

## Sentiment Analysis of Customer Review Using NLP

\*<sup>1</sup>Md Zubair Alam, <sup>2</sup>Muhammad Tarique, <sup>3</sup>Sufi Bilal Ahmad and <sup>4</sup>Shra Fatima

<sup>\*1,2,3</sup> Student, Department of Computer Science and Engineering, Integral University, Lucknow, Uttar Pradesh, India.

<sup>4</sup> Assistant Professor, Department of Computer Science and Engineering, Integral University, Lucknow, Uttar Pradesh, India.

### Article Info.

E-ISSN: 2583-6528

Impact Factor (QJIF): 8.4

Peer Reviewed Journal

Available online:

[www.alladvancejournal.com](http://www.alladvancejournal.com)

Received: 16/Jan/2026

Accepted: 15/Feb/2026

### \*Corresponding Author

**Md Zubair Alam**

Student, Department of Computer Science and Engineering, Integral University, Lucknow, Uttar Pradesh, India.

### Abstract

Sentiment evaluation has emerged as an important thing of natural language processing (NLP), permitting automatic interpretation of subjective statistics from textual data. This paper provides a complete study on sentiment evaluation using machine learning techniques to categorise text into superb, negative, and impartial classes. The proposed device employs preprocessing techniques inclusive of tokenization, stop word removal, stemming, and lemmatization to clean uncooked textual content statistics. Characteristic extraction is performed the usage of term Frequency-Inverse record Frequency (TF-IDF) and Bag-of-Words representations. A couple of type algorithms such as Logistic Regression, Naïve Bayes, support Vector Machines (SVM), and ensemble methods are evaluated. Experimental results reveal that the proposed approach achieves significant accuracy in sentiment classification, with SVM and Logistic Regression displaying superior performance. The device addresses demanding situations in managing noisy information, informal language, and area-particular expressions. This study contributes to the growing body of work in automated sentiment detection and provides insights for real-international applications in customer remarks analysis, social media monitoring, and emblem popularity control.

**Keywords:** Sentiment evaluation, herbal Language Processing, machine getting to know, Text Classification, feature Extraction, TF-IDF, help Vector Machines.

## 1. Introduction

The exponential growth of person-generated content material on social media systems, e-commerce websites, and online evaluate structures has created great quantities of textual facts expressing opinions, feelings, and sentiments [1]. Sentiment analysis, additionally called opinion mining, has come to be an important tool for extracting actionable insights from this unstructured data [2]. Organizations leverage sentiment analysis to apprehend purchaser perceptions, screen brand reputation, analyze market tendencies, and make information-driven selections [3].

Traditional guide analysis of consumer feedback is time-eating, subjective, and impractical for big-scale datasets [4]. Herbal Language Processing (NLP) blended with machine learning gives automatic solutions to classify sentiment polarity effectively and accurately [5]. The primary objective of sentiment analysis is to determine whether a given piece of text expresses superb, negative, or neutral sentiment [6].

### 1.1 Motivation

The motivation for this studies stems from several factors. First, organizations require actual-time sentiment monitoring to respond directly to patron worries and capitalize on positive remarks [7]. Second, the complexity of natural language-along with sarcasm, negation, context-structured meanings, and area-specific terminology-presents significant demanding situations that demand strong computational tactics [8]. Third, Sentiment analysis using Natural Language Processing: A system Learning Approach improvements in machine mastering and deep studying have opened new avenues for improving sentiment category accuracy [9].

### 1.2 Objectives

The number one targets of this research are:

- To develop an give up-to-end sentiment evaluation pipeline capable of processing raw textual content and classifying sentiment with excessive accuracy
- To evaluate and examine a couple of system getting to know algorithms for textual content class

- To investigate the impact of diverse preprocessing and function extraction techniques on version performance
- To pick out demanding situations and limitations in sentiment evaluation and suggest destiny improvements
- To create a sensible machine relevant to real-global scenarios together with purchaser assessment analysis.

### 1.3 Organization

The remainder of this paper is prepared as follows: segment II reviews associated work in sentiment analysis and NLP. Phase III describes the proposed method which include preprocessing, feature extraction, and type algorithms. Segment IV presents the experimental setup and datasets used. Phase V discusses consequences and overall performance assessment. Phase VI concludes the paper and descriptions future research directions.

## 2. Literature Review

Sentiment evaluation has advanced significantly over the last a long time, progressing from rule-based totally and lexicon-based techniques to sophisticated machine gaining knowledge of and deep gaining knowledge of techniques [10].

### 2.1 Traditional Approaches

Early sentiment evaluation structures trusted lexicon-based totally methods that used pre-defined dictionaries of high quality and negative words to compute sentiment rankings [11]. Even as interpretable, those approaches struggled with context-structured meanings, negations, and area-precise vocabulary [12]. Rule-based systems used 6e8e41b7b5d4d34eca2a6bc30dc8f87e linguistic rules to discover sentiment patterns however required vast manual effort and lacked scalability [13].

### 2.2 Machine Learning Approaches

The advent of machine studying transformed sentiment evaluation through allowing information-driven getting to know from categorized examples [14]. Classical algorithms which include Naive Bayes, help Vector Machines (SVM), and Logistic Regression have become famous due to their effectiveness and computational performance [15]. These strategies require careful feature engineering, with TF-IDF and Bag-of-phrases being the maximum broadly followed text representation techniques [16].

Current research by way of Estrella-Ramírez et al. established that SVM, Logistic Regression, and Random Forests obtain spectacular accuracy in textual content class obligations while mixed with suitable characteristic extraction techniques [17]. Their paintings highlighted that FASTTEXT embeddings produce advanced effects in comparison to conventional count Vectorization and TF-IDF techniques [17].

### 2.3 Deep Learning and Transformer Models

Deep state-of-the-art architectures, particularly Recurrent Neural Networks (RNN) and Long Short-term reminiscence (LSTM) networks, addressed limitations today's traditional methods by capturing sequential dependencies and lengthy-variety context in text [18]. Convolutional Neural Networks (CNN) have also been applied efficaciously to text category by learning nearby patterns and hierarchical capabilities [19].

The arrival present day transformer-based totally pre-skilled language fashions together with BERT (Bidirectional Encoder Representations from Transformers), RoBERTa, and GPT has revolutionized NLP obligations inclusive of sentiment evaluation [20]. Those fashions leverage large datasets and

bidirectional context to attain 49a2d564f1275e1c4e633abc331547db overall performance [21]. Albladi's comprehensive evaluate on Twitter sentiment analysis discovered that transformer fashions considerably out carry out conventional tactics in managing casual language, emojis, and context-established sentiments [22].

### 2.4 Preprocessing Impact

Research through Nesca *et al.* emphasized that preprocessing techniques appreciably have an impact on NLP set of rules performance [23]. Their scoping review documented that eighty five.4% of empirical research employed preprocessing to decorate model accuracy [23]. Powerful preprocessing includes tokenization, case normalization, prevent word removal, stemming, lemmatization, and coping with special characters [24].

### 2.5 Challenges in Sentiment Analysis

Despite enormous development, numerous demanding situations persist in sentiment analysis:

- Handling sarcasm and irony, where literal meaning differs from intended sentiment [25]
- Processing multilingual and code-mixed text inclusive of Hinglish (Hindi-English) [26]
- Coping with domain-precise terminology and evolving slang in social media [27]
- Handling class imbalance in datasets in which one sentiment magnificence dominates [28]
- Addressing bias in education information that could cause unfair or faulty predictions [29]

## 3. Proposed Methodology

This segment describes the proposed sentiment analysis gadget structure, which includes 4 important components: statistics collection, preprocessing, characteristic extraction, and type.

### 3.1 System Architecture

The overall machine architecture follows a pipeline approach in which raw textual content information flows through sequential processing ranges before final sentiment category.



### The Pipeline Contains

1. Facts series: gathering classified textual content records from various resources.
2. Preprocessing: cleansing and normalizing uncooked text.
3. Function Extraction: changing textual content to numerical representations.
4. Model education: studying class patterns from categorised records.
5. Assessment: Assessing version overall performance on check data.
6. Deployment: imposing the device for real-time predictions.

### 3.2 Data Collection

The gadget is designed to work with labeled sentiment datasets in which each text pattern is associated with a sentiment label (fine, terrible, or neutral). common information assets encompass:

- Product evaluations from e-commerce structures (Amazon, Flipkart)
- Movie critiques (IMDB dataset).
- Social media posts (Twitter, Reddit).
- Patron remarks and survey responses.

Statistics first-rate is crucial for model performance. The collected dataset should be balanced throughout sentiment training to prevent bias closer to most people magnificence.

### 3.3 Text Preprocessing

Preprocessing transforms uncooked, noisy textual content into clean, based facts suitable for device mastering. The subsequent preprocessing steps are implemented sequentially [30]:

**3.3.1 Tokenization:** The textual content is break up into individual phrases or tokens using whitespace and punctuation as delimiters. This converts sentences into sequences of discrete units that can be analyzed independently.

**3.3.2 Case Normalization:** All textual content is transformed to lowercase to ensure consistency. This prevents the model from treating "excellent" and "appropriate" as extraordinary words.

**3.3.3 Punctuation and Unique Man or Woman Elimination:** Non-alphanumeric characters, punctuation marks, and special symbols are removed as they normally do now not make a contribution to sentiment but add noise.

**3.3.4 Forestall Phrase Elimination:** Commonplace words inclusive of "the", "is", "and", "of" that seem frequently but deliver little sentiment information are removed. This reduces dimensionality and focuses on significant phrases [31].

**3.3.5 Stemming and Lemmatization:** Words are reduced to their root or base paperwork. Stemming applies rule-based truncation (e.g., "walking" → "run"), even as lemmatization makes use of vocabulary and morphological evaluation to go back dictionary paperwork (e.g., "higher" → "true") [32].

**3.3.6 Handling Negations:** Unique attention is given to negation words (not, in no way, no) as they opposite sentiment polarity. Strategies inclusive of negation tagging are carried out top reserve this crucial contextual statistics.

### 3.4 Text Preprocessing

Gadget mastering algorithms require numerical input, necessitating conversion of textual content to vector representations. Primary feature extraction methods are employed:

**3.4.1 Bag-of-phrases (BoW):** This approach represents textual content as a vector where every dimension corresponds to a word in the vocabulary, and the value represents the word's frequency in the document [33]. Even as easy and interpretable, BoW ignores phrase order and context.

**3.4.2 Term Frequency-Inverse Record Frequency (TF-IDF):** TF-IDF improves upon BoW by means of weighting words based totally on their significance [34]. It assigns higher weights to words which can be common in a record, however uncommon across the corpus, efficiently highlighting special words that deliver more sentiment records.

**The TF-IDF Rating for Term in Report from Corpus is calculated as:**

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D)$$

Where,

$$TF(t, d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$$

$$IDF(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Here,  $f_{t,d}$  is the frequency of the term  $t$  in document  $d$ ,  $|D|$  is the total number of documents, and  $|\{d \in D : t \in d\}|$  is the number of documents contain term  $t$ .

### 3.5 Classification Algorithm

A couple of supervised mastering algorithms are implemented and evaluated for sentiment type:

**3.5.1 Naive Bayes:** This probabilistic classifier applies Bayes' theorem with the assumption of characteristic independence [35]. In spite of its simplicity, Naive Bayes performs remarkably well for text type and serves as a robust baseline. The Multinomial Naive Bayes variant is especially desirable for discrete functions inclusive of word counts.

**3.5.2 Aid Vector Machines (SVM):** SVM constructs a hyper plane in high-dimensional characteristic space to maximize the margin between sentiment training [36]. It's far effective for excessive-dimensional text statistics and sturdy in opposition to over fitting. Linear kernel SVM is normally used for textual content classification due to computational performance and robust performance.

**3.5.3 Logistic Regression:** In spite of its call, logistic regression is a classification algorithm that models the opportunity of class membership the use of the logistic feature [37]. Its miles computationally green, interpretable, and plays properly with TF-IDF capabilities. Regularization strategies (L1, L2) help prevent overfitting.

**3.5.4 Random Forest:** This ensemble technique combines multiple selection timber to improve prediction accuracy and robustness [38]. Every tree is trained on a random subset of functions and statistics, and very last predictions are made with the aid of majority balloting. Random forest handles non-linear relationships and characteristic interactions effectively.

### 3.6 Model Training Hyperparameter Tuning

The dataset is cut up into education (70%), validation (15%), and test (15%) units. The schooling set is used to analyze version parameters, the validation set for hyperparameter tuning, and the test set for very last performance assessment. Hyperparameter optimization is executed using grid seek or random seek with move-validation. Key hyperparameters include:

- **For SVM:** kernel type, regularization parameter (C), gamma
- **For Logistic Regression:** regularization power, penalty type (L1/L2)
- **For Random Woodland:** number of bushes, most depth, minimum samples in line with leaf
- **For Naive Bayes:** smoothing parameter (alpha)

## 4. Experimental Setup

### 4.1 Dataset Description

Experiments are performed on publicly to be had sentiment analysis datasets to make sure reproducibility and enable assessment with existing paintings. The primary datasets used consist of:

**Table 1:** Dataset specs for sentiment analysis experiments

Dataset	Size	Classes	Domain
IMDB Reviews	50,000	2 (Pos/Neg)	Movies
Twitter Sentiment	10,000	3(Pos/Neg/Neu)	Social Media
Product Reviews	15,000	3(Pos/Neg/Neu)	E-commerce

Every dataset is preprocessed the use of the pipeline defined in phase III-C, and capabilities are extracted using both BoW and TF-IDF strategies.

### 4.2 Evaluation Metrics

Model overall performance is assessed the usage of trendy type metrics:

**4.2.1 Accuracy:** The share of efficiently labeled times:

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

**4.2.2 Precision:** The percentage of proper nice predictions amongst all high-quality predictions:

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

**4.2.3 Consider (Sensitivity):** The percentage of actual positives correctly recognized:

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

**4.2.4 F1-rating:** The harmonic mean of precision and recall, imparting a balanced degree:

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

For multi-elegance classification, macro-averaged and micro-averaged versions of these metrics are computed to account for class imbalance.

### 4.3 Implementation Report

The proposed device is implemented in Python using the subsequent libraries:

- Scikit-learn: for device gaining knowledge of algorithms and assessment metrics.
- NLTK (natural Language Toolkit): for preprocessing operations.
- Pandas and NumPy: for data manipulation.
- Matplotlib and seaborn: for visualisation.

Experiments are conducted on a machine with Intel Core i5 processor, 8GB RAM, running Ubuntu 20.04. Training instances and computational requirements are recorded for practical feasibility assessment.

## 5. Result and Discussion

### 5.1 Performance Comparison

The table beneath summarizes the overall performance of various type algorithms across information sets:

**Table 2:** Performance Comparison of Classification Algorithms

Algorithm	Accuracy	Precision	Recall	F1-Score
Naïve Bayes	82.3%	81.5%	82.1%	81.8%
Logistic Regression	88.7%	88.2%	88.5%	88.3%
SVM (Linear)	89.4%	89.1%	89.3%	89.2%
Random Forest	86.5%	86.0%	86.3%	86.1%

Outcomes indicate that SVM with linear kernel achieves the highest accuracy (89.four%), followed closely with the aid of Logistic Regression (88.7%) [39]. these findings align with current studies via Abladi *et al.*, who demonstrated that traditional gadget gaining knowledge of strategies stay competitive when combined with effective preprocessing and characteristic extraction [22].

### 5.2 Impact of Feature Extraction Methods

Contrast of BoW and TF-IDF characteristic representations exhibits that TF-IDF constantly outperforms BoW across all algorithms:

- TF-IDF with SVM: 89.4% accuracy
  - BoW with SVM: eighty five.7% accuracy
  - TF-IDF with Logistic Regression: 88.7% accuracy
  - BoW with Logistic Regression: eighty four.2% accuracy
- This development is attributed to TF-IDF's capacity to down-weight commonplace phrases and emphasize different phrases that deliver greater sentiment facts [40].

### 5.3 Preprocessing Impact Analysis

Ablation research had been performed to assess the contribution of individual preprocessing steps:

**Table 3:** Impact of preprocessing steps on classification accuracy.

Preprocessing Configuration	Accuracy
No preprocessing	76.2%
+ Tokenization	78.5%
+ Case normalization	80.1%
+ Stopword removal	84.3%
+ Stemming/Lemmatization	87.6%
Full pipeline	89.4%

Consequences display that every preprocessing step contributes positively to model performance, with forestall phrase elimination and stemming/lemmatization imparting the maximum widespread upgrades [41]. This supports findings from preceding studies emphasizing the vital role of preprocessing in NLP tasks [23].

### 5.4 Error Analysis

Analysis of misclassified examples exhibits commonplace failure styles:

- **Sarcasm and Irony:** "incredible task ruining my order!" labeled high-quality due to "superb" and "process"
- **Negation Handling:** "not terrible in any respect" misclassified as negative due to "not" and "awful"
- **Context-dependent Sentiment:** "This film is unwell!" where "ill" is fantastic slang.
- **Brief Texts:** inadequate context in very brief evaluations reduces accuracy.
- **Mixed Sentiment:** reviews expressing both positive and poor elements undertaking binary classification.

### 5.5 Computational Efficiency

Training and prediction times for different algorithms:

**Table 4:** Computational efficiency comparison.

Algorithm	Training Time (s)	Prediction Time (ms)
Naïve Bayse	2.3	5
Logistic Regression	12.5	8
SVM (Linear)	45.7	12
Random Forest	156.3	45

Naive Bayes gives the fastest training and prediction, making it appropriate for actual-time programs with resource constraints. SVM gives the great accuracy-pace alternate-off for most practical situations <sup>[42]</sup>.

### 5.6 Real-Word Application

The educated version turned into deployed as an internet application the use of Flask framework, permitting users to enter custom text and get hold of actual-time sentiment predictions with self-belief ratings. The gadget effectively processed user critiques from an e-trade platform, offering actionable insights for product improvement and customer support enhancement.

### 6. Conclusion and Future Works

This research provided a complete sentiment evaluation system the use of herbal language processing and gadget gaining knowledge of strategies. The proposed method demonstrated that careful preprocessing, effective feature extraction the usage of TF-IDF, and suitable algorithm selection can gain high accuracy in sentiment category obligations.

#### Key Findings Encompass

- SVM with linear kernel and Logistic Regression are distinctly effective for sentiment analysis, achieving accuracies above 88%
- TF-IDF function representation extensively outperforms easy Bag-of-phrases approach.
- Comprehensive preprocessing including tokenization, normalization, prevent phrase elimination, and lemmatization is vital for most advantageous overall performance.
- Traditional gadget gaining knowledge of strategies remain competitive and computationally green in comparison to deep gaining knowledge of techniques for lots practical applications.

#### 6.1 Limitations

Several barriers must be mentioned:

- The system struggles with sarcasm, irony, and context-structured expressions.
- Cutting-edge implementation focuses on English textual content; multilingual and code-combined text support is restricted.
- Binary and three-magnificence classification may additionally oversimplify complicated sentiment nuances.
- Domain model calls for retraining whilst shifting from one domain to every other.

#### 6.2 Future Research Directions

Future paintings will discover several promising instructions:

1. Imposing deep getting to know fashions such as LSTM, BiLSTM, and transformer-primarily based architectures (BERT, RoBERTa) to seize semantic context higher.
2. Extending to aspect-primarily based sentiment evaluation to identify sentiment towards specific product capabilities.

3. Incorporating emoji, emoticon, and hashtag analysis for social media sentiment.
4. Developing multilingual models to handle code-combined text which include English.
5. Adding give an explanation for potential capabilities using interest mechanisms or LIME/SHAP to interpret model selections.
6. Enforcing real-time sentiment monitoring with time-series analysis for trend detection.
7. Exploring few-shot and zero-shot mastering for area model with limited labelled information.

The proposed gadget gives a stable basis for sensible sentiment analysis packages and demonstrates the continued relevance of classical machine learning strategies in the technology of deep gaining knowledge of.

### References

1. Ferdous SM. *et al.* Sentiment analysis in the transformative era of machine learning and large language models. *Statistics, Optimization & Information Computing*. 2025; 12(4):1015-1039.
2. Geeks for Geeks. What is sentiment analysis? Machine learning approaches and applications, 2020. Retrieved from <https://www.geeksforgeeks.org/machine-learning/what-is-sentiment-analysis/>
3. Convin AI. Unlocking NLP sentiment analysis: Its impact and insights for business intelligence. Convin Blog, 2023. Retrieved from <https://convin.ai/blog/nlp-sentiment-analysis-insights>
4. Project Guru. Introduction to sentiment analysis in natural language processing (NLP), 2025. Retrieved from <https://www.projectguru.in/introduction-to-sentiment-analysis-in-natural-language-processing-nlp/>
5. Netguru. Introduction to sentiment analysis in NLP: Techniques and applications, 2025. Retrieved from <https://www.netguru.com/blog/sentiment-analysis-nlp>
6. Gigaspaces. What is sentiment analysis? Key components and challenges, 2025. Retrieved from <https://www.gigaspaces.com/data-terms/sentiment-analysis>
7. Sentisum. Customer sentiment analysis: Definition, DIY template, and best practices, 2024. Retrieved from <https://www.sentisum.com/customer-sentiment-analysis>
8. HelloIntern. Sentiment analysis interview questions and answers, 2025. Retrieved from <https://hellointern.in/blog/sentiment-analysis-interview-questions-and-answers-25327>
9. Codezup. NLP for sentiment analysis: A step-by-step guide to accuracy, 2025. Retrieved from <https://codezup.com/nlp-sentiment-analysis-step-by-step-guide/>
10. arXiv. Comprehensive study on sentiment analysis: From rule-based methods to large language models. arXiv:2409.09989, 2024.
11. Wikipedia. Sentiment analysis: History and approaches, 2006. Retrieved from [https://en.wikipedia.org/wiki/Sentiment\\_analysis](https://en.wikipedia.org/wiki/Sentiment_analysis)
12. Training Coders arts. 5 sentiment analysis project ideas for practical experience with machine learning, 2023

- Retrieved from <https://www.training.codersarts.com/post/5-sentiment-analysis-project-ideas-for-practical-experience-with-machine-learning>
13. Upgrad. Sentiment analysis projects with code for beginners in 2026. Retrieved from <https://www.upgrad.com/blog/sentiment-analysis-projects-topics-for-beginners/>
  14. Career Karma. Sentiment analysis projects for beginners and experts, 2021. Retrieved from <https://careerkarma.com/blog/sentiment-analysis-projects/>
  15. Interview Query. Top 12 sentiment analysis projects and datasets, 2022. Retrieved from <https://www.interviewquery.com/p/sentiment-analysis-projects-and-datasets>
  16. Project Pro. 10 sentiment analysis project ideas with source code, 2025. Retrieved from <https://www.projectpro.io/article/sentiment-analysis-project-ideas-with-source-code/518>
  17. Estrella-Ramírez J. *et al.* Automated machine learning for text classification using hyper-heuristics. *Engineering Applications of Artificial Intelligence*, 140, 108547, 2025.
  18. Programming World. How to perform sentiment analysis with Python and NLP, 2025. Retrieved from <https://www.programmingworld.tech/blog/how-to-perform-sentiment-analysis-with-python-and-nlp>
  19. Encord. Text classification: Techniques, advancements, and workflows, 2025. Retrieved from <https://encord.com/blog/text-classification/>
  20. GitHub. NLP-Sentiment: Sentiment analysis using natural language processing, 2019. Retrieved from <https://github.com/bmclannahan/NLP-Sentiment>
  21. Neptune AI. 10 NLP projects to boost your resume, 2023. Retrieved from <https://neptune.ai/blog/10-nlp-projects>
  22. Albladi A. *et al.* Sentiment analysis of Twitter data using NLP models: Asystematic review. *IEEE Access*, 13, 10883978, 2025. <https://doi.org/10.1109/ACCESS.2025.10883978>
  23. Nesca M. *et al.* A scoping review of preprocessing methods for unstructured text data in healthcare. *BMC Medical Informatics and Decision Making*. 2022; 23(1):185.
  24. Hassan Farid. Overview of NLP preprocessing techniques for sentiment analysis, 2024. Retrieved from <https://hassanfarid.hashnode.dev/overview-of-nlp-preprocessing-techniques>
  25. Geeks for Geeks. Text preprocessing in NLP: Essential techniques, 2024. Retrieved from <https://www.geeksforgeeks.org/nlp/text-preprocessing-for-nlp-tasks/>
  26. ACL Anthology. Sentiment analysis for low-resource languages, 2025. Retrieved from <https://aclanthology.org/2025.dravidianlangtech-1.48.pdf>
  27. Soft web Solutions. NLP for sentiment analysis: How it can impact businesses, 2025. Retrieved from <https://www.softwebsolutions.com/resources/nlp-for-sentiment-analysis/>
  28. The Smart Cube. Aspect-based sentiment analysis techniques and applications, 2021. Retrieved from <https://www.thesmartcube.com/ai-lab/experiments/aspect-based-sentiment-analysis/>
  29. Science Direct. Recent advancements and challenges of NLP-based sentiment analysis, 2025. Retrieved from <https://www.sciencedirect.com/science/article/pii/S2949719124000074>
  30. Geeks for Geeks. Aspect modelling in sentiment analysis, 2021. Retrieved from <https://www.geeksforgeeks.org/machine-learning/aspect-modelling-in-sentiment-analysis/>
  31. PyPI. Aspect-based sentiment analysis library documentation, 2021. Retrieved from <https://pypi.org/project/aspect-based-sentiment-analysis/>
  32. GitHub Topics. Sentiment analysis projects and repositories, 2024. Retrieved from <https://github.com/topics/sentiment-analysis>
  33. Learning Labb. Sentiment analysis using machine learning: Projects and how to do, 2025. Retrieved from <https://learninglabb.com/sentiment-analysis-using-machine-learning-project/>
  34. arXiv. Text classification using machine learning methods: Feature extraction comparison. arXiv:2502.19801, 2025.
  35. *Journal of Engineering Science*. Sentiment analysis using NLP: An objective and comparative study, 2024. Retrieved from <https://journal.esrgroups.org/jes/article/view/8180>
  36. Scribd. News articles sentiment analysis using natural language processing, 2025. Retrieved from <https://www.scribd.com/document/846222940/>
  37. IAPRESS. Sentiment analysis in the transformative era of machine learning, 2024. Retrieved from <http://www.iapress.org/index.php/soic/article/view/2113>
  38. arXiv. Are we really making much progress in text classification? A comparative analysis. arXiv:2204.03954v6, 2022.
  39. NCBI. A scoping review of preprocessing methods for unstructured text data. *PMC10476151*, 2023. Retrieved from <https://pmc.ncbi.nlm.nih.gov/articles/PMC10476151/>
  40. JMIR. The use of natural language processing to interpret unstructured patient feedback. *Journal of Medical Internet Research*. 2025; 27(1):e72853.
  41. arXiv. A novel approach to sentiment analysis using quantum machine learning. arXiv:2412.12731, 2024.
  42. Resume Worded. NLP natural language processing engineer resume examples, 2025. Retrieved from <https://resumeworded.com/nlp-natural-language-processing-engineer-resume-example>