*A R DIGITECH*
*International Journal Of Engineering, Education And Technology (ARDIJEET)*
*www.ardigitech.in ISSN 2320-883X, VOLUME 13 ISSUE 01, 01/01/2025 pub: 26/02/2025*

# Deep Learning Based Sentiment Analysis On Social Media

[1]Ranjit Bhikaji Kamble, [2]Prof. Pallavi P. Rane, [3]Prof. Nilesh N. Shingne, [4]Prof. Harshal S. Deshpande
[1]rbk141291@gmail.com, [2]koltepallavi200@gmail.com, [3]shingne.nilesh236@gmail.com,
[4]harshaldeshpande4891@gmail.com
[1] M.E. CSE Student, Rajarshi Shahu College of Engineering, Buldhana, Maharashtra
[2,4]Assistant professor, Rajarshi Shahu College of Engineering, Buldhana, Maharashtra
[3]Assistant professor, Sanmati Engineering College, Washim, Maharashtra

## Abstract

Social media is a major data-sharing platform in today's world. Technological progress has made immense amounts of information available for analysis, making this a popular area of study. Users share their views on platforms like X (formerly Twitter), Facebook, and Instagram. X, in particular, is a rich source of data, making its analysis a priority. Sentiment analysis is a common method for classifying emotions in subjective text, using machine learning algorithms like Support Vector Machine, Naive Bayes, Long Short-Term Memory, and Decision Tree Classifier. This paper presents a general approach to X sentiment analysis using Flask. Flask's built-in capabilities are used to categorize text sentiment as positive, negative, or neutral, and to make API calls to the X Developer account for data retrieval. The analyzed data is then displayed on a webpage, showing a pie chart of the sentiment distribution (positive, negative, and neutral) for a given search term.
***Keywords: Data-sharing, Flask, Analysis, Data, Sentiment***

## 1. Introduction

"X Feed" can refer to both the social media platform and, separately, fabricated news or propaganda spread through various media, including traditional outlets like print and television, and online platforms. The goal of such fabricated content is typically to deceive readers, damage reputations, or undermine democratic processes. The rise of social media has amplified the spread of this misinformation, as it provides an open platform for sharing opinions and views, sometimes giving fabricated stories more visibility than original reporting. Research efforts are focused on mitigating the impact of this misinformation, which can range from outlandish claims to politically motivated falsehoods.
Artificial Intelligence (AI) and Natural Language Processing (NLP) are key tools in identifying and verifying information to combat the spread of fabricated news. However, accurate detection remains difficult. These technologies enable the development of systems that classify and authenticate news by comparing it to verified data. This review examines various methods for predicting fabricated news and generating accurate headlines and articles. By analysing related news and headlines, content can be categorized (e.g., "agree," "disagree," "conflict") and sentiment identified (positive, negative, or neutral).

### 1.1 Dataset Description: X Feed Challenge (FNC-1) Data

Datasets are essential for accurate detection of fabricated news. These datasets often contain words and phrases categorized by sentiment (positive, negative, neutral) and can be sourced from various platforms, including X, Facebook, online articles, and customer feedback. Some datasets include the main theme of news articles along with labelled information. Machine learning and deep learning algorithms use these datasets to train models. These trained models then compare and validate content against authentic data for reliable identification of fabricated news.
The main objective is to solve the issue coming with relative words. Distribution across some instance is presented in table 1.

**Table 1.Stance labels in training dataset**

| Stance category | Percentage | Description |
|---|---|---|
| Agree | 7.36% | Headline agrees with the claim made in the news article |
| Disagree | 1.68% | Headline disagrees with the claim made in the news article |
| Discuss | 17.82% | Headline discusses same topic as news article |

| | | |
|---|---|---|
| Unrelated | 73.13% | Headline does not discuss same topic as news article |

## 2. Related Work

M. Granik and V. Mesyura [3] propose a simple approach for incorrect news detection using naive Bayes classifier. It was implemented as an application system and tested in comparison with a dataset of instances which was generated through various medium. There classification accuracy for incorrect news is not perfect and detected only 4.9% of incorrect information.

H. Gupta, M. S. Jamal, S. Madisetty and M. S. Desarkar, [9] provides a framework based on different learning approach that impact with various problems including accuracy less percentage, time lag (BotMaker) and high processing time required to handle thousands of tweets in 1 sec. To do this they collected many samples of tweets and characterized them with spam tweets and derived lightweight features along with major positive, negative or neutral words. They were able to achieve an accuracy of 91.65% and surpassed the existing solution by approximately18%.

C. Buntain and J. Golbeck [10] design a method for automatic detection of X Feed on X Feed by self-learning for predicting accuracy in various trained dataset. They apply this method to for identifying retweeted threads and conversation and extract the features for classifying purpose.

S. B. Parikh and P. K. Atrey [11] aiming to present a realistic characteristics of news story in the current environment and combined with various related content. Studying such existing X Feed and creating the model for it helps to rectify the match content and rectify the actual news from the data.

Sobhani, P., Inkpen, D., try to design a framework for natural language processing for converting the textual data to machine readable format was achieved in this system. Whereas NLP is an area for computer science and artificial intelligence combination concerned and that processing of two technologies was used in the design structure [19].

Many NLP landscape was evolved at great occurrence and Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, M., Kuksa, P Collobert (2011) [22] proposed a Natural Language Processing similar from scratch which defines unified neural network architecture and its algorithms that are applied to various NLP tasks.

Also in consideration with that a pre-neural network techniques which focuses on developing extensive domain specific features was also introduced.

## 3. Problem Statement

Identifying fabricated news is challenging, primarily because of the need to understand current trends, which has led to extensive research in real-time information gathering. Even with curated datasets, detecting fabricated or inaccurate news is difficult due to rapidly changing methods and the dynamic nature of data collection. Integrating advanced technologies like Natural Language Processing (NLP) and machine learning is crucial for distinguishing unreliable information from credible sources. Fabricated news receives significant attention on social media, often fuelled by the political landscape and its negative consequences. The complexity of detecting such news makes it a serious concern for systems designed to combat misinformation. Furthermore, user engagement on social media platforms is a key factor. Analysing engagement patterns at both the individual and group levels is essential, as the growing number of active users adds to the complexity of the problem. This increased engagement highlights the need for robust systems to address the challenges posed by fabricated news.

## 4. Research Methodology

### 4.1 Data Collection and Preparation:

- Dataset Acquisition: Gather a relevant dataset of text data (e.g., reviews, tweets, articles) labeled with sentiment (positive, negative, neutral, or more granular emotions). Publicly available datasets like IMDB movie reviews, Stanford Sentiment Treebank, or Twitter sentiment datasets are often used.
    - Data Cleaning: Preprocess the text data. This may include:
    - Removing irrelevant characters, HTML tags, or URLs.
    - Handling missing values.
    - Converting text to lowercase.
    - Removing stop words (common words like "the," "a," "is" that don't carry much sentiment information).

- Stemming or lemmatization (reducing words to their root form).

- Data Splitting: Divide the dataset into training, validation, and test sets. A common split is 80% for training, 10% for validation, and 10% for testing. Stratified splitting is recommended to maintain class balance across the sets.
- Text Representation: Convert text data into numerical form that deep learning models can understand. Common methods include:
  - Word Embeddings (Word2Vec, GloVe, FastText): Represent words as dense vectors capturing semantic relationships.
  - Character Embeddings: Represent characters as vectors, useful for handling misspellings and out-of-vocabulary words.
  - TF-IDF (Term Frequency-Inverse Document Frequency): Weights words based on their frequency in the document and across the corpus.

**4.2 Model Selection and Design:**

- Deep Learning Architecture: Choose an appropriate deep learning architecture for sentiment analysis. Popular choices include:
- Convolutional Neural Networks (CNNs): CNNs can also be effective for text classification by identifying patterns of n-grams.

**4.3 Model Training and Evaluation:**

- Training: Train the chosen deep learning model on the training data. Use the validation set to monitor performance and prevent overfitting. Early stopping (stopping training when validation performance plateaus) can be used.
- Evaluation: Evaluate the trained model on the held-out test set to assess its generalization performance. Common metrics include:
  - Accuracy: The percentage of correctly classified instances.
  - Precision: The proportion of true positives among the predicted positives.
  - Recall: The proportion of true positives among the actual positives.
  - F1-score: The harmonic mean of precision and [1] recall.
  - AUC-ROC: Area under the Receiver Operating Characteristic curve.
  - Confusion Matrix: Visualize the model's performance by examining a confusion matrix, which shows the counts of true positives, true negatives, false positives, and false negatives.

**4.4 Model Deployment and Monitoring:**

- Deployment: Deploy the trained model for real-world use. This might involve creating an API or integrating the model into an application.
- Monitoring: Continuously monitor the model's performance after deployment to ensure it maintains accuracy and address any issues that arise. Retraining the model with new data may be necessary periodically.

## 5. Proposed Methods

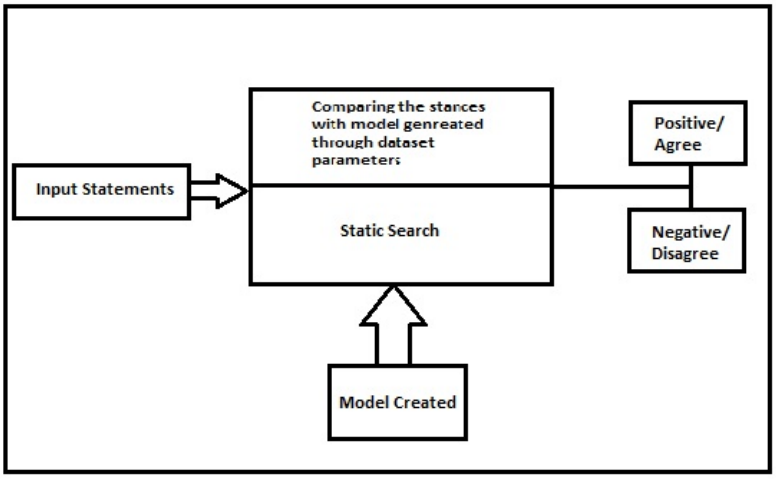


**Figure 1: System Design**

This project analyses sentiment and language for specific keywords in Twitter data, but several enhancements could broaden its functionality. Currently, it only processes tweet text. Future development could include other media types like images, videos, and multimedia. Hashtags could also be incorporated to improve sentiment categorization. Code optimization could reduce complexity and improve efficiency, leading to a faster and more user-friendly system. Visualizations could be enhanced for better interpretation, and the static webpage could be made dynamic for a more interactive experience. Downloadable results in formats like PDF or JPG would facilitate sharing. Integration with platforms like Tableau, R, or Power BI would further improve visualization quality and clarity. Migrating the project to Apache Spark could leverage its scalability and data visualization libraries for more engaging and accurate results. These enhancements are planned for future iterations of the project.

### Steps for the process

In static part, training and used 3 out of 4 Naïve Bayes algorithms for classification.
**Step 1**: In first step, extracting features from the already pre-processed dataset. These features are; Bag-of-words, positive and negative words.
**Step 2**: Here building all the classifiers for predicting the X Feed detection. The extracted features are fed into different classifiers. Using Naive-bayes algorithm and sklearn libraries. Each of the extracted features was used in all of the classifiers.
**Step 3**: Once fitting the model, comparing the f1 score and checked the confusion matrix.
**Step 4**: After fitting all the classifiers, best performing models were selected as candidate models for X Feed classification.
**Step 5**: Finally selected model was used for X Feed detection with the probability of truth.
**Step 6**: Our finally selected and best performing classifier was naives-bayes which was then saved on disk. It will be used to classify the X Feed.
It takes a news article as input from user then model is used for final classification output that is shown to user along with probability of truth.

## 6. Conclusion

Twitter sentiment analysis is becoming increasingly important in data analysis. As users globally share opinions and interpret information on the platform, its significance continues to grow. This paper presents a simple approach to tweet analysis using Flask. Tweets are collected via the Twitter API and Tweepy, then classified as positive, negative, or neutral, with the language of each tweet also identified. A user-friendly webpage, connected to Python code, allows users to input a term or phrase for analysis. The output includes sentiment details, the Twitter handle, and the tweet's timestamp. TextBlob simplifies preprocessing, ensuring efficient sentiment categorization. Flask was chosen to avoid using machine learning. By eliminating model training and testing, this method provides efficient results without relying on machine learning algorithms for accuracy. This approach facilitates real-time analysis and innovative methodologies, offering a fresh perspective on tweet analysis. This versatile system has the potential for broad application across industries. It provides a practical and accessible solution for companies and enhances user experience, making it a valuable tool for practitioners and stakeholders in various sectors.

### *References*

[1] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, “X Feed Detection on Social Media: A Data Mining Perspective” arXiv:1708.01967v3 [cs.SI], 3 Sep 2017
[2] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, “X Feed Detection on Social Media: A Data Mining Perspective” arXiv:1708.01967v3 [cs.SI], 3 Sep 2017
[3] M. Granik and V. Mesyura, "X Feed detection using naive Bayes classifier," 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), Kiev, 2017, pp. 900-903.
[4] X Feed websites. (n.d.) Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Fake_news_website. Accessed Feb. 6, 2017 [5] Cade Metz. (2016, Dec. 16). The bittersweet sweepstakes to build an AI that destroys X Feed.
[5] Conroy, N., Rubin, V. and Chen, Y. (2015). “Automatic deception detection: Methods for finding X Feed” at Proceedings of the Association for Information Science and Technology, 52(1), pp.1-4.
[6] Markines, B., Cattuto, C., & Menczer, F. (2009, April). “Social spam detection”. In Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web (pp. 41-48)
[7] Rada Mihalcea , Carlo Strapparava, The lie detector: explorations in the automatic recognition of deceptive language, Proceedings of the ACL-IJCNLP
[8] Kushal Agarwalla, Shubham Nandan, Varun Anil Nair, D. Deva Hema, “X Feed Detection using Machine Learning and Natural Language Processing,” International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7, Issue-6, March 2019

[9] H. Gupta, M. S. Jamal, S. Madisetty and M. S. Desarkar, "A framework for real-time spam detection in X Feed," 2018 10th International Conference on Communication Systems & Networks (COMSNETS), Bengaluru, 2018, pp. 380-383 [11] M. L. Della Vedova, E. Tacchini, S. Moret, G. Ballarin, M. DiPierro and L. de Alfaro, "Automatic Online X Feed Detection Combining Content and Social Signals," 2018 22nd Conference of Open Innovations Association (FRUCT), Jyvaskyla, 2018, pp. 272- 279.

[10] C. Buntain and J. Golbeck, "Automatically Identifying X Feed in Popular X Feed Threads," 2017 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, 2017, pp. 208-215.

[11] S. B. Parikh and P. K. Atrey, "Media-Rich X Feed Detection: A Survey," 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, 2018, pp. 436-441

[12] Scikit-Learn- Machine Learning In Python [15] Dataset- X Feed detection William Yang Wang. " liar, liar pants on _re": A new benchmark dataset for X Feed detection. arXiv preprint arXiv:1705.00648, 2017.

[13] Titcomb, J., Carson, J.: www.telegraph.co.uk. X Feed: What exactly is it – and how can you spot it?

[14] Allcott, H., Gentzkow, M.: Social media and X Feed in the 2016 election Technical report, National Bureau of Economic Research (2017)

[15] Langin, K.: http://www.sciencemag.org. X Feed spreads faster than true news on X Feed—thanks to people, not bots (2018)

[16] Wardle,C.: Fake News. It's complicated.First Draft New (2017). https://firstdraftnews.com/fake-news-complicated/

[17] Throne, J., Chen, M., Myrianthous, G., Pu, J., Wang, X., Vlachos, A.: 2017. X Feed Detection using Stacked Ensemble of Classifiers. In ACL.

[18] Davis, R., Proctor, C.: 2017. X Feed, Real Consequences: Recruiting Neural Networks for the Fight against X Feed. https://web.stanford.edu/class/cs224n/reports/2761239.pdf

[19] SemEval-2016: Semantic Evaluation Exercises International Workshop on Semantic Evaluation (SemEval-2016). Sobhani, P., Inkpen, D., Zhu, X.: A Dataset for Multi-Target Stance Detection. http://www.aclweb.org/anthology/E17-2088.

[20] Pang, B., Lee, L.: 2008. Opinion mining and sentiment analysis. http://www.cs.cornell.edu/home/llee/omsa/omsa.pdf

[21] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, M., Kuksa, P.: 2011. Natural Language Processing (Almost) from Scratch. Journal of Machine Learning Research 12 (2011) 2493-2537.

[22] Johnson, J.: 2016. The Five Types of X Feed. https://www.huffingtonpost.com/dr-john- johnson/the-five-types-of-fake-ne_b_13609562.html

[23] As X Feed spreads lies, more readers shrug at the truth. In https://www.nytimes.com/ 2016/12/06/us/ fake-news-partisan-republican-democrat.html.