EARLY PREDICTION OF PUBLIC OPINION TRENDS IN THE 2024 U.S. PRESIDENTIAL ELECTION USING TOPIC MODELING, DENDROGRAM CLUSTERING, AND SENTIMENT ANALYSIS

by Seba Susan

Submission date: 27-May-2026 08:26AM (UTC+0530)
Submission ID: 2970281116
File name: ShreyaSrivastava24ITY20_MtechThesisForPlag.docx (926.91K)
Word count: 13547
Character count: 70026

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

For as long as elections have existed, people have tried to predict their outcomes. Pollsters call thousands of strangers. Pundits analyze debate performances. Strategists track rally crowds and yard signs. But all these methods share a common limitation: they are slow, expensive, and often miss the voices of ordinary people who do not answer unknown numbers or attend political events.

Then came social media. Platforms such as Twitter, now known as X, were a game changer. Thousands of individuals were commenting on political discourse on a daily basis. No carefully selected survey answers, no practiced debate responses, but an honest, unfiltered and impromptu response to debates, news stories, campaign events and anything else that crossed the eyes of public figures and politicians. A man on the bus, viewing the debate on his phone can instantaneously send a tweet out to the entire world. This tweet may be read by someone just about anywhere, including us the researcher.

Social media data is useful not only because it is massive, although this is definitely the case. What makes it so invaluable is that it is genuine. While a subject being questioned by a pollster may state what they believe the correct response is, someone posting to Twitter, or a similar platform, are much more likely to say what they actually mean. In this situation, the public use their own words, they state what interests them or infuriates them. This is what researchers seek when searching for authentic opinion data.

There is no denying that social media data is chaotic. Misspellings, slang, sarcastic remarks, inside jokes and acronyms are abundant. The very nature of tweets, with a 280-character limit, also mean that context may be missing. Nonetheless, the site of the US election debate, with all the noise accompanying it, and perhaps because of it, is Twitter. By 2024 it had become an indispensable fixture in the simultaneous production and contestation of narratives by citizens.

This data can be collected that is the relatively easy part. What is difficult is what to do with it. On a busy day during an active election campaign, an enormous number of posts can be published far too many for even the most dedicated research team to manually

work through. To tackle this difficulty, a number of methods have been used in large text analysis, first of which being topic modeling: an approach that seeks to understand what subjects individuals are discussing. The second is sentiment analysis, which helps answer the question: how do people feel about what they are talking about?

When analyzed together side-by-side but without making one subordinate to the other, both provide complementary pictures of discourse. One identifies dominant topics. The other identifies sentiments. Neither, however, provides the complete story in isolation. The dataset of tweets that this thesis analyzed is composed of tweets published within the three months prior to the 2024 U.S. Presidential Election-collected in May, June, and July of 2024 [10]. There are several caveats to this study's claims about elections. It cannot predict with definitive accuracy the winner of an election. Elections depend on the million of votes cast by individuals in voting booths distant from Twitter. However, it can reasonably ask a less ambitious, though still meaningful, question: Do patterns of discourse on Twitter provide a glimpse into voter perception of candidate visibility and appeal? If topic modeling and sentiment analysis both simultaneously lead to the same candidate, what conclusion can be drawn?

1.2 LITERATURE REVIEW

I had a literature review of other researchers' work before my own. Here I have summarized what I found, divided by subject for clarity.

1.2.1 Using social media for election prediction

Twitter has been exploited as an object of electoral study for over 15 years. Tumasjan et al. [1] analyzed tweets relating to the German federal election and found that Twitter is generally an effective reflection of general opinions in a similar manner as opinion polls, although they acknowledge in later work that the strength of this claim has been tested and questioned, it is undeniable that opened the door to Twitter based electoral analysis. In similar temporal proximity, DiGrazia et al. [2] studied the relationship between election result outcomes and Twitter mentions in United States and found that there was a positive relationship between Twitter mentions and candidate votes. While there is clearly a link between mentions and results, O'Connor et al. [3] compared Twitter sentiment to opinion poll and found ambiguous results and hypothesized that: the structure and context in which twitter and opinion polls are representatives of underlying social phenomena are not equivalent, Looking at election campaigning in the 2011 German federal election,

Jungherr et al. [4] asserted that Twitter users were not representative of the overall population, this is an important limitation. Barbera et al. [5] introduced methods to measure political ideology from Twitter data. His work was instrumental in helping

researchers understand the political inclination of social media users. Wisnu et al. [37] authors studied the effect of social media on the creation of public opinion in the Java election of 2018. They used machine learning algorithms such as Naive Bayes and Support Vector Machine (SVM) to classify the sentiment into positive or negative. The study also applied topic modeling through LDA to identify dominant topics from positive and negative sentiments for each political party. Ilyas et al. [38] considered the tweets with the term "Brexit" and adopted LDA to find the most common themes of conversation for each day. The purpose was to see if the trends of subjects that the model found reflected real world events in the days running up to Brexit.

1.2.2 U.S. Election Studies

The US elections of 2012, 2016, 2020 and 2024 have been studied extensively. DiGrazia et al. [2] found that mentions on Twitter predicted vote shares in the 2012 election. Metaxas et al. [6] studied the 2012 election and observed that sentiment analysis was behind polling they argued that Twitter users are not a random sample of voters. Bovet et al. [7] analyzed twitter sentiment during the 2016 election and found it to be biased towards Trump. Wang et al. [8] used deep learning to analyze the tweets from the 2016 election and their model was able to predict the outcome accurately. Anwar et al. [11] applied topic modeling on twitter user profile description who had mentioned QAnon in their tweets in 2020 US Presidential Election moreover they performed frequency analysis which shows most frequently used keywords in their profile description the aim of this is to find political connection of these users. Samsir et al. [9] performed a machine learning analysis of tweets on the 2024 election, they found Naïve Bayes to work well on short-text data. For my own analysis, I used the large corpus of 2024 election tweets provided by Balasubramanian et al.(2024) [10]. Wei et al. [36] did something a bit different, they compared facial expressions in videos and sentiment in text of tweets, they found that the visual cues of emotion gave them information that was not captured by the text alone.

1.2.3 Lexicon based Sentiment analysis approaches

Social media sentiment analysis, unlike traditional polling methodologies, captures spontaneous, large-scale, and diverse expressions from the voter [39]. Recently, scholars have increasingly used sentiment analysis methods to explore social media data in the context of elections [40-42]. The techniques applied range from simple keyword matching and lexicon-based analysis to complex natural language processing (NLP) approaches such as deep learning [41, 42]. Tools based on lexicons such as VADER, AFINN, TextBlob, and SentiWordNet have been widely used to determine the polarity (positive, negative or neutral) of user-generated content.

The Lexicon based sentiment analysis approach is not as complicated as machine learning. It doesn't require training data. VADER was built specifically for social media text [12]. It understands things such as all-caps and exclamation marks (Elbagir and Yang, 2020). VADER was first introduced by Hutto and Gilbert (2014) [13]. They showed that it performs as well as human raters on social media text. AFINN gives a score to each word on the scale between minus five and plus five. AFINN was developed by Nielsen (2011) [14] by manually rating words for the sentiment intensity. Tayal et al. [16] demonstrated the use of AFINN for vaccine-related tweet analysis and also proposed a fuzzy aggregation approach combining AFINN, VADER and TextBlob scores. Loria (2018) described the capabilities of TextBlob [17] for sentiment analysis. Baccianella et al. (2010) introduced SentiWordNet [18], SentiWordNet is different. It gives each word three scores – positive, negative, and objective. Using this lexicon based approaches: VADER, AFINN, TextBlob, SentiWordNet prior studies have examined political discourse, analyzed support for candidates, and even attempted to forecast election outcomes [18, 19].

1.2.4 Latent Dirichlet Allocation (LDA) for Topic Modeling

Topic modeling is used to analyze social media election discussions before the election to identify key issues voters are talking about, track how public concerns evolve over time, and understand emerging trends or shifts in sentiment [21, 22]. It allows to find dominant themes without reading manually, providing useful insights to researchers, media and political campaigns [23]. Twitter is a popular social media platform where users interact on a daily basis, making it a valuable source for topic modeling to find emerging discussions and public concerns [15, 25]. Topic modeling aims to extract the latent topics from large collections of textual documents. Latent Dirichlet Allocation (LDA) is one of the most effective techniques [26, 27]. Moodley and Marivate (2019) ran LDA on the South African election [24] news articles, and found themes of corruption and economic issues.

1.2.5 Coherence, Prevalence and Hierarchical Clustering

LDA does not always find meaningful topics. Automatic evaluation of topic coherence was introduced by Newman et al.(2010) [29]. Mimno et al. (2011) proposed a coherence measure based on word co-occurrence [28]. Syed and Spruit (2017) compared several coherence measures [30] and found that the normalized pointwise mutual information is the best. Lau et al. (2014) studied the interpretation of coherence scores, they gave practical recommendations for researchers [31]. Kontoghiorghes and Colubi (2023) have put forward new methods on the front of prevalence metrics, their work helps researchers to understand which topics dominate a corpus [32]. Hierarchical agglomerative clustering has been used to visualize topic associations; this clustering produces a dendrogram that

visually represents the similarity between topics and helps to merge closely related ones [35]. Kaufman and Rousseeuw (2009) wrote the classic book on clustering methods [33]. Murtagh and Legendre (2014) reviewed agglomerative clustering algorithms [34].

1.2.6 Gaps in the Literature

After reading all these papers, I noticed several gaps. My thesis tries to fill these gaps.

- Most researchers employ one or the other method, but not both simultaneously. If both are used on the same set, the two techniques are usually applied in series and thus there is no clear way of confirming that both techniques lead us to the same conclusion independently.
- Most sentiment analysis research used a single lexicon; however, each lexicon had its own particular biases. For VADER it has been established by Hutto and Gilbert [13]. for AFINN by Nielsen [14] and for SentiWordNet by Baccianella et al. [18]. This gives us an incomplete picture of the opinion of the public.
- Most research on the 2024 US election has focused on the final weeks before voting day. The earlier period – May to July 2024 – has not been studied much, Balasubramanian et al. [10] reported this, Wei et al. [36] also noted this.
- My research could not identify enough studies which applied both LDA topic modelling and multiple-lexicon sentiment analysis only for the 2024 US Presidential Election and only during the specified three month pre-election time period. I applied four sentiment lexicons independently and also applied LDA topic modeling independently, then I compared what each method revealed about the same two candidates, I did not make one method dependent on the other [43.44].

This work builds on all of this. I am not inventing new methods I am applying existing methods in a combination that I have not seen elsewhere. We used the same dataset and an early time window. I studied a recent high-stakes election and we kept the two methods independent so I could compare what they revealed.

1.3 Identification of Problem and Issues

Before I explain what, I did in this thesis, I need to be clear about what motivated me. I spent a lot of time reading papers in two different areas. The first area is sentiment analysis of political tweets. The second area is topic modeling of political tweets. Each area has its own gaps. There is also a gap that applies to both areas. I will discuss them separately.

1.3.1 Issues in Existing Sentiment Analysis Research

This is the area where I used four lexicons to compare sentiment toward Trump and Biden.

So many papers use just one sentiment lexicon. They pick VADER. Or they pick TextBlob. Or, choose another one. And report the result, as if only that lexicon was correct.

Each lexicon has its own character, as VADER was designed for social media, so it knows slang, capitalization. AFINN is too simple and has a number for each word. SentiWordNet is more granular, but also more complicated. As the lexicons may disagree with each other, a tweet labelled as positive by VADER may be labelled neutral by TextBlob, etc. If I only used one lexicon, there is no way to know if the results aren't biased to this specific lexicon. Indeed, that is a true literature gap.

1.3.2 Issues in Existing Topic Modeling Research

Coming to the topic modeling part. This is the place where I use the LDA to extract topics from the discussions.

Meaningless Topics from LDA

The LDA is a very strong tool. However, it may sometimes perform bad. So the generated topics from LDA are some kinds of random words group, the meanings of them are meaningless. A researcher can even believe that they have found something new and significant, but in the fact, it is a noise. Many researches are aware this problem in literature. But they still present the topics derived from LDA without inspecting them. I decided to overcome this problem and used topic coherence scores to check whether a topic makes sense or not.

Lack of Clustering Validation for Topic Distinctiveness

When the researchers checked for coherence, they usually did not do much beyond that. They calculated the coherence number and did no further work. Coherence, though, does not provide any relationship between the topics. I could have two very similar and also completely coherent topics. Alternatively, one topic could be entirely different from all others. I needed to find out which topics were different from others. Dendrograms and hierarchical clustering is used to identify how separate a topic is from all others visually. It appeared there was little of that with election tweets.

1.3.3 The Early Pre-Election Window Has Been Largely Ignored

This issue applies to both sentiment analysis and topic modeling. Most research on election-related tweets focuses on the final weeks before voting day. Some papers even analyze tweets from after the election. That is useful for understanding what happened. But it does not help with early indicators.

If I are a political strategist, I do not want to know in November who is going to win. I want to know in July or August if candidate is gaining momentum or losing it. I want early signals so I can adjust campaign strategy. The three months from May to July 2024 is exactly that kind of early window. But there weren't many studies that looked at this particular time period. Balasubramanian et al. [10] collected the election tweets from this window. But data collection is not data analysis. So whether I am talking about sentiment analysis or topic modeling, the early window of May to July 2024 has been largely ignored. This is a clear gap in the literature.

1.3.4 How This Thesis Addresses These Gaps

For the gap of using a single lexicon, I used four different sentiment lexicons. VADER, AFINN, TextBlob, and SentiWordNet. I normalized their scores so they could be compared fairly. If all four said the same thing, I could be confident in the result.

Coming to meaningless LDA topics, I used coherence scores and calculated them for every topic. Only topics with good coherence were considered meaningful.

In order to validate topic distinctiveness, I used hierarchical clustering and generated dendrograms for the topics and for the top words in each topic.

I have now had the opportunity to examine which topics are genuinely distinct. In order to examine a case where the "early window" was deliberately not ignored I select tweets in May-July 2024. This was three months prior to November and is earlier than a lot of

existing research and therefore has the potential to capture early "signals" of sentiment or discussion trends which is precisely what I wanted to test, rather than developing new techniques or "reinventing the wheel".

CHAPTER 2

Problem Formulation, Methodology and Solution Approach

2.1 Statement of the Problem

A research thesis necessitates a well-formed research question. Here is my research question; it's quite straightforward. Social media platforms like Twitter (now X) are host to the thoughts and feelings of people with respect to a particular candidate. Many thousands if not millions of these statements are submitted daily on an election cycle. In theory, it seems logical that these vast datasets will enable researchers to grasp the views of the populace. The practical truth is that these very unstructured messy data are more difficult to decipher than expected.

Two popular methods have emerged from research aiming to analyze such data. The first of these methods is topic modeling, which identifies what people are discussing and the second method of analysis is sentiment analysis, which identifies the emotions of the posters with regard to the subjects being discussed. While both of these methods have been utilized independently in numerous analyses of election campaigns, this is why neither of these methods alone tells the full story: and where the problem is in using both together, there tends to be some dependency between them. They find topics first, then analyze sentiment on those topics. That is fine for some questions. But it does not tell me whether the two methods, when kept completely separate, point to the same conclusions.

There is another problem. Most sentiment analysis studies rely on just one lexicon. They pick VADER or TextBlob or something else. But different lexicons have different biases. A result from one lexicon might not hold up when tested with another. How can I be sure my finding is real and not just a quirk of that particular lexicon?

There is also a problem with timing. Most research on election tweets focuses on the final weeks before voting day. Some papers even analyze tweets from after the election. That tells what happened. But it does not tell what was happening early. The early window of May to July 2024, three full months before the election, has been largely ignored. If social media provides some early warnings regarding candidate momentum, then this is the time where these warnings have an impact. Finally the problem of topic validation has to be faced. LDA topic modeling is quite popular. However it produces many word collections that are like random word lists. Many of them report the topics but do not validate. Even

when they check coherence, they rarely go further. They do not ask whether the topics are truly distinct from each other or overlapping So the problem is-

The problem this thesis addresses is that existing research on election-related tweets has not systematically applied independent topic modeling and multi-lexicon sentiment analysis to the early pre-election window of the 2024 US Presidential Election, nor has it validated topic coherence and distinctiveness using clustering methods.

I do not know if topic modeling and sentiment analysis, when kept independent, point toward the same candidate and if using multiple lexicons changes the sentiment result. moreover what topics dominated the early discussion period of the 2024 election and if those topics are actually coherent and distinct from each other. This thesis tries to answer these questions.

2.2 Formulation of the Problem

I have now stated the problem, and will formalize it more by breaking it down into answerable questions. I also need to define what I am measuring and what I am comparing. I split the problem into two parts. The first part deals with topic modeling. The second part deals with sentiment analysis. Both parts are independent. Neither depends on the other.

2.2.1 Topic Modeling Formulation

The topic modeling part of my thesis tries to answer three specific questions.

Identifying the Main Discussion Themes

What are the major discussion themes in the tweets from May to July 2024 on the US Presidential Election? I wanted to find this out without guessing ahead of time. So I used LDA. The model takes all the tweets as input. It outputs a set of topics. Each topic is a list of words that tend to occur together . I asked the model to produce three topics . I picked this number after trying a few options and seeing which gave the most sensible results.

Measuring Semantic Meaningfulness

Is there actual semantic meaning behind these words or is it just gibberish. LDA does not guarantee you the topics are meaningful. Therefore, there needs to be a measure of quality. The measure is coherence. Coherence is a measurement that explains how often words in a topic have appeared next to one another within tweets. The higher the coherence, the

more likely the words are truly a topic and less likely they are a result of noise. I set a minimum.

Evaluating Topic Distinctiveness

How much separate are these two topics? Maybe the topics are very similar, while at the same time both are well defined. So I wanted to know how separate are they actually? And I performed hierarchical clustering. For this task, I have taken top words of each topic, created vectors and calculated distance between these vectors using cosine distance. The result is illustrated by dendrograms where topics which belong together form branches.

2.2.2 Sentiment Analysis Formulation

The section about sentiment analysis of my thesis tries to answer other kind of questions.

Comparing Sentiment Between Candidates

The question of if Donald Trump and Joe Biden are sentiment-wise different and in which direction they are different. To answer this question I divided the tweets into two groups. One group contained tweets referring to Trump and the other contained tweets referring to Biden. I used lists of keywords for this division. After that I let each tweet through 4 different lexicons of sentiment. Every lexicon returned a score to me. Positive scores referred to a positive sentiment, negative to a negative one.

Cross-Lexicon Agreement

How much is each lexion in agreement? I needed to verify that there was consistency. What if VADER thought Trump tweets were positive and Text Blob thought the tweets were negative? Then there would be an issue, and it would be unclear which tool to trust. For that reason I compared the results from all four of the lexions to one another and I wanted to feel confident that, if all four of them agreed, then I had a good estimate of the positivity or negativity of the tweets; otherwise the conflict itself should be noted.

Score Normalization Across Different Scales

The biggest technical difficulty: How do I compare scores from lexicons that are on different scales? VADER and TextBlob output scores from -1 to 1, AFINN scores from -5 to 5 and SentiWordNet scores from 0 to 1. A score of 0.5 from VADER can't really be compared with a score of -0.3 from TextBlob, so I normalized all scores. I applied a formula to transform all scores onto the 0 to 1 scale so that a score of 0.5 in VADER meant the same as a score of 0.5 in TextBlob. This enabled comparison of lexicons, as well as averaging across lexicons.

2.2.3 Early Window Specification

Both parts of my thesis have one thing in common. The time frame of the data used. The window used is from May 1, 2024 - July 31, 2024. This is about three months before the election.

There is a reason behind using this time frame. Most researchers tend to use tweet data from a few weeks leading up to election day. I wanted to go a little earlier. I wanted to check to see if sentiment signals and discussion patterns were already occurring at this earlier point. If they are present, this could suggest that twitter is an early indicator. If not, this finding would be useful to report as well. Thus, the problem formulation is straightforward. Obtain all tweet data from May through July of 2024. Run topic modeling on that data separately. Run sentiment analysis separately. Note what each model determines, and then compare.

The overall problem formulation is that if the topic modeling using LDA proves Trump-related topics are the most cohesive and distinct, and if all four of the sentiment lexicons also find a more positive sentiment towards Trump than towards Biden, then the two individual methods are found to have convergence. This convergence would provide evidence that twitter was signaling in a consistent way Trump's trajectory in the early pre-election period.

If, instead, the two analyses have divergence, it would also be worth investigating. I will, however report whichever direction the results lean toward.

2.3 Methodology (Solution Approach)

This section describes everything I did to answer the questions. I will walk through the entire process step by step. The methodology has two independent branches. The first branch is topic modeling. The second branch is sentiment analysis. They do not depend on each other. I will explain each branch separately, starting from the common dataset preparation steps.

2.3.1 Data collection

I used a publicly available dataset of Tweets (now known as X) collected by researchers who mined the tweets related to the 2024 US Presidential Election [10]. I employ the data set of Balasubramanian et al. (2024) including more than 22 million posts[10] from 1 May 2024 to 31 July 2024. The dataset is available on GitHub [46] and is stored in different directories named “part-{partnumber}”. Each directory has chunk files. Each chunk file

has about 50,000 tweets. Each file contains tweets of a specific period of time. For my work I have used the file "may-july-19.csv" from the directory part-1. This file has tweets from May 1 to July 31 of 2024. This is the early pre-election window that I am interested in. The file has 30 different columns. These columns include tweet text, user info, timestamps and other metadata. I only pulled the 'text' column since I wanted to analyze the content of the tweets. This column contains the actual tweet posts. The keywords and hashtags used to collect the original dataset were "Biden", "Trump", "MAGA", "Joe", "President", "GOP", "Donald", "Harris", "Kamala" and "conservative". The hashtags included "#maga", "#trump2024", "#trump", "#bidenharris2024", "#biden", "#donaldtrump", "#biden2024", "#gop" and "#joebiden".

2.3.2 Data Preprocessing

Raw tweets are messy. They contain all kinds of things that are not useful for analysis. So I had to clean them before I could do anything meaningful. I performed several preprocessing steps. The steps applied were the same for both topic modeling branch and sentiment analysis branch. Let's discuss step by step.

2.3.2.1 Filter out URLs and Mentions and Retain Hashtags-

First, I kept only the tweets that were in English. Most sentiment lexicons are for English language. The discourse about US election is in English and other languages would have introduced noises rather than giving value. So I filtered them out.

Tweets often contain links to other websites. They also contain @username mentions. These do not carry sentiment or topic meaning. A URL tells me nothing about whether the tweet is positive or negative. A mention just tells someone is replying to someone else. Neither is useful for my analysis. So I removed them entirely.

I made a conscious decision to keep hashtags. It was a big deal. #MAGA #Biden2024 Campaign slogans and keywords Removing them would have lost valuable information. A tweet saying "I support #MAGA" is different from a tweet saying "I support Trump". The hashtag carries its own meaning. So I kept them.

Removing special characters and punctuation. Symbols, punctuation marks, and special characters do not add meaning for topic modeling or lexicon-based sentiment analysis. They are just an obstruction.

2.3.2.2 Lowercasing and removing stopwords -

This was straightforward but necessary. The words Trump and trump should be thought of in the same. By lowercasing all words, I eliminate the issue of treating capitalized words differently than lowercased words.

Stopwords are common words that appear very frequently but carry almost no meaning. Words like "the", "and", "of" etc. all fall under this category. Without removing stopwords, the most frequent words in the corpus would be things like "the" and "and". That tells me nothing about the topics. With stopwords removed, the frequent words become things like "trump", "biden", "vote", "election". That is useful information. I used a standard stopword list from the Natural Language Toolkit. This list is widely used in text processing. It covers common English stopwords.

2.3.2.3 Lemmatization-

The last, and probably most significant step in my cleaning pipeline was lemmatization. The fundamental issue that lemmatization tries to solve is morphological variation; that is, that there are numerous surface form variants which all relate to the same underlying concept. A verb such as "run" will manifest in different forms such as "running," "runs," or "ran," depending on the context and whether the tense is present or past. The adjective "good" changes to "better" or "best" when used comparatively, and the plural of "woman" is "women." Without normalization these different variants represent completely distinct words to the model, and so the model incorrectly gives each concept too little weight.

Lemmatization reduces all such surface forms to one base form, and the model counts all instances of the same concept under this single base form, rather than giving them different meanings in different grammatical contexts. This task was performed using the spaCy library which contains a pre-trained English model which attempts to identify the correct base form of any word given its context. I fed each token in every tweet to the model, replaced it with the identified lemma and discarded the inflected original form. All other stages of the cleaning process were performed in sequence-filtering out non-English tweets, deleting URLs and mentions, deleting punctuation, converting all text to lower case, deleting stopwords and finally applying lemmatization. After these stages had been performed on each tweet the text was much more compact. For example, a tweet like: "I really love #MAGA! Trump2024!!" would have been transformed into "love maga trump2024".

Final dataset sizes-

The size of the dataset reduced after preprocessing. For the topic modeling branch, I ended up with 45,264 clean tweets. For the sentiment analysis branch, I ended up with 41,768 clean tweets. The slight difference came from slightly different preprocessing choices. In the sentiment analysis branch, I was stricter about keeping only tweets that had clear candidate keywords. Some of the tweets that were kept in the topic modeling preprocessing were filtered out from the sentiment analysis branch because they did not mention Trump or Biden clearly enough.

2.3.3 Approach of topic modelling

This is the first independent part of my methodology. I used Latent Dirichlet Allocation (LDA) to find the main themes of discussion in the tweets .

2.3.3.1 LDA (Latent Dirichlet Allocation)

LDA is a generative probabilistic model. It was introduced by Blei et al. (2003) [26]. The name sounds complicated but the idea is actually quite simple.

To get an intuition for what LDA is doing, it helps to think about how the model imagines documents came to exist in the first place. It works under the assumption that every document was produced by a hidden generative process: before a single word was written, a blend of topics was drawn for that document, and then each individual word was created by first picking a topic from that blend and then sampling a word from that topic's characteristic vocabulary. A tweet that is heavily focused on campaign rhetoric would draw most of its words from a politics-heavy topic, while occasionally pulling a word from some other topic too [26, 27].

LDA gains the ability to run the process in reverse. Given only the finished documents the actual words on the page it tries to reconstruct the hidden topic structure that most plausibly gave rise to those word patterns. It does this by starting with a random guess about which topic each word belongs to, then iteratively refining those guesses. At each step it asks: given everything else I currently believe about the topic assignments in this corpus, what is the most probable topic for this particular word? It updates the assignment, moves to the next word, and repeats this across the entire dataset many thousands of times until the assignments stop changing meaningfully [26].

This iterative refinement process is known as Gibbs sampling [26]. The underlying logic is intuitive even if the mathematics is not: words that keep showing up together across many different documents are almost certainly being generated by the same topic, so they get pulled toward the same assignment. Words that rarely appear alongside each other drift

into different topics. Over enough iterations the model settles into a stable configuration where each topic is represented by a coherent cluster of co-occurring words, and each document is described by a mixture of those topics.

2.3.3.2 Creating the Document-Term Matrix -

Before I could run LDA, I needed to convert the text into numbers. I created a Document-Term Matrix. Each row in this matrix is a single tweet. Each column represents a unique word in the entire vocabulary of the data set. The value in each cell is the frequency of that word in that tweet [26]. For my dataset, the matrix size was 45,264 rows by 12,822 columns.

LDA needs you to specify the number of topics to find beforehand [26]. I tried different numbers. I tried 2 topics, 3 topics, 4 topics, 5 topics. I looked at the lists of words that LDA gave me for each topic. I also looked at how different the topics were from each other. I tried a few options and settled on three topics. Three topics provided a good balance of clarity and separation. The topics were distinct enough to interpret but not so many as to be confusing.

Training the LDA model

I trained the LDA model on the Document-Term Matrix. The model was allowed to run until it reached a stable state. After training, it produced a topic-word distribution. This distribution tells you, for each topic, which words are most likely to appear. I extracted the top 100 words for each topic.

2.3.3.3 Visualizing Topics with Word Clouds-

Numbers are hard to look at. A list of the top 100 words for each topic is just a block of text. It does not tell you much at a glance. So I needed a better way to see which words were truly important in each topic.

I generated a word cloud for each topic. A word cloud is a way of visualizing text data. Words that occur frequently, or have higher probability in the topic are shown in larger font sizes. Words that are less important are shown in smaller font sizes. This makes it easy to spot the dominant terms at a single glance.

For each of the three topics, I fed the top 100 words and their LDA probabilities into a word cloud generator. The generator placed the words in a random layout. However, the size of each word in the cloud was relative to the probability that the word represented it. The more relevant the word, the larger the font size it was displayed in. Less relevant words seemed to blend into the background. This allowed me to very quickly see the

general idea of what each topic was by looking at the larger words in the cloud and also see more nuanced, secondary aspects to the topic through the medium-sized words. The smallest words seemed to provide cues for the edges or limits of the topic. Also, using the clouds enabled comparison of topics; when I was comparing two topics I could quickly determine whether they contained similar words or if the topics were distinct from one another. If a word appeared equally large in two different topic clouds it was important to both topics. Unique to that topic were words that were big in only one cloud.

Sometimes, the model will create word groupings that seem to be random noise. I needed a way to determine the meaningful topics from the meaningless ones. Coherence scores helped me do this. I also wanted to know which topics appeared most frequently. Prevalence scores helped me with that.

2.3.3.4 Coherence and Prevalence Score-

Coherence captures something intuitive: if the words assigned to a topic genuinely belong together conceptually, they should tend to appear near each other in the actual data. A topic whose top words include "trump", "maga", "rnc" and "conservative" is likely coherent because those words naturally cluster in the same kinds of tweets. A topic whose words almost never appear together in any real tweet is almost certainly noise rather than a meaningful theme [28, 30].

The measure of coherence I employed is based on Normalized Pointwise Mutual Information (NPMI) [28, 30]. NPMI measures the strength of association between two words [28]. For two words, w_i and w_j, NPMI describes the probability of their co-occurrence normalized by the probability of their co-occurrence . Positive NPMI means they co-occur more often than random, negative NPMI means they co-occur less often. I computed the average NPMI over all word pairs for a topic with N top words. Let N be the number of top words considered per topic. The coherence score of topic k is then given by:

$$Coherence(k) = \frac{2}{N(N-1)} \sum_{i<j} NPMI\ (w_i, w_j) \tag{2.1}$$

I used N = 10 for my coherence calculations. Higher coherence scores mean the topic is more meaningful. Lower scores mean the topic is noisy.

Coherence tells you about quality. Prevalence tells you about quantity. The prevalence score of a topic is the average probability of that topic across all tweets. For each tweet,

the LDA model assigns a probability distribution over the three topics. The prevalence score is the average of these probabilities shown in Eq. (2.2)

$$Prevalence\ (k) = \frac{1}{T}\sum_{t=1}^{T} \theta_{t,k} \tag{2.2}$$

Here, T is the total number of tweets and $\theta_{t,k}$ is the probability of topic k in tweet t.

A topic with high prevalence appears in many tweets. A topic with low prevalence appears in only a few tweets. Neither coherence nor prevalence alone is enough. A topic could be very coherent but very rare. That might not be important. A topic could be very common but completely random. That is also not useful. So I looked at both together.

2.3.3.5 Hierarchical Clustering of Topics and Keywords

Coherence scores tell me if a topic is meaningful. But they do not tell me how topics relate to each other. Two topics could both be coherent but also very similar. Or one topic could be completely different from the others. I wanted to see which topics were truly distinct. Hierarchical clustering helped me answer this.

I represented each topic as a vector. The vector was built from the top 100 words of that topic. Each word was weighted by its probability from the LDA model. So important words had higher weights. Less important words had lower weights. I then computed the distance between each pair of topic vectors. The measure of distance I used was the cosine distance. Cosine distance is the angle between two vectors. It can be computed as:

$$Cosine\ Distance\ (A, B) = 1 - \frac{A.B}{||A||.||B||} \tag{2.3}$$

Here A, B are two vectors

If two vectors point in the same direction , the cosine distance is small . This means that the topics are similar . If two vectors point in opposite directions, the cosine distance is large. This means the topics are different.

Then I applied hierarchical agglomerative clustering. This is a bottom up approach [33,34]. It starts with each topic as its own cluster. Then it repeatedly merges the two closest clusters. It continues until only one cluster remains. I used average linkage [33]. This means the distance between two clusters was the average distance between all pairs of topics in those clusters. The final product of such a process is a dendrogram, that is to say a branched tree-like structure, the height at which any two items are joined reflects the amount of dissimilarity between them and two items that join the same branch near the

top of the tree are genuinely dissimilar, those that join near the bottom are genuinely similar [33, 34].

I also wanted to see which words within a topic were unique. I took the top 15 words from each topic and converted each word into a vector using the SpaCy library [35]. SpaCy has pre-trained word embeddings. These embeddings are a representation of the meaning of words.

Then I applied the same hierarchical clustering approach. Again I used cosine distance and average linkage. I generated dendrograms for each topic separately. The dendrograms showed the points of clusters of words and points of individual words. Fig 2.1 shows the entire workflow of the topic modeling approach.

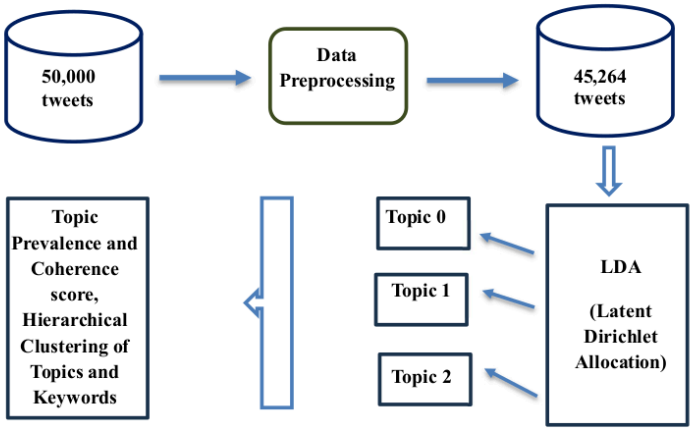


Fig 2.1 Step-by-step pipeline followed for data preprocessing and LDA-based topic extraction [43]

2.3.4 Sentiment Analysis Approach

This is the second independent branch of methodology. While topic modeling tells me what people are talking about, sentiment analysis tells me how they feel about it. I used four different sentiment lexicons VADER, SentiWordNet, TextBlob, AFINN. I wanted to see if they all agreed.

2.3.4.1 Grouping tweets by candidate-

I had to filter tweets containing both Trump's name and 'Trump' tag, and containing both Biden's name and 'Biden' tag. I cannot suppose all election related tweets mentioned both

candidates. I built a set of keywords; the list of keywords was based on the knowledge on the terms in the initial dataset [10], then some hashtags and slogans encountered during an initial phase of exploration.

The keywords for Trump included: 'trump', 'donaldtrump', 'maga', 'trump2024', 'trumptrain', 'makeamericagreatagain', 'realdonaldtrump', 'trump rally', 'trump campaign', 'maga2024', 'americafirst', 'keepamericagreat', 'saveamerica', 'trump supporters', 'rnc', 'gop', 'conservative', 'republikan', 'donald', 'trumpbiden', 'trumpharris', 'bidentrump', and 'harristrump'.

The keywords for Biden included: 'biden', 'joebiden', 'biden2024', 'bidenharris', 'presidentbiden', 'teamjoe', 'votebiden', 'joebidenharris', 'biden administration', 'darkbrandon', 'letsgobiden', 'bidenomics', 'biden campaign', 'joe biden', 'biden supporters', and 'joe'.

If a tweet contained any of the Trump keywords, I put it in the Trump group. If it contained any of the Biden keywords, I put it in the Biden group. Some tweets contained keywords for both candidates. Those were included in both groups for their respective analyses. After grouping, I had 19,323 Trump-related tweets and 22,445 Biden-related tweets.

SentiWordNet

SentiWordNet is a lexical resource. It assigns three scores to each word in the WordNet database [18]. These scores are positivity, negativity, and objectivity. The positivity and negativity scores each range from 0 to 1. The objectivity score is 1 minus the sum of positivity and negativity.

The positive ($posScore_i$) and negative ($negScore_i$) sentiment scores were extracted for each word w_i in the tweet [45]. The mean positive and negative sentiment scores of all the words in a tweet were calculated by applying Eq. (2.4) and Eq. (2.5), respectively, where N represents the total number of words in the tweet. The net sentiment score (difference between positive and negative sentiment scores) for the tweet [45] was determined as per Eq. (2.6).

SentiWordNet is especially beneficial in view of the lexical sentiment scoring provided by it.

$$AvgPosScore = \frac{1}{N}\sum_{i=1}^{N}(posScore_i) \quad (2.4)$$

$$AvgNegScore = \frac{1}{N}\sum_{i=1}^{N}(negScore_i) \quad (2.5)$$

$$SWN\_Net = AvgPosScore - AvgNegScore \quad (2.6)$$

VADER

VADER was built specifically for social media text. It understands capitalization, punctuation, and emoticons [13]. For example, "GOOD!!!" is more positive than "good". VADER captures this.

For any tweet j, VADER calculates compound sentiment scores ranging from -1 (the most negative) to +1 (the most positive) [13]. The avg value of compound scores for all tweets M belonging to each candidate category was calculated as per Eq. (2.7).

$$Vader\_Avg = \frac{1}{M} + \sum_{j=1}^{M}(CompundScore_j) \quad (2.7)$$

AFINN

AFINN is a simple lexicon. Each word in the AFINN lexicon is assigned a score between -5 and +5 [14]. Negative numbers mean negative sentiment. Positive numbers mean positive sentiment. The magnitude indicates intensity. For example, "good" might be +3. "Excellent" might be +5. "Bad" might be -3. "Terrible" might be -5. The average AFINN score for a candidate group was computed in Eq. 2.8.

$$Afinn\_Avg = \frac{1}{M} + \sum_{j=1}^{M}(AfinnScore_j) \quad (2.8)$$

TextBlob

TextBlob evaluates the sentiment polarity within the interval [-1, +1] by employing both rule-based and pattern-based methods [17]. The sentiment polarity of a set of tweets was calculated by applying Eq. (2.9), in which M denotes the number of tweets in a candidate-specific group.

$$TextBlob\_Avg = \frac{1}{M} + \sum_{j=1}^{M}(Polarity_j) \quad (2.9)$$

2.3.4.2 Normalizing Sentiment Scores Across Lexicons

The scoring range differs from one lexicon to another. SentiWordNet scores are between 0 and 1. VADER scores are between -1 and +1. AFINN scores are between -5 and +5. TextBlob scores are between -1 and +1. You cannot directly compare a VADER score of 0.5 with an AFINN score of 2.0. They mean different things.

So I normalized all scores to a common scale [44]. I chose the range [0, 1]. 0 represents the most negative possible score. 1 represents the most positive possible score.

The normalization score calculation was shown in Eq. (2.10), with x being the average sentiment score value obtained for the candidate and the minimum (min(x)) and maximum (max(x)) values are defined by their corresponding sentiment score range shown in Table 2.1. This ensures that all lexicons contribute equally when comparing sentiment values among the candidates

$$NormalizedScore = \frac{x - \min(x)}{\max(x) - \mathrm{mi}\quad(x)} \tag{2.10}$$

This allowed me to compare results across lexicons and also calculate an average normalized score of all lexicon for each candidate. Fig 2.2 shows the workflow of the sentiment analysis approach.

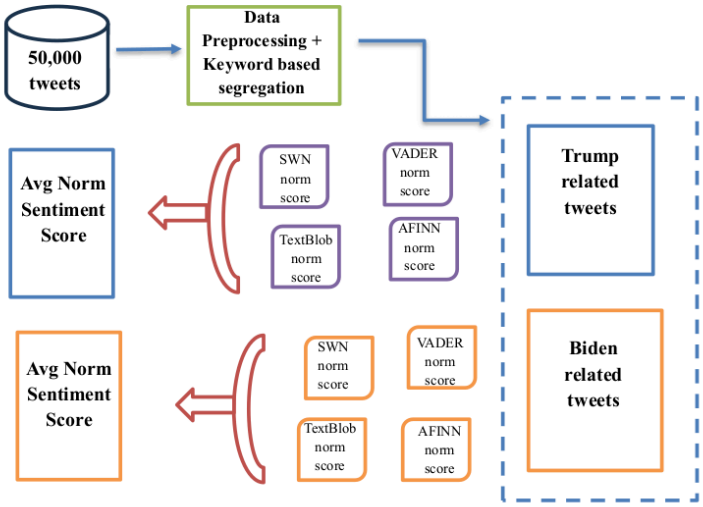


Fig 2.2 Overall process flow for lexicon-based sentiment analysis with score normalization [44]

Table 2.1 Scoring intervals of the four lexicon-based sentiment analysis tools used in this study [44].

Lexicon based Method	Range of their sentiment score
VADER	[-1,1]
SentiWordNet	[0,1]
TextBlob	[-1,1]
AFINN	[-5,5]

The two branches of the study Topic Modeling approach and Sentiment Analysis approach shared the identical basic dataset and same data preprocessing step. However, after that point, they were kept separate, and there was no interaction between the two parts. Results obtained during topic modeling didn't affect sentiment analysis and vice versa. This was a conscious choice since I needed to analyze what each part could yield independently.

Having covered what I did, it is time to proceed to what I got. First, there will be the results of the topic modeling followed by the results of sentiment analysis. After that, both of them will be compared.

Chapter 3

Results and Discussion

In this chapter I will present all results of my experiments. I have organized it into two main parts. The first part covers topic modeling results. The second part covers sentiment analysis results. Both parts are presented independently. At the end of each part, I discuss what the findings mean and then bring both together in a final discussion.

All experiments were done in Python on Google Colab. For the topic modeling, I used various libraries like Scikit-Learn, Gensim, Matplotlib and SciPy. Sentiment analysis was done using the VADER, AFINN, TextBlob, SentiWordNet libraries and Matplotlib for visualizations, Section 3.1 and 3.2 results are drawn from and expand upon the findings reported in our published conference paper [43, 44].

3.1 Topic Modeling Results

LDA was applied to the preprocessed dataset of 45,264 tweets. The number of topics to be extracted from the tweets was set to three. This value was chosen after trying other values. Three topics seemed to be the most ideal .

3.1.1 Three Topics Found Using LDA Analysis

Once the LDA algorithm was executed, I had the topic-word distribution matrix. From this matrix, I picked out the 100 best representing words per topic. Using this list of words, I produced word clouds where the size of a word shows how significant it is. Larger words are more significant.

Topic 0

Words in the word cloud for Topic 0 included “biden”, “trump”, “president”, “donald”, “schiff”, “attempt”, “people” this topic is related to political leadership and some events both "biden" and "trump" were used in the same topic and the words “Schiff” and “attempt” were used, which means there are specific things or controversies this topic included people mentioning political events related to both candidates, the prevalence score for Topic 0 was 0.2690 and the coherence score was 0.2523 [43].

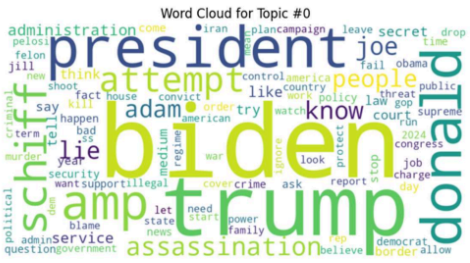
Topic 1

In the word cloud for Topic 1, there were such words as "biden", "trump", "vote", "want", "joe", "win", "people", "harris", "president", and "lose" this topic is concerned voting and elections results such terms as "win" and "lose" drew attention right away people discussed the winner and the loser of the presidential contest, the word "want" implied that some preferences were expressed therefore, this topic focused on the race for power. There was no mention of any policy or event it is all about the result, the prevalence rate of Topic 1 was 0.4069 and Coherence was 0.5428 [43].

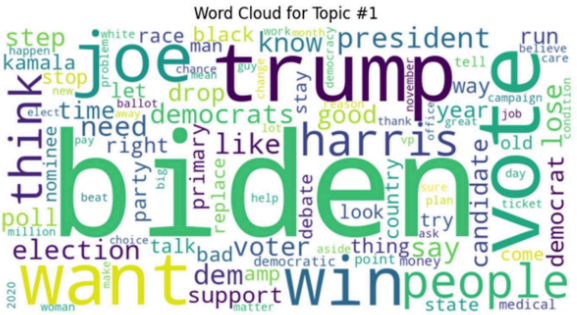
Topic 2

The word cloud for Topic 2 contained such words as "trump", "maga", "rnc", "gop", "conservative", and "republican" Topic 2 was devoted to Trump and his MAGA ideology. In particular, MAGA was shown in the word cloud very prominently one could see that it was a central theme, but not just one side note to the main idea words like "rnc" and "gop" implied that this topic included discussions not only about Trump himself but also about the Republican Party as an organization, the prevalence score for Topic 2 was 0.3241 and the coherence score was 0.5752 [43].

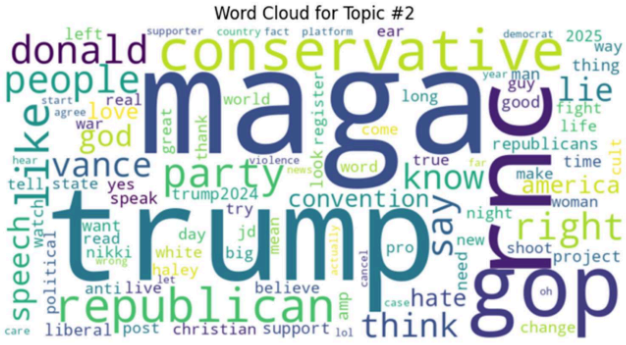
Fig 3.1 shows the word cloud of the three topics. What struck me when looking at these three word clouds was that Trump appeared in all three topics but in completely different roles. In Topic 0, he appeared alongside Biden in discussions of political events. In Topic 1, he appeared alongside Biden in discussions of winning and losing. But in Topic 2, he appeared alone with MAGA and Republican Party terms. Biden was nowhere in Topic 2. It was assumed that Trump had a separate, unique forum for discussion from Biden's. People talked about Trump in relation to the MAGA movement differently than they talked about elections in general.



(a)



(b)



(c)

Fig 3.1 . Visual depiction of the three LDA-generated topics using word clouds [43]

3.1.2 Coherence and Prevalence Scores Analysis

Coherence is a measure of the importance of the topic in question it reflects the degree to which words in a particular topic co-occur in original messages high coherence means the presence of real words, while low coherence implies noise [28, 29].

If coherence indicates how good a topic is, prevalence measures how "wide" it is, i.e. what percentage of the entire corpus of tweets it covers; a highly prevalent topic is widely spread across tweets, a lowly prevalent topic is narrowly confined neither coherence nor prevalence provides a complete description, however [28, 32]. A topic could be very coherent but appear in only a handful of tweets. That would be interesting but not broadly important. A topic could be very prevalent but have low coherence. That would mean many people talked about something but the discussion was scattered and unfocused. So I looked at both together. Here are the scores I got Table 3.1 shows coherence and prevalence score of respective topics .

Table 3.1 Summary of LDA output showing representative keywords, coherence scores, and prevalence scores for each topic [43]

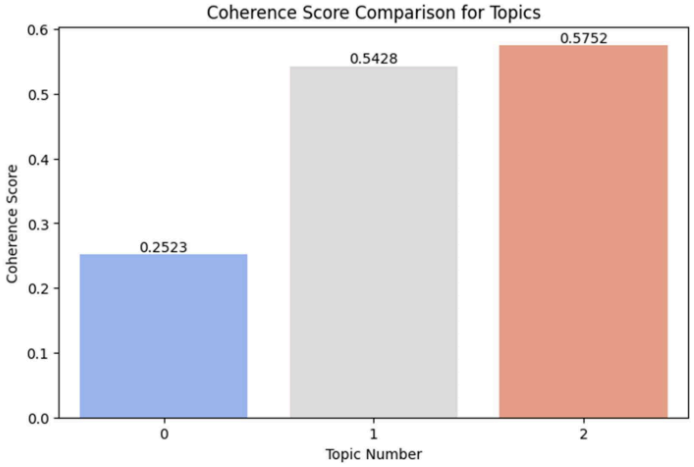
Topic Category	**Representative Keywords**	**Topic Coherence**	**Topic Prevalence**
Topic 2	trump, maga, rnc, gop, conservative, republician	0.5752	0.3241
Topic 1	biden, trump, vote, want, joe, win, people, harris, president, lose	0.5428	0.4069
Topic 0	biden, trump, president, donald, amp, schiff, attempt, , people	0.2523	0.2690

Topic 2 had the highest coherence score at 0.5752. This was significantly higher than the other two topics. This means that, when referring to MAGA or Trump, consistent language usage can be observed, as "trump," "maga," "rnc," "gop," "conservative," and "republican" occurred together often in the same tweets, and they are supposed to be together. It is a clear sign of an organized debate.

Topic 1 was the dominant topic, receiving the highest prevalence score of 0.4069. As the main essence of the elections is to determine winners and losers, it was expected for such a topic to receive the highest score. However, its coherence score was 0.5428.This is still good but lower than Topic 2. Why? Because discussions about winning and losing can take many forms. Some people talked about Trump winning. Some talked about Biden winning. Some talked about polling. Some talked about predictions. Language was more varied, so the coherence was lower, even though the prevalence was higher

Topic 0 had the lowest coherence score at 0.2523 and the lowest prevalence score at 0.2690. This topic was the least meaningful and the least frequent. The words in this topic did not appear together consistently. The discussions were scattered. It could either reflect political discussions where the topics of conversation failed to create further dialogue. Alternatively, it could be noise within the data which LDA could not categorize into either Topic 1 or Topic 2.

Figure 3.2 visually compares coherence and prevalence scores between the topics. Most interesting, Topic 2 was found to have the highest semantic consistency, though not the highest prevalence. The people talking about Trump and MAGA were using clear and relevant terminology. Their discussions were more structured than discussions about voting outcomes or political events.



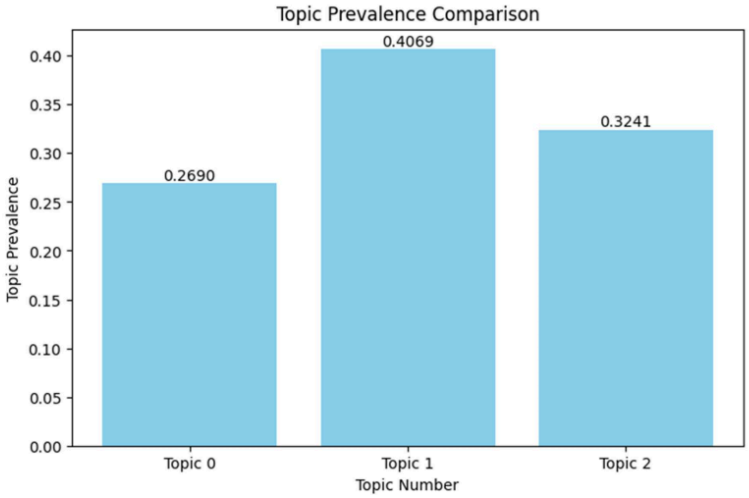


Fig 3.2 Bar chart comparing Topic Coherence and Prevalence Scores across all three topics [43]

3.1.3 Results of Hierarchical Clustering

Coherence scores tell me if a topic is meaningful. But they do not tell me how topics relate to each other. Two topics could both be coherent but also very similar. Then again, they might not be separate topics at all. They could be just variations of the same underlying theme. Hierarchical clustering helped me see which topics were actually different from each other.

Clustering the three topics

Each topic was represented by vectors containing 100 most probable words of the topic [43]. I then computed cosine distance for those vectors. Cosine distance is the angular measurement between two vectors [33]. Small distance means the vectors are similar. Large distance means they are different. I performed hierarchical agglomerative clustering using the average linkage [34]. This produces a dendrogram. A dendrogram is a tree-like diagram. The vertical axis shows the distance at which clusters merge. Topics that merge low on the tree are very similar. Topics that merge high on the tree are very different.

The dendrogram I obtained showed something clear and interesting Fig 3.3 shows the dendrogram of topics. Topic 2 formed a distinct, separate cluster. It was well separated from Topics 0 and 1. Topics 0 and 1 were closer to each other. They merged on a very smaller distance. So Topics 0 and 1 had some overlapping discussion themes. They were not separated. But Topic 2 was not. So what does it mean? It means that the discussion around Trump-MAGA is semantically a very specific discourse and is not a variant of the general election discourse and is not a subset of the discourse about voting. It is a discourse in its own right. A discourse by people interested in discussing Trump and MAGA is semantically different than the one by people who just want to discuss about who is going to win the elections, as they use different words.

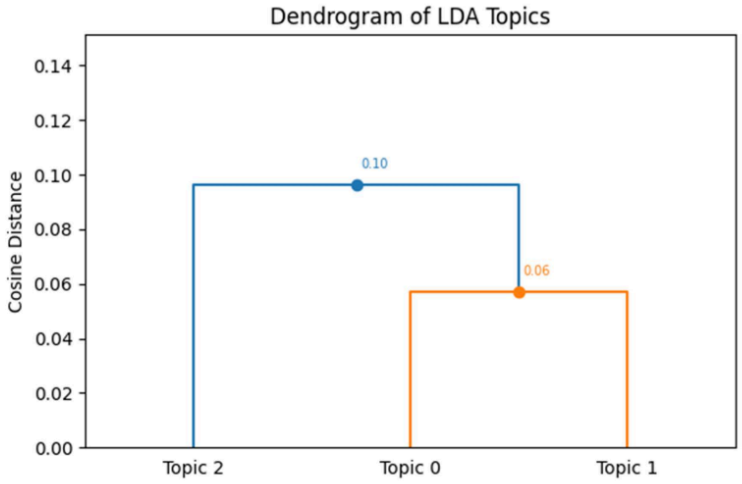


Fig 3.3 Hierarchical clustering dendrogram showing similarity relationships among the three LDA topics [43].

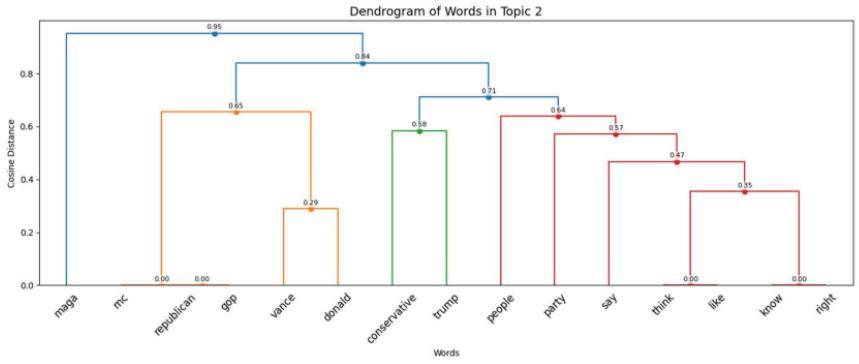
Clustering top words in each topic.

I also wanted to find out which words are unique for a specific topic. Sometimes within a topic, many words group closely together. And other times there is a specific keyword which stands on itself and other keywords might not have similar words attached. This can suggest whether a certain keyword is just some synonym of a keyword and might not be very central to the topic. From all the topics I took the top 15 most important words

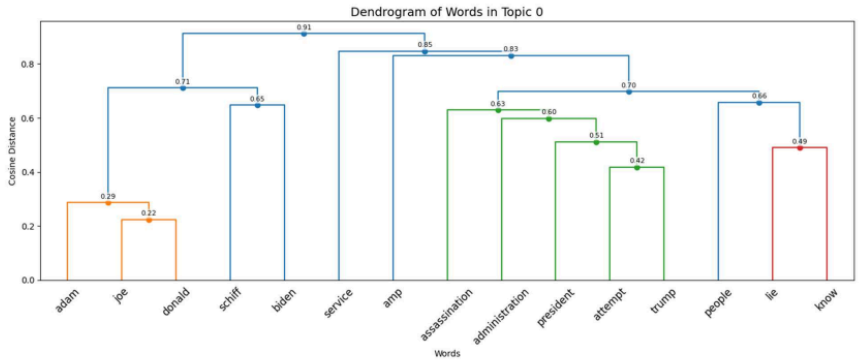
from each, I turned these words into vectors and used the spaCy library and their pre-trained word embeddings [43]. Word embeddings represent the semantics of the words. Similar words will have close vectors and dissimilar words will have further apart vectors. I then used the same approach for hierarchical clustering, with cosine distance as distance and average linkage as metric and drew a dendrogram for each topic (Fig 3.4).

It was the topic 2 word dendrogram that was most telling. "MAGA" formed its own separate cluster with no related words even close to it in the dendrogram like: "Trump", "GOP" and "republican". I was surprised to find this, as I predicted MAGA would be a keyword related to Trump, and I assumed that they would be grouped with other words representing the republicans. However this dendrogram suggests that MAGA had semantic agency. MAGA was not simply another slogan tacked onto Donald Trump; it was something people felt stood separately from the President. The act of people Tweeting the word "MAGA" did not represent the person of the President; it represented the broader message that went along with him. A political ideology. A movement. The fact that the word MAGA formed a separate cluster in Topic 2 also suggests the effectiveness of Trump's campaign message. The slogan of the President's campaign was not simply some marketing catch-phrase; it was a movement.

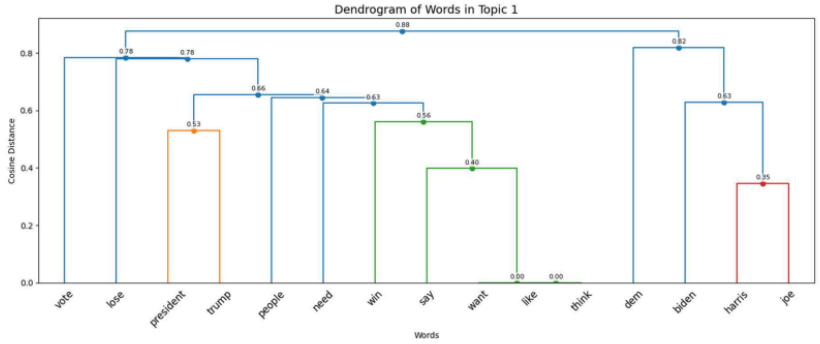
For Topic 0 and Topic 1, the word level dendrograms was less clustered. It seemed as if words where somewhat more spread out and loosely aggregated with each other. This finding was coherent with the finding from coherence scores; Topic 0 and Topic 1 had less well-defined themes than Topic 2.



(a)



(b)



(c)

Fig 3.4. Word-level hierarchical clustering results for (a) Topic 2, (b) Topic 0, and (c) Topic 1 [43]

3.1.4 Discussion of Results of Topic Modeling

Coming to these topic modeling results mean: the first thing I saw about these results was the fact that Trump was present in each of these topics; but his presence was mixed. The first topic he was along with Biden discussing political events and, in the second topic he was once again along with Biden discussing wins or losses, but the last one, topic two, completely excluded Biden in favor of just MAGA and Republican party. Unlike Trump, he completely excluded from the second topic. This seems to imply he has his own personal space. There really was a topic in the group that was only regarding Trump, it didn't use the term Biden once, unlike, it seems, Biden cannot make a comment with out a mention of the other.

Second thing I found interesting about the results were the coherency scores for each of the topics. The Trump and MAGA topic showed that they understood exactly what this topic was about. By coherency I mean that it is possible to know what the topic is regarding. No random words are thrown around; there is a specific meaning/direction to them.

Lastly, I examined the grouping that the results implied. The second topic is clearly separate of the two other topics; topics 0 and 1. And yet within the Trump-MAGA topic, I noticed that the MAGA part of the topic was not along with anything else. Which seems to me that it is not simply a different term that could have been used in place of Trump; MAGA itself has it's own distinct group and gravity.Putting all of this together, the topic modeling analysis suggested that Trump and his MAGA campaign were discussed in a focused, coherent, and distinct way. The discussion had structure. It had consistency. It had a clear identity.

3.2 Results of sentiment analysis

I now turn to the second independent branch of my work. I looked at sentiment about Trump and Biden with four different lexicons. First, I classified tweets by candidate using keyword lists Fig 3.5 shows Tweets distribution among Trump and Biden, after classification I had 19,323 tweets related to Trump and 22,445 tweets related to Biden [44]. Notice that Biden had more tweets overall. People mentioned Biden more frequently. But as I will see, frequency is not the same as sentiment.

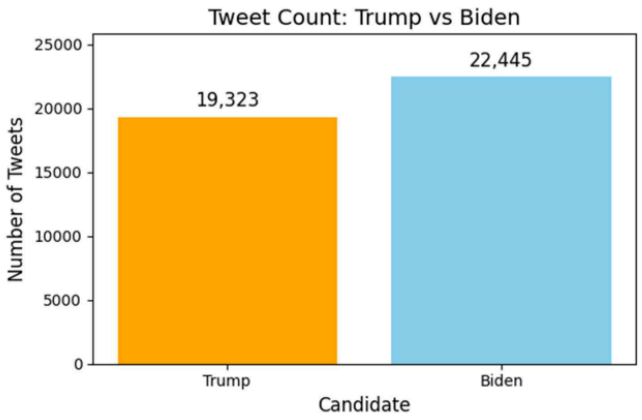


Fig 3.5 Bar chart showing the number of tweets associated with each candidate [44]

3.2.1 Results from SentiWordNet

SentiWordNet gives each word three scores. Positive, negative, and objective [18]. For each tweet I calculated the average positive score for the words in the tweet, the average negative score for the words in the tweet and then I calculated the net score as the average positive score minus the average negative score. If the net score is positive then the tweet is more positive than negative and vice versa. For Trump-related tweets, the average positive score was 0.046546, the average negative score was 0.038172 and the net score was 0.008374, this is positive and coming to tweets about Biden had an average positive score of 0.042608, the average negative score was 0.038892 and the net score was 0.003716, this is positive too, but not in the league of Trump both candidates had positive net sentiment according to SentiWordNet but Trump's net sentiment was more than double that of Biden, this suggested a favorable lean toward Trump [44].

3.2.2 Results from VADER

VADER is specifically designed for social media text. It understands capitalization, punctuation, and emoticons [13]. VADER gives each tweet a compound score between -1 and +1. Negative means negative sentiment. Positive means positive sentiment. For Trump-related tweets, the average VADER score was 0.012611,this is positive, albeit just above zero and the average VADER score for tweets about Biden was -0.050956, this is negative the difference was clear and consistent. Trump tweets were on the positive side

of neutral and Biden tweets were on the negative side of neutral, VADER showed a clear preference for Trump over Biden [44].

3.2.3 Results from AFINN

AFINN gives each word a score between -5 and +5 [14]. Negative numbers mean negative sentiment. Positive numbers mean positive sentiment. The magnitude indicates intensity. I summed the scores of all words in each tweet. Then I averaged these sums across all tweets in each candidate group. The average AFINN score for tweets about Trump was -0.340475.and the average AFINN score for tweets related to Biden was -0.743016, both scores were negative neither candidate received positive sentiment according to AFINN but here is the important part Trump's score was less negative than Biden's, Here the negativity toward Trump was milder and the negativity toward Biden was stronger and more intense, therefore Trump still came out ahead he was less disliked than Biden [44].

3.2.4 Results from TextBlob

TextBlob gives each tweet a polarity score between -1 and +1 [17]. Negative means negative sentiment. Positive means positive sentiment. The mean TextBlob score for Trump-related tweets was 0.049540 and the average score of TextBlob for tweets referring to Biden was 0.018301 both scores were positive but Trump's score was higher than Biden's, TextBlob agreed with SentiWordNet and VADER and Trump came out ahead [44].

3.2.5 Normalization of Sentiment Scores

Each lexicon uses a different scoring range. SentiWordNet net scores from -1 to +1 VADER ranges from -1 to +1 AFINN ranges from -5 to +5 TextBlob ranges from -1 to +1, so I normalized all scores to a common scale from 0 to 1 where zero represents the most negative possible score for that lexicon and one represents the most positive possible score this rendered all scores directly comparable after normalization for each and every lexicon used, Trump's normalized score outstripped Biden's; there was no lexicon where Biden outscored Trump it was the reliability of the pattern across four distinct tools that built my confidence if only one of the lexicons had registered Trump outstripping Biden, I would have been suspicious; it would have simply been a peculiarity of that tool, nothing more, however to register it across all four is something else. It could not have happened accidentally [44].

3.2.6 Aggregated sentiment comparison

I also averaged the normalized score across the four lexicons for each candidate, creating a single overall sentiment measure, the average normalized score for Trump was

.376350.Biden's average normalized score was .353272 here Trump scored higher on the overall sentiment than Biden, the difference was not huge but it was marginal about 0.023, which is roughly 2.3% on the normalized scale, but it was consistent and it was present across every single lexicon this means that between May and July 2024 the average sentiment of the public on Twitter towards Trump was more positive than towards Biden. Biden had more tweets mentioning him but when people did discuss Trump, the tone was more favorable Trump discussions were fewer, but higher quality [44].

Table 3.2 shows average sentiment scores, normalized score of each lexicon based sentiment analysis and also shows the average normalized score of both candidates.

3.2.7 Discussion of Sentiment Analysis Findings

The first thing that struck me was the disagreement between lexicons about whether sentiment was positive or negative. VADER and TextBlob and SentiWordNet showed positive scores for Trump. AFINN showed negative. So which one is correct? There is no single correct answer. Positive and negative are defined in a different way in each lexicon. The important thing is not the absolute sign of the score. What really matters is the comparison of candidates with each other. All four lexicons agreed that Trump had higher scores than Biden. That is the real finding. The second thing I noticed was the tweet counts. Biden had more tweets. People mentioned him more often. But the sentiment was less positive. This is interesting. Frequency does not equal favorability. A candidate can be talked about a lot but not liked. Another candidate can be talked about less but liked more. The third thing I noticed was the margin. The difference between Trump and Biden was small. This was not a landslide in sentiment. It was marginal. But in an election that could be decided by small margins, even a small difference matters. And the fact that the difference was consistent across all four lexicons makes it more meaningful.

Table 3.2 Raw and normalized sentiment scores for both candidates across all four lexicons. The average normalized score is also shown [44]

Sentiment Metric	Trump	Biden
Average SentiWordNet Positive Score	0.046546	0.042608
Average SentiWordNet Negative Score	0.038172	0.038892
Average SentiWordNet Sentiment Score	0.008374	0.003716
Average VADER Sentiment Score	0.012611	-0.050956
Average AFINN Sentiment Score	-0.340474	-0.743016
Average TextBlob Polarity Score	0.049540	0.018301
Normalized SentiWordNet Score	0.008374	0.003716
Normalized VADER Score	0.506306	0.474522
Normalized AFINN Score	0.465952	0.425698
Normalized TextBlob Score	0.524770	0.509151
Combined Average Normalized Sentiment Score	**0.376350**	**0.353272**

3.3 Discussion

Now I want to bring both parts together. Topic modeling and sentiment analysis were done independently. They did not influence each other. Yet both pointed in the same direction.

Topic Modeling Result

What the topic modeling revealed, above all else, was that the conversation around Trump and MAGA had a qualitatively different character from the rest of the corpus. It was tight, focused and internally consistent — the kind of discourse where people repeatedly reach for the same vocabulary because they are genuinely part of a shared narrative. Strikingly, "MAGA" did not simply cluster with "Trump" as a near-synonym; it formed its own

distinct grouping, suggesting it had taken on a life beyond being a label for one candidate [43].

Sentiment Analysis

Across four lexicons the sentiment of Trump was higher than Biden. The four lexicons displayed differing results. Some showed both candidates with positive sentiment, others negative sentiment, others Trump with positive sentiment and Biden with negative. However, Trump was consistently higher than Biden. The topic modeling showed Trump was discussed in a narrow focused topic. The sentiment analysis illustrated talk of Trump was generally more positive than talk of Biden. Together these show Trump was not merely discussed but talked about in a valuable positive structured manner and the tone of the discussion was generally positive.

Early Indicator

Data was collected between May 2024-July 2024. Three months prior to the election. If there is anything that social media will predict, it will be during this three month window. What my results demonstrate is that despite a greater number of total tweets, Donald Trump had the more coherent discussion, and a more positive discussion. The conversation quality was more important than the number of tweets, and the quality belonged to Trump.

Limitations

This doesn't mean that Twitter sentiment will always predict election results. This doesn't mean that every Trump tweet was positive. and the conversation about Biden wasn't positive at all. Users of Twitter aren't a representative sample of all voters [3,4,6]. Twitter posters are, on average, the most passionate and extreme voters. The discussion is useful only as an early indicator, and is not a substitute for polls.

Key Takeaway

Political operatives and campaigns may rely on polling that is collected and analyzed in a way that can take hours or days. Social media analysis can be conducted in nearly real-time. If the results of the two forms of analysis converge on the same candidate weeks or months before the election, that insight is invaluable. These two forms of analysis do exactly that. Donald Trump was found to have a more coherent conversation, and a more

positive conversation as well. Both methods are independent and came up with the same conclusion.

3.4 Implementation Notes

All experiments were executed on the Google Colab platform. I was using the free version, a regular CPU runtime. Since lexicon based sentiment analysis and LDA topic modelling do not require any GPU and the datasets were not too large to be put into memory, I didn't require a specific configuration.

For the topic modelling branch, the Document-Term Matrix was composed of 45 264 lines and 12 822 columns, I used wordcloud library of Python to build the word clouds and have personalized the colors and the maximum number of words to be display for a best visibility and I plotted the dendrograms using the Matplotlib library and the Hierarchical clustering functions from Scipy [43].

For the sentiment analysis branch, I am processing 41,768 tweets through four lexicons Most of that time was spent on SentiWordNet lookups. SentiWordNet is slower than the other lexicons because it requires, I used the WordNet database to look up the meaning of each word. VADER, AFINN, and TextBlob are faster because they keep scores in memory and look. I also saved intermediate results as CSV files after each major step. This way I could rerun only the steps that needed correction, instead of starting over. For example, after completing the initial sentiment analysis, I realized that I needed to normalize the scores, so I could load the raw scores from a CSV file and simply rerun the normalizing step. That saved time.

Chapter 4

Conclusions, Future Scope and Social Impact

This is the final chapter of my thesis. In this chapter I will summaries what I have done and found out and what it means. I will also discuss the limitations of my work and what can be done next. I will finally touch upon the potential applications of this research to society outside of the academic realm.

4.1 Conclusion

I will reiterate my aim. I aimed to find out if social media data from Twitter (now X) can reveal early signals of public mood with regards to the 2024 US Presidential election. I considered three months from May 2024 up to July 2024. This was approximately 3 months before the actual election date. Most researchers focus their data collection to the last couple of weeks leading up to election day, and I aimed to collect data at a much earlier date to see if patterns of discourse and sentiment signals already existed. Instead of relying solely on one method I incorporated two separate approaches. The first was topic modeling using Latent Dirichlet Allocation. The second was sentiment analysis using four different lexicon-based tools. I kept both methods completely separate. Neither influenced the other. Then I looked at what each method found and compared them.

4.1.1 Summary of Topic Modeling Results

I used LDA on 45,264 cleaned tweets. Three main topics were extracted by this model. The first topic was political leadership and current affairs. The second topic was about voting and election results. The third topic was Trump and his MAGA campaign.

The most coherent topic was the third one with 0.5752 [43] which is the measure of how often the words in the topic co-occurred in the original tweets, meaning that the discussion was focused and organized and people were talking about Trump and MAGA in a consistent language. The most prevalent topic was the second one with 0.4069 meaning that the discussions about winning and losing were in more tweets than any other topic. But the third topic was not far behind at 0.3241.

Then I performed hierarchical clustering on the three topics. In the dendrogram it was possible to see the third topic being one distinct and different group. The topic was distinct semantically. It was not a variation of the other two topics. I did also cluster top words for each topic (top 15 words for each topic). In the third topic the word MAGA was it's own distinct cluster separate even to closely associated words Trump, GOP, republican. This

was a surprise and I learnt from this that MAGA was not merely a tag for the term Trump, it was itself, its own narrative, concept and discussion.

4.1.2 Sentiment Analysis results summary

Sentiment analysis was applied to all of the tweets on the dataset with respect to each candidate individually using the 4 lexicons, Upon filtering the tweets by keywords, 19,323 tweets were related to Donald Trump while 22,445 tweets were related to Joe Biden; though Joe Biden received more mention it does not mean he was more favored. All 4 lexicons were tested, yielding: SentiWordNet assigned a score of 0.008374 to Trump and 0.003716 to Biden; both positive with Trump being higher. VADER assigned a score of 0.012611 to Trump and -.050956 to Biden; again Trump being positive, using AFINN gave Trump -0.340475 and Biden -0.743016; Trump was once more favored. For the final lexicon TextBlob; Trump was assigned 0.049540 and Biden 0.018301, both positive but Trump more favored, by every single one of the 4 lexicons Trump received the better positive score every single time when it was positive and Donald Trump also had the better positive score in each and every comparison, once scaled to a 0-1 scale: Trump 0.376350 and Biden 0.353272; very close, and the score is still higher for Trump [44].

4.1.3 Convergence of Independent Analyses

The two analysis approaches- topic modeling and sentiment analysis thus speak to the same underlying reality without ever sharing information or influencing one another Topic modeling clearly indicated that the Trump-MAGA discourse is the most semantically coherent and conceptually separate strand within the entire corpus, and even MAGA established its own conceptual niche within the Trump-MAGA topic and Sentiments on Trump-associated tweets are thus more positive than on Biden-associated tweets for all four sentiment measures, although again [43, 44].

4.2 Future Scope

As every research is a building brick in the scientific building, no project can be stated as complete. However, the suggestions mentioned below are obvious extensions to the present work and are based on its limitations and strengths.

4.2.1 Expanding the Time Window

I have confined my study on the pre-election period from May to July. It is intended to analyze the trends of topics and their sentiments over the campaign period from early 2024 till election November 2024. It will help in analyzing if the initial trends will remain till the final and thus reflect the impact.

4.2.2 Include User Meta-Data

This analysis of tweet contents do not differentiate based on its creator. With additional user features such as number of followers, verified account, account age etc., it will be possible to determine influential accounts and thereby differentiate between human users and bots. The treatment of all tweets as equal under the belief of equal impact is a limitation I wish to overcome.

4.2.3 Multiple Social Media Platform Analysis

I had analyzed data from twitter only; however, social media discourse involves many platforms. Facebook, Reddit, Instagram are among few of these platforms where such a political discourse occurs with its unique way of interaction and platform-specific language and features. Comparative study of twitter against another platform will surely enhance the study to understand the transferability of findings across the platforms.

4.3 Social Impact

Having looked at the technical aspects of my research I want to take a step back and address why this research is important for society. For me at least the point of doing research isn't to publish paper after paper, it's that I believe analyzing social media may have some benefit for people to understand their political landscape better.

4.3.1 Relevance to Political Strategists and Campaign Teams

Campaigning is a very draining business and cost an immense amount of money. Strategists must be ever aware of where the most amount of money needs to be spent and what messages are most impactful. The current most obvious way to do this is through polling, but polling has some serious flaws: it is notoriously slow; polling can take days, weeks even to conduct and analyze and by the time the poll returns the electorate may already have changed their minds due to new events or topic discussions. Analyzing social media can potentially offer a feedback loop in a near real-time fashion identifying when topics suddenly start gaining candidate-focused traction, and how people's sentiment towards a candidate is shifting days, or even weeks, earlier than the results of any poll. The work presented has showed that these topic and sentiment signals are already evident in the pre-election period which could potentially offer campaign strategists another tool with which to direct campaign resources without it being necessarily a crystal ball.

4.3.2 Relevance to Journalists and Political Analysts

Journalists and political analysts' greatest challenge is accurately representing what their audience wants to know and what they consider to be the important issues. While polls are often used, interviews and questionnaires can be a more timely and accurate measure,

although both are often time consuming to conduct, can be subject to bias due to sampling and the questions asked may lead the respondent's answer. Social media platforms provide millions of individual voices freely giving their opinion; topic modeling may be one method of sorting through large amounts of online discourse, allowing for efficient identification of key topic areas without extensive manual input. A journalist may find that computational methods allow them a snapshot overview of a public discussion post-event, or debate, almost instantly.

4.3.3 Relevance to Voters and Citizens

The functionality of a democracy can be influenced by how informed their citizens are. With the deluge of information coming from the media the current environment often leads to information overload, and the spread of countless counter arguments. The research presented here can represent one small step in making this vast quantity of information more manageable. From the individual voters' perspective, to filter down a large amount of online discussion, say thousands of individual political tweets, to what the primary topics and the prevailing sentiment really is provides them with some of the key components to gauge their political environment from.

4.3.6 Final Remarks

A note of caution is needed here. Social media analysis is not a panacea. Twitter users are not representative of the general voter [3,4,6]. Twitter users are younger, better educated, and more politically involved than the general public. My findings are representative of discussion on twitter and not for the general electorate as a whole. It would be disastrous if political strategists used social media data and solely depended on it. It is necessary to use traditional polls as well; social media is a supplement, not a replacement.

With the limitations considered, social media analysis has a place for determining public opinion. It is not perfect. But it is fast and cheap. It can give a voice to those people whose voices are not represented in a typical poll. And most importantly, in the study I performed, it gave us insights into candidate momentum months before the election occurred.

The question I started with was modest: does social media carry any signal worth paying attention to before an election? Having worked through the data, I think it does. Not a definitive signal, not a replacement for other evidence, and certainly not a crystal ball. But a real signal nonetheless one that in this case pointed in the same direction as the eventual outcome, and did so three months early.

EARLY PREDICTION OF PUBLIC OPINION TRENDS IN THE 2024 U.S. PRESIDENTIAL ELECTION USING TOPIC MODELING, DENDROGRAM CLUSTERING, AND SENTIMENT ANALYSIS

ORIGINALITY REPORT

2%	1%	1%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	ebin.pub Internet Source	<1%
2	www.pbcsf.tsinghua.edu.cn Internet Source	<1%
3	Ben-Arieh, D.. "Data Envelopment Analysis of clinics with sparse data: Fuzzy clustering approach", Computers & Industrial Engineering, 201208 Publication	<1%
4	Uwe Engel, Anabel Quan-Haase, Sunny Xun Liu, Lars Lyberg. "Handbook of Computational Social Science, Volume 1 - Theory, Case Studies and Ethics", Routledge, 2021 Publication	<1%
5	www.repustate.com Internet Source	<1%
6	Marisa Vasconcelos, Jussara Almeida, Marcos Gonçalves, Daniel Souza, Guilherme Gomes. "Popularity dynamics of foursquare micro-reviews", Proceedings of the second edition of the ACM conference on Online social networks - COSN '14, 2014 Publication	<1%
7	ntnuopen.ntnu.no Internet Source	<1%

8	expresser.lkl.ac.uk Internet Source	<1 %
9	jutif.if.unsoed.ac.id Internet Source	<1 %
10	repository.mines.edu Internet Source	<1 %
11	Trada, Parth. "Evaluating Sentiment Analysis Mechanism for Labelled Amazon Reviews", University of Houston-Clear Lake, 2023 Publication	<1 %
12	www.frontiersin.org Internet Source	<1 %
13	era.library.ualberta.ca Internet Source	<1 %
14	elib.dlr.de Internet Source	<1 %
15	escholarship.org Internet Source	<1 %
16	www.imperial.ac.uk Internet Source	<1 %
17	Alisa Kongthon. "Expert identification for multidisciplinary R&D project collaboration", PICMET 09 - 2009 Portland International Conference on Management of Engineering & Technology, 08/2009 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		