Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

SENTIMENT ANALYSIS OF SHOPPER REVIEWS USING X DATA BY MACHINE LEARNING

NAME: SURAJ RAMESH BABU

ID NO: 2513205

LECTURER: MARK HUNTLY

MODULE CODE: ACCA7025

MODULE NAME: APPLICATION OF MACHINE LEARNING

DATE: 17/04/2026

INDEX

SL.NO.	CONTENTS	PAGE.NO.
1	INTRODUCTION	4
2	LITERATURE REVIEW	2-6
3	METHODOLOGY	6-10
4	TRAINING CLASSIFIER	10-12
5	DISCUSSION AND RECOMMENDED SYSTEM	12-13
6	CONCLUSION	14
7	APPENDIX	14-16
8	REFERENCES	16-18

ABSTRACT

With over 500 million messages and opinions published every day, X (previously Twitter) has turned into an amazing find for businesses looking to track their brand's image, assess performance, and gauge consumer attitude toward their goods. Our efforts to develop a machine learning based method for analyzing message sentiment from X are presented in this work. An arbitrary post is fed into this algorithm, which then classifies it according to whether it is positive or negative. We take a supervised text classification strategy and combine new feature engineering approaches with well-known features. [1]

Our technology is ideal for practical implementation in dynamic social media contexts because it can handle enormous volumes of posts in real time. To increase classification accuracy, we also include preprocessing methods that are specific to short-text features like hashtags, mentions, and emoticons. Businesses can obtain insights into user satisfaction and product quality by expanding the system to evaluate consumer evaluations from e-commerce sites. Additionally, the robustness of sentiment prediction across various text domains is improved by using both structured reviews and social media posts. Our project's primary goal was to compare several machine learning algorithms for the sentiment categorization assignment. Based on This research suggests that the proposed machine learning and natural language processing methods are practical and effective ways to analyze sentiment. [2]

Additionally, compared to brief social media posts, the algorithm may capture more context-rich and complex feelings by utilizing review-based information. This enhances the system's capacity to generalize across many areas and writing styles. As a result, when the sentiment classification model is used in practical settings, its overall performance and dependability are much improved.

1.INTRODUCTION:

Social media usage to express feelings, ideas, and views has increased exponentially, making X a treasure trove for brand performance analysis. Compared to official polls and questionnaires, opinions found on X are frequently more informal, honest, and instructive. Millions of consumers share their opinions about the companies they deal with. If these sensations are recognized, businesses may utilize them not only to track the success of their brand but also to pinpoint the elements and times that elicit strong feelings. Products, celebrities, occasions, or political parties are examples of these brands. As a result, specialized teams are frequently tasked with analyzing a candidate's performance in national surveys or assessing public reaction during the release of a product or film.

But with over 500 million posts received every day, this data is already too massive for any staff to manually evaluate. Similarly, it is likely that a set of rules created by hand cannot adequately capture the diversity of posts. It is important to note that deciphering a post's sentiment is significantly more difficult than deciphering a well-formatted paper. Posts on X don't necessarily contain terms from conventional vocabulary (i.e., out-of-vocabulary words) and don't adhere to any formal language structure. Emojis, punctuation, and symbols, such smileys and emoticons, are frequently employed to convey emotions.

In this work, I estimate the sentiment if any that posts on X convey by analyzing their patterns and characteristics Using techniques for natural language processing and machine learning. In particular, we develop a computational model that, depending on the attitudes it expresses, can categorize a particular post as either good, negative, or neutral. Polar posts with distinct sentiments would be included in both positive and negative classes. Nonetheless, a neutral class may include a post that is objective or subjective; the user may express neutrality in their viewpoint or not express any opinion at all. [3]

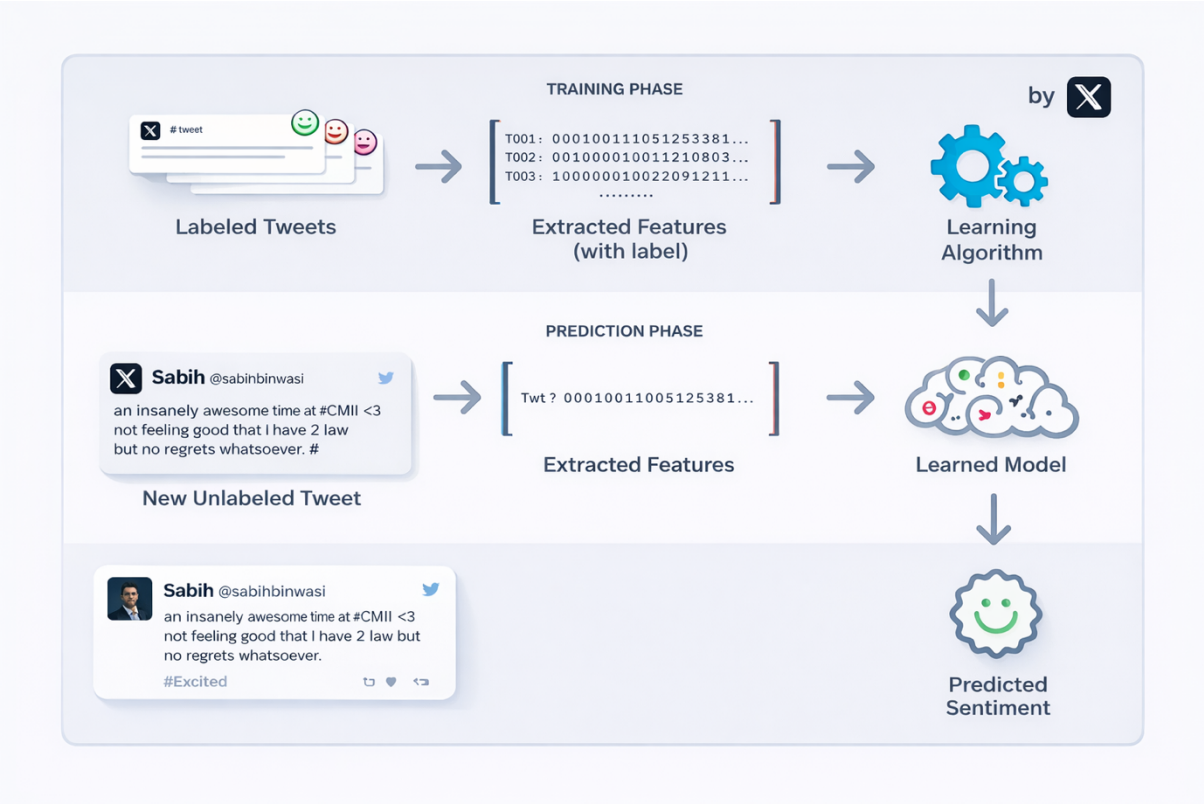

Figure 1: Overview of Supervised Sentiment Classification of X

Before understanding X sentiment analysis research, we must outline the typical process used to address this issue. A common method for sentiment classification of posts on X is supervised text classification, a machine learning technique in which a class predictor is deduced from labeled training data. Figure 1 shows an overview of this method, modified to classify a post's sentiment. A dataset of labeled posts is first assembled. Human annotation classifies each post in this collection as positive, negative, or neutral depending on the sentiment the commenter or commentators determined after examining the post. [4]

For every labeled post, a feature extractor then creates a feature vector, the values of which should describe the sentiment. Following the extraction of feature vectors for every post in the labeled dataset, these vectors are input into a classification algorithm that looks for connections between the labeled sentiment and each value (referred to as a feature) in the vector. SVM (Support Vector Machines) and Naive Bayes classifiers are two common classification techniques used for this purpose. These algorithms capture the link between features and sentiments. These algorithms record the feature-sentiment link and store it in a learning model. The model employs previously learnt relationships to predict sentiment when it is given a new instance.

2.LITERATURE REVIEW:

In the contexts of natural language processing and machine learning, sentiment analysis on popular social media sites such as X (previously Twitter) has been extensively researched. For text classification tasks, early research mostly concentrated on using conventional supervised learning techniques like Naive Bayes, Support Vector Machines (SVM), and Maximum Entropy classifiers. These research showed that basic models can attain good baseline performance in sentiment classification when paired with efficient feature engineering, such as n-grams, part-of-speech tagging and term frequency-inverse document frequency (TF-IDF). [5]

The difficulties related to short-text data, such as postings on X, were examined in further research. These difficulties included misspellings, informal language, abbreviations, and the usage of emojis, hashtags, and mentions. To better capture emotion signals in such chaotic data, researchers developed specific preprocessing methods and feature representations. To increase classification accuracy, machine learning techniques were also integrated with lexicon-based approaches, which rely on pre-established sentiment dictionaries.

For sentiment classification tasks, more recent research has evaluated several conventional machine learning models, like Logistic Regression, Naive Bayes, and Support Vector Machines (SVM).These studies demonstrate that efficient feature engineering methods, such as n-grams, TF-IDF, and sentiment lexicons, are critical to these models' performance. While algorithmic differences might affect accuracy, well-tuned conventional models can produce datasets. Promising outcomes in enhancing performance across a variety of datasets have been demonstrated by hybrid approaches that combine machine learning with language and semantic aspects. For sentiment analysis tasks on platforms such as X, the literature generally shows that traditional supervised learning algorithms are still efficient in terms of computation, effectiveness, and interpretability. [6]

3. METHODOLOGY:

As mentioned in the background section, the proposed model utilizes a standard machine learning approach for sentiment analysis.

3.1 LABELED DATA:

Several datasets are available in the NLTK library. I selected the movie reviews dataset in order to select the appropriate type of data. When evaluating a film, a variety of terms are used, which would be beneficial for the classifier. Additionally, it classifies both positive and negative evaluations, making it simpler for the classifier to train the dataset. The dataset is also well-structured and often utilized in research, making it appropriate for comparing and measuring model performance. Tokenization, stop-word removal, and normalization are

examples of preprocessing techniques that can be successfully used to this dataset to improve feature extraction even more.

3.2 PREPROCESSING:

The text undergoes the following preprocessing stages before the feature extractor may utilize the post to create a feature vector. In these steps, the post's plain text is transformed into processable pieces with additional data that the feature extractor can use. Third-party tools that were specifically designed to handle the peculiar characteristics of post text were employed for each of these processes.
Tokenization, stop word removal, hashtag and mention handling, and text normalization (e.g., lowercase conversion) are further preprocessing processes. Emojis, emoticons, and punctuation are also given special consideration because they frequently convey important sentiment information. Additionally, methods like lemmatization and stemming are used to reduce words to their most basic forms, which enhances feature representation consistency. Together, these preprocessing techniques improve the quality of the input data and help the sentiment classification model perform better. [7]

STEP 1-TOKENIZATION:

The technique of transforming text as a string into a readable components known as tokens is known as tokenization. These components can be words, emojis, URL links, hashtags, or punctuation in the context of a post. For instance, the text "an insanely aweseome..." was divided into "an," "insanely," and "awesome." Spaces are frequently used to divide these components. However, there is frequently no gap between punctuation that ends a phrase, such as exclamation points or full stops. However, since a term used since a term that occurs frequently in the text may have a different sentiment value than a hashtag., hashtags with "#" before the tag must be kept. As a result, tokens are extracted using a platform-specific tokenizer.

Tokenization also aids in maintaining crucial contextual clues, including repeated characters (like "soooo good"), which could suggest a greater mood. Additionally, it makes it possible to distinguish between emoticons and emojis, which are important markers of user emotions in short-text data. A solid basis for ensuing feature extraction and classification tasks is formed by proper tokenization, which guarantees that every significant aspect of the post is efficiently captured. [8]

STEP 2 – PARTS OF SPEECH TAGS:

The features of a word in a sentence are known as Parts of Speech (POS) tags. that are derived from a language's grammatical categories. Because words may have varying sentiment values based on their POS tag, this information is crucial for sentiment analysis. For instance, the noun "good" has no sentiment, whereas the adjective "good" conveys a positive sentiment.

By identifying each word's function inside a phrase, POS tagging enables the model to more accurately assess the context in which the word is used. Additionally, it helps differentiate between less sentimental terms (like conjunctions or determiners) and opinion words (like adjectives and adverbs). Additionally, by emphasizing sentiment-rich sections of speech, using POS data as features might increase classification accuracy. This improves the model's capacity to identify minute differences in sentiment conveyed in brief and casual language.

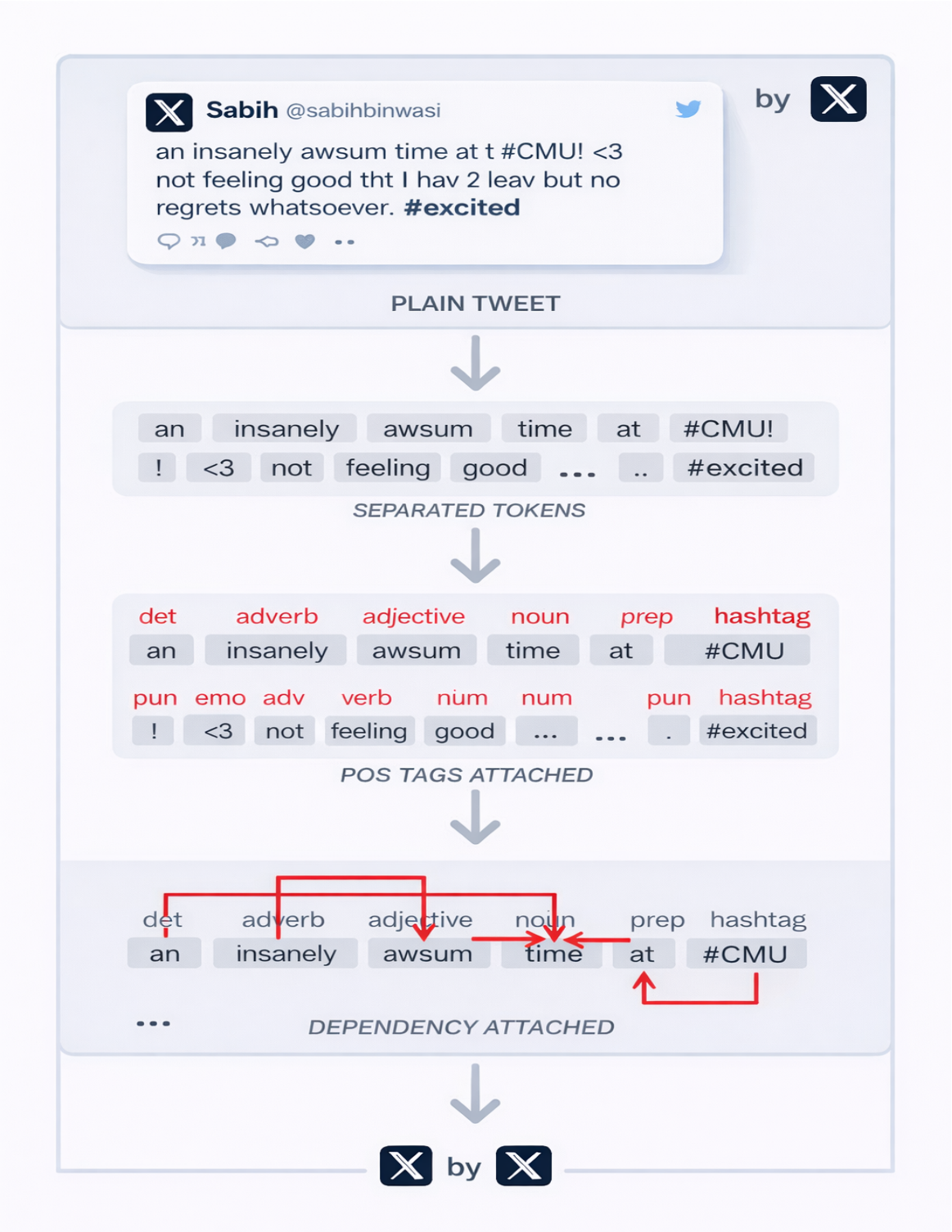

Figure 2 : Tweet Preprocessing and Linguistic Feature Extraction Pipeline

3.3 FEATURE EXTRACTION:

Creating a feature vector from a specific post is known as feature extraction. An number that contributes to assigning a sentiment class to a post is each entry in a feature vector. This contribution can range from substantial, where a feature entry's value significantly affects the true sentiment class, to negligible, in which there is no correlation between the sentiment class and feature value. Using highly correlated data and avoiding the use of "noisy features," the classification algorithm's task is frequently to determine the degree of dependency between features and classes. Word frequencies, n-grams, and sentiment lexicon scores are often employed characteristics in practice that aid in extracting both syntactic and semantic information from the text.

A common cross-validation technique was used to assess performance. Ten non-overlapping subsets of the training dataset were created pseudo-randomly while maintaining the class distribution. Nine folds were used as the test set and one fold as the training set to assess the system's accuracy. This procedure is repeated ten times, with a different fold chosen as the evaluation set each time. I compute an average that serves as a performance metric after all ten accuracy scores have been gathered. By lowering the variance brought on by random data splits, this approach offers a more accurate estimate of model performance. Furthermore, it guarantees that each data point is utilized for both training and validation, resulting in a more reliable and comprehensive assessment of the model.

3.3.1 BAG OF WORDS (BOW) FEATURE SET:

The frequency of tokens (or, in this instance, the presence of a token) is shown by a feature vector in a bag of words (unigrams). Researchers all agree that this feature set ought to be an aspect of the feature vector. based on our analysis of previous work. Every distinct token in the labeled training set is given an entry in the feature vector. The corresponding token is given a binary value of 1 if it appears in a post, and 0 otherwise. Keep in mind that neither the token sequence ordering nor the grammatical structure are maintained. Rather, just a token's autonomous existence is maintained. As can be observed, that column receives a value of 1 since the token "awsum" appears in a particular message. Conversely, the term "hater" has a score of 0 because it does not appear in a post. Keep in mind that since we have a column for "hater" in a feature vector, it would have appeared in some post instances in the training dataset.

For text classification problems, this method is straightforward but quite successful, especially when paired with other features. It enables the model to rapidly recognize significant sentiment-related terms. However, it might miss statements whose meaning depends on sequence because it disregards word order and context. Higher-order n-grams or other linguistic traits are frequently mixed with unigram features to overcome this restriction. The Bag of Words model is nevertheless a powerful foundation despite its simplicity because of its ease of implementation and computing economy.

3.3.2 FEATURE SELECTION:

We chose to eliminate several of the features in this set after realizing that the majority of them had no bearing on the sentiment analysis objective. Previous research has shown that features that are carefully chosen and optimized in the training set and occur less frequently than a particular value can be eliminated. According to the theory, tokens that are uncommon in the training set can be less significant for the task. Eliminating these low-frequency characteristics also contributes to lowering noise and raising the model's overall effectiveness.

Chi-Squared Feature Selection, a widely used technique for evaluating attributes, was applied. This technique assesses the relationship between a feature value and each class by running a classification algorithm. The feature is given a higher rank if it is determined to have a higher reliance (correlation). A more condensed and instructive feature set is produced by eliminating features with lower scores. This lowers computational complexity while also increasing classification accuracy. Additionally, by removing unnecessary and duplicated features, feature selection improves the model's capacity for generalization and helps minimize overfitting.

4. TRAINING CLASSIFIER:

Multiple machine learning algorithms have been used to train the classifier. Below is a brief overview of the classifiers.

NAIVES BAYES Algorithm:

It is a classification procedure based on Bayes' Theorem that requires predictor independence. In short, a Naive Bayes classifier assumes that being aware of one feature in a class has no effect on the existence of any other feature.For instance, if a fruit is spherical, red, and roughly three inches in diameter, it can be classified as an apple. All of these characteristics independently increase the likelihood that this fruit is an apple, even if they rely on one another or the presence of other characteristics. For this reason, it is referred to as "Naive."

Naive Bayes does very well in many real-world applications despite this simplifying assumption, particularly in text classification tasks like sentiment analysis. It functions well with high-dimensional feature spaces like Bag of Words, is computationally economical, and takes comparatively little training data. Furthermore, it is capable of efficiently managing both binary and multi-class classification tasks. It is frequently employed as a baseline model in sentiment analysis systems because of its probabilistic character, which also makes it simple to understand and apply.

SVM CLASSIFIER:

Regression and classification problems can be solved with a method for supervised machine learning named "Support Vector Machine" (SVM).. Nonetheless, categorization difficulties are where it is most frequently employed. Each data item in this algorithm is plotted as a point in n-dimensional space, where n is the number of features you have. Each feature's value is represented by a specific coordinate. Next, we classify by identifying the hyperplane that best distinguishes the two classes.

Finding the optimal hyperplane in a way that maximizes the margin between different classes maximizes the model's capacity to generalize to new data.Support vectors, which are the data points nearest to the hyperplane, are essential for establishing the decision boundary. Additionally, SVM is a potent and popular method for sentiment analysis and text classification tasks since it can handle both linear and non-linear data through the application of kernel functions.

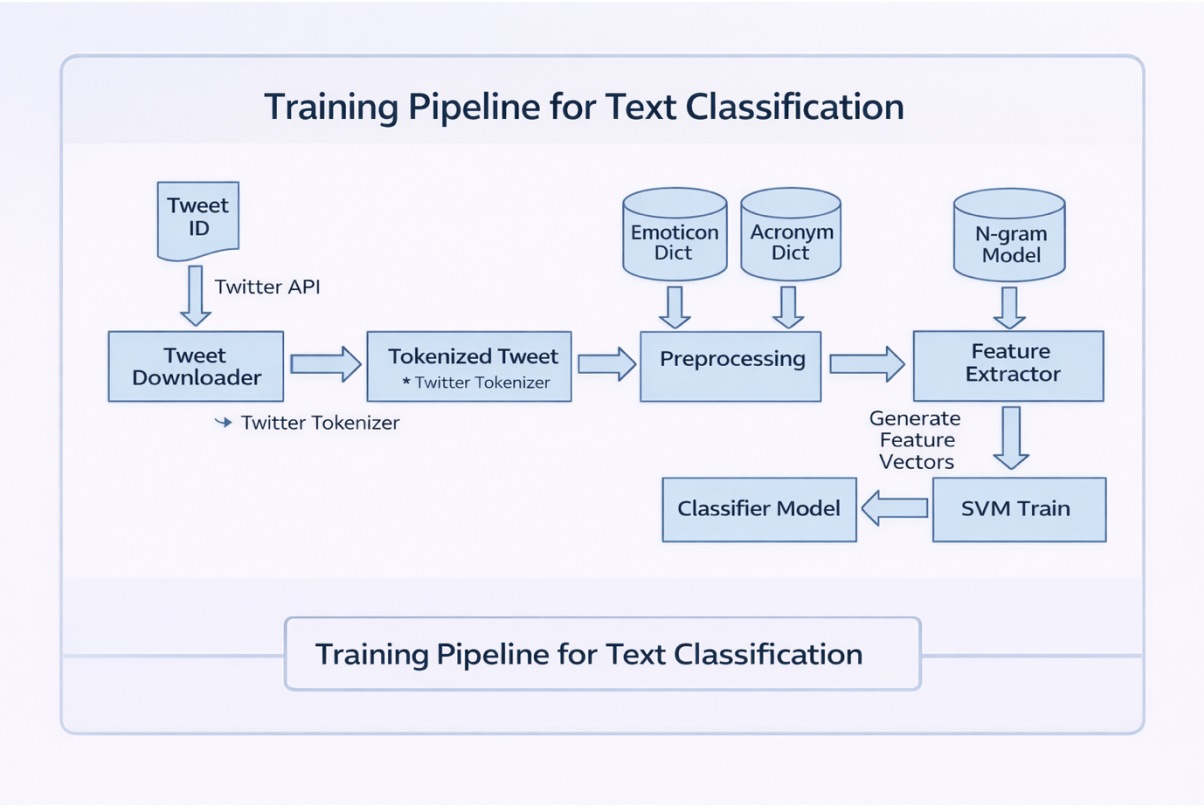

Figure 3 : Sentiment Analysis Training Process

Logistic Regression:

Logistic regression is classified algorithm. It is used to estimate the binary result (1/0, Yes/No, True/False) given a set of independent variables. Binary or categorical outputs are represented

by dummy variables. Logistic regression can be seen as a procedure where the outcome variable is categorical and the dependent variable is the log of chances. particular instance of linear regression. Put simply, it fits data to a logit function to estimate the likelihood of an event occurring.

Because logistic regression produces easily comprehensible probability scores, one of its main benefits is its simplicity and interpretability. Additionally, it works effectively when there is a roughly linear relationship between the target variable and the characteristics. It also functions well with big datasets and is computationally efficient. Regularization strategies like L1 and L2 can be performed to avoid overfitting and boost model generalization. We mix the earlier defined ways to create a sentiment module that works better.. In sentiment classification tasks, the system can increase accuracy and robustness by comparing their performance and utilizing their unique capabilities.

5. DISCUSSION AND RECOMMENDED SYSTEM:

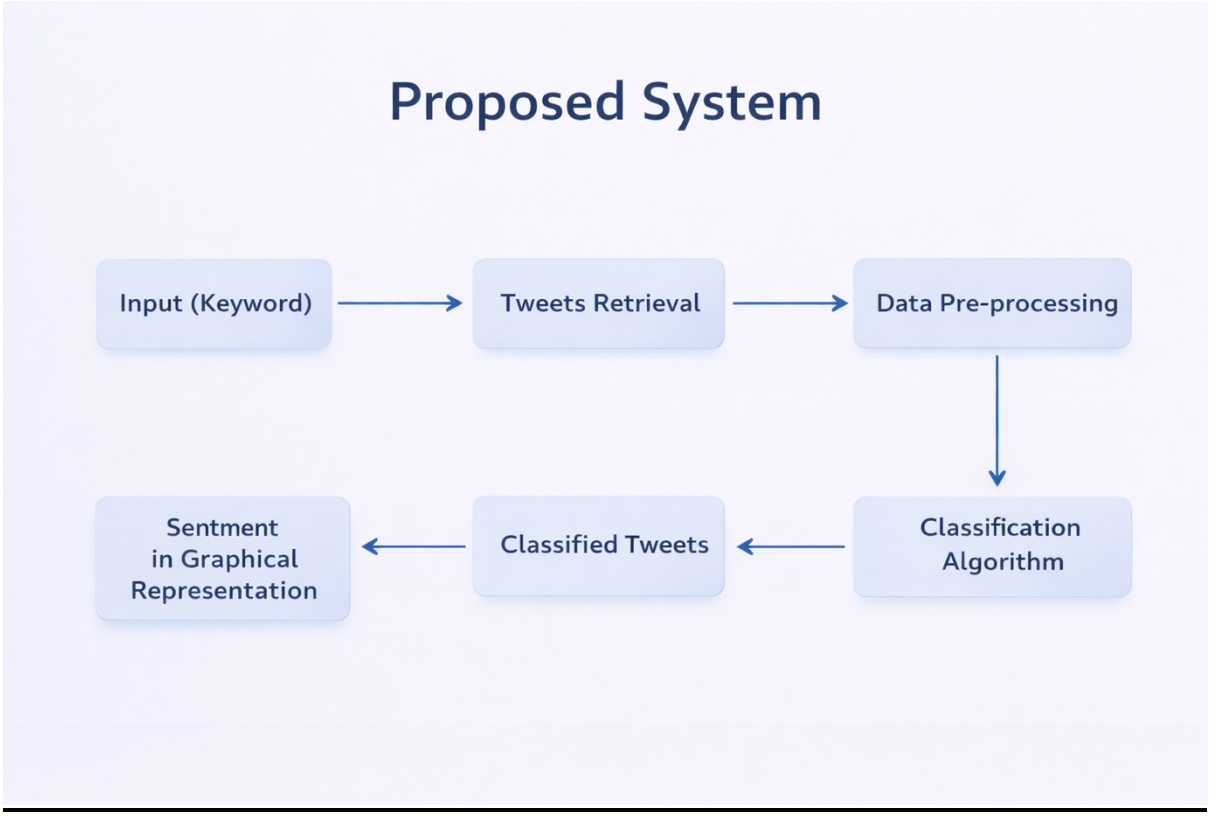

Figure 4: Proposed Architecture for Tweet Sentiment Analysis System

Posts were found in real time using the X (previously Twitter) Public Stream API. We had to use this API to subscribe to keywords so that X could send a subset of articles bearing a keyword in real time. This meant that our tool could not access content that had already been mentioned, and a keyword subscription was needed to record the trend for an event. We noticed

just 4–10% of all posts mentioning the subscribed trending keywords made use of the Public API.We avoided using the paid firehose service, which was necessary to collect a larger share. The stream's size was adequate for our research because we were working with large amounts of data and had little funding.

Metadata about the post was also saved in addition to the post itself. The posting time, location, platform, GPS coordinates, and user details of the account that made the post were all included. Screen name, follower count, location, and account age were among the user's details. It is important to note that not every post included every field. For instance, because users frequently decide not to disclose their location information, only a very small portion of posts included coordinates. In order to preserve data quality, the gathered data also needed to be cleaned and filtered to eliminate spam, duplicate posts, and non-English text. The metadata also made time-based analysis viable, enabling us to examine sentiment patterns over particular time periods. Despite certain limitations in data availability, the collected dataset provided sufficient diversity and volume for building and evaluating the sentiment analysis model.

As soon as our system retrieves the tweet, the text is sent to the sentiment analysis learning model that identified the sentiment it conveyed. The prediction stage was fast enough to stop interfering with the speed at which the tweets were received and stored.

The stream of tweets includes both positive and negative sentiment, which allows us to create a live graph.

5.1 OUTCOME:

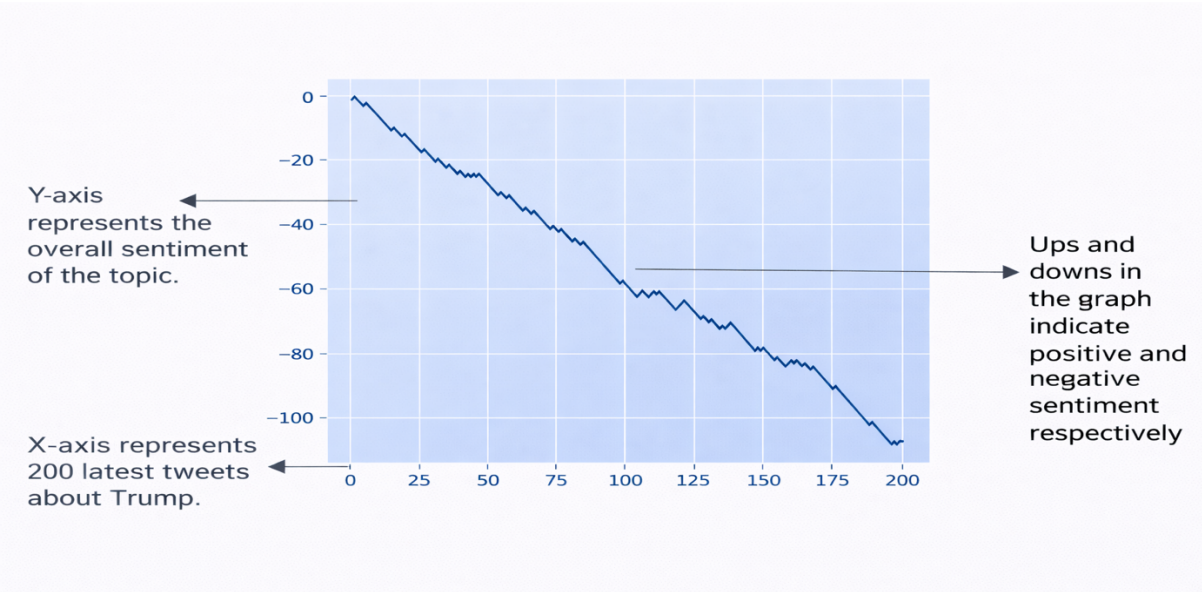

Figure 5: ime-Series Visualization of Tweet Sentiment Polarity

It is possible to make a live charting of real-time Twitter data using matplotlib and the appropriate intuition for the x and y axes. The sentiment of the public regarding Trump following his withdrawal of the United States from the Paris Climate Agreement is depicted in the above picture.

6. CONCLUSION:

In this study, I used supervised text classification techniques to present a machine learning-based method for sentiment analysis of postings on X. To increase classification performance, the suggested model combines well-known techniques like bag-of-words and lexical features with improved tactics like polarity-based grouping and weight-adjusted handling of negation. An efficient and trustworthy sentiment prediction system was created by using Support Vector Machines (SVM) as the main classifier.

The findings show that the accuracy and efficiency of sentiment categorization can be greatly increased by integrating several feature engineering techniques with strong machine learning algorithms. Furthermore, the technology facilitates interactive visualization, which makes it easier to understand sentiment patterns across time and in various circumstances. The same paradigm may also be successfully used for product review analysis, where user satisfaction, product strengths and weaknesses, and corporate decision-making can all be determined by classifying customer comments from e-commerce platforms. There are, still, a number of chances for additional development. By merging several models, ensemble learning techniques could improve prediction performance. By using sophisticated methods to separate and understand compound phrases, hashtags—which frequently contain powerful sentiment signals—can be further investigated. Furthermore, more accurate feature representation may result from improving sentiment lexicons to lower noise. Lastly, the suggested system's generalizability and usefulness across different online platforms would be enhanced by expanding its evaluation to a variety of short-text datasets, such as product reviews.

7. APPENDIX:

Appendix A: Description of the Dataset

Posts gathered from X utilizing the Public Stream API according to predetermined keywords make up the dataset used in this study. Every post contains the primary text as well as related metadata, including location (if available), user information, and a timestamp. Additionally, model training and evaluation were supported by the NLTK movie reviews dataset. The datasets include labeled examples that are divided into three sentiment classes: neutral, negative, and positive.

Appendix B: Steps in Preprocessing

To transform unstructured text into structured input for the model, the preprocessing

Pipeline consists of several steps:

- Text tokenization using words, emojis, hashtags, and symbols
- Elimination of unnecessary characters and stop words
- Normalization techniques include managing repeated characters and lowercasing
- Lemmatization or stemming words to their most basic form
- Managing unique tokens like hashtags, URLs, and mentions

Appendix C: Feature Extraction Techniques

The following feature extraction methods were used:

- Bag of Words (Unigrams) representation
- Frequency-based and binary feature vectors
- Lexical features including sentiment lexicons
- Part-of-Speech (POS) tagging features
- Feature selection using Chi-Squared method

Appendix D: Machine Learning Algorithms

The study implemented and compared multiple supervised machine learning algorithms:

- Naive Bayes Classifier
- Support Vector Machine (SVM)
- Logistic Regression

Appendix E: Evaluation Methodology

Metrics used for evaluation include:

- Accuracy
- Precision
- Recall
- F1-score

Appendix F: Tools and Libraries

The implementation Recommending Following tools and libraries:

- Python programming language
- NLTK (Natural Language Toolkit)
- Scikit-learn for machine learning algorithms
- APIs for real-time data collection from X

Appendix G: Limitations

- Limited access to full data stream due to API restrictions
- Presence of noise and informal language in posts
- Difficulty in handling sarcasm and implicit sentiment
- Limited contextual understanding due to short text format

Appendix H: Future Enhancements

- Implementation of ensemble learning techniques
- mproved handling of hashtags and compound words
- Refinement of sentiment lexicons to reduce noise
- Extension of the system to product review datasets and other social media platforms

This appendix provides supporting details for the methodology, implementation, and evaluation of the sentiment analysis system described in the report.

8. REFERENCES:

1. Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., & Lin, C. J. (2008). LIBLINEAR: A library for large linear classification. The Journal of Machine Learning Research, 9, 1871-1874.

2. E. Palogiannidi, A. Kolovou, F. Christopoulou, F. Kokkinos, E. Iosif, N. Malandrakis, H. Papageorgiou, S. S. Narayanan, and A. Potamianos, "Tweester at SemEval-2016 Task 4: Sentiment Analysis in Twitter Using Semantic-Affective Model Adaptation," in *Proc. Int. Workshop on Semantic Evaluation (SemEval)*, 2016.

3. L. Kong, N. Schneider, S. Swayamdipta, A. Bhatia, C. Dyer, and N. A. Smith, "A dependency parser for tweets," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

A.Harrison, T.Holding(2015) Natural Language Processing tutorials on his website(NLTK).

4. R. Wagh and P. Punde, "Survey on Sentiment Analysis using Twitter Dataset," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2018, pp. 208-211, doi: 10.1109/ICECA.2018.8474783. keywords: {Sentiment analysis;Twitter;Support vector machines;Conferences;Data mining;Machine learning;Sentiment Analysis;Opinion Mining;Social Media;Twitter Data},

5. V. Prakruthi, D. Sindhu and D. S. Anupama Kumar, "Real Time Sentiment Analysis Of Twitter Posts," 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), Bengaluru, India, 2018, pp. 29-34, doi: 10.1109/CSITSS.2018.8768774. keywords: {Twitter;Sentiment analysis;Conferences;Information technology;Data mining;Histograms;Bag of words;Sentiment Analysis;Tweets;Twitter API;Visualization},

6. H. Parveen and S. Pandey, "Sentiment analysis on Twitter Data-set using Naive Bayes algorithm," 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Bangalore, India, 2016, pp. 416-419, doi: 10.1109/ICATCCT.2016.7912034. keywords: {Twitter;Sentiment analysis;Algorithm design and analysis;Data mining;Computer architecture;Prediction algorithms;Hadoop;MapReduce;Twitter;Sentiment Analysis;Naïve Bayes},

7. P. Ray and A. Chakrabarti, "Twitter sentiment analysis for product review using lexicon method," 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), Pune, India, 2017, pp. 211-216, doi: 10.1109/ICDMAI.2017.8073512. keywords: {Sentiment analysis;Dictionaries;Twitter;Batteries;Organizations;Servers;Baita analysis;data preprocessing;data mining;sentiment analysis},

8. J. P. U. S. D. Jayakody and B. T. G. S. Kumara, "Sentiment analysis on product reviews on twitter using Machine Learning Approaches," 2021 International Conference on Decision Aid Sciences and Application (DASA), Sakheer, Bahrain, 2021, pp. 1056-1061, doi: 10.1109/DASA53625.2021.9682291. keywords: {Support vector machines;Sentiment analysis;Analytical models;Social networking (online);Blogs;Training data;Machine learning;Machine learning;SVM;Logistic regression;K-nearest neighbor;Sentiment analysis},

9. Z. Singla, S. Randhawa and S. Jain, "Sentiment analysis of customer product reviews using machine learning," 2017 International Conference on Intelligent Computing and Control (I2C2), Coimbatore, India, 2017, pp. 1-5, doi: 10.1109/I2C2.2017.8321910. keywords: {Support vector machines;Sentiment analysis;Predictive models;Decision trees;Mobile handsets;Feature extraction;Big Data;Big data;text mining;text classification;sentiment analysis;machine learning},

10. A. N. Rohman, R. Luviana Musyarofah, E. Utami and S. Raharjo, "Natural Language Processing on Marketplace Product Review Sentiment Analysis," 2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS), Manado, Indonesia, 2020, pp. 1-5, doi: 10.1109/ICORIS50180.2020.9320827. keywords: {Sentiment analysis;Classification algorithms;Support vector machines;Training

data;Tokenization;Social networking (online);Informatics;sentiment analysis;NLP;Text Mining;Naïve Bayes;K-NN},

11. J. A. Costales, C. M. De Los Santos, J. J. J. E. Catulay and M. G. Albino, "Sentiment Analysis for Twitter Tweets: A Framework to Detect Sentiment Using Naïve Bayes Algorithm," 2022 4th International Conference on Computer Communication and the Internet (ICCCI), Chiba, Japan, 2022, pp. 39-44, doi: 10.1109/ICCCI55554.2022.9850257. keywords: {Sentiment analysis;Social networking (online);Computational modeling;Blogs;Real-time systems;Internet;Reliability;twitter;social media;sentiment analysis;opinion mining;natural language processing;Naïve Bayes algorithm;Python},

12. A. Venkatachalam, R. D and S. N. S. S, "Movie Reviews using Sentiment Analysis with Natural Language API," 2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), Chennai, India, 2023, pp. 1-5, doi: 10.1109/ICONSTEM56934.2023.10142803. keywords: {Sentiment analysis;Social networking (online);Blogs;Natural languages;Software as a service;Knowledge based systems;Motion pictures;Natural Language API;movie reviews;sentiment analysis;lexical analyzer;twitter data},

13. J. A. Banados and K. J. Espinosa, "Optimizing Support Vector Machine in classifying sentiments on product brands from Twitter," IISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications, Chania, Greece, 2014, pp. 75-80, doi: 10.1109/IISA.2014.6878768. keywords: {Twitter;Training;Feature extraction;Sentiment analysis;Accuracy;Support vector machine classification;Support Vector Machine;Sentiment Analysis;Unigram Model},

14. T. P. Sahu and S. Ahuja, "Sentiment analysis of movie reviews: A study on feature selection & classification algorithms," 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, India, 2016, pp. 1-6, doi: 10.1109/MicroCom.2016.7522583. keywords: {Motion pictures;Sentiment analysis;Feature extraction;Entropy;Classification algorithms;Data mining;Speech;Feature selection;Movie Review;Sentiment Analysis;Information Retrieval;Opinion Mining;Classifier},

15. N. Raghuvanshi and J. M. Patil, "A brief review on sentiment analysis," 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, 2016, pp. 2827-2831, doi: 10.1109/ICEEOT.2016.7755213. keywords: {Sentiment analysis;Motion pictures;Social network services;Feature extraction;Blogs;Decision trees;Visualization;Sentiment analysis;Opinion based mining;Naïve Bayes classifier},

16. Lavanya T., Miraclin Joyce Pamila J. C. and Veningston K., "Online review analytics using word alignment model on Twitter data," 2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2016, pp. 1-6, doi: 10.1109/ICACCS.2016.7586388. keywords: {Sentiment analysis;Data mining;Twitter;Feature extraction;Blogs;Algorithm design and analysis;Opinion mining;contextual polarity;Twitter;sentiment analysis},

17. A. M. Rajeswari, M. Mahalakshmi, R. Nithyashree and G. Nalini, "Sentiment Analysis for Predicting Customer Reviews using a Hybrid Approach," 2020 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA), Cochin, India, 2020, pp. 200-205, doi: 10.1109/ACCTHPA49271.2020.9213236. keywords: {Feature extraction;Motion pictures;Sentiment analysis;Machine learning;Machine learning algorithms;Classification algorithms;Support vector machines;Sentiment analysis;Text Classification;SentiWordNet;Lexicon-approach;Neutral reviews},

18. A. Goel, J. Gautam and S. Kumar, "Real time sentiment analysis of tweets using Naive Bayes," 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), Dehradun, India, 2016, pp. 257-261, doi: 10.1109/NGCT.2016.7877424. keywords: {Sentiment analysis;Training;Training data;Twitter;Motion pictures;Databases;Tagging;Twitter;SentiWordNet;Machine Learning;NLTK;Python;Sentiment Analysis},