Analyzing Reddit's Climate Change Discourse: A Distributed Study of Topic Modeling and Sentiment Analysis Techniques

Candidate Numbers: 39884, 48099, 49308, 50250*
ST446 Distributed Computing for Big Data

Abstract

Social media platforms host large-scale discussions that reflect public opinion on global issues. In this project, we develop a distributed pipeline using Apache Spark to analyze Reddit climate change discussions. Comparing multiple topic models and sentiment classifiers, we select LDA and Logistic Regression for their balance of interpretability and performance. The analysis reveals how themes and sentiment around climate change have evolved over time, offering insights into evolving climate discourse.

ACM Reference Format:
Candidate Numbers: 39884, 48099, 49308, 50250. 2025. Analyzing Reddit's Climate Change Discourse: A Distributed Study of Topic Modeling and Sentiment Analysis Techniques. In . ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Reddit has become a prominent platform for large-scale public discourse, offering topic-specific discussions that reflect a wide range of opinions on global issues. Among these, climate change stands out as a topic of recurring concern, often intersecting with politics, science, technology, and personal belief. Reddit provides unfiltered perspectives that are valuable for understanding how people talk about and respond to climate issues. Analyzing such discussions requires both scalable and interpretable methods. This project applies distributed computing techniques to process Reddit comments related to climate change. We extract recurring themes via topic modeling and track sentiment shifts and emotional engagement across topics using sentiment analysis. We compare several topic modeling methods —LDA, NMF, LSA, and BERTopic—using a variety of evaluation metrics to assess their performance. For sentiment analysis, both Logistic Regression and Naive Bayes are tested using VADER sentiment scores. Based on overall performance, we proceed with LDA and Logistic Regression for the final combined analysis, as they offer a strong balance of interpretability, coherence, and computational efficiency. All analysis is conducted within a distributed framework using Apache Spark, allowing for efficient processing of high-volume, unstructured text. By combining model comparison with temporal and thematic analysis, this work contributes to a clearer understanding of public climate discourse and the methodological challenges involved in analyzing it at scale.

1.1 Research Questions

(1) What are the main topics of discussion on Reddit about climate change, and how do these topics vary over time?
(2) Which topic modeling and sentiment analysis methods perform best in terms of coherence, interpretability, and scalability when applied to Reddit data?
(3) Can changes in sentiment within specific topics reveal shifts in public perception or reaction to real-world climate events?

2 Literature Review

2.1 Topic Modeling on Social Media Text

In their comparative study, Egger and Yu [4] apply LDA, NMF, Top2Vec, and BERTopic to short-text data from Twitter, examining their performance in extracting coherent and meaningful topics. While traditional models like LDA and NMF rely on statistical co-occurrence and linear factorization, embedding-based methods such as Top2Vec and BERTopic use semantic representations to capture deeper contextual relationships. The study finds that NMF and BERTopic consistently outperform LDA and Top2Vec in both topic coherence and interpretability. These results highlight the strength of matrix factorization and modern embedding-based methods for analyzing social discourse on social media platforms.

2.1.1 LDA. While LDA is traditionally applied to longer documents, recent work shows that with appropriate pre-processing, standard LDA remains a viable choice for analyzing social media comments.

Rohani et al. [14] present an LDA-based method to detect topics in aviation-related tweets. Analyzing over 90,000 posts, they extract five coherent topics whose temporal trends align with real events like the Paris attacks. By aggregating tweets daily and removing domain-specific stopwords, they achieve meaningful results without modifying standard LDA.

Negara et al. [12] apply LDA to Indonesian-language tweets in domains like economy and technology. They find that LDA produces more coherent and distinct topics than LSI.

Both studies demonstrate that standard LDA, when paired with preprocessing and domain-aware filtering can yield coherent and contextually relevant topic structures from social media data. Additionally, the probabilistic nature and word distributions produced by LDA make its results more transparent and interpretable, allowing easier manual validation and thematic labeling.

2.1.2 NMF. Non-Negative Matrix Factorization (NMF) is a matrix decomposition-based topic modeling method that represents documents and words in a lower-dimensional, non-negative space.

*All authors contributed equally to this project

Conference'17, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Alfajri et al. [1] applied NMF to 1,008 Instagram comments and evaluated topic quality using standard coherence metrics. The highest C_V score (0.60628) and U_MASS score (−13.97) were achieved without applying stemming or stopword removal. Notably, the C_V score peaked when using four topics, while the U_MASS score was highest at seven topics. The paper demonstrates NMF's robustness and effectiveness in identifying coherent and interpretable topics from noisy, user-generated content.

2.1.3 LSA. Latent Semantic Analysis (LSA) is a singular value decomposition (SVD)-based topic modeling technique that identifies co-occurring word patterns in large text datasets. Valdez et al. [15] employ LSA on over 41,000 words from the 2016 U.S. presidential debate transcripts, extracting 25 latent topics. The results show alignment with contemporaneous public policy concerns, such as taxes, healthcare, and foreign policy. The topics also correspond closely to the most googled political issues at the time, supporting the model's external validity.

2.1.4 BERTopic. Recent literature highlights BERTopic's effectiveness in extracting coherent themes from short, unstructured, and noisy social media text. Grootendorst [8] demonstrate this by applying BERTopic to datasets such as 20NewsGroups, BBC News, and a large corpus of tweets by Donald Trump. On the Trump dataset, BERTopic outperformed traditional models like LDA and NMF with a C_NPMI score of 0.066 (vs. -0.011 and 0.009 respectively), despite minimal pre-processing and high text variability. These results emphasize BERTopic's capacity to isolate semantically rich topics in low-context, opinion-heavy environments.

Bhuvaneswari et al. [2] extend BERTopic to a disaster response context, modeling 10,000 tweets filtered for disaster relevance. The study extracts 376 distinct topics, 196 of which directly map to observable impact themes like road closures, power outages, and infrastructure damage. BERTopic achieves coherence scores as high as 0.74 across categories, substantially higher than LDA (0.28–0.34) and LSA (0.21–0.37). The authors highlight the model's ability to disentangle overlapping narratives and structure chaotic tweet streams into distinct, interpretable themes.

Together, these studies underscore BERTopic's strength in high-noise and domain-specific contexts, where standard models fail to determine coherent topics.

2.2 Sentiment Analysis on Social Media Text

As social media continues to generate massive streams of public opinion, sentiment analysis has become central to understanding trends, behaviors, and reactions in real time. Sentiment analysis involves the identification and extraction of subjective information from text, providing insights into people's opinions on various topics [7].

2.2.1 VADER. To quantify sentiment in social media content, sentiment scoring tools are commonly used. Fu [7] utilizes VADER (Valence Aware Dictionary and Sentiment Reasoner) to understand the sentiment of long COVID tweets. The study grouped sentiment scores into three categories - positive, negative, and neutral - and found that 45.3% of tweets had negative sentiment. The study recommended that future research monitor sentiment spikes to better understand the events or topics driving these changes.

2.2.2 Models. Various machine learning models have been applied to sentiment analysis of social media data. Addressing the scale and complexity of this data, Elzayady et al. [5] implemented a sentiment classification system using Apache Spark, applying Naïve Bayes, Logistic Regression, and Decision Tree algorithms on over 1.5 million tweets. Their comparative analysis found that Naïve Bayes and Logistic Regression consistently outperformed Decision Trees in classification accuracy, with performance improving alongside dataset size. Moreover, the study emphasized the value of distributed processing, showing that increasing Spark cluster nodes led to faster execution—demonstrating both the effectiveness of model selection and the critical role of scalable infrastructure. Ramadhan et al. [13] utilized multinomial logistic regression to classify tweets of names of candidates in the Jakarta governor election. The findings indicated that the model could classify tweets with 74% accuracy, and they found that changing the number of fold in k-fold validation did not impact the accuracy. Cordero and Bustillos [3] found Naive Bayes outperformed other models in terms of accuracy, precision, recall, and F1 score. Given the consistent performance of both Logistic Regression and Naive Bayes in previous studies, these models will be adopted in the present analysis.

3 Dataset Description

The data used in this study is derived from the comments.csv file within the *Reddit Climate Change Dataset*, sourced from Kaggle. It contains a comprehensive archive of Reddit comments that include the terms "climate" and "change," collected up to September 1, 2022. The comments file includes 4,600,698 comments, approximately 4.11 GB of uncompressed data. Each row represents a single Reddit comment and includes ten columns of metadata, such as the comment ID, subreddit identifiers (both ID and name), timestamp of creation, permalink, and a numerical score (upvotes minus downvotes). The main textual content of each comment is stored in the *body* column, which serves as the primary input for this study.

A precomputed sentiment score is provided for each comment, ranging from -1 (very negative) to 1 (very positive). The dataset does not include usernames or personally identifiable information, preserving user anonymity. The chronological coverage and subreddit diversity allow for longitudinal and cross-community analysis of climate discourse.

4 Methodology

4.1 Distributed Computing Setup

All analyses were conducted in Jupyter Notebooks on Google Cloud Platform (GCP), with distributed data processing managed via Apache Spark on GCP Dataproc clusters. The modeling pipelines integrated Spark for large-scale data handling with Python-based topic modeling and sentiment analysis using libraries such as *pyspark*, *scikit-learn*, *sentence-transformers*, and *bertopic*. Coherence metrics were computed using a combination of *gensim*, *MulticlassClassificationEvaluator*, and custom evaluation functions.

GCP cluster configurations were selected based on the memory and compute needs of each model. Traditional models (LDA, NMF) ran efficiently on clusters with 2 vCPUs and 8GB RAM per node, using Spark MLlib and no partitioning. LSA required data partitioning into 64 partitions as well as a larger worker node (4

vCPUs, 16GB RAM). The BERTopic pipeline required a larger master node (16 vCPUs, 64GB RAM) to support the transformer-based embeddings, UMAP, and HDBSCAN, which were executed outside of Spark due to their CPU and memory demands. Sentiment analysis for logistic regression and Naive Bayes models also utilized 16 vCPUs, 64GB, and 8 partitions to accommodate hyperparameter tuning and ensure efficient model training.

4.2 Data Preprocessing

The pre-processing pipeline is inspired by the workflow presented in [11], which emphasizes cleaning, tokenization, stopword removal, lemmatization, and word vector representation.

Due to the memory constraints of our GCP Dataproc cluster, the full dataset could not be processed without causing node failures. Through iterative experimentation, we determined that sampling 20% of the data allowed for stable and efficient execution of most models while maintaining variety in language and topics.

4.2.1 Data Cleaning and Tokenization. The Reddit comment text was first converted to lowercase and stripped of URLs, non-alphabetic characters, and excess whitespace. This normalization ensures consistency and removes irrelevant symbols. Tokenization was then applied to split the cleaned text into individual words. All punctuation and numerical characters were removed at this stage.

4.2.2 Stopword Removal and Text Normalization. The default NLTK English stopword list, augmented with a custom list for Reddit-specific, web-related, and formulaic artifacts (e.g., mathematical symbols, CSS tags, metadata tokens) was used. Tokens shorter than three characters were also excluded to minimize sparsity.

Although lemmatization is often used to group forms of words under a shared lemma, we found that applying lemmatization reduced topic coherence. This reduction is likely due to the loss of contextually specific word forms, which are crucial in topic modeling algorithms like LDA and NMF that rely on fine-grained frequency distributions. Therefore, lemmatization was omitted from the main topic modeling pipeline. However, for LSA, which is more sensitive to high-dimensional inputs and computational cost, lemmatization was retained to aggressively reduce feature space and enable feasible execution of singular value decomposition (SVD) within a PySpark row matrix structure.

4.2.3 Count Vectorization. Count Vectorization was used to convert each document into a vector of word counts. Only words that appeared in at least 50 documents and in less than 80% of the total documents were kept. This helped remove very rare words and very common ones, both of which add noise and reduce model quality.

4.2.4 Sentiment Analysis. In the dataset, there is a column "sentiment" which contains VADER sentiment scores ranging from -1 (most negative) to 1 (most positive). In order to classify sentiments, we first dropped the rows that did not have a sentiment value, and a total of 57,131 null values were removed. We then added a new column with categorical labels for sentiment, where we defined the custom thresholds as positive is greater than 0.35, negative is less than -0.35, and otherwise it is neutral.

The resulting vocabulary included 26,919 unique tokens across the sampled dataset. For BERTopic, however, no preprocessing was performed. This is because BERTopic uses transformer-based embeddings which operate on raw, unprocessed text to preserve contextual meaning. Preprocessing removes all context and would reduce model efficiency. Hence raw text is used as input for BERTopic.

4.3 Topic Modeling

4.3.1 Models.

Latent Dirichlet Allocation (LDA). LDA is a widely used probabilistic model for unsupervised topic modelling. It assumes that each document in a collection is a mixture of underlying topics, and each topic is represented by a specific distribution over words. The goal of LDA is to uncover these hidden topic structures by analyzing how words are distributed across the documents.

LDA was applied to a 20% stratified sample of the Reddit climate change dataset, comprising approximately 920,000 comments. Data pre-processing followed the steps outlined in that section, to ensure high-quality input for topic inference.

The LDA model was trained using PySpark's distributed `LDA` implementation with $k = 10$ topics and 20 iterations, initialized with a fixed random seed for reproducibility.

After training, we extracted topic descriptors by listing the most probable words for each topic. We also computed topic distributions per document to identify dominant themes.

We explored several LDA configurations to balance topic interpretability and model performance. Our baseline setup with k=10 topics and `minDF=50` produced clear, distinct topics that aligned well with real-world themes. Lowering the `minDF` threshold introduced noisier topics with overlapping content, while increasing the number of topics (k) beyond 10 led to more fragmented and less interpretable results. Additional experiments with lemmatization and more restrictive document frequency settings did not yield improvements and, in some cases, reduced topic quality due to the informal nature of Reddit language. Based on these observations, we selected k=10 and `minDF=50` as the most robust and interpretable configuration for downstream analysis.

Non-Negative Matrix Factorization (NMF). NMF is a matrix factorization technique commonly used for topic modelling. It factorizes a non-negative document-term matrix $\mathbf{X} \in \mathbb{R}_{+}^{m \times n}$ into two lower-dimensional non-negative matrices: $\mathbf{W} \in \mathbb{R}_{+}^{m \times k}$, representing document-topic associations, and $\mathbf{H} \in \mathbb{R}_{+}^{k \times n}$, representing topic-term distributions, such that: $\mathbf{X} \approx \mathbf{WH}$.

Each topic corresponds to a sparse, interpretable distribution over words (rows of $\mathbf{H}$), and each document is modeled as a weighted combination of these topics (rows of $\mathbf{W}$).

We applied NMF to the same 20% sample of the Reddit climate change dataset to ensure a direct comparison with LDA. Preprocessed tokens were pulled from Spark into Python, concatenated into strings, and vectorized using a `TfidfVectorizer` with `minDF=50`, `maxDF=0.8`, and a domain-aware stopword list. This produced a sparse TF-IDF matrix $\mathbf{X}$ suitable for decomposition. The `NMF` model from `scikit-learn` was trained with $k = 10$ components, `init="nndsvda"`, a fixed random seed, and up to 200 iterations. After training, dominant topics for each comment were inferred from $\mathbf{W}$, while top keywords per topic were extracted from $\mathbf{H}$.

We reused the same sample, tokenization, and document frequency thresholds as in LDA to maintain consistent input features and enable an apples-to-apples comparison across models. This ensured differences in results were due to modeling approach rather than pre-processing differences.

Latent Semantic Analysis (LSA). Latent Semantic Analysis (LSA) is a matrix decomposition technique that uncovers latent semantic structure in text data by projecting high-dimensional term-document matrices into a lower-dimensional topic space using Singular Value Decomposition (SVD). Given a matrix $A \in \mathbb{R}^{m \times n}$ of term frequencies, LSA computes $A = U\Sigma V^\top$, where U represents documents in topic space, Σ is a diagonal matrix of singular values, and $V^\top$ contains topics represented as weighted combinations of words Kalepalli et al. [10].

We applied LSA to a 20% sample of the Reddit climate change dataset (920,000 comments), using the same preprocessing pipeline as other models. To ensure computational feasibility, we limited the vocabulary size during vectorization by setting `vocabSize=10000` and `minDF=100` in `CountVectorizer`. This reduced dimensionality and controlled memory usage during matrix construction. Additionally, LSA's topic components tend to be less sparse, sometimes blending themes. To reduce this, we applied lemmatization for LSA to reduce vocabulary redundancy and focus dimensionality on semantically relevant terms.

Due to the lack of sparse matrix support in PySpark's `computeSVD`, the term-document matrix was converted to an RDD of dense vectors. We partitioned the dataset into 64 partitions before SVD computation to improve task scheduling and avoid executor overload. Despite these optimizations, attempting LSA on the full dataset consistently led to out-of-memory crashes and executor failures due to the `RowMatrix` exceeding supported size. Through experimentation, we found 20% to be the largest sample size that allowed stable execution under our GCP cluster constraints.

We selected $k = 7$ components based on qualitative coherence and runtime constraints. To improve interpretability, we used a LLM (via Groq) to generate topic labels based on the top words in each LSA component. This is further explained below.

BERTopic. BERTopic uses transformer-based embeddings to represent documents semantically, applies dimensionality reduction and clustering to form dense topic groups, and then uses class-based TF-IDF (c-TF-IDF) to extract important and interpretable keywords for each topic [9]. Given computational constraints (cluster with n2-standard-16 master node), a lightweight embedding model with strong performance on the MTEB leaderboard was selected: all-MiniLM-L6-v2. Multiple submodels were incorporated in line with BERTopic best practices (Maarten Grootendorst, 2023):

- **Representation Models**: An ensemble of KeyBERT and Part-of-Speech (POS) filtering was used as a representation model. The combination of these two models makes sense: KeyBERT ensures semantic relevance of selected words while POS filtering ensures syntactic quality, e.g., filters for nouns, which are more informative for topic labels. *As an additional extension, the Groq (LLM) API was used to generate natural language topic labels to improve human interpretability.*
- **Vectorization**: Since BERTopic does not preprocess text natively, CountVectorizer was applied to remove English stopwords and ignore infrequent terms.
- **Dimensionality Reduction**: UMAP was applied to project the high-dimensional embeddings into a lower-dimensional space and facilitate clustering.
- **Clustering**: HDBSCAN was used for clustering with a min_cluster_size of 50, controling the granularity of topic formation by avoiding overly small clusters.

To choose the optimal hyperparameters for clustering and HDBSCAN, the created pipeline was first run on a subset of 10,000 comments. After that, the BERTopic model was scaled to 100,000 Reddit comments, using 16 vCPUs and 64GB of RAM. *Note: Using PySpark partitioning for BERTopic was not possible, as BERTopic relies on transformer embeddings and clustering that require global context. In addition to that, Clustering (UMAP + HDBSCAN) needs access to all embeddings at once, not partitioned data.*

4.4 Sentiment Analysis

4.4.1 Models. For multi-class classification for sentiment analysis the three classes are relatively balanced: 36.89% of the comments are negative, 36.27% positive, and 26.82% are neutral.

VADER. VADER is a rule-based sentiment analysis tool specifically designed for social media data, and it is part of the natural language toolkit (nltk) in Python. It utilizes a dictionary of words mapped to sentiments, in addition to analyzing capitalization, punctuation, and emoticons which are common in social media text. It returns a sentiment score between -1, the most negative, and 1, the most positive. The sentiment column in our dataset was calculated using VADER. [7]

Logistic Regression. Logistic Regression is a supervised machine learning model used to predict the probability of a binary or multi-class outcome based on input features. It is trained using stochastic gradient descent and cross-entropy loss. [13]

For this study, the data was preprocessed as described above, and then we use Multiclass Logistic Regression with the model classifying comments as positive, negative, or neutral. The input features for the model are derived from the output of the TF-IDF transformation.

Hyperparameter tuning was performed for three parameters: `regParam`, `elasticNetParam`, and `maxIter`. The regularization parameter (`regParam`) is the L2 regularization strength. The elastic net parameter (`elasticNetParam`) is the mixed ratio between L1 and L2 regularization where 0.0 is pure L2 regularization and 0.5 is a balance between the two. The maximum iterations (`maxIter`) is the number of iterations for algorithm convergence.

A train-validation split was applied which evaluates each combination of parameters once. The parameter grid included two values for each parameter, resulting in a total of eight combinations. Model performance was evaluated using the F1 score, which balances precision and recall. The best model was selected based on the highest train-validated F1 score. The best performing model, based on the highest train-validation F1 score, had the following hyperparameters: `regParam` = 0.0, `elasticNetParam` = 0.0, and `maxIter` = 50.

Naive Bayes. Naive Bayes is a generative probabilistic classifier that utilizes Bayes Theorem and assumes independence between features. Bayes' Theorem is used to calculate the posterior probability of a class given input features:

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

where A and B are events, and $P(B) \neq 0$.

In this study, we ran the multinomial Naive Bayes model, which is well-suited for discrete features like term frequencies from TF-IDF, which was our input feature. We utilized a 3-fold cross validation and a parameter grid to tune the smoothing parameter. Smoothing is used to avoid zero probabilities for unseen features in the test data.

Model performance was evaluated using the F1 score, which balances precision and recall. The optimal model, selected based on the highest cross-validated F1 score, used a smoothing parameter of 2.0.

4.5 Flexibility of proposed solution

The combined topic modeling and sentiment analysis pipeline is adaptable to other social media datasets, given the input text is clean and well-formatted (to ensure reliable performance from BERTopic), written in English, and similar in style to Reddit discourse, i.e. informal language and relatively short text units. Under these conditions, the approach can generalize well to platforms such as Twitter, YouTube, or online forums. With sufficient computational resources and GPU access, the pipeline can scale to larger datasets.

5 Numerical Results and Discussion

5.1 Topic Modeling Evaluation

5.1.1 Evaluation Metrics. Two types of metrics were used for the sentiment analysis models: (I) Model Evaluation Metrics and (II) Distributed Computing Metrics.

- **C_V Coherence** C_V Coherence measures semantic similarity of top topic words using NPMI and cosine similarity. Higher values (closer to 1) indicate better coherence. Computed via Gensim's `CoherenceModel()`.
- **U_Mass Coherence** U_Mass Coherence evaluates topic coherence using document co-occurrence counts from the training corpus. It penalizes topics with word pairs that rarely co-occur. Lower scores (closer to 0) are better.

$$C_{\text{UMass}} = \frac{1}{|W|} \sum_{i=2}^{|W|} \sum_{j=1}^{i-1} \log \frac{D(w_i, w_j) + \epsilon}{D(w_j)}$$

  where $D(w_i, w_j)$ is the document count containing both w_i and w_j.
- **C_NPMI Coherence** C_NPMI extends U_Mass by normalizing PMI scores between topic words to [-1, 1], allowing comparison across models. Higher values (closer to 1) indicate better topic coherence.

$$\text{NPMI}(w_i, w_j) = \frac{\log\left(\frac{P(w_i, w_j)}{P(w_i)P(w_j)}\right)}{-\log P(w_i, w_j)}$$

- **Topic Imbalance** Measures the skewness of topic sizes i.e., how evenly documents are distributed across topics. It is computed as the ratio of the largest to the smallest topic (excluding outliers). The topic imbalance score ranges from $[1, \infty)$, while a lower score indicates a better balance.

$$\text{Imbalance} = \frac{\max(\text{topic sizes})}{\min(\text{topic sizes})}$$

- **Topic Diversity** Measures the proportion of unique words across all topic top-n word lists. Higher diversity (closer to 1) suggests less redundancy in topic descriptions.

$$\text{Diversity} = \frac{\text{Number of unique words across topics}}{\text{Total number of words across topics}}$$

- **Runtime (min)** Total time taken from embedding to topic labeling.
- **Dataset Size (GB)** Models were run on subsets of at least 100,000 comments. Larger subsets were used when feasible.
- **No. of Partitions** Data partitioning was applied to LDA and LSA. As explained in section 4.3.1, data partitioning was not possible for BERTopic and NMF.
- **No. of vCPUs** As topic model memory needs differ strongly between models, they were run on separate clusters with different master node configurations including the number of virtual CPUs in use.
- **RAM** As topic model memory needs differ strongly between models, they were run on separate clusters with different master node configurations including memory (RAM).

5.1.2 ***Model Comparison.*** The following table shows the results for each topic modeling method across evaluation metrics.

Table 1: Comparison of Topic Modeling Methods Across Evaluation Metrics

Metric	LDA	NMF	LSA	BERTopic
CV Coherence	0.51	**0.62**	0.50	0.55
UMass Coherence	-2.19	-2.16	**-1.81**	-6.16
C_NPMI Coherence	0.04	**0.13**	0.03	0.02
Topic Imbalance	**10.66**	22.27	27.47	50.30
Topic Diversity	0.64	**0.90**	0.51	0.74
Runtime (min)	15	**3**	46	17
Dataset Size (GB)	0.47	0.47	0.47	0.10
No. of Partitions	1	1	64	/
Master vCPUs	2	2	4	16
Master RAM	8GB	8GB	16GB	64GB
Worker vCPUs	2	2	4	2
Worker RAM	8GB	8GB	12GB	8GB

LSA performed the weakest overall- it produced overlapping topics with low diversity and required significantly more runtime and memory due to its reliance on dense matrix operations, making it impractical for large-scale, noisy datasets like Reddit.

BERTopic achieved a higher CV coherence (0.55) and topic diversity (0.74) score than LDA, its significantly lower U_Mass (−6.16), near-zero C_NPMI (0.02), and extremely high topic size imbalance (50.30) suggest that it struggled to generate well-separated or consistently interpretable clusters on this dataset. Additionally, BERTopic

required substantially more computational resources (16 vCPUs, 64GB RAM) and complex model tuning, whereas LDA performed well with much more modest infrastructure.

Because we are working with social media text, which is often noisy and informal, we cannot fully rely on automatic evaluation metrics alone. Human interpretability is another critical dimension that cannot be entirely captured by numbers, but it is essential to assess whether the topics are meaningful, distinct, and coherent in real-world contexts. Both LDA and NMF uncovered important themes in the Reddit climate discourse; however, LDA offered more distinct and interpretable topics, clearly separating areas such as scientific discourse, U.S. politics, veganism, and lifestyle activism. In contrast, NMF showed notable topic overlap, particularly among opinion-driven categories like general reflections, belief systems, and climate skepticism. This suggests that while NMF may capture subtle emotional tone, LDA provides better thematic variety and structure across climate-related domains. Therefore, we chose to proceed with LDA for the final analysis due to its stronger balance between topic coherence, diversity, and interpretability.

5.1.3 *Analysis of Results.*

Key Topics Identified. The LDA model uncovered ten coherent and thematically distinct topics that collectively capture the multifaceted nature of climate discourse on Reddit. Several topics are centered around policy and politics, such as Topic1 (*Political Opinions & Division*) and Topic8 (*U.S. Political Figures*), which highlight ideological polarization and the influence of individual leaders. Scientific framing is also prominent, as seen in Topic3 (*Scientific Discourse*) and Topic5 (*Carbon, Energy & Emissions*), where users reference empirical data, energy systems, and emissions science. Notably, the model captured culturally specific perspectives such as Topic4 (*Gender, Life & Climate*) and Topic6 (*Veganism & Lifestyle Activism*), illustrating how climate change intersects with identity, lifestyle, and social values. Finally, broad and introspective commentary appears in Topic 9 (*General Reflections & Concerns*), representing the affective and moral dimension of user engagement. The interpretability and diversity of these topics suggest that the model successfully captured both global themes and niche subcultures present in the Reddit climate corpus.

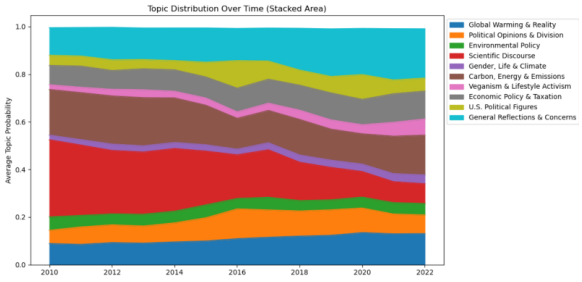

Figure 1: Topic Distribution over time

Topic Trends Over Time. The stacked area plot in Figure 1 illustrates how the prominence of LDA-derived topics evolved over time. *Scientific Discourse* dominated Reddit climate discussions in the early years (2010–2015), with users focused on evidence, data, and rebutting denialism. From 2016 onward, this theme declined as the conversation broadened. By 2018, *General Reflections & Concerns* became the most dominant topic, reflecting increased emotional, moral, and societal engagement with climate change. To investigate these shifts more precisely, we ranked topics within each year based on their average probability. Results confirm the transition from *Scientific Discourse* (2010–2017) to *General Reflections & Concerns* (2018–2022) as the leading theme. Meanwhile, topics like *Carbon, Energy & Emissions* and *Environmental Policy* maintained consistent presence, suggesting steady interest in technical and policy-based solutions. Politically oriented topics—including *Political Opinions & Division* and *U.S. Political Figures*—gained momentum post-2016, mirroring real-world polarization. Finally, lifestyle and social themes such as *Veganism & Lifestyle Activism* and *Gender, Life & Climate* trended gradually upward, indicating broader engagement with intersectional climate discourse.

To further investigate temporal topic dynamics, we identified the five largest year-over-year spikes and drops in topic prevalence. Notable spikes included *U.S. Political Figures* in 2016 and 2020—both U.S. election years—and *General Reflections & Concerns* in 2018, reflecting rising emotional engagement. The most significant drops occurred in *Scientific Discourse*, which declined steadily from 2015 to 2018, indicating a shift away from scientific narratives. We also computed topic volatility by measuring the standard deviation of each topic's annual prevalence. *Scientific Discourse* emerged as the most volatile theme, while *Environmental Policy* and *Gender, Life & Climate* were the most stable. Lastly, by comparing topic distributions before and after 2020, we observed that the COVID-19 pandemic triggered a decline in *Scientific Discourse* and a rise in more introspective and policy-oriented topics such as *General Reflections & Concerns*, *Economic Policy & Taxation*, and *Veganism & Lifestyle Activism*. These trends underscore the responsiveness of public climate discourse to global events and shifting societal priorities.

5.2 Sentiment Analysis Evaluation

5.2.1 ***Evaluation Metrics.***

Two types of metrics were used for the sentiment analysis models: (I) Model Evaluation Metrics and (II) Distributed Computing Metrics.

Model Evaluation Metrics:

- **Accuracy**: The proportion of correctly classified comments:

$$Accuracy = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{FP} + \text{TP} + \text{FN}}$$

  where TN = True Negatives, TP = True Positives, FP = False Positives, FN = False Negatives.
- **Precision**: The ratio of true positives to total predicted positives:

$$Precision = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- **Recall**: The ratio of true positives to total actual positives:

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **F1 Score**: The balance of both precision and recall:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Training Runtime (sec)**: The time taken to fit the model with the training data, excluding preprocessing time.
- **Testing Runtime (sec)**: The time taken to transform the model with the testing data, excluding preprocessing time.

Distributed Computing Metrics:

- **Training No. Comments**: 727,553. 80% of data was used for training.
- **Testing No. Comments**: 182,427. 20% of data was used for testing.
- **No. of Partitions**: 8 partitions for parallel processing and scalability.
- **No. of vCPUs**: 16 virtual CPUs, based on a cluster of n2-standard-4 machine with 1 master and 3 worker nodes.
- **RAM (GB)**: 64 GB. 16 GB per node and 4 nodes in total.

Table 2: Comparison of Sentiment Analysis Methods

Metric	Logistic Regression	Naive Bayes
Accuracy	**0.7499**	0.554
Precision	**0.7533**	0.607
Recall	**0.7499**	0.554
F1 Score	**0.7513**	0.559
Training Runtime (sec)	1536.44	**596.71**
Testing Runtime (sec)	0.07	**0.06**

5.2.2 ***Model Comparison.*** The comparison between Logistic Regression and Naive Bayes for sentiment analysis is presented in Table 2. The performance metrics clearly indicate that Logistic Regression outperforms Naive Bayes for all performance metrics, making it the more suitable choice for multi-class sentiment classification of Reddit comments. However, the one advantage of Naive Bayes is the training runtime, which is approximately 60% faster.

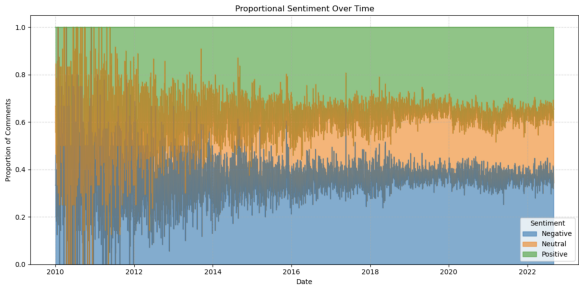

Figure 2: Ratio of Reddit comment sentiment over time.

5.2.3 ***Sentiment Analysis Over Time.*** To analyze how public sentiment of climate change evolves over time, we analyzed the sentiment by date. The number of comments steadily increases, with spikes often aligning with major climate events. These spikes tend to reflect higher negative sentiment, suggesting concern and frustration.

To better understand the distribution of sentiments over time, we then normalized the daily sentiment counts by the total number of comments per day. The resulting stacked area chart shown in Figure 2 shows significant decrease in neutral sentiment from 2010, while both negative and positive sentiment have steadily increased, indicating that discourse around climate change is growing more polarized over time.

We then calculated the dominant sentiment for each day based on the highest comment count. Negative sentiment was dominant 2,229 days, positive sentiment was dominant 2,148 days, and neutral sentiment was only dominant 248 days. This suggests negative sentiment is overall the most prominent, though positive sentiment is almost as dominant. The relatively low dominance of neutral sentiment further underscores the highly charged nature of climate change conversations, where opinions are often strongly held.

5.2.4 ***Sentiment Analysis of Comment Spikes.*** We detected spikes in daily comment volume using z-scores to identify days with unusually high comment counts. These spikes were always linked to real-world climate events, as shown in Table 3, which lists the top 8 comment spikes, sentiment ratios, and the corresponding events.

Table 3: Comparison of Sentiment Analysis of Comment Spikes

Date	No. Comments	Pos Ratio	Neg Ratio	Event
24/9/19	2432	0.31	0.41	Global Climate Strike
2/6/17	2004	0.42	0.32	US Leaves Paris Agreement
25/9/19	1885	0.31	0.41	Global Climate Strike
27/9/19	1471	0.31	0.40	Global Climate Strike
26/9/19	1445	0.35	0.40	Global Climate Strike
28/9/19	1384	0.32	0.39	Global Climate Strike
9/8/21	1382	0.36	0.39	Severe Heatwave in US
10/11/16	1324	0.36	0.40	Trump Elected US President

5.3 Combined Topic-Sentiment Trends

We combined the outputs of LDA (selected for its balance of coherence and interpretability) and Logistic Regression (the best performing model for sentiment analysis) by predicting a dominant topic and sentiment label for each Reddit comment. Merging these results, we analyze how sentiment differs across topics and evolves over time, offering insights into public climate discourse. The topic labels generated using an LLM representation model based on the topics produced by LDA along with their corresponding keywords, is provided in Table 4 in the Appendix.

5.3.1 ***Sentiment Distribution Across Topics:*** The grouped bar chart 3 shows how sentiment varies across different discussion topics based on the number of Reddit comments labeled as Positive, Neutral, or Negative. From the graph we can conclude that **Understanding the World's Reality** and **How Scientists View the World** have the highest overall engagement, with a large number of comments across all sentiment classes. Notably, negative sentiment dominates both topics, suggesting these may be areas of controversy or disagreement. **Trump's Anti-World Beliefs** shows a strong skew toward negative sentiment, with negative comments more than double the positive.

Expressing Political Party Views is one of the few topics where positive sentiment dominates, suggesting affirmation, support, or pride in one's political alignment. However, **US-China**

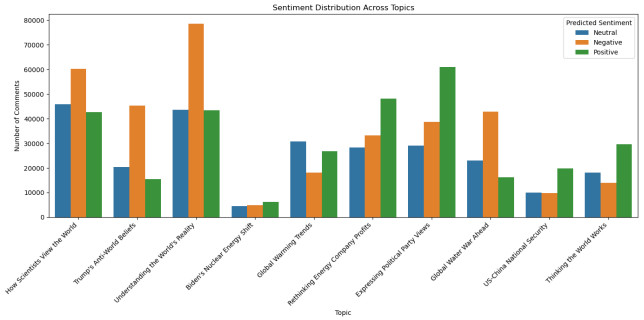

Figure 3: Sentiment Distribution Across Topics

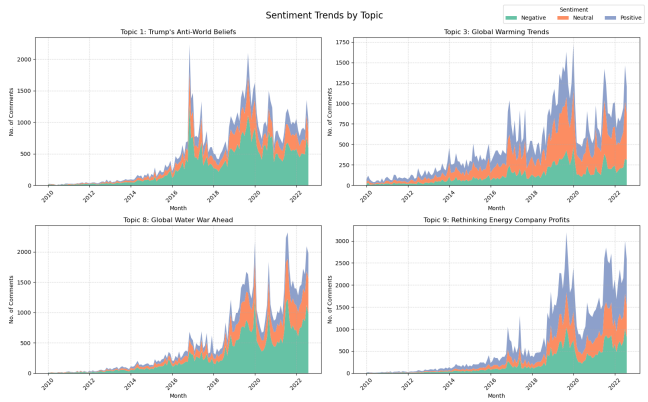

Figure 4: Sentiment Trend for Relevant Topics over Time

National Security and **Thinking the World Works** show relatively lower total volume but lean more positively. Additionally, **Global Water War Ahead** is heavily skewed negative, consistent with concern-driven discourse on climate-related issues.

Therefore Reddit users are generally more likely to post when they disagree about a topic. This trend is visible in topics with higher negative comment volumes, reinforcing the idea that negativity is often more vocal online.

5.3.2 ***Sentiment Trends over Time per Topic:*** The stacked area plots in figure 4 show how Reddit sentiment has evolved over time across key topics. Notable trends include:

- Consistent Volume Growth: All topics show a clear rise in comment volume starting around 2016–2017, reflecting heightened public discourse.
- Topic 1: Trump's Anti-World Beliefs: Peaks around 2016 and 2020 align with U.S. presidential election cycles, showing a high volume of negative sentiment.
- Topic 3: Global Warming Trends: Displays a more balanced sentiment mix, but still skewed negative. Discussion intensifies post-2016, reflecting growing climate awareness.
- Topic 8: Global Water War Ahead: Predominantly negative steady rise, reflecting concerns about water scarcity.
- Topic 9: Rethinking Energy Company Profits: An increase in positive sentiment in recent years suggests shifts in perception around clean energy.
- Prevalence of Negative Sentiment: Across most topics, negative sentiment tends to dominate. This reflects an observed tendency in online platforms where users are more likely to speak out when they disagree — negative opinions often drive stronger engagement than approval.

6 Conclusion and Future Work

6.1 Key Findings

This study analysed public discourse on climate change through Reddit comments by combining topic modeling and sentiment analysis in a distributed computing framework. After comparing four different topic modeling approaches, LDA was selected due to its computational efficiency and interpretability, ignoring the better performance of other models with regards to some of the chosen coherence metrics. Logistic Regression and Naive Bayes models were compared for sentiment analysis, with Logistic Regression outperforming Naive Bayes in all evaluation metrics. Sentiment analysis over time revealed that both positive and negative sentiment grew, with neutral decreasing, indicating a growing polarization in climate change discourse. Spikes in comment volume were consistently linked to major real-world climate events. Overall, this research demonstrates the potential of combining scalable NLP techniques with sentiment analysis to better understand public discourse around climate change on social media.

6.2 Limitations

6.2.1 Bias in Data. Reddit users are not demographically representative of the general public—nearly 40% are aged 18–35, two-thirds are male, and about half are based in the U.S.—which limits the generalization of findings. [6]

6.2.2 Computational Constraints. Due to limited resources, particularly the lack of GPU access, topic modeling was restricted to data subsets, and a lightweight embedding model was used for BERTopic. More advanced models could have improved topic coherence but were not feasible within the available infrastructure.

6.3 Future Research

6.3.1 Cross-Subreddit Comparative Analysis. The current analysis treats Reddit as a unified corpus, but future work could compare topic distributions and sentiment patterns across subreddits. A cross-subreddit comparison would allow for identifying how focus and emotional tone differ across audience segments, enabling more targeted insight into the discourse.

6.3.2 Integration of External Knowledge. Integrating external data sources such as news articles or real-world events could provide valuable additional context for interpreting topic dynamics. An extended pipeline could incorporate named entity recognition and event linking to systematically map Reddit discussions to real-world developments.

6.3.3 Alternative Sentiment Tools. This study used VADER sentiment scores; future research could compare results using transformer-based models (e.g., BERT) or APIs like Hugging Face or Google Cloud NLP to assess accuracy and performance.

References

[1] Alfajri Alfajri, Donny Richasdy, and Muhammad Bijaksana. 2022. Topic Modelling Using Non-Negative Matrix Factorization (NMF) for Telkom University Entry Selection from Instagram Comments. *Journal of Computer System and Informatics (JoSYC)* 3 (09 2022), 485–492. doi:10.47065/josyc.v3i4.2212

[2] A Bhuvaneswari, M Kumudha, et al. 2024. Topic Modeling Based Clustering of Disaster Tweets Using BERTopic. In *2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon)*. IEEE, 1–6.

[3] Jorge Cordero and José Bustillos. 2021. Sentiment analysis based on user opinions on twitter using machine learning. In *International Conference on Applied Technologies*. Springer, 279–288.

[4] Roman Egger and Joanne Yu. 2022. A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts. *Frontiers in Sociology* 7 (2022). doi:10.3389/fsoc.2022.886498

[5] Hossam Elzayady, Khaled Badran, and Gouda Salama. 2018. Sentiment Analysis on Twitter Data using Apache Spark Framework. In *2018 International Conference on Computer Engineering Systems (ICCES)*. 171–176. doi:10.1109/ICCES.2018.8639195

[6] Exploding Topics. 2024. Reddit User Statistics: How Many People Use Reddit in 2024? https://explodingtopics.com/blog/reddit-users Accessed: 2025-05-04.

[7] YB Fu. 2022. Investigating public perceptions regarding the long COVID on Twitter using sentiment analysis and topic modeling. *Med Data Min* 5, 4 (2022), 24.

[8] Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. arXiv:2203.05794 [cs.CL] https://arxiv.org/abs/2203.05794

[9] Maarten Grootendorst. 2024. BERTopic: The Algorithm. https://maartengr.github.io/BERTopic/algorithm/algorithm.html. Accessed: 2025-05-05.

[10] Yaswanth Kalepalli, Shaik Tasneem, Pasupuleti Durga Phani Teja, and Suneetha Manne. 2020. Effective comparison of LDA with LSA for topic modelling. In *2020 4th International conference on intelligent computing and control systems (ICICCS)*. IEEE, 1245–1250.

[11] Amna Meddeb and Lotfi Ben Romdhane. 2022. Using Topic Modeling and Word Embedding for Topic Extraction in Twitter. *Procedia Computer Science* 207 (2022), 790–799. doi:10.1016/j.procs.2022.09.134 Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 26th International Conference KES2022.

[12] Edi Surya Negara, Dendi Triadi, and Ria Andryani. 2019. Topic modelling twitter data with latent dirichlet allocation method. In *2019 International Conference on Electrical Engineering and Computer Science (ICECOS)*. IEEE, 386–390.

[13] WP Ramadhan, STMT Astri Novianty, and STMT Casi Setianingsih. 2017. Sentiment analysis using multinomial logistic regression. In *2017 International Conference on Control, Electronics, Renewable Energy and Communications (ICCREC)*. IEEE, 46–49.

[14] Vala Ali Rohani, Shahid Shayaa, and Ghazaleh Babanejaddehaki. 2016. Topic modeling for social media content: A practical approach. In *2016 3rd international conference on computer and information sciences (ICCOINS)*. IEEE, 397–402.

[15] Danny Valdez, Andrew C Pickett, and Patricia Goodson. 2018. Topic modeling: latent semantic analysis for the social sciences. *Social Science Quarterly* 99, 5 (2018), 1665–1679.

A Appendix

Topic Name	Top Keywords
US-China National Security	bernie, national, said, trump, government, bot, china, security, energy, states
Trump's Anti-World Beliefs	trump, think, white, years, said, world, believe, time, anti, know
How Scientists View the World	science, know, think, real, scientists, things, world, way, say, scientific
Global Warming Trends	warming, global, years, earth, ice, temperature, data, carbon, time, emissions
Thinking the World Works	time, world, really, think, good, way, things, well, work, lot
Expressing Political Party Views	think, party, want, government, vote, political, things, trump, say, way
Understanding the World's Reality	think, world, believe, know, years, really, way, time, things, say
Biden's Nuclear Energy Shift	nuclear, biden, energy, power, solar, wind, coal, joe, plants, fossil
Global Water War Ahead	water, world, global, food, population, countries, years, due, war, human
Rethinking Energy Company Profits	carbon, money, emissions, think, need, energy, oil, world, companies, way

Table 4: Topic modeling using LDA and LLM representation Model

Dataset link: https://www.kaggle.com/datasets/pavellexyr/the-reddit-climate-change-dataset