ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue IX, September 2026

Deep Sequence Sentiment Modeling via Graph-Theoretic Negation Boundary Resolution and Multi-Classifier Ensemble for E-Commerce Opinion Spam Detection

Dr. Shrinivas Biradar[1], Dr. G T Raju[2], Divakar K M[3]

[1]Assoc. Professor, Dept. of Computer Science & Design, SJC Institute of Technology, Chickaballapur,Visvesvaraya Technological University, Belagavi, Karnataka, India

[2]Professor, Dept. of Information Science and Engineering, SJC Institute of Technology, Chickaballapur, Visvesvaraya Technological, University, Belagavi, Karnataka, India

[3]Research Scholar, Dept. of Computer Science and Engineering, SJC Institute of Technology, Chickaballapur,Visvesvaraya Technological University, Belagavi, Karnataka, India

DOI: https://doi.org/10.51583/IJLTEMAS.2026.150900140

Received: 11 October 2026; Accepted: 16 October 2026; Published: 23 October 2026

ABSTRACT

Traditional lexicon-based and frequency-domain sentiment analysis paradigms exhibit systemic vulnerabilities when processing syntactic negation windows, resulting in severe polarity inversion and degraded classification boundaries. While standard Bag-of-Words (BoW) frameworks dominate baseline Natural Language Processing (NLP), they inherently decouple sequence order, neutralizing contextual dependencies. This investigation introduces an Enhanced Negation Detection and Scope Resolution Framework tailored for unstructured electronic and social commerce streaming data. By formulating negation propagation as a deterministic parse-tree traversal algorithm, the system calculates shift coefficients over long-range dependencies, resolving semantic polarity inversions before feature extraction. The methodology evaluates a 62,000-record dataset spanning multi-channel mobile hardware reviews. A robust Multi-Classifier Ensemble—incorporating Multinomial Naïve Bayes, Bernoulli Naïve Bayes, and Logistic Regression with Sigmoid mapping, and Linear Support Vector Classifiers with hinge-loss optimization, and Stochastic Gradient Descent—is deployed over parameterized validation distributions. Empirical metrics demonstrate that a prioritized 80:20 training-to-testing tensor partition combined with our negation handler yields optimal convergence, with Logistic Regression achieving an absolute peak accuracy of 82.74%. These results provide an empirically verifiable blueprint for protecting e-commerce platforms from malicious opinion spamming and manipulative market dynamics.

Keywords: Computational Linguistics, Opinion Spam Detection, Negation Scope Resolution, Stochastic Gradient Descent, Supervised Ensemble Topologies, Convex Optimization.

INTRODUCTION

The contemporary digital architecture is characterized by an exponential expansion of unstructured, user-generated web content, generating multi-quintillion-byte streaming data arrays daily. Within this landscape, electronic commerce networks rely heavily on customer feedback arrays to calibrate predictive market models. However, standard text mining engines face significant bottlenecks due to intrinsic linguistic phenomena, primarily domain dependency, implicit semantic orientation, and negation handling.

Negation tracking presents a critical bottleneck in sequence classification. When a negation operator (e.g., not, never, cannot) is introduced into a text segment, it projects a downstream modification field—known as the negation window—that can completely invert the underlying polarity of sentiment-bearing tokens. Standard Bag-of-Words (BoW) representation vectors discard token position matrices, completely ignoring these polarity shifts. Consequently, sophisticated malicious actors exploit these linguistic vulnerabilities by

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue IX, September 2026

generating deceptive positive feedback sequences that encode underlying negative sentiments, effectively bypassing traditional fraud detection filters.

This study systematically addresses the challenge of polarity inversion in deceptive opinion spam. The technical contributions of this work are fourfold:

1) We formalize a deterministic tree-traversal negation scope resolution algorithm that computes semantic inversion across localized constituent nodes.

2) We clean and construct a specialized 62,000-record streaming review corpus focused on the smartphone domain utilizing automated web-scraping interface engines.

3) We execute a comprehensive mathematical evaluation of six discrete machine learning optimization routines to isolate optimal decision boundaries.

4) We perform rigorous hyperparameter partition tuning (70:30, 75:25, and 80:20) to map model convergence characteristics and eliminate overfitting states.

Mathematical Formalization of Classifiers And System Architecture

The analytical pipeline processes raw text strings through a deterministic pre-processing layer, constructs numerical token tensors, resolves negation fields, and executes classification via an ensemble system.

The Data Pre-processing Pipeline

Let the raw input corpus be represented as a set of documents D = {d1, d2... dm}. Each document undergoes a sequential morphological reduction pipeline:

d_hat_i = psi_lemma(psi_token(psi_stop(psi_punct(psi_HTML(d_i))))) (1)

Where psi_HTML maps the text to an empty string space for all regular expression sequences matching HTML tags, psi_punct strips non-alphanumeric ASCII vectors, psi_stop filters out high-frequency closed-class lexical items based on standard NLTK stop-word matrices, psi_token splits continuous text strings into individual tokens, and psi_lemma projects inflected word forms to their standard dictionary base form using morphological context analysis.

Multinomial Naïve Bayes Classification

The Multinomial Naïve Bayes model computes class probabilities by modeling token counts within text documents under an assumption of conditional independence. Given a document tensor d represented by feature counts, the posterior probability for class c is formulated as:

P(c | d) = [P(c) * PROD_ {j=1} ^n P (x_j | c) ^ (f_j)] / P (d) (2)

Where f_j represents the frequency weight of token x_j within document d. To prevent zero-probability bottlenecks across unobserved terms, Laplacian smoothing is applied to the feature conditional probability parameters:

P (x_j | c) = (N_jc + alpha) / (N_c + alpha * |V|) (3)

Where N_jc is the cumulative count of feature x_j in class c, N_c is the aggregate count of all features in class c, |V| is the total vocabulary dimensions, and alpha = 1 sets the default smoothing parameter.

Bernoulli Naïve Bayes Classification

For feature spaces restricted to multivariate Bernoulli distributions, feature presence or absence is modeled

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue IX, September 2026

as a binary state vector B. The characteristic decision probability calculation is expressed as:

$$P(c \mid d) \sim P(c) * PROD_{j=1}^{n} [b_j * P(x_j \mid c) + (1 - b_j)(1 - P(x_j \mid c))] \quad (4)$$

Where b_j is a binary indicator variable tracking the occurrence (b_j=1) or non-occurrence (b_j=0) of the target feature x_j.

Logistic Regression with Sigmoid Mapping

Logistic Regression projects linear combinations of feature vectors into a bounded probability distribution utilizing a continuous, differentiable transcendental function. Let $z = w^T * x + b$, where w represents the primary parameter weight vector and b acts as the bias scalar component. The functional mapping is derived via:

$$sigma(z) = 1 / (1 + e^{(-z)}) = 1 / (1 + e^{-(w^T * x + b)}) \quad (5)$$

The optimization routinely updates the weights by minimizing the negative log-likelihood function governed by Binary Cross-Entropy Loss optimization metrics.

Linear Support Vector Classifier (SVC)

The core objective of the Linear Support Vector Machine is to maximize the geometric margin separating the binary classes while introducing a soft-margin penalty for misclassification. The structural optimization target optimizes a convex objective function governed by hinge loss parameters:

$$\min_{w, b} [(1/2) * ||w||^2 + C * SUM_{i=1}^{N} \max(0, 1 - y_i(w^T * x_i + b))] \quad (6)$$

Where C serves as the regularization hyperparameter that balances margin maximization against training error tolerances.

Stochastic Gradient Descent (SGD) Optimization

For large-scale data arrays, computing full-batch gradients becomes computationally prohibitive. The SGD optimization engine evaluates the empirical loss function iteratively over randomized single samples or mini-batches:

$$w^{(k+1)} = w^{(k)} - eta * grad_w L_i(w^{(k)}) \quad (7)$$

Where eta > 0 represents the dynamically decaying step-size learning rate, and L_i indicates the local loss value computed for the i-th randomized sample vector.

Ensemble Voted Topology

The ensemble configuration aggregates the output arrays of the baseline models to construct a more resilient voting architecture. For a hard-voting classification framework, the final prediction function is determined by a majority threshold:

$$Y_hat = mode(M_MNB(x), M_BNB(x), M_LR(x), M_SVC(x), M_SGD(x)) \quad (8)$$

III. ALGORITHMIC NEGATION SCOPE RESOLUTION

Negation phenomena are systematically categorized into syntactic negation (explicit lexical markers like no, not, deny) and morphologic negation (affixation strings using semantic prefixes such as un-, in-, dis-). To track negation propagation, we construct a parse tree T for each processed statement segment. Let score(n) define the sentiment vector returned from the SentiWordNet lexical structure for node n.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue IX, September 2026

Mathematical Logic Formulation

Let V_L represent the set of active leaf nodes located downstream from a matched parent operator. If a node satisfies the boolean function neg_word(node) -> True, the sentiment score optimization operates via two operations:

1) The base operator node has its absolute polarity zeroed out to prevent redundant extraction weights: score(node) = 0

2) The downstream directional right-sibling leaf array elements n in r_leafs(node) undergo a linear transformation: score(n) = score(n) * (-1)

B. Programmatic Execution Structure

The explicit algorithmic logic for processing these shifts across the recursive sentence tree nodes is defined in the foal structure below:

```
procedure ProcessNegationScope(node: TreeNode; var residual_window: Integer);
begin
  if IsLeafNode(node) and (residual_window > 0) then
  begin
    if EvaluateNegationMarker(node) then
    begin
      SetNodeSentimentScore(node, 0.0);
      foreach sibling_node in ExtractRightLeaves(node) do
      begin
        InvertNodePolarity(sibling_node);
      end;
      residual_window := residual_window - 1;
    end;
  end
  else
  begin
    foreach child_node in GetNodeChildren(node) do
    begin
      ProcessNegationScope(child_node, residual_window);
    end;
  end;
end;
```

Empirical Evaluation and Analytical Results

Corpus Parameters

The system was validated using a 62,000-record mobile electronics dataset collected via the SNScrape API interface. The feature engineering framework relies on SentiWordNet definitions to calculate base word tokens into positive and negative index matrices.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue IX, September 2026

Negation Resolution Metrics

To establish baseline performance values, the negation handling modules were evaluated across all six machine learning architectures. Table I isolates the exact accuracy profiles obtained during this verification phase.

TABLE I. Negation Handling Baseline Accuracy Profiles

Machine Learning Model Topology	Model Accuracy (%)
Multinomial Naïve Bayes Classifier	72.8096
Bernoulli Naïve Bayes Classifier	74.3202
Logistic Regression Classifier	74.6223
Linear Support Vector Classifier	72.6586
Stochastic Gradient Descent Classifier	73.716
Consolidated Voted Ensemble Framework	74.1691

Evaluation Over Variable Data Distributions

To optimize structural generalization boundaries and eliminate overfitting trends, the dataset was processed across three distinct training-to-testing ratio configurations (70:30, 75:25, and 80:20). Performance matrices were tracked using standard information retrieval metrics across the execution domains.

TABLE II. Fake Review Detection Results across Parameter Partitions

Partition	ML Classifier Base	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
70% Train / 30% Test	Naïve Bayes	79.775	86.7625	91.325	88.9852
	Logistic Regression	81.675	82.475	91.16	86.6002
	Support Vector Machine	79.775	86.7625	91.325	88.9852
75% Train / 25% Test	Naïve Bayes	80.1125	86.5625	91.75	89.0806
	Logistic Regression	81.25	81.8375	91.16	86.2474
	Support Vector Machine	78.9	86.3875	91.325	88.7876
80% Train / 20% Test	Naïve Bayes	81.3125	87.9375	92.4375	90.1312
	Logistic Regression	82. 7375	82.7625	91.16	86.7582
	Support Vector Machine	80.0625	86.6875	91.4375	88.999

Data Stratification Analysis

An comparative evaluation across the configurations proves that the 80:20 training-to-testing layout delivers optimal system performance. While constrained training allocations (70%) induce minor underfitting states due to insufficient sample representation, expanding the training allocation to a robust 80% optimization window provides the network parameters with enough linguistic context to capture subtle negation boundaries. This distribution balances model generalization against overfitting constraints, yielding an absolute peak accuracy of 82.7375% when using the Logistic Regression pipeline.

CONCLUSION

This investigation demonstrates that incorporating a deterministic, graph-theoretic negation handling layer into text processing pipelines successfully addresses the issue of polarity inversion within e-commerce

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue IX, September 2026

review corpora. Standard sequence analysis methods consistently fail when encountering complex negation fields, leaving online platforms vulnerable to deceptive opinion spamming. By isolating negation scope windows and executing directional transformations on downstream token sentiment values before classification modeling, our methodology establishes clear, highly accurate decision boundaries.

Across all evaluated scenarios, the Logistic Regression model optimization routine systematically outperformed alternative machine learning structures. It demonstrated superior performance due to its continuous, differentiable sigmoid mapping design and resilient convergence behavior when processing multi-dimensional text data. Future research directions will focus on replacing manual lexicographical dictionary evaluations with automated, deep contextual embeddings using transformer-based attention layer architectures to further improve classification boundaries.

REFERENCES

1. M. S. Kumar, V. S. Raj, and P. Sharma, "Fake Review Detection System: A Machine Learning Approach with Advanced Linguistic Cues," in Proceedings of the IEEE International Conference on Advanced Computing, 2025, pp. 112–119.
2. A. R. Joshi, S. P. Patel, and N. M. Mehta, "Enhancing Fake Review Detection Using Linguistic Exaggeration, BERT Embeddings, and Fuzzy Logic," IEEE Access, vol. 13, pp. 4521–4533, 2025.
3. K. L. Naidu and R. K. Singh, "Fake Review Detection in E-Commerce Using Machine Learning and NLP Technique," in Proceedings of the International Conference on Innovations in Data Science, 2025, pp. 88–95.
4. T. H. Nguyen and P. Hajek, "Psychological and Linguistic Approach for Fake Review Detection Using Sentiment and Emotional Consistency," IEEE Trans. Comput. Soc. Syst., vol. 12, no. 2, pp. 310–324, Apr. 2025.
5. S. Khan and M. Ali, "Improving Sentiment Analysis Using Negation Scope Detection and Negation Handling via Machine Learning," Int. J. Cognitive Comput., vol. 22, no. 1, pp. 45–59, Jan. 2024.
6. H. Zhang, L. Wang, and Y. Liu, "Fake Review Detection Model Based on Comment Content and Reviewer Behavior," IEEE Access, vol. 12, pp. 98124–98138, 2024.
7. R. Gupta and S. Chakraborty, "Machine Learning Approach for Negation Handling in Sentiment Analysis Using Linguistic Features," in Proceedings of the 5th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2024, pp. 1–7.
8. M. Fatima and A. Ahmad, "Negation Handling for Sentiment Analysis Task: Approaches and Performance Analysis," Comput. Sci. Rev., vol. 51, p. 100612, Feb. 2024.
9. J. R. Smith and L. M. Garcia, "Enhancing Fake News and False Opinion Detection with Sentiment Analysis Using Machine Learning Classifiers," in Proceedings of the IEEE Seventh International Conference on Image Information Processing (ICIIP), 2024, pp. 241–246.
10. S. Pandey, S. Sagnika, and B. S. P. Mishra, "A Technique to Handle Negation in Sentiment Analysis on Movie Reviews," 12th International Multi-Conference on Information Processing, 2016.
11. K. Thevar, V. Thakur, T. Shukla, D. Vishwakarma, and K. Ghag, "Handling Polarity Shift for Opinion Mining," in Proceedings of the IEEE International Conference on Computing Methodologies and Communication (ICCMC), 2017.
12. C. Diamantini, A. Mircoli, and D. Potena, "A Negation Handling Technique for Sentiment Analysis," International Conference on Collaboration Technologies and Systems, 2016.
13. W. Sharif, N. A. Samsudin, M. M. Deris, and R. Naseem, "Effect of Negation in Sentiment Analysis," 6th International Conference on Innovative Computing Technology, 2016.
14. P. Sharma and N. Mishra, "Feature level sentiment analysis on movie reviews," 2nd International Conference on Next Generation Computing Technologies (NGCT), 2016.
15. K. V. Ghag and K. Shah, "Negation Handling for Sentiment Classification," International Conference on Computing Communication Control and Automation (ICCUBEA), 2016.
16. C. T. C and S. Joseph, "A syntactic approach for aspect based opinion mining," in Proceedings of the IEEE 9th International Conference on Semantic Computing (IEEE ICSC), 2015.

ISSN 2278-2540 | DOI: 10.51583EMAS | Volume XV, Issue IX, September 2026

17. L.-F. Huang, X. Liu, and V. Ng, "Associating sentimental orientation of Chinese neologism in social media data," IEEE 19th International Conference on Computational Systems and Communications (ICCSC), 2014.
18. V. Kharde and S. Sonawane, "Sentiment analysis of twitter data: a survey of techniques," arXiv preprint arXiv:1601.06971, 2016.
19. Twitter Usage Statistics, https://about.twitter.com/company, December 2015.
20. S. K. Yadav, "Sentiment analysis and classification: A survey," International Journal of Advance Research in Computer Science and Management Studies, 2015.
21. A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," Expert Systems with Applications, vol. 57, pp. 117-126, 2016.
22. S. Biradar, G. T. Raju, and D. K. M, "Twitter Data Pre-Processing and Detection of Fake Reviews," Seybold Report Journal, vol. 19, no. 1, Jan. 2024.
23. S. Biradar, G. T. Raju, and D. K. M, "Negation Handling and Domain Generalization in Sentiment Analysis Using Machine Learning Models," IEEE International Conference on Knowledge Engineering and Communication Systems, Chikkaballapur, India, 2024.
24. T. Singh and M. Kumari, "Role of Text Pre-Processing in Twitter Sentiment Analysis," 12th International Multi-Conference on Information Processing, 2016.
25. M. A. Hadi et al., "Go-Food Sentiment Analysis Using Twitter Data, Compared the Performance of the Random Forest Classifier with That of the Linear Support Vector Classifier," MIMSE-I-C 2022, ACSR 102, pp. 3-3, 2022.
26. P. Ficamos and Y. Liu, "A Topic based Approach for Sentiment Analysis on Twitter Data," International Journal of Advanced Computer Science and Applications, vol. 7, no. 12, 2016.
27. Y. Qi and Z. Shabrina, "Sentiment Analysis using Twitter data: a complete application of lexicon- and machine-learning-based approach," Springer, 2023.
28. M. Hamza and S. Gupta, "A Comparison of Sentiment Analysis Classifiers on Twitter Data using Machine Learning," 2018.
29. A. Krouska, C. Troussas, and M. Virvou, "The Effect of preprocessing techniques on Twitter Sentiment Analysis," 2016.
30. H.-T. Duong and T.-A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," Springer, 2020.
31. K. Kim, "An improved semisupervised dimensionality reduction using feature weighting: Application to sentiment analysis," Elsevier, 2023.
32. J. Salminen, C. Kandpal, A. M. Kamel, S.-G. Jung, and B. J. Jansen, "Creating and detecting fake reviews of online products," Elsevier, 2022.
33. N. C. S. Ram, G. Vakati, J. V. Nadimpalli, Y. Sah, and S. K. Datla, "Fake Reviews Detection Using Supervised Machine Learning," International Journal for Research in Applied Science & Engineering Technology, vol. 10, no. 5, May 2022.
34. S. Yadav, Dr. G. Dharmela, and K. Mistry, "Fake Review Detection Using Machine Learning Techniques," JETIR, vol. 8, no. 4, April 2021.
35. M. Ennaouri and A. Zellou, "Fake Reviews Detection through Machine learning Classifiers: A Systematic Literature Review," Springer, 2021.
36. H. Le and B. Kim, "Detection of Fake Reviews on Social Media Using Machine Learning Classifiers," Issues in Information Systems, vol. 21, no. 1, pp. 185-194, 2020.
37. M. Crawford, T. M. Khoshgoftaar, J. D. Prusa, A. N. Richter, and Hamzah, "Survey of Review Spam detection Using Machine Learning Techniques," Springer, 2015.
38. G. Bathla, P. Singh, R. K. Singh, E. Cambria, and R. Tiwari, "Intelligent fake reviews detection based on aspect extraction and analysis using deep learning," Springer Nature, 2022.
39. D. V. Lindberg and H. K. H. Lee, "Optimization under constraints by applying an asymmetric entropy measure," J. Comput. Graph. Statist., vol. 24, no. 2, pp. 379–393, Jun. 2015.
40. B. Rieder, Engines of Order: A Mechanology of Algorithmic Techniques. Amsterdam, Netherlands: Amsterdam Univ. Press, 2020.