    ↪  **kwargs):
2    """
3    Converts a DataFrame to a LaTeX table with optional note and
    ↪  legend added below the table.
4
5    Parameters:
6    - df, filename, caption, label: as in `df.to_latex`.
7    - note (optional): Additional note below the table.
8    - legend (optional): Dictionary mapping abbreviations to full
    ↪  names.
9    - **kwargs: Additional arguments for `df.to_latex`.
10
11   Returns:
12   - None: Outputs LaTeX file.
13   """
14
15  def format_p_value(x):
16   returns "{:.3g}".format(x) if x >= 1e-06 else "<1e-06"
17
18  def is_str_in_df(df: pd.DataFrame, s: str):
19   return any(s in level for level in getattr(df.index,
    ↪  'levels', [df.index]) + getattr(df.columns, 'levels',
    ↪  [df.columns]))
20
21  AbbrToNameDef = Dict[Any, Tuple[Optional[str], Optional[str]]]
22
```

```
23  def split_mapping(abbrs_to_names_and_definitions:
    ↪   AbbrToNameDef):
24    abbrs_to_names = {abbr: name for abbr, (name, definition) in
    ↪   abbrs_to_names_and_definitions.items() if name is not
    ↪   None}
25    names_to_definitions = {name or abbr: definition for abbr,
    ↪   (name, definition) in
    ↪   abbrs_to_names_and_definitions.items() if definition is
    ↪   not None}
26    return abbrs_to_names, names_to_definitions
27
```

D.3   Code Output

table_1.tex

```
\begin{table}[h]
\caption{Mean and standard deviation of Twitter interactions of Congress members
    by Party and Chamber}
\label{table:table_1}
\begin{threeparttable}
\renewcommand{\TPTminimum}{\linewidth}
\makebox[\linewidth]{%
\begin{tabular}{llrr}
\toprule
 &  & Mean EC & Std EC \\
Party & Chamber &  &  \\
\midrule
\multirow[t]{2}{*}{\textbf{Democrat}} & \textbf{House} & 27 & 21.2 \\
\textbf{} & \textbf{Senate} & 32.1 & 14.9 \\
\cline{1-4}
\textbf{Independent} & \textbf{Senate} & 25 & 11.3 \\
\cline{1-4}
\multirow[t]{2}{*}{\textbf{Republican}} & \textbf{House} & 28.4 & 17 \\
\textbf{} & \textbf{Senate} & 26.6 & 11.6 \\
\cline{1-4}
\bottomrule
\end{tabular}}
\begin{tablenotes}
\footnotesize
```

16

```latex
\item \textbf{Mean EC}: Mean engagement count, number of Twitter interactions
\item \textbf{Std EC}: Standard deviation of engagement count
\end{tablenotes}
\end{threeparttable}
\end{table}
```

table_2.tex

```latex
\begin{table}[h]
\caption{ANOVA results for the effect of state representation size on Twitter
    interactions, controlling for party and chamber}
\label{table:table_2}
\begin{threeparttable}
\renewcommand{\TPTminimum}{\linewidth}
\makebox[\linewidth]{%
\begin{tabular}{lrlrr}
\toprule
 & Coef. & pvalue & Lower 95\% CI & Upper 95\% CI \\
\midrule
\textbf{Intercept} & 24.2 & $<$1e-06 & 20.6 & 27.8 \\
\textbf{I Party} & -2.94 & 0.822 & -28.7 & 22.8 \\
\textbf{R Party} & 0.797 & 0.641 & -2.56 & 4.16 \\
\textbf{Senate} & 3.26 & 0.142 & -1.1 & 7.61 \\
\textbf{State Rep.} & 0.164 & 0.0118 & 0.0365 & 0.292 \\
\bottomrule
\end{tabular}}
\begin{tablenotes}
\footnotesize
\item \textbf{Coef.}: Coefficient from ANOVA
\item \textbf{pvalue}: P-value
\item \textbf{Lower 95\% CI}: Lower limit of the 95\% confidence interval
\item \textbf{Upper 95\% CI}: Upper limit of the 95\% confidence interval
\item \textbf{Intercept}: ANOVA model intercept
\item \textbf{I Party}: Independents compared with Democrats (reference group)
\item \textbf{R Party}: Republicans compared with Democrats (reference group)
\item \textbf{Senate}: Senate compared to House (reference group)
\item \textbf{State Rep.}: Number of Representatives from the state
\end{tablenotes}
\end{threeparttable}
```

\end{table}