Sentiment Analysis of Tweets to Discover Regional and Demographic Trends

Vishnu Alla
College of Computing
Michigan Technological University
Email: valla1@mtu.edu
Team: Data Smugglers

Vineeth Karjala
College of Computing
Michigan Technological University
Email: vkarjala@mtu.edu
Team: Data Smugglers

Surya Vakkalagadda
College of Computing
Michigan Technological University
Email: svakkala@mtu.edu
Team: Data Smugglers

***Abstract*—Sentiment analysis is essential for understanding public opinion across digital platforms. In this project, we evaluated multiple machine learning and deep learning models for sentiment classification using a labeled Twitter dataset from Kaggle. After preprocessing the tweets, we extracted features using Term Frequency–Inverse Document Frequency (TF-IDF) and Bag-of-Words (BoW). Models including Support Vector Machines (SVM), Random Forests, Naive Bayes, Feedforward Neural Networks, and LSTMs were trained and evaluated using F1 score. Although the Feedforward Neural Network achieved the highest F1 score, SVM with TF-IDF offered a strong balance of performance and efficiency, making it the preferred model. We applied the selected SVM to two additional datasets, one on airline sentiment and another on public reactions to Squid Game with geographic metadata. Visualization techniques helped reveal sentiment patterns across topics and regions. Our findings demonstrate the practicality and scalability of SVM based sentiment analysis in real-world applications.**

***Index Terms*—sentiment analysis, twitter, machine learning, natural language processing, data mining**

I. Introduction

With the growth of social media, platforms like Twitter (now X) have become key sources for analyzing public sentiment across diverse topics. Given the high volume of short, informal and emotionally expressive content, Twitter provides a rich dataset for sentiment analysis.

As unstructured text data grows, using machine learning (ML) and deep learning (DL) techniques has become important for building scalable sentiment classification systems. Classical ML models such as Support Vector Machines (SVM), Random Forests, and Naive Bayes, when combined with feature extraction methods like Bag-of-Words (BoW) and TF-IDF offer efficient solutions. In parallel, DL models like Feedforward Neural Networks and Long Short-Term Memory (LSTM) networks capture complex linguistic patterns.

This project evaluates a range of ML and DL models on a labeled Twitter dataset to identify the most effective model using F1 score as the performance metric. These models can help in identifying sentiment trends over time, detecting shifts in public perception, and even uncovering regional sentiment patterns when augmented with temporal or geographic metadata. These insights can be helpful in fields like marketing and policymaking to social research.

To demonstrate real-world applicability, the selected model was applied to two additional Twitter datasets: one analyzing airline-related sentiment, and another exploring geographic sentiment surrounding the Netflix series Squid Game. Visualizations using Power BI revealed meaningful trends in public opinion across domains and regions.

The dual objectives of this study are to identify an optimal sentiment classification model and to illustrate the broader utility of sentiment analysis in mining large-scale public opinion.

II. Related Works

Sentiment analysis has been a central topic in natural language processing (NLP) and data mining, evolving over the past decade. Initial approaches were largely lexicon-based or rule-driven, relying on predefined sentiment scores assigned to words or phrases [5]. While useful in early applications, these methods did not have the flexibility to handle informal, ambiguous language, particularly common on platforms like Twitter (now X).

With the shift toward supervised learning, traditional machine learning models such as Support Vector Machines (SVM) [1], Naive Bayes, and Random Forests emerged as effective tools for sentiment classification. These models, when paired with text vectorization techniques like Bag-of-Words (BoW) and Term Frequency–Inverse Document Frequency (TF-IDF), allowed for structured representation of text.

More recently, the field has seen substantial progress through deep learning with the introduction of neural network architectures capable of capturing syntactic and semantic patterns. Feedforward Neural Networks and Long Short-Term Memory (LSTM) networks [2] have proven effective for modeling sequential and contextual information in the text. However, their computational demands and sensitivity to hyperparameter tuning often limit their practicality in resource-constrained or real-time applications.

Current research increasingly focuses on leveraging pre-trained language models such as Word2Vec [3], GloVe [4], and transformer-based architectures like BERT, which deliver state-of-the-art results in sentiment classification tasks. Despite their impressive accuracy, these models introduce trade-offs re-

TABLE I
DATASET DESCRIPTION

Dataset	Usage	Size	Input Feature	Target Variable
Dataset 1	Training & Testing & Validation	4869	Tweet text	Sentiment Label
Dataset 2	Airline Sentiment Analysis	14640	Tweet text	Inferred sentiment
Dataset 3	Squid Game Sentiment by Location	56149	Tweet text	Inferred sentiment

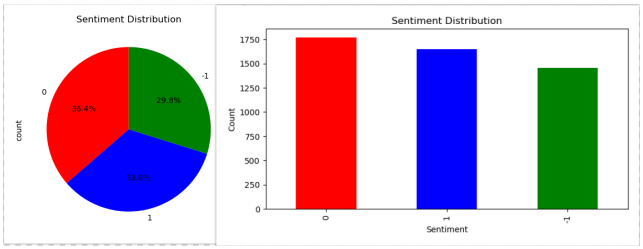

Fig. 1. Sentiment label distribution

lated to interpretability, deployment complexity, and inference speed.

While much of the literature emphasizes boosting classification performance, fewer studies explore how sentiment analysis can be translated into actionable insights. The integration of sentiment classification with temporal, geographic, or demographic data is still relatively underexplored. Studies like the Twitter Sentiment Geographic Index (TSGI) [6] have shown promise in mapping sentiment spatially, but broader adoption remains limited.

This project addresses that gap by comparing both traditional and deep learning approaches using F1 score as the primary evaluation metric.

III. DATA DESCRIPTION

This project utilizes three Twitter-based text datasets for sentiment analysis. The first dataset is labeled and used for training, validation, and testing of various machine learning and deep learning models. The remaining two datasets are unlabeled and are used solely for inference after the model has been trained. Table 1 describes about the datasets. The distribution or ratios of sentiment label in dataset 1 are as presented in Figure 1.

Across all datasets,only the tweet text was used as the input feature. For Datasets 2 and 3, the trained model was applied to predict sentiment labels. These predictions were then analyzed to identify sentiment patterns related to airlines and geographic locations, respectively.

a) ***Dataset 1*** *– Labeled Twitter Dataset:* Used to train and evaluate the models, this dataset includes sentiment-labeled tweets categorized as positive, neutral, or negative. It is relatively balanced and serves as the foundation for learning sentiment patterns in text.

b) ***Dataset 2*** *– Airline Twitter Dataset:* This dataset contains a large collection of tweets about various airlines. Though originally labeled, we disregarded existing labels to simulate a real-world inference scenario. The trained model was used to classify sentiment for each tweet. Later, results were visualized using a 100% stacked column chart to compare sentiment distribution across different airlines.

c) ***Dataset 3*** *– Squid Game Tweets by Location:* This dataset was created by extracting tweets mentioning *Squid Game* from various geographic regions. Using the trained sentiment model, each tweet was classified and visualized using a Power BI map. This helped identify how sentiment toward the show varied globally, offering insights into regional differences in public opinion.

IV. METHOD

A. Data Pre-processing

All three datasets underwent a comprehensive pre-processing pipeline to prepare them for sentiment classification. These steps ensured the data was clean, consistent, and suitable for both traditional machine learning models and neural network-based architectures.

- Text Cleaning: All tweets were first converted to lowercase to ensure uniformity. URLs, user mentions (@username), hashtags (#hashtag), emojis, special symbols, and punctuation were removed using regular expressions. This step reduced noise and preserved the semantic structure of the text.
- Tokenization & Lemmatization: The cleaned text was tokenized into individual words. Common English stopwords (e.g., “the”, “is”, “in”) were removed to focus on informative content. Lemmatization was then applied to reduce words to their base or dictionary form (e.g., “running” $\rightarrow$ “run”), improving generalization across similar terms.
- Label Encoding: In the labeled dataset, categorical sentiment labels — *negative*, *neutral*, and *positive* — were mapped to numeric values 1, 0, and 1, respectively. This encoding made the data compatible with classification algorithms.
- Location Filtering (for Squid Game Dataset): Tweets from Dataset 3 were filtered based on the validity of the user-reported location field. A combination of heuristics, the GeoText library, and spaCy’s Named Entity Recognition (NER) was used to identify and retain only tweets with recognizable geographic references (e.g., cities, countries, landmarks).
- Vectorization: Cleaned tweets were transformed into fixed-length numerical vectors using two feature extraction methods:
  - Bag of Words (BoW): Encoded text as token occurrence counts across the corpus.
  - TF-IDF: Applied term-frequency inverse-document-frequency weighting to reduce the influence of common but uninformative words.

TABLE II
FEATURE EXTRACTION TECHNIQUES

Technique	Description
Bag of Words (BoW)	Represents text by counting the frequency of each word in the corpus vocabulary. Simpler and fast but ignores word importance and context.
TF-IDF	Computes term frequency-inverse document frequency, giving more weight to rare but important words. Helps reduce the dominance of common but less informative words.

TABLE III
HYPERPARAMETER TUNING RANGES

Model	Input features	Hyperparameters tuned on
SVM	TF-IDF vectors	Cost [0.1, 1, 10] Kernel [linear, RBF]
Random Forest	TF-IDF vectors	n_estimators [100, 200] max_depth [None, 10, 20]
Naive Bayes	TF-IDF vectors	Alpha [0.1, 0.5, 1.0]
SVM	BOW vectors	Cost [0.1, 1, 10] Kernel [linear, RBF]
Random Forest	BOW vectors	n_estimators [100, 200] max_depth [None, 10, 20]
Naive Bayes	BOW vectors	Alpha [0.1, 0.5, 1.0]
Feedforward Neural Network	TF-IDF vectors	learning_rates = [0.001, 0.0005, 0.01] hidden_dims = [64, 128, 256] batch_sizes = [32, 64]
LSTM	TF-IDF vectors	learning_rates = [0.001, 0.0005, 0.01] hidden_dims = [64, 128, 256] batch_sizes = [32, 64]

- Data Splitting: The labeled dataset (Dataset 1) was initially split into 80% training and 20% test sets. The training set was further divided into 80% training and 20% validation subsets. These partitions were used for model training, hyperparameter tuning, and unbiased performance evaluation.

This pre-processing pipeline ensured that all models received high-quality, normalized input for optimal performance during training and inference.

B. Feature Extraction

Two widely-used text vectorization techniques were employed to convert cleaned textual data into machine-readable format:

These feature vectors served as input to all supervised learning models for training and evaluation.

C. Model building and Hyperparameter Tuning

To identify the most effective sentiment classification model, we developed and tuned both traditional machine learning and deep learning models. The evaluation metric used for comparison was F1 Score, prioritizing a balance between precision and recall.

While the Feedforward Neural Network achieved the highest F1 score, its implementation and computational complexity were higher compared to classical models. Therefore, after evaluating both performance and feasibility for broader sentiment analysis tasks, we selected the Support Vector Machine (SVM) model due to its near-competitive performance (only 0.95% lower in F1 score) and simpler deployment pipeline.

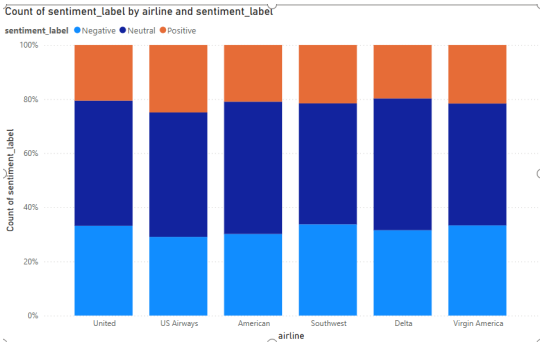

Fig. 2. 100% Stacked Column Chart of Sentiment by Airline

V. EXPERIMENT AND RESULTS

To demonstrate the real-world applicability of the developed sentiment analysis models, we conducted two experiments using unlabeled Twitter datasets. The goal was to showcase how these models can be leveraged to derive insights from large volumes of text data. All visualizations were created using Power BI, enabling clear interpretation and communication of the results. .

A. Experiment 1: Airline Sentiment Distribution

In the first experiment, we applied the trained model to the airline Twitter dataset, which contained over 14,000 tweets mentioning major U.S. airlines. After classifying each tweet into positive, neutral, or negative sentiment, we visualized the results using a 100% stacked column chart and clustered column chart. This 00% stacked column chart provides a normalized view of sentiment distribution across airlines, allowing direct comparison regardless of the total tweet volume per airline. Figure 2 and 3 highlights the results of this experiment.

Key observations include:

- United Airlines had the highest volume of tweets, with a significant portion expressing neutral or negative sentiments, suggesting frequent dissatisfaction or unresolved issues.
- American Airlines, US Airways, and Southwest displayed a similar sentiment profile, with negative tweets dominating.
- Virgin America, although receiving fewer tweets, stood out with a notably higher share of positive sentiments, reflecting comparatively better public perception or customer experience.

This experiment highlights how normalized sentiment comparisons can provide actionable insights for airline brands. For instance, marketing or customer service teams could use such findings to monitor brand health, benchmark against competitors, and prioritize response strategies.

TABLE IV
BEST HYPERPARAMETERS AND MODEL PERFORMANCE COMPARISON

Model	Tuned Hyperparameters	Input features	F1 Score	% Difference from Best Model
SVM	C = 10, kernel = RBF	TF-IDF vectors	0.632	-2.17
Random Forest	n_estimators: 100, max_depth: None	TF-IDF vectors	0.615	-4.8
Naïve Bayes	Alpha = 0.5	TF-IDF vectors	0.606	-6.2
SVM	C = 10, kernel = RBF	BOW vectors	0.616	-4.64
Random Forest	n_estimators: 100, max_depth: None	BOW vectors	0.622	-3.72
Naive Bayes	Alpha = 1	BOW vectors	0.620	-4.02
Feedforward Neural Network	Learning_rate = 0.001, hidden_dim = 64, batch_size = 32, epochs = 100	TF-IDF vectors	0.646	–
LSTM	Learning_rate = 0.001, hidden_dim = 64, batch_size = 64, epochs = 100	TF-IDF vectors	0.620	-4.02

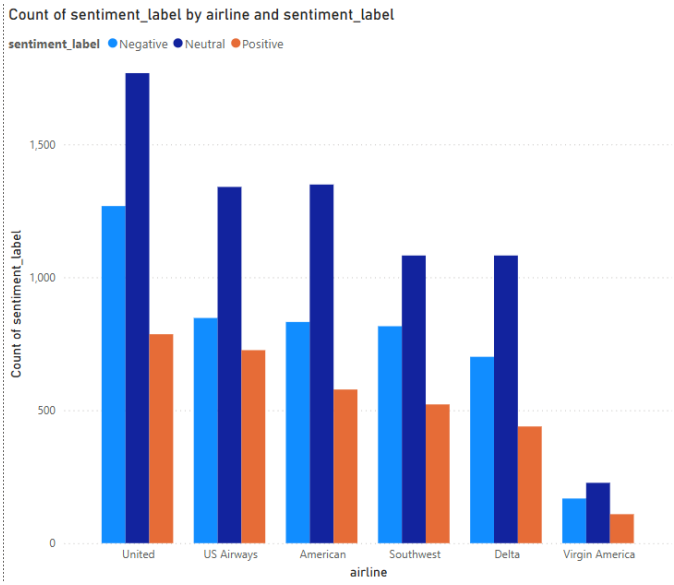

Fig. 3. Enter Caption

B. Experiment 2: Sentiment of 'Squid Game' Tweets by Location

The second experiment focused on analyzing global sentiment toward the Netflix show *"Squid Game"*. Using a dataset of over 56,000 tweets, we applied the trained model to infer sentiment and then visualized the results geographically using a Power BI map based on the user's location metadata.

Figure 4 illustrates the geospatial sentiment distribution for tweets referencing *Squid Game*.

Key insights from the map visualization:

- A majority of tweets from North America and Western Europe showed positive sentiment, suggesting high levels of audience engagement and favorable reception.
- Neutral and negative sentiments were scattered across other parts of the world, potentially influenced by regional content preferences, cultural context, or local media coverage.

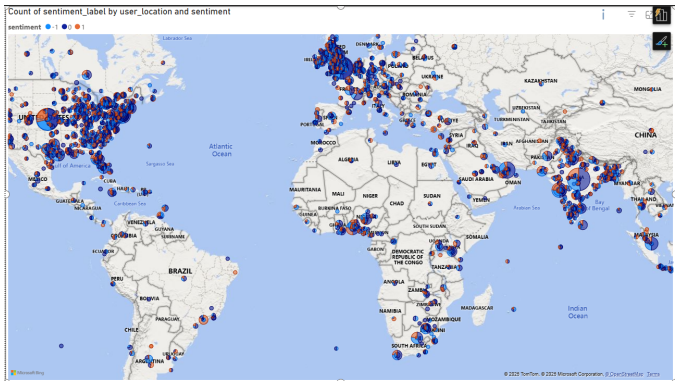

Fig. 4. Map of Squid Game Tweet Sentiments by User Location

- The model effectively captured regional sentiment variance, enabling a broader understanding of how public opinion toward the same media content may differ globally.

This kind of geo-sentiment mapping can be particularly valuable for entertainment platforms, market analysts, or regional marketing teams aiming to tailor their strategies based on how content resonates across different geographies.

These experiments demonstrate the practical utility and generalizability of the sentiment classification model. By extending model predictions to new, unlabeled datasets and pairing them with effective visualization tools like Power BI, we show how sentiment analysis can uncover trends, patterns, and actionable insights at scale.

VI. CONCLUSION

This data mining project successfully explored sentiment analysis on Twitter data to uncover patterns in public opinion. By developing and evaluating multiple machine learning and deep learning models, we identified a Support Vector Machine (SVM) trained on TF-IDF features as the optimal balance between performance and complexity.

The trained model was applied to two real-world scenarios: analyzing sentiment toward major U.S. airlines and under-

standing geographic sentiment distribution regarding the show *Squid Game*. These experiments revealed meaningful trends in user perception and demonstrated the model's ability to generalize to new, unlabeled datasets.

The use of Power BI for visualization further enhanced the interpretability of results, highlighting sentiment proportions across entities and locations. These findings illustrate the practical utility of sentiment analysis in applications such as brand monitoring, audience analysis, marketing strategy, and content promotion.

Overall, this project reinforces the value of combining data mining, natural language processing, and visual analytics to extract actionable insights from social media data.

REFERENCES

[1] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[4] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proc. 2014 Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[5] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.

[6] MITRE Corporation, "Twitter Sentiment Geographic Index (TSGI)," Harvard Center for Geographical Analysis, 2013. [Online]. Available: https://gis.harvard.edu/twitter-sentiment-geographic-index-tsgi

[7] "Sentiment Analysis in Power BI," ClearPeaks. [Online]. Available: https://www.clearpeaks.com/sentiment-analysis-in-power-bi/

[8] L. Mitchell, K. Dodge, M. S. Golder, and C. M. Danforth, "The Geography of Happiness: Connecting Twitter sentiment and expression, demographics, and objective characteristics of place," *arXiv preprint arXiv:1302.3299*, 2013. [Online]. Available: https://arxiv.org/pdf/1302.3299

DECLARATION OF INDIVIDUAL CONTRIBUTIONS

Project: Sentiment Analysis of Tweets to Discover Regional & Demographic Trends

Team: Data Smugglers

We, the members of Team Data Smugglers, hereby declare that this submission represents our own work and that the contributions of each team member are accurately and honestly described below. We confirm that we have each contributed substantially to this project as outlined.

INDIVIDUAL CONTRIBUTIONS

Vishnu Alla

- Design, implementation, and evaluation of machine learning models (SVM, Random Forest, Naïve Bayes).
- Hyperparameter tuning and optimization of ML algorithms.
- Development of PowerBI dashboards and visualizations for sentiment analysis results.
- Coordinated team efforts and organized tasks for smooth project progress.
- Contributed to report writing and presentation preparation.

Vineeth Karjala

- Design, implementation, and evaluation of deep learning models (Feedforward Neural Network, LSTM).
- Neural network architecture design and optimization.
- Creation of PowerPoint presentations for project milestones and final presentation.
- Comparative analysis between traditional ML and deep learning approaches.

Surya Vakkalagadda

- Data collection, cleaning, and preprocessing of all three Twitter datasets.
- Text normalization, tokenization, and feature extraction (TF-IDF, BoW).
- Implementation of data filtering and location extraction procedures.
- Literature review, writing and compilation of the project report and documentation.