Social Media Analytics

FINAL PROJECT

Temporal and structural analysis of the American political subreddit

---

Authors:

Federico Bidone, 892054, f.bidone1@campus.unimib.it

Paola Maria Cavana, 859341, p.cavana1@campus.unimib.it

A.Y.: 2023/2024

Contents

1. Introduction

The project aims to examine the comments and communities of the subreddit r/politics over the period from 2007 to 2022. r/politics is the main subreddit for global political discussion, with a predominant focus on American politics. With its 8.5 million members, it ranks in the top 1% of the largest subreddits in the world.

In this analysis our objectives are:

1. Identifying predominant figures over time
2. Assessing emotional trends
3. Understanding the polarization over the years
4. Detecting the presence of distinct communities

In particular we performed a social analysis of the content, through the sentiment analysis, the emotions recognition, keywords analysis and NER analysis, and an analysis of the network, which includes metrics calculation and community detection.

2. Data collection

Due to the large amount of data collected over such an extended period, it was not feasible to rely directly on Reddit's APIs to retrieve all comments. Therefore, it was decided to use Pushshift. Pushshift is a project dedicated to archiving and analyzing data, focusing mainly on social media platforms like Reddit. Every month, Pushshift performs a complete dump of the top 20,000 subreddits. User u/Watchful1 has extracted, subdivided, and individually repackaged each subreddit, making them available on accademictorrents.com. This allows anyone to download only the posts and comments of the subreddits of interest.

The files are in NDJSON format, compressed with zstandard. NDJSON, an acronym for Newline Delimited JSON, is a convenient format for storing or transmitting structured data that can be processed one record at a time. Each line is a valid JSON value, allowing you to access individual lines and work only with those of interest.

3. Data Processing

After defining the project's objective and the resources used, the next step was the implementation of the code for data processing. This code was written in Python, a programming language widely used for data analysis.

The code begins with the import of the necessary libraries, including datetime, zstandard, os, orjson, csv, and logging.handlers. These libraries provide the necessary functionalities to read and decode files compressed with Zstandard, manipulate data, and write the results to a CSV file.

Subsequently, the code defines the variables for the name of the input file, the fields to be extracted from each JSON object, the directory of the input file, and the date range. These variables are easily customizable, allowing the code to be adapted to different input files and date ranges.

The code then defines two functions, read_and_decode and read_lines_zst, to read and decode the blocks from the file compressed with Zstandard and to read the lines from the file. These functions use a chunk-based approach to read the data, reducing memory usage and improving efficiency.

Finally, the code processes each year in the specified date range, filtering comments based on their creation date and writing the relevant data to a CSV file for each year. During this process, the code keeps track of the number of lines read, the number of bytes processed, and the number of lines that failed to be decoded. This information can be useful for monitoring the progress of data processing and identifying any problems.

4. Data Analysis

The fields extracted from each JSON object include author, body, created_utc, parent_id, id, score, author_fullname, and ups. These fields represent various dimensions of the data, including the author of the comment, the content of the comment, the creation date, the ID of the parent comment, the ID of the comment,

the score of the comment, the full name of the author, and the number of “upvotes”. These fields provide a comprehensive overview of the data and allow for in-depth analysis.

In the analysis phase, the focus shifts to processing the data for each year in the specified range. For each year, CSV files containing comments and submissions are read. These data are then filtered to remove rows where the author or body of the comment has been deleted or removed. The dates are converted to datetime format.

For comments, the fields extracted include author, body, date, parent_id, id, score, author_fullname, and ups. These fields provide a comprehensive overview of the data and allow for in-depth analysis. For submissions, the fields extracted include `author`, `created_utc`, `id`, `num_comments`, `title`, `score`, `author_fullname`, and `ups`.

Subsequently, a sentiment analysis is performed on the comments using the VADER package. VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. It uses a combination of a sentiment lexicon, which is a list of lexical features (e.g., words) which are generally labeled according to their semantic orientation as either positive or negative, and five simple heuristics, which encode how contextual elements increment, decrement, or negate the sentiment of text1. VADER not only tells about the Positivity and Negativity score but also tells us about how positive or negative a sentiment is. This tool returns a sentiment score for each comment, which is then added to the DataFrame of comments.

In addition, the full ID is split into type and ID, allowing to distinguish between comments and posts.

Finally, a DataFrame of connections is created for each year, mapping each comment to its author and the post or comment it responds to. This DataFrame of connections provides a representation of the interactions between users in the subreddit.

5. Data Storage and Graph Creation

In this phase of the project, the focus shifts to storing the processed DataFrames and creating directed graphs for each year in the specified range.

For each year, the DataFrames of comments and submissions are saved in pickle files. These files can be easily loaded at a later time for further analysis.

Subsequently, each pickle file is opened, converted into a DataFrame, and a boolean mask is applied to keep only the rows where the author of the comment is not the same as the author of the comment or post it responds to. The modified DataFrame is then saved with the same name as the original file.

The DataFrame is then grouped by author and parent_author and the number of interactions and the average sentiment for each pair is calculated. A new column interaction_sentiment_weighted is also created, which represents the average sentiment weighted for the number of interactions.

Finally, a complete directed graph is created using the NetworkX library. This graph represents all interactions between users in the subreddit for a given year, with the number of interactions and the average sentiment as attributes of each edge. The graph is then saved in a pickle file.

This process is repeated for each year in the specified range. At the end of each iteration, data that are no longer needed are deleted to save memory.

6. Social Network Analysis

Given the limited computational capacity available, analyzing the entire set of graphs spanning from 2007 to 2022 is not feasible. Therefore, we have come up with a new strategy to deal with this limitation. Our solution involves implementing a function that prioritizes the examination of the most connected components within the graphs. Subsequently, to simplify the computational process, the function selectively focuses on a subset, specifically narrowing down the analysis to a manageable scale by considering only 10,000 nodes. This strategic reduction in scope allows us to

glean meaningful insights from the data without compromising the integrity of our findings.

As part of future developments, a potential avenue could involve a more detailed examination of the entire graphs. This approach would require enhanced computational resources and efficiency, allowing for a comprehensive analysis of each graph in its entirety.

6.1. SNA Metrics

In this step for each graph obtained, we calculated the following parameters:

- Number of nodes
- Number of edges
- Assortativity
- Node connectivity
- Edge connectivity
- Graph density
- Overall reciprocity
- Number of strongly connected component
- Average degree

And since we talk about graphs, it was important to consider the following centrality metrics:

- Degree Centrality: that is defined as the number of edges incident to a node.
- Closeness Centrality: measures the proximity of a node to all other nodes in a network, not just those to which it is directly connected.
- Betweenness centrality that reports the importance of a node in communications with other nodes (i.e. how much a node is passing).

For each graph we got the top 10 of each parameter written above.

For the SNA metrics, we have generated a file named `"SNA_metrics.txt"` that encapsulates comprehensive information for each graph spanning the period from 2007 to 2022. This file serves as a consolidated repository, providing detailed metrics and analytical data relevant to social network analysis. The inclusion of this

file enhances the efficiency of data retrieval and analysis, offering a centralized resource for evaluating the SNA metrics associated with each graph in the comprehensive dataset.

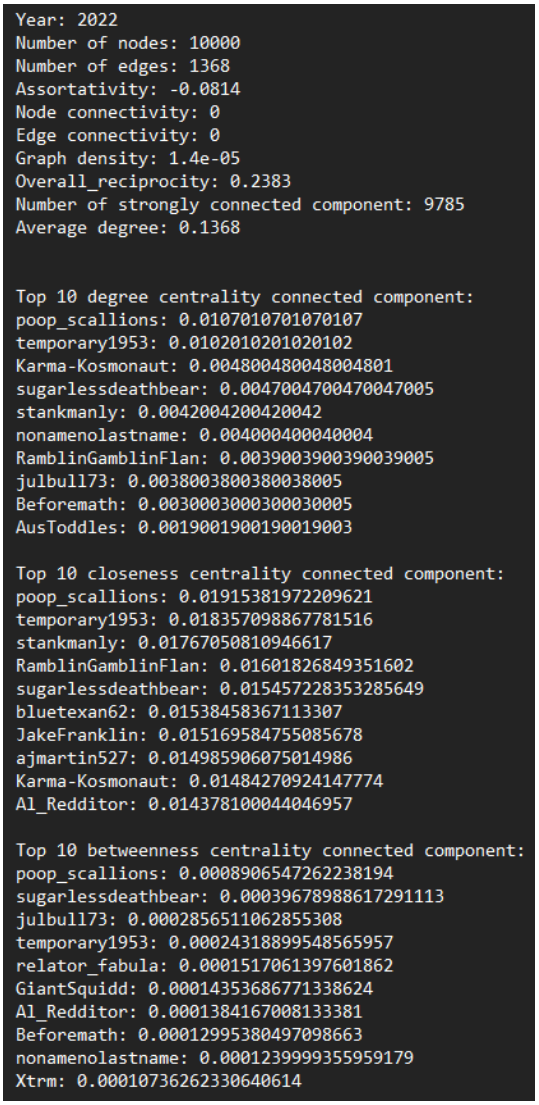
```
Year: 2007
Number of nodes: 10000
Number of edges: 105097
Assortativity: -0.1413
Node connectivity: 0
Edge connectivity: 0
Graph density: 0.001051
Overall_reciprocity: 0.33518
Number of strongly connected component: 2999
Average degree: 10.5097


Top 10 degree centrality connected component:
qgyh2: 0.22792279227922793
Flemlord: 0.15291529152915292
NoMoreNicksLeft: 0.1343134313431343
sid13: 0.11121112111211122
anonymgrl: 0.10621062106210621
maxwellhill: 0.10571057105710571
democracy101: 0.09700970097009701
aletoledo: 0.0968096809680968
innocentbystander: 0.086008600860086
ayrnieu: 0.0818081808180818

Top 10 closeness centrality connected component:
qgyh2: 0.39800665711142075
Flemlord: 0.3530403181000591
sid13: 0.3489618376370904
anonymgrl: 0.3448792620470025
maxwellhill: 0.3425136032186546
democracy101: 0.3383221687766122
abudabu: 0.3300392425429904
NoMoreNicksLeft: 0.3247820965809331
rmuser: 0.3232161062869827
Bestman0: 0.3231093990878124

Top 10 betweenness centrality connected component:
Flemlord: 0.03610628869656135
NoMoreNicksLeft: 0.03224540531377084
qgyh2: 0.02062244134203257
aletoledo: 0.01867173859738821
ayrnieu: 0.01743439756877785
innocentbystander: 0.015359356598151897
deuteros: 0.012521549211755849
rollingdivision: 0.011975438839106133
malcontent: 0.011558732654104897
shiner_man: 0.010328052315531968
```

This figure shows the analysis of the metrics of the 2007 graph in the SNA_metrics.txt file



This figure shows the analysis of the metrics of the 2022 graph in the SNA_metrics.txt file

For example, analyzing assortativity measurements for each year in our graphs from 2007 to 2022 reveals a variation in assortativity trends over the years. Negative assortativity suggests that nodes tend to connect with others having different characteristics rather than similar ones. There is a notable variation over the years, with some significant fluctuations. For instance, the year 2013 exhibits a particularly low assortativity value, indicating a pronounced tendency towards connecting nodes with different characteristics during that specific period. Additionally, 2015 shows a slightly more positive assortativity value compared to surrounding years, suggesting an increased preference for connecting similar nodes in that year. In general, the prevailing negative trend may suggest a tendency in our network to connect nodes with different characteristics over the years. However, it is important to consider

further analyses and contextualizations to fully understand the meaning of these variations.

Another important observation is that for all the graphs under consideration, both node connectivity and edge connectivity are consistently reported as 0. This implies that the removal of a single node or edge does not result in a disconnection of the network. The network remains resilient to the loss of individual nodes or edges, indicating a robust structure.

In addition, we observe a general trend of decreasing network density, and since we are analyzing the politics subreddit, this could be explained by several dynamics:

1. Fragmentation of discussions: This could mean that participants in the subreddit are focusing on specific topics or that divergent opinions tend to isolate themselves in subgroups of users.
2. Polarization: If discussions are polarizing, with groups of users primarily interacting with each other, the network could become more fragmented.
3. Moderation and subreddit rules: Changes in moderation policies or subreddit rules could influence the structure of discussions. For example, restrictions on certain types of content might lead to increased fragmentation.
4. External events and political cycles: For instance, during elections or periods of political tension, a shift in subreddit dynamics might occur.

6.2. Community Detection

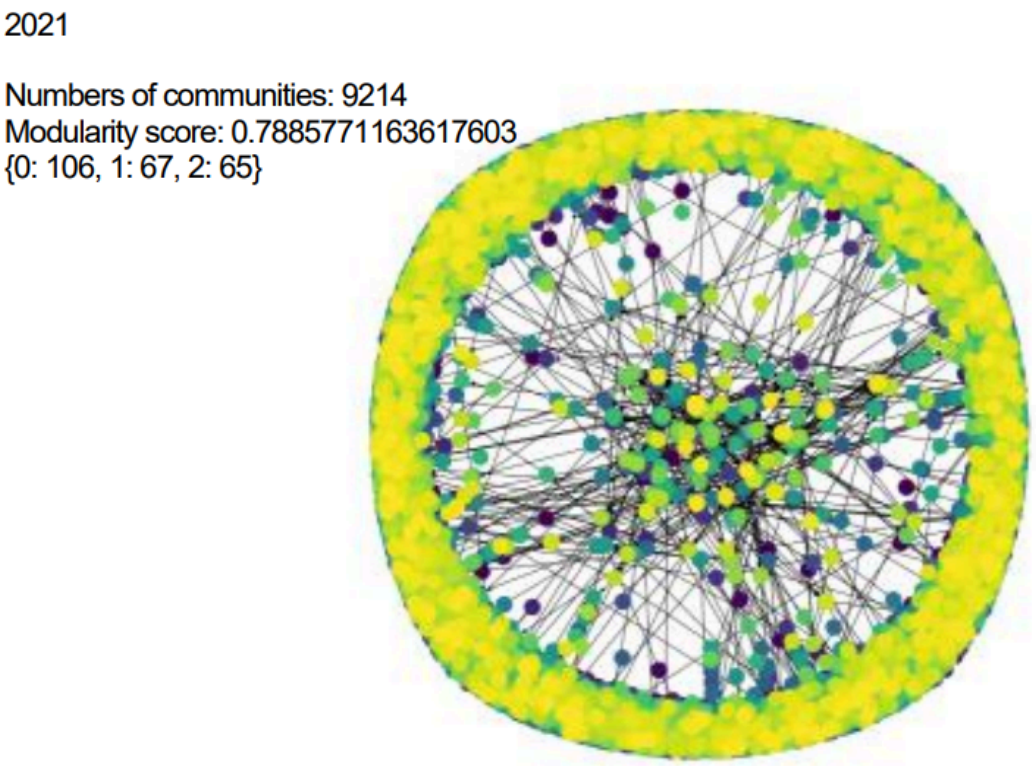
Online social platforms have made it possible for people around the world to interact and form relationships with others who have similar interests. This can be observed in real life, we tend to develop and maintain relationships with others that are similar to us. People with similar interests tend to gravitate towards each other and become associated in communities. Community detection can be used in machine learning to detect groups with similar properties and extract groups for various reasons.

Also for the community detection, we utilized the subgraph consisting of 10,000 nodes.

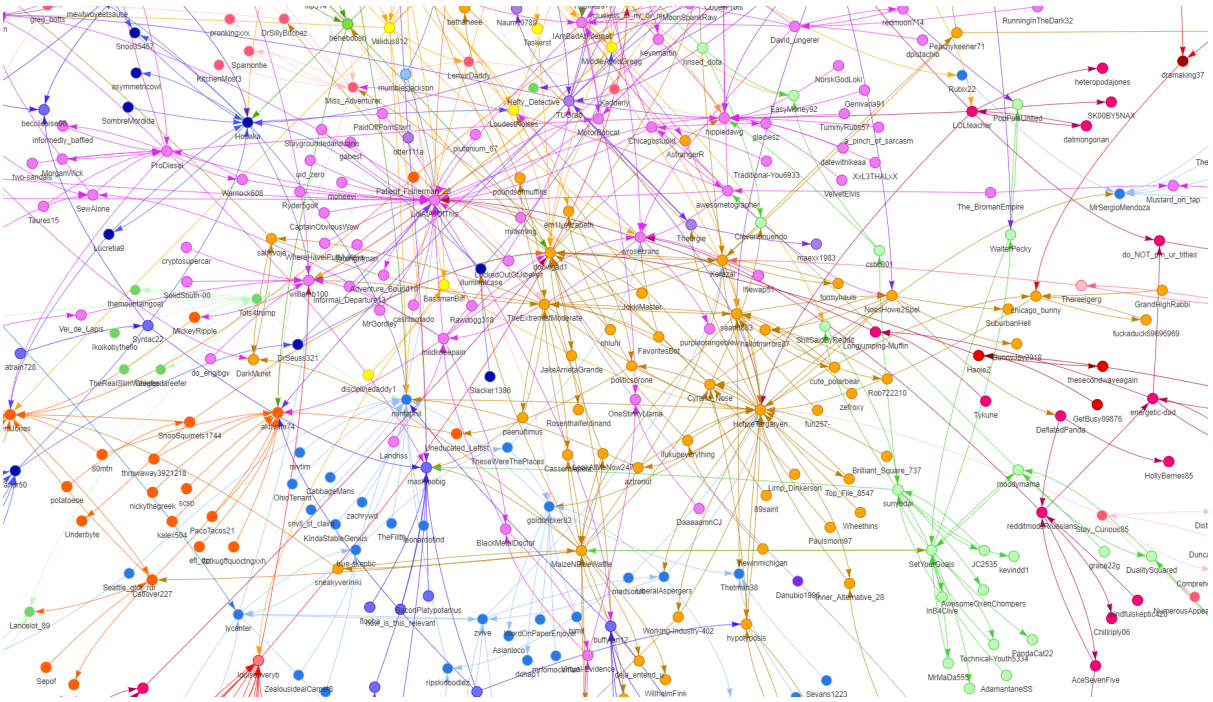
In our project the identification of communities in graphs was achieved through the use of the Clauset-Newman-Moore greedy modularity maximization algorithm.

Greedy modularity maximization begins with each node in its own community and repeatedly joins the pair of communities that lead to the largest modularity until no further increase in modularity is possible (a maximum).

So in conclusion for each graph we identified all the communities, then we made available two different visualizations through two libraries: networkx library and the pyvis library (all results can be found in the files: `community_nx` and `community_pyvis`). Then we have calculated the modularity score and extracted the three main communities.



This figure shows the communities of the 2021 graph in the community_nx.pdf file



This figure shows part of the communities of the 2021 graph in the community_pyvis.html

2007	• Numbers of communities: 3027 • Modularity score: 0.175941639308155 • {0: 2088, 1: 1976, 2: 1577}
2008	• Numbers of communities: 3047 • Modularity score: 0.26185088880544927 • {0: 2110, 1: 1891, 2: 1285}
2009	• Numbers of communities: 4592 • Modularity score: 0.39573127743664 • {0: 1034, 1: 977, 2: 704}
2010	• Numbers of communities: 6226 • Modularity score: 0.5041104116183034 • {0: 523, 1: 343, 2: 343}
2011	• Numbers of communities: 7766 • Modularity score:

	0.6574859358919425 • {0: 290, 1: 194, 2: 168}
2012	• Numbers of communities: 8563 • Modularity score: 0.6940896517153748 • {0: 164, 1: 154, 2: 135}
2013	• Numbers of communities: 8041 • Modularity score: 0.6471478831051454 • {0: 266, 1: 215, 2: 183}
2014	• Numbers of communities: 7309 • Modularity score: 0.5483770912494692 • {0: 437, 1: 232, 2: 219}
2015	• Numbers of communities: 7765 • Modularity score: 0.6035389917874171 • {0: 521, 1: 169, 2: 144}
2016	• Numbers of communities: 7992 • Modularity score: 0.528195001866968 • {0: 377, 1: 348, 2: 174}
2017	• Numbers of communities: 8203 • Modularity score: 0.5502106981403051 • {0: 243, 1: 186, 2: 180}
2018	• Numbers of communities: 8250 • Modularity score: 0.5267516836596412 • {0: 324, 1: 232, 2: 150}
2019	• Numbers of communities: 8763 • Modularity score: 0.6274548112544055

	• {0: 186, 1: 133, 2: 132}
2020	• Numbers of communities: 9135 • Modularity score: 0.7178301601210002 • {0: 148, 1: 89, 2: 82}
2021	• Numbers of communities: 9214 • Modularity score: 0.7885771163617603 • {0: 106, 1: 67, 2: 65}
2022	• Numbers of communities: 9104 • Modularity score: 0.7406963894873595 • {0: 96, 1: 86, 2: 85}

- Increasing community diversity: The number of communities consistently grows over the years, indicating a progressively diverse range of topics and discussions within the subreddit.
- Modularity score fluctuations: The modularity score variations suggest changing patterns in community structure. Higher scores imply more distinct communities.
- Community size dynamics: Shifts in the distribution of community sizes might indicate shifts in focus within the subreddit.

Let's focus specifically on some years:

- 2007:
    - Community Diversity: With 3027 communities, the subreddit in 2007 already demonstrates a considerable level of diversification in topics or interests.
    - Moderate Modularity: The modest modularity score of 0.1759 suggests a moderate community structure, indicating some level of thematic cohesion within the subreddit.
- 2011:

    - Exponential Growth: The number of communities experiences significant growth, reaching 7766. This exponential increase might indicate an expansion in the range of political discussions or an influx of varied perspectives.
    - High Modularity: The modularity score of 0.6575 reflects a stronger community structure, suggesting a more prominent separation of discussions into clearly defined groups..
- 2015:
    - Shift in Community Sizes: Notably, the sizes of communities change, with one community (Community 0) having 521 members.
    - Balanced Modularity: The modularity score of 0.6035 suggests a well-balanced structure, where discussions are diverse yet maintain a degree of cohesion.
- 2020:
    - Highest Modularity: The modularity score peaks at 0.7178 in 2020, indicating a strong community structure.
    - Reduced Community Sizes: Community sizes decrease, potentially reflecting more focused discussions or increased specialization within communities.
- 2021:
    - Peak Modularity Score: With a modularity score of 0.7886, 2021 stands out as a year with the highest level of community segregation. Examining the nature of these distinct communities may provide insights into polarized topics or events.
    - Smaller Communities: Community sizes further decrease.

7. Social Content Analysis

To thoroughly analyze the politics subreddit, we employed keyword analysis, conducted polarization analysis, emotion detection and NER analysis.

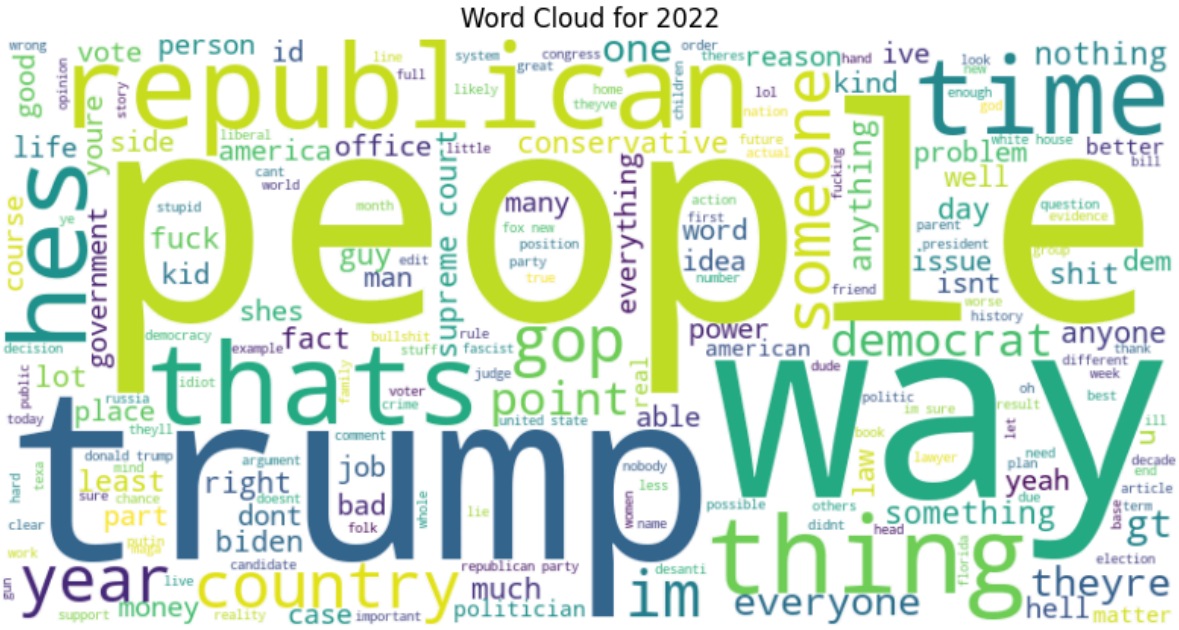
7.1. Keywords

For this part of analysis, we first created a function that:

- removes special characters,
- tokenizes the text using the `word_tokenize` function,
- performs part-of-speech tagging using the `pos_tag` function,
- extracts keywords that are nouns or adjectives,
- and finally returns the list of keywords.

Subsequently, from each dataframe containing comments (`politics_comments`) , we extracted the top 1 million rows based on the highest "score." From this subset, we removed stop words, applied the previously described function, and initially generated a word cloud. Following this, we present the twenty most frequently used words within a specific year.

In the following section, we'll show only the results for the 2007 and 2022 years, while the comprehensive set of data can be found in the attached PDF file labeled "`keywords.pdf`".



This figure shows the wordcloud of the year 2022

Top 20 Keywords for 2022:

1. people: 173463
2. trump: 123178
3. republicans: 65410
4. im: 63667
5. time: 61373

6. right: 53196
7. other: 52414
8. more: 52165
9. republican: 50578
10. way: 47843
11. state: 47029
12. years: 46234
13. thats: 43960
14. good: 42370
15. party: 41876
16. gop: 41727
17. hes: 41472
18. election: 39051
19. same: 38938
20. thing: 36011

This figure shows the wordcloud of the year 2007

Top 20 Keywords for 2007:

1. people: 42389
2. paul: 22167
3. im: 19050
4. government: 17845

5. other: 17784
6. ron: 16835
7. more: 15031
8. time: 13826
9. good: 13699
10. way: 13586
11. gt: 13304
12. us: 12984
13. war: 12462
14. right: 11894
15. thats: 10964
16. money: 10801
17. same: 10682
18. many: 10557
19. point: 9879
20. youre: 9723

In our analysis of the American political subreddit, we noted that some keywords emerge as recurring themes. In particular, words such as "Trump", "people", "way", "money", "government", "Republican", "state" and "right" are constantly present. These words reflect crucial aspects of political discourse in the community. " Trump" highlights the impact and influence of the former president in the discussions. "People" and "government" emphasize attention to civic engagement and the role of the state. " Way" and "money" may indicate debates on political strategies and financial considerations. The frequent appearance of the words "Republican" and "right"may indicate a distinct emphasis on ideological dimensions within the discussions. The term "Republican" is associated with a major political party in the United States known for its conservative stance on various issues.

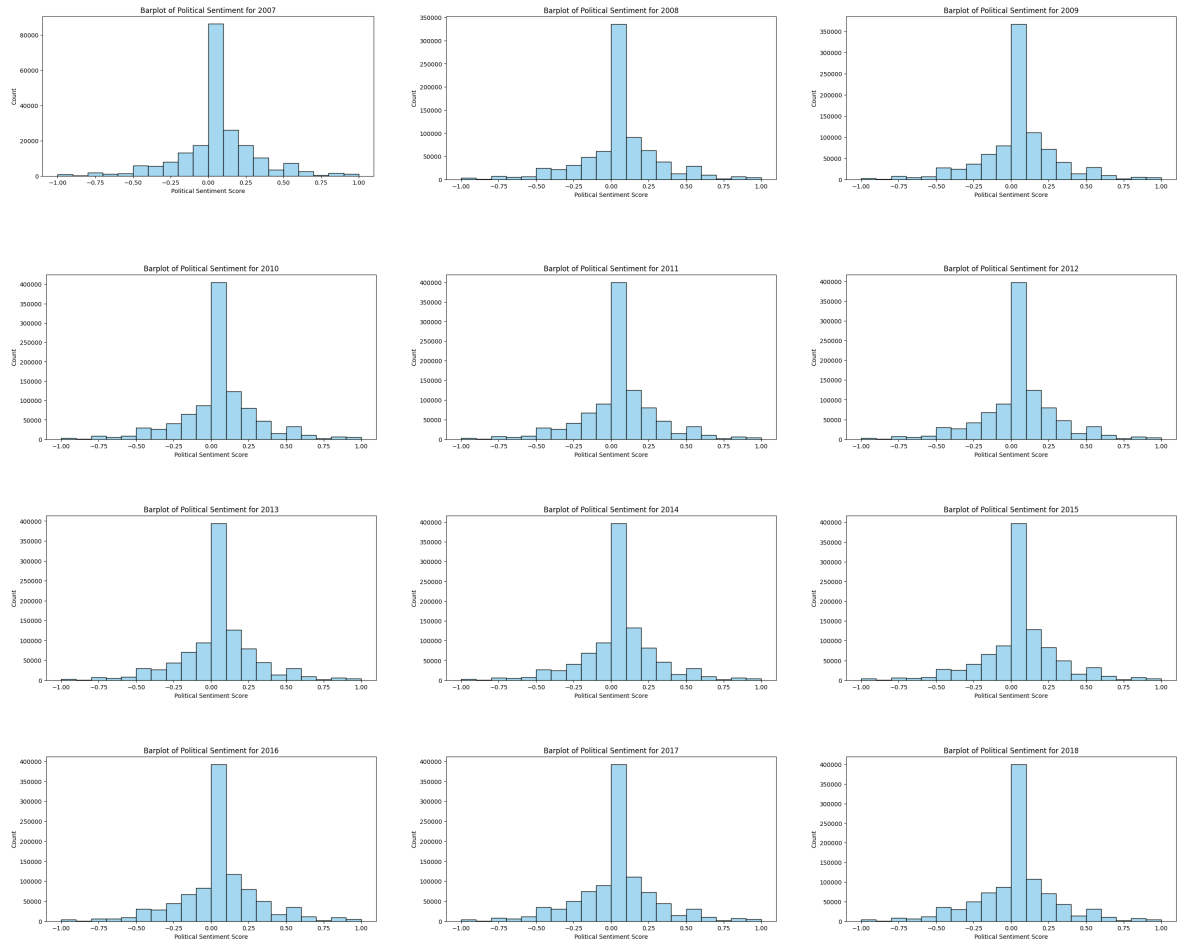
7.2. Polarization

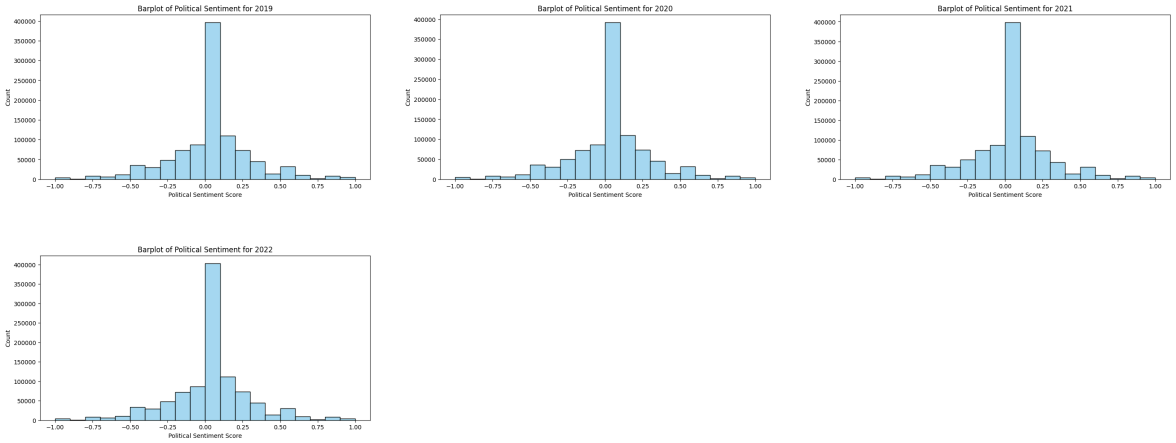
To analyze sentiment polarity, we utilized TextBlob, a Python library for natural language processing tasks. TextBlob offers a simple API for tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification,

translation, and more. Also in this case we extracted 1 million rows with the highest score.

For sentiment analysis, TextBlob employs a pre-trained machine learning model to assign polarity scores to text. The sentiment polarity ranges from -1 to 1, where -1 indicates extremely negative sentiment, 0 denotes neutrality, and 1 signifies extremely positive sentiment. Interpreting the results involves understanding that a polarity score closer to -1 suggests a predominantly negative sentiment, while a score closer to 1 indicates a predominantly positive sentiment. A score near 0 signifies neutral sentiment. Therefore, when analyzing comments in a political context, a polarity score of -1 may indicate strong disapproval or negativity towards the discussed political subject, while a score of 1 may suggest strong approval or positivity.

The distribution of the results is made clear through the implementation of a bar plot.

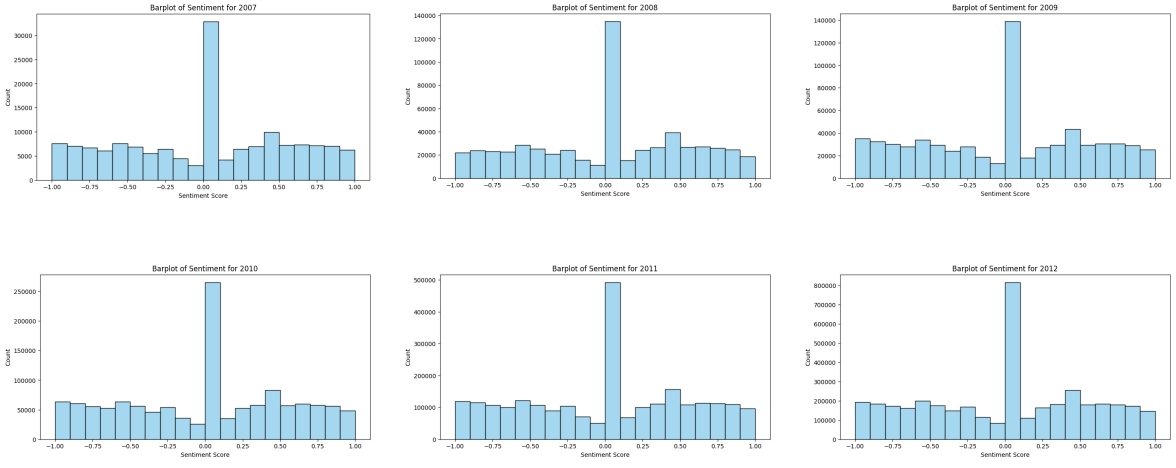


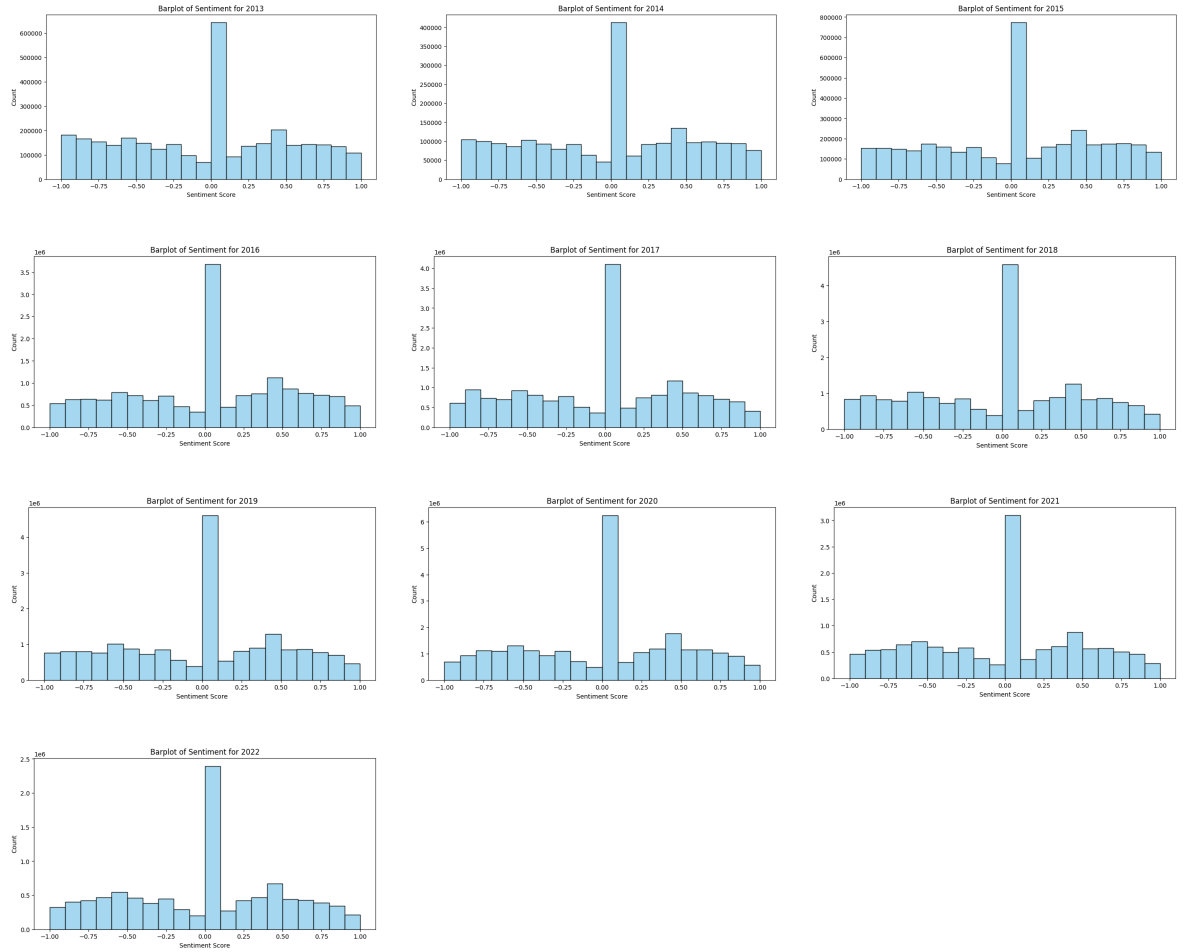


This figure shows the results of polarization with TextBlob

We have observed a consistent pattern in the bar plots generated for the polarization analysis of the political subreddit using the TextBlob library. Across the years from 2007 to 2022, we consistently find peaks between 0 and 0.10 in polarization scores. This suggests that the overall sentiment of the political discussions within the subreddit tends to be relatively neutral or mildly positive. The sentiment scores, ranging from -1 to 1, with values closer to 1 indicating positive sentiment, imply a generally balanced or slightly positive language use. It's important to note, however, that sentiment analysis tools have limitations, and they may not fully capture the complexities of political discourse, sarcasm, or context-specific sentiments.

We had concerns that TextBlob, in our case, might not be the most suitable library for polarization analysis, prompting us to conduct a verification using VADER. Below are the resulting graphs.





This figure shows the results of polarization with VADER

Even with the implementation of VADER for polarization analysis, the observed peak between 0 and 0.10 persists in the generated graphs. This consistency could be attributed to the inherent challenges of assessing sentiment within the context of political discussions on the subreddit. Political discourse often involves nuanced language, diverse perspectives, and complex emotions that may not be fully captured by polarization analysis algorithms. Moreover, the nature of online political discussions may include a mix of positive and negative sentiments, leading to an overall neutral or moderately positive sentiment score. It's essential to consider the limitations of sentiment analysis in capturing the intricacies of political conversations, where sentiment can be highly subjective and context-dependent. Also these results may be influenced by the moderation policies and guidelines of the political subreddit. Negative sentiments could potentially be limited or moderated to maintain a more civil and constructive discourse within the community. Subreddits

often enforce rules to curb inflammatory language, personal attacks, or overly negative expressions, which could impact sentiment scores.

7.3. Emotion Recognition

In addition to the sentiment analysis in the text, it is also very important the emotion recognition. In fact you can express a negative comment, but it's different if you make an angry comment or disgusted comment.

For the Emotion Recognition we use the NRCLex library. The NRC Emotion Lexicon is a list of English words and their associations with eight basic emotions:

- anger
- fear
- anticipation
- trust
- surprise
- sadness
- joy
- disgust

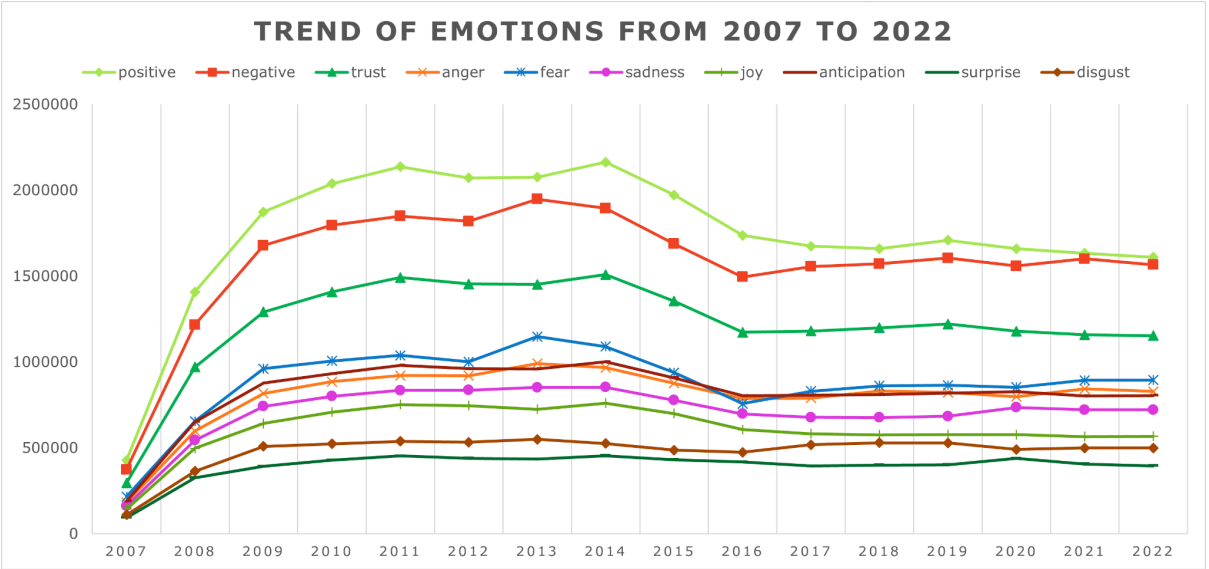
And two sentiments:

- negative and
- positive.

Also for the Emotion recognition part we extracted 1 million rows with the highest score.

For each data frame we get the frequencies of each emotion and the top emotions. Then we added two different visualization in order to better analyze the data:

- table (a column for emotions and a column for the absolute frequencies of each emotion)
- pie chart (all charts can be found in the file: `emotion_pie.pdf`)



This figure shows the trend of emotions from 2007 to 2022

	positive	negative	trust	anger	fear	sadness	joy	anticipatic	surprise	disgust
2007	428508	373586	295701	182956	215904	162321	144154	190454	93783	110127
2008	1408080	1216883	970925	598635	655175	544275	497529	651780	326280	364509
2009	1873103	1678146	1291517	817569	961094	742086	641599	877407	392286	509158
2010	2038706	1795884	1408297	885112	1006098	800373	708712	932166	428572	523885
2011	2137295	1849153	1492034	921771	1038320	835277	752560	980659	453712	538084
2012	2071475	1819634	1454305	920550	1000467	835810	746099	960935	439340	533031
2013	2075656	1947793	1452128	991526	1148328	852259	724936	960419	434591	549978
2014	2162693	1894606	1509030	967750	1090434	853113	761062	1001381	454670	526212
2015	1972225	1688337	1354111	875840	937478	777855	699895	907235	431380	487083
2016	1737143	1494955	1173065	786873	757506	697737	607036	803496	417744	474783
2017	1673992	1555542	1180206	790115	830085	677698	582635	806254	394887	518556
2018	1659317	1571427	1198792	830822	861289	676766	576366	811437	400175	529777
2019	1709184	1605596	1221427	822419	864102	684142	576622	819786	401616	528714
2020	1659741	1559054	1179275	798376	853394	735208	576664	826549	439129	490905
2021	1633624	1600366	1158123	843511	894019	722171	565312	802659	405772	499692
2022	1610138	1566469	1152905	828879	894903	721766	567329	804301	395935	499964

This figure shows for each year from 2007 to 2022 the absolute frequencies of the 8 emotions and two feelings

In our emotional sentiment analysis, a noteworthy trend has emerged consistently across all years: the prevalence of a positive sentiment. The detection of positive emotions consistently yields the highest percentage, underscoring an overall optimistic tone within the discussions. This could indicate that despite the often contentious nature of political discourse, users tend to express positivity or favorable sentiments when engaging with political topics. Additionally, the emotions most frequently detected are "trust" and "fear." The recurrent presence of "trust" may signify a foundation of belief and reliance within the community, reflecting an element of trust in the political discussions taking place. Conversely, the recurring detection of "fear" could point to the apprehensions and concerns inherent in political conversations, possibly stemming from uncertainties or potential

consequences associated with political events and decisions. Together, these findings provide valuable insights into the emotional landscape of the political subreddit, highlighting a balance of positive sentiments with underlying themes of trust and apprehension.

7.4. NER

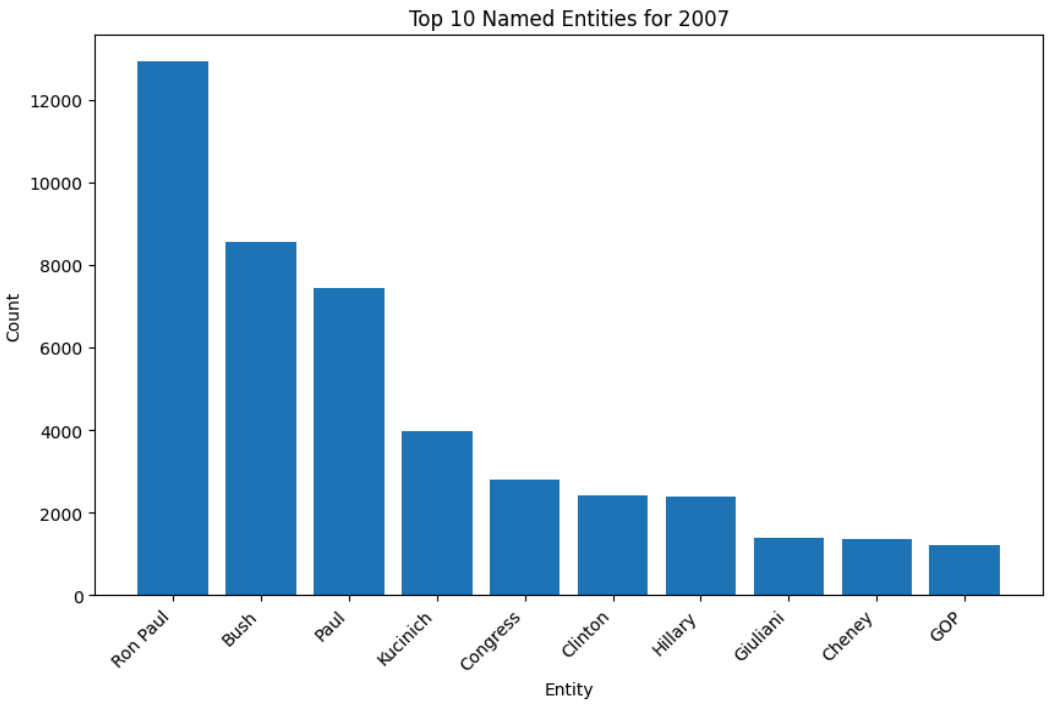
To enhance our analysis, we performed Named Entity Recognition (NER) on our dataset (a subset of 1 million rows with the highest score) to identify and extract proper nouns and organizational names. This process involved systematically identifying and categorizing entities within the text data. By conducting NER, we aimed to gain insights into the prevalence and distribution of specific names and organizational entities present in our dataframe.

In the context of our analysis, the NER phase proved to be pivotal, particularly considering the nature of the dataset related to the political subreddit. Recognizing and categorizing proper nouns and organizational names within this domain is of critical importance. This step allows us to shed light on the specific individuals, entities, and organizations that are frequently mentioned or play a notable role within political discussions on the subreddit.

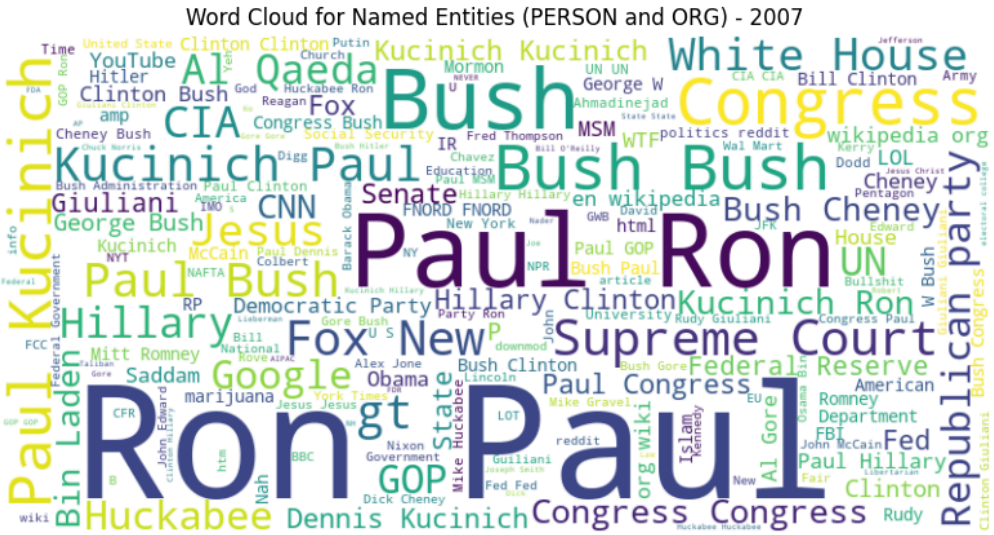
The Named Entity Recognition (NER) process was conducted using the Python library spaCy. To enhance the interpretability of the NER results, we implemented two distinct visualizations: a bar plot and a word cloud.

The bar plot provides a structured representation of the frequency and distribution of identified named entities. In addition, the word cloud visualization offers a more visually impactful representation of the identified entities.

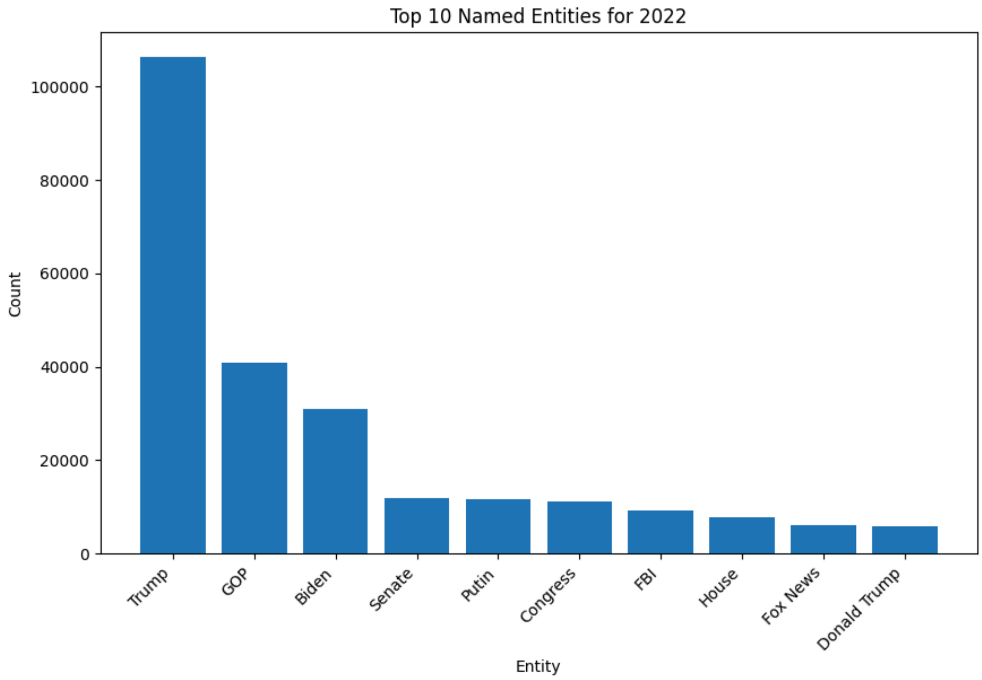
These dual visualizations aim to provide a comprehensive perspective on the identified named entities, combining both quantitative and visual insights. Together, they contribute to a more thorough exploration of the specific individuals and organizations that play a significant role in the political discussions within the analyzed subreddit.



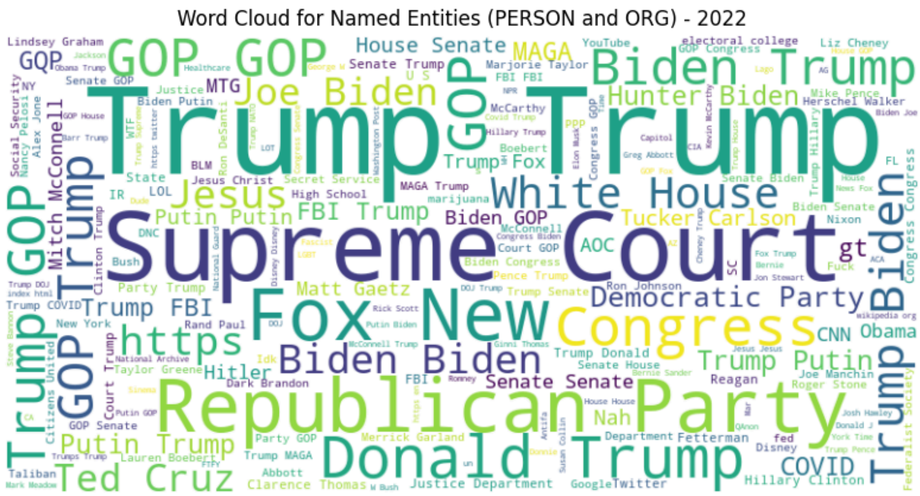
This figure shows the top 10 entities in 2007



This figure shows the word cloud of entities in 2007



This figure shows the top 10 entities in 2022



This figure shows the word cloud of entities in 2022

As the results are many to show, below we decided to show only the first two entities for each year, in any case all the results can be found in the `entities.pdf` file.

2007	Ron Paul	Bush
2008	McCain	Obama
2009	Bush	Obama
2010	Bush	Obama

2011	Ron Paul	Obama
2012	Romney	Obama
2013	GOP	Obama
2014	GOP	Obama
2015	Trump	Sanders
2016	Trump	Clinton
2017	Trump	GOP
2018	Trump	GOP
2019	Trump	GOP
2020	Trump	Biden
2021	Trump	GOP
2022	Trump	GOP

In the analysis of Named Entity Recognition (NER), the most frequently occurring proper names/entities from 2007 to 2022 are consistently identified as Trump, Obama, and GOP. This prominence reflects the enduring influence and significant presence of these entities in American political discourse during the specified period. The frequent appearance of "Trump" and "Obama" is indicative of their respective presidencies. The inclusion of "GOP" (Grand Old Party), referring to the Republican Party, underscores the enduring relevance of political parties in shaping discussions within the subreddit. The recurrence of these entities suggests a sustained focus on key political figures and parties over the years, highlighting their enduring impact on the political landscape and the ongoing relevance of their associated policies and ideologies within the subreddit's discussions.

8. Conclusion

In conclusion, this project aimed to delve into the comments and communities within the r/politics subreddit spanning from 2007 to 2022. Throughout our analysis, we pursued several key objectives. We identified predominant figures over

the years, assessed emotional trends within discussions, delved into the evolving polarization of sentiments, and detected the presence of distinct communities within the subreddit. Our methodology included a comprehensive social analysis, incorporating sentiment analysis, emotion recognition, keyword analysis, and Named Entity Recognition (NER), alongside a network analysis that involved metrics calculation and community detection.

The insights gained from this multifaceted approach provide a nuanced understanding of the dynamics and nuances within the r/politics subreddit.