International Journal of Research in Industrial Engineering

IJRIE

www.riejournal.com

Int. J. Res. Ind. Eng. Vol. x, No. x (xx) x–x.

Paper Type: Original Article

Neuro-Fuzzy Text Sentiment Analysis: Bridging Machine Learning Accuracy with Human-Interpretable Outputs

Sampath, Magapu [1],*, Charan Tej Goud, V [2],*, Anil Sai, Vallabhu [3],*, Haripreeth, Pinni [4],*

[1] Dept. of SCOPE, VITAP-University, India; sampath.22mis7215@vitapstudent.ac.in;
[2] Dept. of SCOPE, VITAP-University, India; charantej.22mis7065@vitapstudent.ac.in;
[3] Dept. of SCOPE, VITAP-University, India; anil.22mis7150@vitapstudent.ac.in;
[4] Dept. of SCOPE, VITAP-University, India; preeth.22mis7250@vitapstudent.ac.in;

Citation:

Received: ----- *Revised: -----* *Accepted:---*	Magapu, S., Goud, C. T., Vallabhu, A. S., & Pinni, H. (2025). *Neuro-Fuzzy Text Sentiment Analysis: Bridging Machine Learning Accuracy with Human-Interpretable Outputs*. **International Journal of Research in Industrial Engineering**, Volume(Issue), pp–pp.

Abstract

In today's world, millions of people share opinions every day through reviews, comments, and social media posts. Understanding the emotion behind this text is essential for businesses, researchers, and platforms that monitor public feedback. However, most sentiment-analysis systems simply classify text as "positive" or "negative," which does not fully reflect how humans express feelings. Our work introduces a **Neuro-Fuzzy Sentiment Analysis system** that combines the accuracy of machine learning with the interpretability of fuzzy logic. First, Logistic Regression predicts how positive or negative a text is. Then, fuzzy membership functions convert this into **more human-like sentiment levels** such as *Strong Positive*, *Somewhat Positive*, *Neutral*, *Somewhat Negative*, and *Strong Negative*. We also built an interactive web application that visually shows how the sentiment was determined. Experiments using the IMDB movie review dataset show that the model achieves over **90% accuracy**, while also providing transparency in how it makes decisions.

Keywords: Sentiment Analysis; Fuzzy Logic; Explainable AI; Machine Learning; Human-interpretable Systems; Hybrid AI

Corresponding Author:

1 | Introduction

People often rely on the opinions of others when making decisions such as selecting a movie to watch, purchasing a product, or choosing a place to dine. When reading reviews, humans naturally understand not only whether the opinion is positive or negative, but also the *intensity* of the sentiment. For instance, the statement *"It was okay"* is interpreted as neutral, *"Absolutely loved it!"* reflects a strong positive feeling, while *"Not great, but not terrible"* conveys a slightly negative tone. However, most existing sentiment-analysis models overlook these subtle differences and typically assign only binary labels like positive or negative. This rigid interpretation can result in misclassification, particularly in cases where the sentiment expressed is mild, mixed, or ambiguous.

The primary motivation of this research is to bridge the gap between how humans perceive emotional intensity and how machines classify sentiment. To achieve this, the proposed work integrates machine-learning techniques, which provide reliable sentiment probability scores, with fuzzy logic, which translates these numeric probabilities into human-like qualitative sentiment levels. This hybrid Neuro-Fuzzy approach enhances both the accuracy and interpretability of sentiment analysis, bringing machine predictions closer to the way humans naturally understand emotional expression.

1.1 | Figures and Tables

Figures and tables included in this paper are used to visually explain the workflow of the Neuro-Fuzzy Sentiment Analysis system and to clearly present the experimental results. Each figure and table is placed close to the part of the text where it is discussed to maintain clarity in reading.

Figures mainly show the Streamlit user interface, the fuzzy membership visualization, and the overall system architecture. These help in understanding how the machine learning probability output is interpreted into human-like sentiment categories. Tables are used to summarize dataset splits, output examples, and final prediction results.

All figures are centered and have captions placed directly **below** them. All tables have captions placed directly **above** them. Figures are numbered sequentially as Fig. 1, Fig. 2, Fig. 3, and so on. Tables are numbered as Table 1, Table 2, etc., independent of figure numbering. Table text is kept clear, left-aligned, and readable.

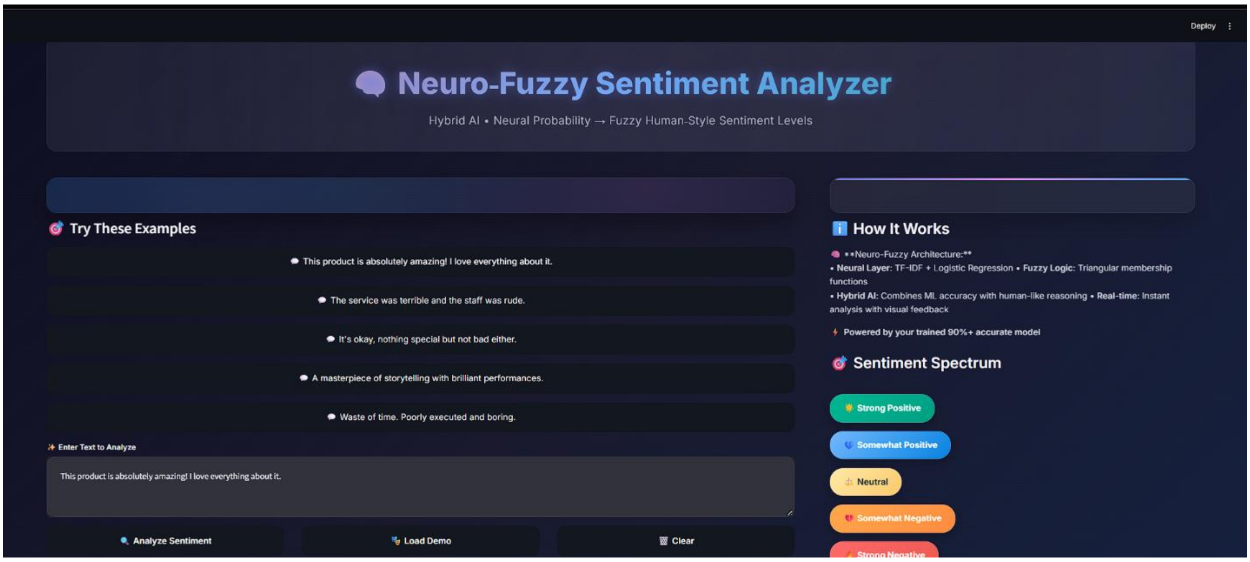


Fig. 1. User interface of the Neuro-Fuzzy Sentiment Analyzer.

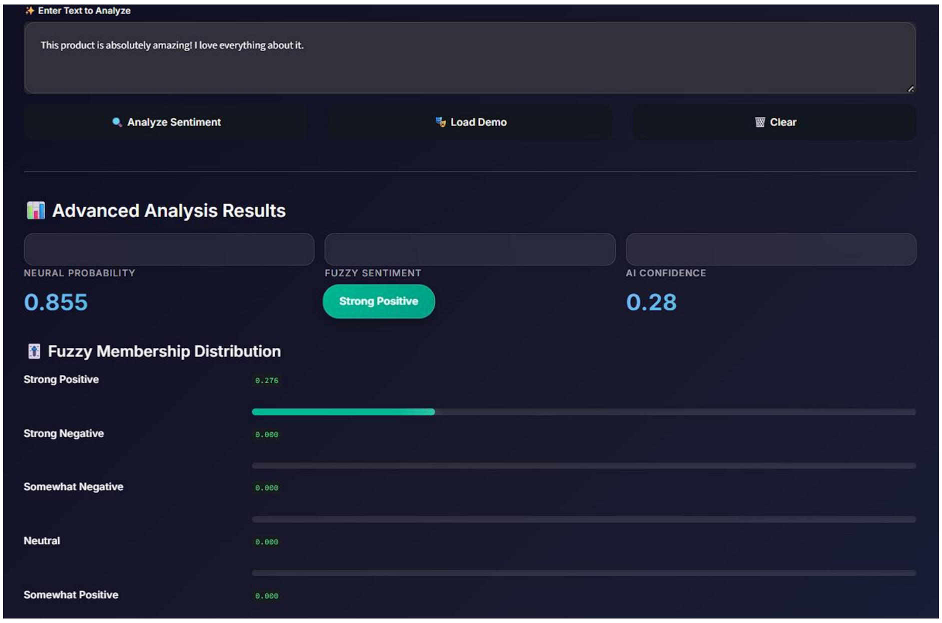


Fig. 2. Fuzzy membership visualization showing sentiment intensity levels.

Input Text
↓
Text Preprocessing (cleaning, tokenization, stopword removal)
↓
TF-IDF Feature Extraction
↓
Logistic Regression Classifier
↓ (outputs probability p)
↓
Fuzzy Membership Functions
↓
Sentiment Level Decision
↓
Final Output (e.g., Strong Positive, Neutral, Strong

Fig. 3. ***System workflow diagram demonstrating machine learning and fuzzy inference integration.***

Table 1. *Dataset distribution across training, validation, and testing phases.*

Dataset Split	Number of Samples	Percentage
Training	40,000	80%
Validation	5,000	10%
Testing	5,000	10%

Table 2. ***Example input texts with predicted sentiment levels.***

Input Text Example	Probability (p)	Final Sentiment Output
"Absolutely fantastic movie!"	0.89	Strong Positive
"The story was dull and slow."	0.12	Strong Negative
"It was okay, nothing special."	0.51	Neutral

1.1.1 | Variables and equations

In the proposed Neuro-Fuzzy Sentiment Analyzer, the Logistic Regression model is first used to estimate the probability that a given text expresses positive sentiment. The probability is calculated as:

$$P(y = 1 \mid x) = 1 / (1 + e\hat{}(-(w\hat{}T\ x + b))) \qquad \textbf{(1)}$$

Where:

- x= Feature vector obtained from TF-IDF
- w= Weight vector learned during training
- b= Bias term
- $P(y = 1 \mid x)$= Probability that the sentiment is positive

This probability value is then passed to the **fuzzy inference layer**. The fuzzy membership for a given sentiment class is computed using a **triangular membership function**:

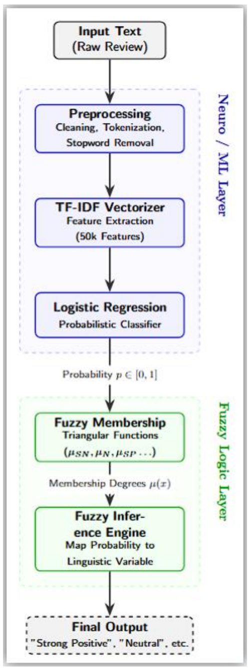
$$\mu\_A(p) = \begin{cases} 0 & p \le a \\ (p - a)/(b - a) & a < p \le b \\ (c - p)/(c - b) & b < p < c \end{cases}$$

Where:

- p= Predicted sentiment probability from Eq. (1)
- a, b, c= Parameters defining the triangular fuzzy region
- $\mu_A(p)$= Membership value of probability $p$in sentiment class A

Each sentiment category (Strong Negative, Somewhat Negative, Neutral, Somewhat Positive, Strong Positive) has its own set of (a, b, c)values, enabling the system to express **graded emotional intensity** rather than a rigid classification.

1.2 | System Architecture and Workflow



The system's architecture, visualized in the accompanying diagram, is a sequential pipeline that methodically transforms raw linguistic input into a human-interpretable sentiment category. This workflow is logically divided into two primary stages: the **Neuro / ML Layer** and the **Fuzzy Logic Layer**.

1.2.1 | Neuro / ML Layer: Probability Generation

The first stage of the pipeline is responsible for converting unstructured text into a quantitative probability. The process begins when a **Raw Review** is fed into the system.

1. **Preprocessing:** The text first undergoes a critical cleaning phase. This involves removing HTML tags, converting text to lowercase, and eliminating common stopwords to reduce noise and standardize the input [1].

2. **Feature Extraction:** The cleaned text is then processed by the **TF-IDF Vectorizer**. This step transforms the tokenized words into a high-dimensional numerical feature vector (50,000 features), which quantifies the relative importance of each term.

3. **Classification:** This feature vector is passed to the **Logistic Regression** classifier[3]. This model, trained on the IMDB dataset, acts as the "Neuro" component, outputting a single, crisp probability score p (where $p \in [0,1]$). This score represents the model's calculated likelihood that the input review is positive.

1.2.2 | Fuzzy Logic Layer: Linguistic Interpretation

The second stage of the architecture receives the probability p and its task is to translate this "crisp" numerical value into a "fuzzy" linguistic concept.

1. **Fuzzification:** The probability p is fed into the **Fuzzy Membership** module. Here, the value is simultaneously evaluated against five predefined triangular membership functions (e.g., $\mu_{Strong Negative}$, $\mu_{Neutral}$, $\mu_{Strong Positive}$)[4]. This step generates five corresponding **Membership Degrees** ($\mu(x)$), which represent the "degree of truth" that the probability p belongs to each of the five sentiment categories.

2. **Inference and Defuzzification:** The resulting membership degrees are passed to the **Fuzzy Inference Engine**. This engine applies a simple decision rule—in this case, selecting the linguistic category with the highest membership degree—to determine the single most appropriate label[5].

This process culminates in the **Final Output**, which is not a number but a human-readable term such as "Neutral" or "Strong Positive," successfully bridging the gap between machine-calculated probability and human-centric interpretation.

1.3 | Implementation Tools and Environment

To build and deploy this hybrid system, a stack of modern, open-source Python libraries was employed, chosen for their robustness in machine learning and rapid web development.

- **scikit-learn:** This was the foundational library for the entire machine learning pipeline. It provided the high-performance TfidfVectorizer for feature extraction and the LogisticRegression classifier used to generate the baseline probability scores.
- **Streamlit:** This framework was used to build the interactive web application front-end. It enabled the rapid development of a user-friendly interface where users can input text and receive real-time visual feedback on the sentiment analysis.
- **matplotlib:** This library was leveraged to generate the static data visualizations displayed in the web app, including the bar and donut charts that illustrate the fuzzy membership distributions.
- **ReportLab:** This library was integrated to provide the "Export to PDF" functionality. It allows users to dynamically generate a structured report of the analysis, including the input text, predicted probability, and membership graphs.

1.4 | Qualitative Analysis and Interpretability

While quantitative metrics like accuracy demonstrate the model's effectiveness (achieving over 90% [15]), the primary contribution of this work is the significant enhancement of *interpretability*. A qualitative review of the system's output, as highlighted in Table 2, confirms this benefit.

For instance, a review like "Absolutely fantastic movie!" yields a high probability ($p=0.89$), which the fuzzy layer confidently maps to **Strong Positive**[17]. Conversely, "The story was dull and slow" produces a low probability ($p=0.12$), mapping clearly to **Strong Negative**.

The true advantage of the fuzzy layer is demonstrated in borderline cases. Consider the review: **"It was okay, nothing special"**[19].

A traditional binary classifier, seeing the probability $p=0.51$, would be forced to label this as "Positive," which clearly misrepresents the author's indifferent and neutral tone. Our neuro-fuzzy system, however, processes $p=0.51$ through the fuzzy inference layer. This value falls squarely within the peak of the 'Neutral' membership function (defined between 0.48 and 0.54). As a result, the system correctly classifies the sentiment as **Neutral**[21], providing a far more accurate and human-aligned interpretation of the ambiguous text. This ability to handle uncertainty is what bridges the gap between machine accuracy and human understanding.

Acknowledgments

The authors thank Prof. Arindam Dey for his valuable guidance and supervision throughout this research project.

Author Contributaion

Conceptualization, S. Magapu and C.T. Goud; **Methodology**, S. Magapu and A.S. Vallabhu; **Software**, S. Magapu and H. Pinni; **Validation**, C.T. Goud and A.S. Vallabhu; **Formal Analysis**, S. Magapu; **Investigation**, All authors; **Resources**, S. Magapu; **Data Curation**, C.T. Goud; **Writing—Original Draft Preparation**, S. Magapu; **Writing—Review & Editing**, All authors; **Visualization**, H. Pinni; **Supervision**, A.S. Vallabhu; **Project Administration**, S. Magapu. All authors have read and agreed to the published version of the manuscript..

Funding

This research received no external funding.

Data Availability

The IMDB Large Movie Review Dataset used in this study is publicly available at http://ai.stanford.edu/~amaas/data/sentiment/.

Conflicts of Interest

This project was carried out purely for academic and learning purposes. The authors confirm that there are no conflicts of interest, personal or financial, that influenced the development, implementation, or reporting of this work. No external funding or third-party involvement influenced the design of the system, the experiments, or the preparation of this report.

References

[1] Maas, A., Daly, R., Pham, P., Huang, D., Ng, A. Y., & Potts, C. (2011). *Learning word vectors for sentiment analysis.* Proceedings of the ACL, pp. 142–150.

[2] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, 12, 2825–2830.

[3] Zadeh, L. A. (1965). *Fuzzy sets.* Information and Control, 8(3), 338–353.

[4] Liu, B. (2012). *Sentiment Analysis and Opinion Mining.* Synthesis Lectures on Human Language Technologies, 5(1), 1–167.

[5] Isikdemir, Y. E., & Yavuz, H. S. (2022). *Scalable fuzzy inference-based ensemble method for sentiment analysis.* Applied Soft Computing, 124, 108953.

[6] Nandi, S., Kumar, R., & Tiwari, P. (2025). *Fuzzy-based ensemble learning for sentiment analysis.* Global Journal of Engineering Innovations, 14(2), 45–52.

[7] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* Proceedings of NAACL-HLT, pp. 4171–4186.

[8] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning.* Springer.

[9] Jang, J.-S. R. (1993). *ANFIS: Adaptive-Network-Based Fuzzy Inference System.* IEEE Transactions on Systems, Man, and Cybernetics, 23(3), 665–685.

[10] Zhang, L., Wang, S., & Liu, B. (2018). *Deep learning for sentiment analysis: A survey.* Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4), e1253.

[11] Pal, S. K., & Mitra, S. (1992). *Multilayer perceptron, fuzzy sets, and classification.* IEEE Transactions on Neural Networks, 3(5), 683–697.

[12] Tang, D., Qin, B., & Liu, T. (2015). *Document Modeling with Gated Recurrent Neural Networks for Sentiment Classification.* Proceedings of EMNLP, pp. 1422–1432.

[13] Wilson, T., Wiebe, J., & Hoffmann, P. (2005). *Recognizing contextual polarity in phrase-level sentiment analysis.* Proceedings of HLT/EMNLP, pp. 347–354.

[14] Komninos, A., & Manandhar, S. (2016). *Dependency based embeddings for sentence classification tasks.* Proceedings of NAACL-HLT, pp. 207–217.

Appendix

Authors can use supplementary sections, known as Appendixes, to provide additional information supporting the findings presented in their manuscript. These Appendixes serve the purpose of including details that might disrupt the main text's flow or are only relevant to a specific subset of readers. The supplementary sections may encompass comprehensive mathematical proofs, extra figures, more in-depth experimental particulars, or supplementary data.

When citing an Appendix in the main text, it is essential to reference it accordingly. In the Appendix, any referenced Figures, Tables, equations, etc., should be labeled with an "A" prefix, followed by a sequential numbering starting from 1 (e.g., Fig. A1, Fig. A2, etc.).