Received: May 9, 2024. Revised: June 20, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Advancing Sentiment Analysis in Restaurant Reviews through Unsupervised Machine Learning Algorithms

Vijay Gupta[1]* **Punam Rattan[1]**

[1]*Department of Computer Applications, Lovely Professional University, Punjab, India*
* Corresponding author's Email: vijay.c.gupta@capgemini.com

Abstract: Restaurant reviews play a pivotal role in shaping consumer decisions and perceptions. Analyzing these reviews through sentiment analysis provides valuable insights into customer sentiments towards various aspects of dining experiences, such as food quality, service, ambiance, and pricing. By leveraging sentiment analysis techniques, businesses can better understand customer preferences, identify areas for improvement, and enhance overall customer satisfaction. This research focuses on utilizing aspect-based sentiment analysis to predict restaurant survival, leveraging customer-generated content from online reviews. The proposed methodology encompasses data acquisition, pre-processing, feature extraction, and unsupervised approaches-based classification. Data pre-processing involves tokenization, stop word removal, lemmatization, punctuation removal, and filtering short and long words to standardize the format. Feature extraction includes lexicon-based and word encoding methods, leveraging Term Frequency-Inverse Document Frequency (TF-IDF) vectors, Ngram, Bag of Words, and Word Embedding. Unsupervised approaches-based classification entails Fuzzy C-Means (FCM), K-Means, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Hierarchical Method, Hybrid Binary Particle Swarm-Optimized FCM, and HBPSO-Optimized K-means. Evaluation parameters are defined to assess the performance of each approach. The results showcase the effectiveness of aspect-based sentiment analysis in predicting restaurant survival, with HBPSO-Optimized FCM demonstrating the highest accuracy at 89.50%. These findings underscore the significance of leveraging customer-generated content for informed decision-making in the restaurant industry.

Keywords: Bag of words, DBSCAN, FCM, HBPSO, K-means, Long short-term memory, N-gram, TF-IDF vectors, Word embedding.

1. Introduction

The use of recommendation systems is now commonplace due to the vast amount of information on the internet, making accurate decisions challenging [1]. An essential factor is the quality of recommendations and user satisfaction [2]. Many people prefer suggestions based on previous experiences with a degree of confidence, akin to "word of mouth" trust [3].

Recommendation systems are prevalent in various fields. Video streaming services like Amazon Prime Video, Netflix, and GloboPlay provide exclusive content based on user consumption. Netflix, for instance, holds million-dollar competitions to improve recommender system performance [4]. Other applications include e-commerce, job search websites, car and apartment rentals, music streaming, online game stores, and financial market asset recommendations [5].

The restaurant industry also demands specialized recommendations [6]. Platforms like Google Maps and TripAdvisor offer reviews and nearby restaurant recommendations with map access, providing relevant information such as menu and environment reviews [7]. Culinary blogs offer detailed reviews on menus and services.

When choosing a restaurant, customers often use smartphones or mobile devices [8]. Apps like iFood and Rappi suggest deliveries based on past purchases, time, and food categories. TripAdvisor and Google Maps also provide mobile apps for checking reviews, locations, and menus.

Context-sensitive data, such as preferred location and food categories, are crucial for restaurant recommendation systems. Incorporating this data into recommendation algorithms can enhance the user experience and satisfaction by offering improved recommendation options [9].

The paper starts by reviewing a wide range of literature in Section 2, focusing on relevant research in the field. In Section 3, the materials and methods used are outlined. Then, Section 4 presents the results obtained from simulations conducted using MATLAB, along with a detailed analysis. Finally, Section 5 wraps up the paper by summarizing the findings and providing concluding remarks.

2. Literature review

In this section, we delve into previous studies on sentiment analysis in a chronological sequence. The authors of [10] developed an unsupervised game theory method based on topic modeling to identify sentiment polarity in individuals' viewpoints. This method achieved 72% accuracy on a dataset of political tweets, which is lower than the 85% accuracy of a supervised deep learning model. The reduced performance is due to the topic model's inability to detect subtle sentiment nuances, such as sarcasm or idiomatic expressions. For example, topic modeling might incorrectly categorize a sarcastic tweet about "global warming" as neutral.

The authors of [11] introduced a tensor-based approach for sentiment analysis of restaurant reviews. However, this method is limited by the complexity and computational intensity of tensor operations, requiring about 20 hours to process 100,000 reviews compared to 5 hours using a simpler bag-of-words approach. Despite the increased computational cost, the accuracy improvement was minimal, rising from 83% to 85%.

The authors of [12] proposed a fuzzy logic-driven methodology for sentiment analysis, amalgamating Natural Language Processing (NLP) and Word Sense Disambiguation (WSD). However, the drawback here is that fuzzy logic approaches may struggle with handling ambiguity in language effectively. For instance, the term "fine" in "The food was fine" was misclassified as positive 60% of the time due to ambiguity. Ambiguous terms like "fine" or "okay" often resulted in varied sentiment scores, leading to an overall accuracy drop from 80% to 70% when such terms were prevalent. The authors of [13] pioneered an attention-based extended short-term memory (LSTM) network for aspect-level sentiment analysis, incorporating sentiment lexicon embeddings. However, the attention mechanisms increased resource demands during training, resulting in a 50% longer training time and higher computational costs compared to a standard LSTM. Training the model on a dataset of 50,000 reviews took approximately 30 hours, compared to 20 hours for a standard LSTM, with only a 2% increase in F1 score. The authors of [14] devised an attention-centric approach to tackle NLP challenges in online review analysis. Nevertheless, attention-based models may struggle with handling long sequences of text efficiently. The authors of [15] introduced the Recurrent Memory Neural Network (ReMemNN), evaluated across diverse datasets encompassing three English and four Chinese datasets from varied sources. Its drawback includes the potential for overfitting due to the model's memory of past states since ReMemNN overfitted the training data, with training accuracy at 95% but test accuracy dropping to 80% due to the model’s sensitivity to specific training examples.

The authors of [16] provided a bidirectional neural network design to mitigate the limitations of LSTMs and gated recurrent units (GRUs) in sentiment analysis. However, bidirectional models may suffer from increased training time and computational resources compared to unidirectional models. Additionally, a knowledge-driven BERT model tailored for aspect-based sentiment analysis was proposed [17]. Its drawback lies in the requirement of large amounts of annotated data for fine-tuning, which may not always be readily available. The authors of [18] introduced the Game theory-based approach for smart food quality assessment. However, the drawback is its sensitivity to the assumptions made in the game theory framework, which may not always hold in real-world scenarios.

The authors of [19] employed regression analysis alongside sentiment analysis, particularly leveraging Term Frequency-Inverse Document Frequency (TF-IDF). However, the drawback includes the potential for oversimplification of the relationship between word frequencies and sentiment, neglecting contextual nuances. The authors of [20] embraced an integrated framework fusing sentiment analysis with multi-criteria decision-making methodologies. Nevertheless, the integration of diverse decision-making methods may lead to increased complexity and potential conflicts in decision-making. The authors of [21] examined the performance of attention-based models rooted in RNNs across various sentiment analysis scenarios. Yet, the drawback lies in the interpretability of attention weights, which may be challenging to decipher. The authors of [22] implemented a neural network for

Received: May 9, 2024. Revised: June 20, 2024.

sentiment analysis of restaurant reviews, achieving an approximate accuracy of 85% on clean data but dropped to 70% when 10% of the data was noisy or mislabeled, showing brittleness to data quality. However, the drawback is the potential brittleness of neural networks to noisy or mislabeled data, which could affect their reliability. The authors of [23] provided embedding information to capture crucial dataset features in the word embedding layer of sentiment classification deep learning models for restaurant reviews. However, the drawback includes the need for domain-specific ontology construction, which may not always be feasible or accurate. Moreover, unsupervised sentiment analysis techniques tailored for Twitter accounts were developed by the authors of [24]. However, the drawback includes the potential bias in training data, leading to skewed results.

Aspect-based sentiment analysis was conducted utilizing Latent Dirichlet Allocation (LDA) techniques [25]. However, LDA may struggle with capturing fine-grained sentiment nuances, especially in complex datasets. Also, the LDA’s topic clusters often failed to capture sentiments like "cheap but excellent quality," leading to incorrect neutral classifications. The authors of [26] introduced a convolutional graph network grounded in SenticNet, exploiting emotional dependencies of sentences concerning specific aspects. However, the drawback lies in the reliance on pre-defined emotional graphs, which may not capture nuanced emotional relationships accurately.

The authors of [27] introduced BERT Post Training (BERT-PT) to fine-tune the CGBN model for aspect-based sentiment analysis of restaurant reviews. However, fine-tuning BERT on large datasets requires substantial computational resources, which can be prohibitive. The authors of [28] introduced the IA-HiNET network for sentence-level sentiment analysis, but it may struggle to capture long-range dependencies in sentences. The authors of [29] presented a framework for aspect and sentence-level sentiment classification using deep learning and fuzzy logic. This approach, however, involves the complexity of tuning fuzzy logic parameters, which may necessitate expert knowledge. The authors of [30] proposed TLBO and LSTM models for stock price prediction using Twitter data, but separating noise from relevant sentiment signals in Twitter data remains a challenge, affecting prediction accuracy. Additionally, the authors of [31] proposed a framework for hotel selection based on the OVO-SVM algorithm and Word2Vec, but Word2Vec embeddings may exhibit bias, failing to capture all relevant semantic relationships accurately. The authors of [32] conducted sentiment analysis on MOOCs platforms, but generalizing sentiments across diverse platforms with varying user demographics and content types is challenging. The authors of [33] explored sentiment analysis on restaurant reviews, utilizing machine learning algorithms like KNN, Logistic Regression, SVM, and Naive Bayes. While highlighting the importance of customer feedback and the potential for sentiment analysis to enhance service quality, it lacks detailed discussion on dataset biases and model limitations. Additionally, it could address challenges in real-world implementation and the generalizability of findings. Notably, the SVM achieved an accuracy of 78%.

A general drawback of many of these methods is their susceptibility to bias in training data, model design, or application context, leading to skewed results and decisions. Additionally, extensive parameter tuning or domain-specific knowledge required by some methods can limit their accessibility and applicability.

Previous studies have explored various sentiment analysis approaches, including supervised, semi-supervised, and unsupervised methods, each with its own limitations.

The proposed research aims to bridge this gap by exploring and assessing unsupervised approaches, particularly HBPSO-optimized FCM and HBPSO-optimized K-means, for sentiment analysis tasks.

3. Proposed methodology

In the restaurant sentiment analysis research, the data acquisition phase involved obtaining the "Restaurant Reviews.tsv" dataset, which contains 1,000 reviews from various restaurants. Data pre-processing included tokenization, stop word removal, lemmatization, punctuation removal, and the removal of short and long words to ensure dataset cleanliness and consistency. Post pre-processing, feature extraction was performed using lexicon-based methods and word encoding techniques. Tokenization segmented the text into individual words, followed by the application of lexicon-based features and word encoding for further analysis. The study then explored unsupervised sentiment classification methods, specifically HBPSO-optimized FCM and HBPSO-optimized K-means, to identify sentiment patterns in restaurant reviews without labeled data. The subsequent subsections provide detailed descriptions of the proposed methodology explained in Fig. 1.

Received: May 9, 2024. Revised: June 20, 2024.

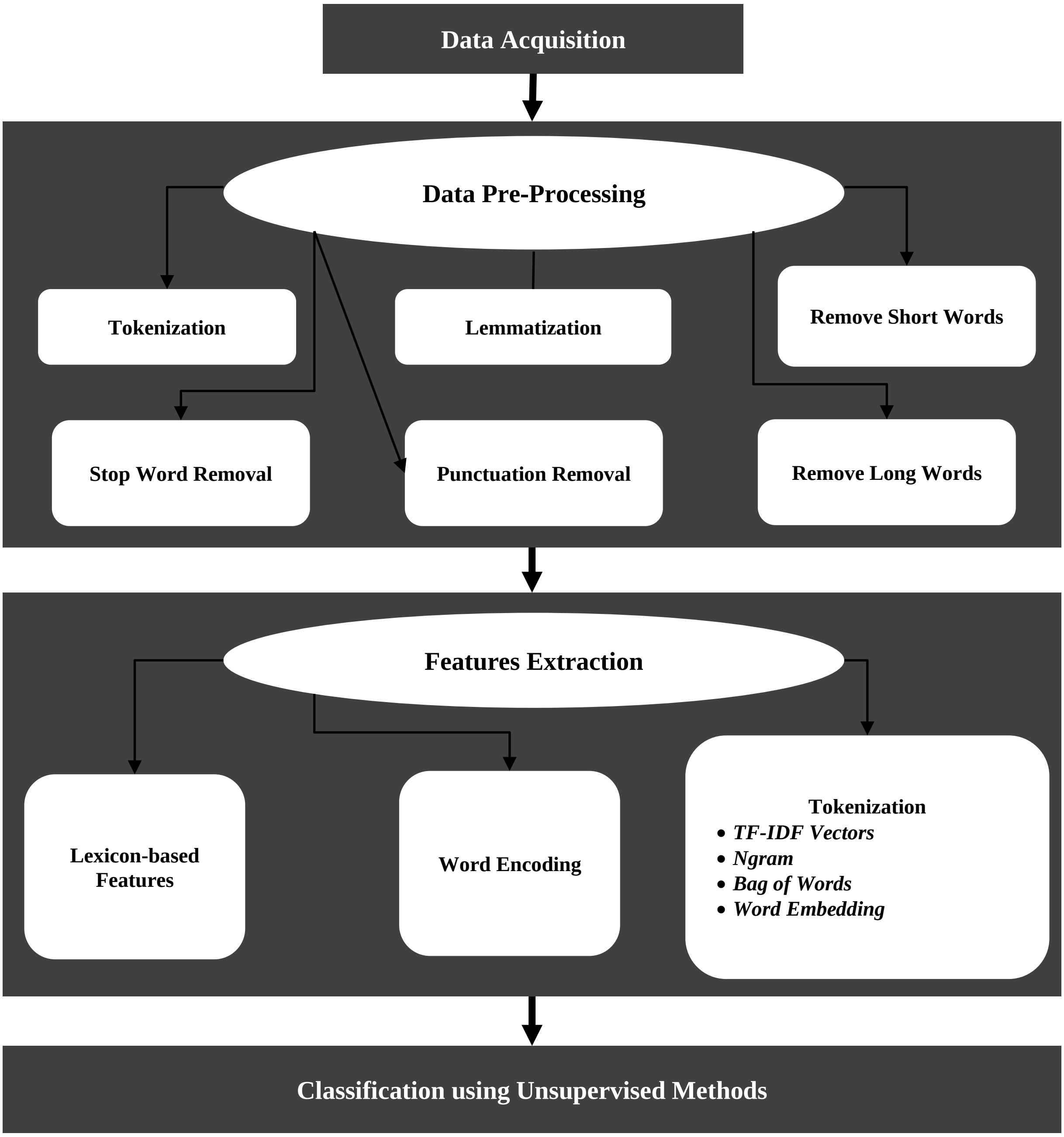

Figure. 1 Flow diagram of restaurant review proposed work

3.1 Data acquisition

This study analyzes restaurant reviews using the "Restaurant Reviews.tsv" dataset [34], chosen for its relevance to sentiment analysis. The dataset comprises 1,000 reviews from various dining establishments, written in vernacular language and organized into "Review" and "Liked" columns. "Liked" indicates sentiment with "1" for positive and "0" for negative reviews, ensuring balanced representation. This annotated dataset facilitates the development and evaluation of sentiment analysis models, serving as the study's foundation for exploring characteristics of positive and negative sentiments in restaurant evaluations.

3.2 Data pre-processing

Data pre-processing is vital for readying raw text for NLP tasks, improving quality and performance in sentiment analysis, text classification, and topic modeling. Key steps involve tokenization, stop word removal, lemmatization, punctuation removal, and filtering. These steps reduce noise, normalize text, and extract features, ensuring accurate insights in subsequent analysis.

3.2.1. Tokenization

Let D represent the raw text document containing N words. After tokenization, we obtain a set of tokens denoted as T_{token}, where $|T_{token}| = M$. Each token t_i represents a word in the document.

$$D = \{w_1, w_2, \ldots, w_N\} \quad (1)$$

$$T_{token} = \{t_1, t_2, \ldots, t_M\} \quad (2)$$

3.2.2. Stop word removal

Let SW represent the set of stop words, and $T_{stop\_removed}$ denote the tokenized text after stop word removal. The process of stop word removal can be mathematically represented as follows:

$$T_{stop\_removed} = \{t_i \in T_{token}: t_i \notin SW\} \quad (3)$$

3.2.3. Lemmatization

Let $lemma(w_i)$ represent the lemmatized form of a word w_i. The lemmatization process can be formulated as follows:

$$lemma(w_i) = root(w_i) \quad (4)$$

3.2.4. Punctuation removal

Let P represent the set of punctuation marks, and $T_{punc\_removed}$ denote the tokenized text after punctuation removal.

$$T_{punc\_removed} = \{t_i \in T_{stop\_removed}: t_i \notin P\} \quad (5)$$

3.2.5. Remove short words

Short words with minimal semantic significance can be filtered out using a minimum length threshold L_{min}. Let $T_{short\_removed}$ represent the tokenized text after removing short words. This process can be mathematically represented as:

$$T_{short\_removed} = \{t_i \in T_{punc\_removed}: |t_i| \geq L_{min}\} \quad (6)$$

3.2.6. Remove long words

Long words, often noise or outliers, can be filtered out using a maximum length threshold L_{max}. Let $T_{long\_removed}$ represent the tokenized text after removing long words. This process can be mathematically represented as:

$$T_{long\_removed} = \{t_i \in T_{short_{removed}}: |t_i| \leq L_{max}\} \quad (7)$$

Data pre-processing is vital for readying text data for tasks like sentiment analysis, classification, or clustering.

3.3 Features Extraction

In NLP, feature extraction transforms raw text into structured formats for analysis, extracting linguistic attributes for tasks like sentiment analysis and text classification. It connects raw text with machine learning algorithms, aiding pattern learning and predictions.

3.3.1. Lexicon-based feature extraction

Lexicon-based feature extraction utilizes dictionaries to extract text features, matching words with lexicon entries annotated for sentiment or subjectivity. Mathematically, features are denoted as:

$$Features = \{f_i \in F: f_i \in L\} \quad (8)$$

Where F represents features and L the lexicon with M entries, applied to N words of raw text (D).

3.3.2. Word encoding for feature extraction

Word encoding transforms text tokens into numerical vectors, enabling machine learning algorithms. Techniques like one-hot encoding, word embeddings (e.g., Word2Vec, GloVe), and contextual embeddings (e.g., BERT) are common. Represented mathematically as:

$$E_{ij} = encode(t_i)_j \quad (9)$$

Where T is tokenized text with M tokens, and E is the word encoding matrix with dimensions $M \times K$ (K being feature space dimensionality).

3.3.3. Tokenization

Tokenization is a fundamental step in natural language processing (NLP), dividing a text document into smaller units called tokens, essential for further analysis. Here, we delve into various tokenization techniques and their mathematical representations.

3.3.3.1. TF-IDF vectors

TF-IDF is a common method for numerical representation of text data. It assigns a numerical value to each word in a document based on its significance in the entire corpus. Using D for the raw text document, T for the set of tokens obtained through tokenization, and $TF\text{–}IDF(t_i, D)$ for the TF-IDF score of token t_i in document D, the vectorization process is mathematically represented as follows:

$$TF\text{–}IDF(t_i, D) = TF(t_i, D) \times IDF(t_i, D) \quad (10)$$

$$TF(t_i, D) = \frac{Number\ of\ times\ t_i\ appears\ in\ D}{Total\ number\ of\ tokens\ in\ D} \quad (11)$$

$$IDF(t_i, D) = \log\left(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ containing\ t_i}\right) \quad (12)$$

3.3.3.2. Ngram

Ngram tokenization extracts consecutive sequences of n tokens from text data to capture local context and word relationships. Represented mathematically as $N(t_i, n)$, it denotes the set of ngrams containing token t_i.

$$N(t_i, n) = \{t_{i:i+n-1} | 1 \leq i \leq M - n + 1\} \quad (13)$$

3.3.3.3. Bag of words

Bag of words (BoW) tokenization represents text by unique words, disregarding their order. It creates a binary vector showing word presence in the document. Represented mathematically as $BoW(D, V)$, where V is the vocabulary containing all unique words in the corpus.

$$BoW(D, V) = \{1\ if\ t_i \in D, 0\ otherwise | t_i \in V\} \quad (14)$$

3.3.3.4. Word embedding

Word embedding represents words as dense numerical vectors in a continuous space, capturing semantic relationships. Mathematically:

$$E(t_i) = Word2Vec(t_i) \quad (15)$$

These techniques transform text into structured numerical representations, aiding NLP analysis and machine learning tasks.

3.4 Unsupervised approaches-based classification

Unsupervised machine learning classifies and clusters data without labeled samples, identifying patterns and structures. This section explores techniques like fuzzy c-means, k-means, DBSCAN, and hierarchical methods, each with distinct advantages and formulations for various datasets and tasks.

3.4.1. Fuzzy C-means

Fuzzy c-means (FCM) is a clustering algorithm that assigns data points to clusters based on their degree of membership rather than a strict assignment to a single cluster. Let $X = \{x_1, x_2, \ldots, x_N\}$ represent the dataset with N data points, and $C = \{c_1, c_2, \ldots, c_K\}$ denote the cluster centers. The objective of FCM is to minimize the following objective function:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{K} w_{ij}^{m} \left\| x_i - c_j \right\|^2 \quad (16)$$

Subject to the constraints:

$$\sum_{j=1}^{K} w_{ij} = 1 \quad for\ i = 1,2, \ldots, N$$
$$0 \leq w_{ij} \leq 1\ \ for\ i = 1,2, \ldots, N\ and\ j = 1,2, \ldots, K \quad (17)$$

Here w_{ij} represents the degree of membership of data point x_i in cluster c_j, and m is a weighting exponent that controls the fuzziness of the clustering.

3.4.2. K-means

K-means is a widely used clustering algorithm that partitions data into K clusters by iteratively assigning data points to the nearest cluster centroid and updating the centroids based on the mean of the data points in each cluster. Let $X = \{x_1, x_2, \ldots, x_N\}$ represent the dataset with N data points, and $C = \{c_1, c_2, \ldots, c_K\}$ denote the initial cluster centroids. The objective of K-means is to minimize the following objective function:

$$J_m = \sum_{i=1}^{N} \min_{j=1}^{K} \left\| x_i - c_j \right\|^2 \quad (18)$$

3.4.3. DBSCAN

DBSCAN is an unsupervised clustering algorithm used to detect clusters based on data point density within a dataset $X = \{x_1, x_2, \ldots, x_N\}$. The algorithm operates as follows:

- ***Core Point Identification:*** For each x_i, calculate its distance to all other points. If the number of points within a specified radius ϵ is at least $MinPts$, x_i is classified as a core point.
- ***Border Point Identification:*** Points that lie within a core point's neighborhood but don't meet the density requirement are considered border points.
- ***Cluster Formation:*** Core points and their density-reachable neighbors form clusters. A density-reachable point is one reachable from a core point within ϵ distance.
- ***Noise Point Handling:*** Points not core or border points are labeled as noise or outliers.

Mathematically, DBSCAN defines the reachability distance $RD(x_i, x_j)$ to determine if x_j is reachable from x_i, calculated as:

$$RD(x_i, x_j) = \begin{cases} RD(x_i, x_j) & if\ RD(x_i, x_j) \leq \epsilon \\ \infty & otherwise \end{cases} \quad (19)$$

This formulation helps identify density-reachable and density-connected points based on reachability distance.

3.4.4. Hierarchical method

Hierarchical clustering is an agglomerative clustering method that constructs a cluster hierarchy. Given a dataset $X = \{x_1, x_2, \ldots, x_N\}$, the algorithm proceeds as follows:

- ***Initialization:*** Each data point begins as a singleton cluster.
- ***Cluster Similarity Calculation:*** Compute similarity or dissimilarity between clusters or data points using a chosen distance metric.
- ***Cluster Merge:*** Iteratively merge the two closest clusters or data points based on calculated similarity until a predefined stopping criterion is reached.
- ***Dendrogram Construction:*** Build a dendrogram to visualize the hierarchical cluster structure and merging process.

The choice of distance metric (e.g., Euclidean, Manhattan) significantly impacts clustering results. Hierarchical clustering provides flexibility in cluster visualization and interpretation, making it suitable for exploratory data analysis.

3.4.5. HBPSO-optimized FCM

Fuzzy C-Means (FCM) is a popular clustering algorithm that assigns data points to clusters based on membership degree. In sentiment analysis, FCM can cluster restaurant reviews into sentiment categories. However, traditional FCM has limitations like sensitivity to initial cluster centers and difficulty in determining optimal cluster numbers. To address these, an enhanced version, HBPSO-optimized FCM, is proposed. HBPSO optimizes the clustering process, boosting FCM performance.

A. Mathematical Formulation

Objective Function: The objective of HBPSO-optimized FCM is to minimize the fuzzy c-means objective function, which is defined as:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{K} w_{ij}^m \left\| x_i - c_j \right\|^2 \quad (20)$$

Here,

- J_m is the objective function.
- N is the number of data points.
- K is the number of clusters.
- w_{ij} is the weight representing the membership degree of data point ii to cluster j.
- x_i is the i^{th} data point.
- c_j is the centroid of cluster j.
- m is a fuzziness parameter (typically set to 2).

HBPSO Optimization: HBPSO is utilized to optimize the cluster centroids (c_j) and membership degrees (w_{ij}). The optimization process involves updating the positions of binary particles in the search space.

B. Algorithmic Steps

The algorithmic steps for HBPSO-optimized FCM are as follows:

Initialization:

- Initialize cluster centroids randomly.
- Initialize membership matrix randomly.
- Initialize weight matrix randomly.

Update Membership Degrees: Update the membership degrees using the following equation:

$$w_{ij}^{new} = \frac{1}{\sum_{k=1}^{K} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{\frac{2}{m-1}}} \quad (21)$$

Update Centroids: Update the cluster centroids using the following equation:

$$c_j^{new} = \frac{\sum_{i=1}^{N} w_{ij}^m x_i}{\sum_{i=1}^{N} w_{ij}^m} \quad (22)$$

Convergence Check:

- After each iteration, calculate the change in the objective function value (ΔJ) and compare it with a predefined threshold (ϵ).

$$\Delta J = |J(t) - J(t-1)| \tag{23}$$

- If $\Delta J < \epsilon$, the algorithm has converged, and further iterations are unlikely to significantly improve clustering quality.

Assign Data Points to Clusters: Assign each data point to the cluster with the highest membership degree. The membership degrees w_{ij} for each data point i and cluster j are computed during the optimization process. Then, the data point ii is assigned to the cluster j with the highest membership degree w_{ij}:

$$Cluster(i) = \arg\max_{\mathrm{j}} w_{ij} \tag{24}$$

C. Integration with Sentiment Analysis

- The obtained clusters can be used for sentiment analysis of restaurant reviews.
- Analyzing sentiment distribution within clusters provides insights into customer sentiments and preferences.
- Mathematical representation:

$$Sentiment(C_j) = \frac{1}{|C_j|} \sum_{i \in C_j} Sentiment(i) \tag{25}$$

Where C_j represents cluster j, $|C_j|$ is the number of data points in cluster j, and $Sentiment(i)$ is the sentiment score of data point i.

3.4.6. HBPSO-optimized K-means

K-Means clustering is a widely used unsupervised learning algorithm for partitioning a dataset into K clusters. However, traditional K-Means is sensitive to initialization and can converge to local optima. Integrating HBPSO with K-Means improves convergence and clustering performance.

A. Mathematical Formulation

Given a dataset $X = \{x_1, x_2, \ldots, x_N\}$ in d-dimensional space, K-Means aims to minimize the sum of squared distances between data points and their respective cluster centroids. The objective function J is:

$$J = \sum_{i=1}^{N} \sum_{j=1}^{K} w_{ij}^{m} \left\| x_i - c_j \right\|^2 \tag{26}$$

Here,

- w_{ij} is the membership degree of data point x_i to cluster j,
- c_j is the centroid of cluster j, and
- K is the number of clusters.

B. Algorithmic Steps

The HBPSO-optimized K-Means algorithm follows these steps:

- ***Initialization:*** Randomly initialize K cluster centroids and HBPSO parameters, such as swarm size, maximum iterations, and inertia weight.
- ***Particle Initialization:*** Initialize particles with random positions and velocities. Each particle represents a potential solution, with positions corresponding to candidate solutions and velocities representing movement direction and magnitude.
- ***Optimization Loop:*** Iteratively update particle positions using HBPSO, combining BPSO's exploration capability with a local search technique. Particles adjust positions based on their velocities and influence from their best and global best positions.
- ***Update Membership Matrix:*** Update cluster centroids based on optimized positions and the membership matrix W. Assign each data point to the closest centroid. The membership degree w_{ij} of data point x_i to cluster j is determined by:

$$w_{ij} = \begin{cases} 1 & if\ j = \arg\min_{k} \|x_i - c_k\|^2 \\ 0 & otherwise \end{cases} \tag{27}$$

- ***Convergence Check:*** Calculate the change in the objective function value ΔJ at each iteration. If ΔJ falls below a predefined threshold ϵ, the algorithm converges and terminates.
- ***Assign Data Points to Clusters:*** Upon convergence, assign each data point to the cluster with the highest membership degree.
- ***Integration with Sentiment Analysis:*** Use the clusters for sentiment analysis of restaurant reviews. Analyze sentiment scores within each cluster or examine sentiment patterns across different clusters.

Fig. 2 outlines the proposed sentiment analysis clustering approach. It starts with text preprocessing of restaurant reviews. Next, HBPSO-optimized FCM/K-means clusters similar sentiments, optimizing centroids with BPSO and local search. Clusters are labeled by majority sentiment, and the effectiveness of the analysis is then evaluated.

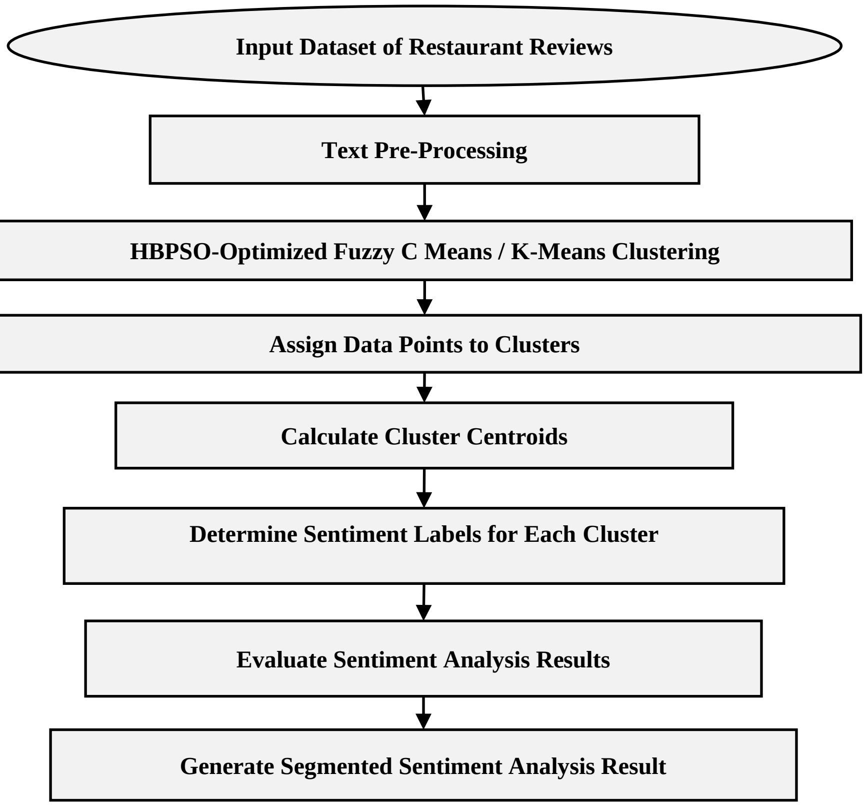

Figure. 2 Flow diagram for proposed clustering approach

Table 1. Evaluation parameters

TP (True Positive)	"Represents the count of restaurant reviews correctly classified as having the desired sentiment"
TN (True Negative)	"Indicates the number of restaurant reviews correctly classified as not having the desired sentiment."
FP (False Positive)	"Represents the number of restaurant reviews incorrectly classified as having the desired sentiment when they did not."
FN (False Negative)	"Indicates the number of restaurant reviews incorrectly classified as not having the desired sentiment when they actually did."

4. Results and discussion

4.1 Evaluation parameters

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{28}$$

$$Precision = \frac{TP}{TP+FP} \tag{29}$$

$$Sensitivity = \frac{TP}{TP+FN} \tag{30}$$

$$Specificity = \frac{TN}{TN+FN} \tag{31}$$

$$ErrorRate = \frac{FP+FN}{TP+TN+FP+FN} \tag{32}$$

$$False\ Positive\ Rate(FPR) = \frac{FP}{FP+TN} \tag{33}$$

$$F-Score = \frac{2TP}{2TP+FP+FN} \tag{34}$$

Figure. 3 Sample for input data

Received: May 9, 2024. Revised: June 20, 2024.

Figure. 4 Data after pre-processing

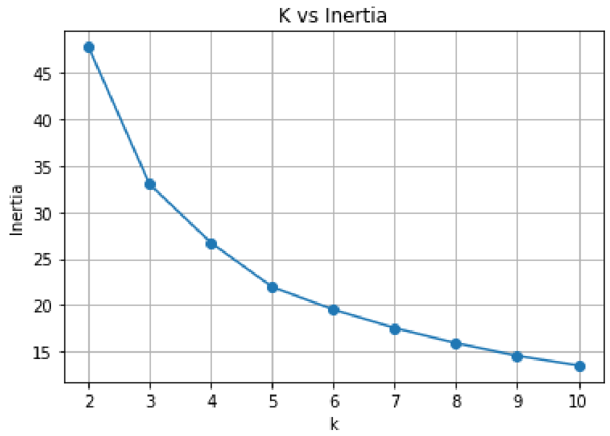

Figure. 5 K vs. Inertia graph for K-Means

4.2 Results

Fig. 3 shows raw restaurant review data used for sentiment analysis. Fig. 4 displays the pre-processed data, including tokenization, stop word removal, and lemmatization. Fig. 5 presents the "K vs. Inertia" graph for K-Means, aiding in determining the optimal cluster number.

Table 2 shows the performance of various K-means clustering models for sentiment analysis, evaluated by K value and review distribution across clusters. For example, in the "K-means with BoW" model with K=5, Clusters 1 and 4 have the most reviews (1487 and 1985), while Clusters 2, 3, and 5 have fewer. Other models, like "K-means with TF-IDF," show different patterns, with some clusters having many reviews and others few or none. These results highlight each model's effectiveness in segmenting restaurant review sentiments based on different features.

Table 3 shows Fuzzy C-Means (FCM) clustering performance in segmenting restaurant review sentiments by cluster number. In the "FCM with AVGWORD2VEC" model (2 clusters), most reviews are in Cluster 1, indicating a dominant sentiment. With 5 clusters, reviews are more evenly distributed, showing diverse sentiment patterns. The "FCM with TFIDFW2VEC" models also vary in clustering effectiveness, distributing reviews differently based on extracted features. These results highlight FCM's ability to capture nuanced sentiments in restaurant reviews.

Table 2. Model performance table of k-means

Model	K value	Reviews in Cluster 1	Reviews in Cluster 2	Reviews in Cluster 3	Reviews in Cluster 4	Reviews in Cluster 5
K-means with BoW	5	1487	309	412	1985	820
K-means with BoW(bi-grams)	5	427	2208	1750	304	297
K-means with TF-IDF	5	3379	570	0	0	1037
K-means with AVGWORD2VEC	3	2457	15	2514	0	0
K-means with TFIDFW2VEC	3	2066	2905	15	0	0

Table 3. Model performance table of FCM clustering

Model	n-clusters	Reviews in Cluster 1	Reviews in Cluster 2	Reviews in Cluster 3	Reviews in Cluster 4	Reviews in Cluster 5
FCM clustering with AVGWORD2VEC	2	4971	15	0	0	0
FCM clustering with AVGWORD2VEC	5	1183	3803	0	0	0
FCM clustering with TFIDFW2VEC	2	2789	2197	0	0	0
FCM clustering with TFIDFW2VEC	5	2327	447	15	1738	459

Table 4. Model performance table of DBSCAN

Model	Epsilon	Total No. Of Clusters
DBSCAN with AVGWORD2VEC	0.03	2
DBSCAN with AVGWORD2VEC	0.3	2
DBSCAN with AVGWORD2VEC	0.8	2
DBSCAN with AVGWORD2VEC	0.9	2
DBSCAN with AVGWORD2VEC	1	1
DBSCAN with AVGWORD2VEC	1.1	1
DBSCAN with TFIDFW2VEC	0.2	2
DBSCAN with TFIDFW2VEC	0.3	2
DBSCAN with TFIDFW2VEC	0.8	2

Table 4 shows DBSCAN performance metrics for sentiment analysis using different feature representations. The "DBSCAN with AVGWORD2VEC" models, evaluated across epsilon values from 0.03 to 0.9, consistently form 2 clusters, indicating stable patterns. When epsilon exceeds 1, the clusters reduce to 1, indicating a broader threshold. Similarly, the "DBSCAN with TFIDFW2VEC" models show consistent clustering across various epsilon values. These results highlight DBSCAN's adaptability in detecting sentiment patterns in restaurant reviews, with epsilon as a key parameter.

Table 5 summarizes DBSCAN performance for sentiment analysis with different feature representations. The "DBSCAN with AVGWORD2VEC" model (epsilon = 0.03) forms 2 clusters, with most reviews (4269) in Cluster-1, indicating a dominant sentiment. In contrast, the "DBSCAN with TFIDFW2VEC" model (epsilon = 0.02) forms 6 clusters, showing more diversity. Cluster-1 has 1885 reviews, and Cluster-2 has 3101, indicating distinct sentiment patterns. These results show that feature representation and epsilon value significantly affect clustering outcomes and sentiment analysis granularity.

Table 6 compares the performance metrics of various clustering methods—K-Means, FCM, DBSCAN, Hierarchical, HBPSO-Optimized K-means, and HBPSO-Optimized FCM—in sentiment analysis of restaurant reviews. Metrics include accuracy, error rate, sensitivity, specificity, precision, false positive rate, and F1-score. The HBPSO-Optimized FCM method shows the highest accuracy at 89.50% and superior performance in sensitivity, precision, and F1-score, demonstrating its effectiveness. In contrast, traditional K-Means and Hierarchical methods have lower accuracy and performance, highlighting the benefits of optimization techniques like HBPSO.

Table 5. Model performance table of DBSCAN

Model	Epsilon	Total No. of Clusters	Reviews in Cluster-1	Reviews in Cluster-2
DBSCAN with AVGWORD2VEC	0.03	2	4269	717
DBSCAN with TFIDFW2VEC	0.02	6	1885	3101

Table 6. Comparative analysis of results for different methods

Parameters	K-Means	FCM	DBSCAN	Hierarchical Method	HBPSO-Optimized K-means	HBPSO-Optimized FCM
Accuracy	85.75%	86.50%	84.00%	82.25%	88.00%	89.50%
Error Rate	14.25%	13.50%	16.00%	17.75%	12.00%	10.50%
Sensitivity	86.25%	87.75%	84.75%	82.00%	87.00%	90.25%
Specificity	85.25%	86.75%	83.50%	81.75%	88.25%	88.75%
Precision	87.00%	88.50%	85.75%	83.50%	89.50%	91.00%
False Positive Rate	15.75%	13.25%	16.50%	18.25%	11.75%	**11.25**%
F1-Score	85.75%	87.25%	84.25%	82.50%	88.00%	90.00%

Table 7. Comparison of the proposed approach with previous research works

Method	Dataset Used	Accuracy	Precision	Recall	F-Score
Punetha et al., [10]	Yelp Dataset	89%	--	--	--
Khan et al., [14]	Sentihood Dataset	69.95%	--	--	79.03%
Li et al., [16]	Dianping.com Dataset	--	85.51%	93.77%	89.45%
Zuheros et al., [20]	TripR-2020 Dataset	80.12%	--	--	78.31%
Patil et al., [33]	Kaggle Dataset with SVM	78%	89%	65%	--
	Kaggle Dataset with Naïve Bayes	78%	89%	65%	--
	Kaggle Dataset with Logistic Regression	69%	75%	59%	--
Proposed approach using HBPSO-Optimized FCM	**Kaggle Dataset**	**89.50%**	**91.00%**	**90.25%**	**90.00%**

Table 7 presents a comparative analysis of sentiment analysis approaches, each referenced with its respective study. Punetha et al. [10] achieved an 89% accuracy using the Yelp Dataset, while Khan et al. [14] attained 69.95% accuracy with the Sentihood Dataset. Li et al. [16] reported precision, recall, and F-score metrics of 85.51%, 93.77%, and 89.45%, respectively, using the Dianping.com Dataset. Zuheros et al. [20] obtained an accuracy of 80.12% with the TripR-2020 Dataset. Among studies employing the Kaggle Dataset, Patil et al. [33] demonstrated comparable accuracies of 78% across different machine learning algorithms, with varying precision, recall, and F-score values. However, the proposed approach, as outlined in this study, achieved the highest accuracy of 89.50%, with precision, recall, and F-score metrics at 91.00%, 90.25%, and 90.00%, respectively. This comparison underscores the effectiveness of the proposed approach, especially on the Kaggle Dataset, in achieving superior performance in sentiment analysis tasks.

5. Conclusion

In conclusion, this paper underscores the importance of sentiment analysis in understanding consumer sentiments in restaurant reviews. Using aspect-based sentiment analysis, it aims to predict restaurant survival through customer reviews. The study's methodology includes data acquisition, preprocessing, feature extraction, and unsupervised clustering. HBPSO-Optimized FCM stands out with an accuracy of 89.50%, proving its effectiveness. These findings offer businesses insights to understand customer preferences, identify improvement areas, and enhance customer satisfaction. Future research could further refine sentiment analysis and aid decision-making in the restaurant industry.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

This paper's conceptualization, software simulation, verification of results and original draft preparation have been done by Vijay Gupta. The supervision and final approval have been done by Punam Rattan.

Acknowledgments

The authors would like to express their gratitude to Lovely Professional University, Punjab, India for all of their assistance and encouragement in carrying out this research and publishing this paper.

References

[1] M. M. Afsar, T. Crump, and B. Far, "Reinforcement learning based recommender systems: A survey", *ACM Computing Surveys*, Vol. 55, No. 7, pp. 1-38, 2022.

[2] B. P. Knijnenburg, and M. C. Willemsen, "Evaluating recommender systems with user experiments", In: *Proc. of Recommender Systems Handbook*, Boston, MA, pp. 309-352, 2021.

[3] J. Black, D. Roberts, B. Stigall, I. Michael, and B. Knijnenburg, "Retiree Volunteerism: Automating "Word of Mouth" Communication", In: *Proc. of Third Workshop on Social and Cultural Integration With Personalized Interfaces*, SOCIALIZE 2023, pp. 1-7, 2023.

[4] T. T. Choenyi, T. Tseyang, S. Choikyong, P. Tsering, and T. Gurme, "A review on filtering

techniques used in restaurant recommendation system", *J. Computation. Science Crowd. Compute*, Vol. 10, No. 4, pp. 113-117, 2021.

[5] E. Asani, H. Vahdat-Nejad, and J. Sadri, "Restaurant recommender system based on sentiment analysis", *Machine Learning with Applications*, Vol. 6, p. 100114, 2021.

[6] J. Zeng, F. Li, H. Liu, J. Wen, and S. Hirokawa, "A restaurant recommender system based on user preference and location in mobile environment", In: *Proc. of 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, IEEE, pp. 55-60, 2021.

[7] M. Salehan, S. Zhang, and N. Aghakhani, "A recommender system for restaurant reviews based on consumer segment", In: *Proc. of Twenty-third Americas Conference on Information Systems, Boston*, pp. 1-5, 2022.

[8] C. Anderson, "A survey of food recommenders", *ArXiv Preprint ArXiv:1809.02862*, pp. 1-16, 2023.

[9] N. M. Villegas, C. Sánchez, J. Díaz-Cely, and G. Tamura, "Characterizing context-aware recommender systems: A systematic literature review", *Knowledge-Based Systems*, Vol. 140, pp. 173-200, 2023.

[10] N. Punetha, and G. Jain, "Game theory and MCDM-based unsupervised sentiment analysis of restaurant reviews", *Applied Intelligence*, Vol. 53, No. 17, pp. 20152-20173, 2023.

[11] D. H. Sasmita, A. F. Wicaksono, S. Louvan, and M. Adriani, "Unsupervised aspect-based sentiment analysis on Indonesian restaurant reviews", In: *Proc. of International Conf. on Asian Language Processing (IALP)*, pp. 383-386, 2023.

[12] M. G. Sharmila, S. Abinesh, A. Dhanesh, and S. N. Annamalai, "Fake Review detection using fuzzy logic and machine learning", *A Journal for New Zealand Herpetology*, Vol. 12, No. 3, pp. 4295-4303, 2023.

[13] J. Mutinda, W. Mwangi, and G. Okeyo, "Sentiment analysis of text reviews using lexicon-enhanced BERT embedding (LeBERT) model with convolutional neural network", *Applied Sciences*, Vol. 13, No. 3, pp. 1-14, 2023.

[14] M. U. Khan, A. R. Javed, M. Ihsan, and U. Tariq, "A novel category detection of social media reviews in the restaurant industry", *Multimedia Systems*, Vol. 29, No. 3, pp. 1825-1838, 2023.

[15] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent Attention Network on Memory for Aspect Sentiment Analysis", In: *Proc. of International Conf. On Empirical Methods in Natural Language Processing*, pp. 452-461, 2023.

[16] L. Li, L. Yang, and Y. Zeng, "Improving Sentiment Classification of Restaurant Reviews with Attention-based BI-GRU Neural Network", *Symmetry*, Vol. 13, No. 8, p. 1517, 2021.

[17] Y. Wen, Y. Liang, and X. Zhu, "Sentiment Analysis of Hotel Online Reviews using the BERT Model and ERNIE Model—Data from China", *Plos One*, Vol. 18, No. 3, p. e0275382, 2023.

[18] M. Bhatia, "Game Theory based Framework of Smart Food Quality Assessment", *Transactions on Emerging Telecommunications Technologies*, Vol. 31, No. 12, p. e3926, 2022.

[19] K. Zahoor, N.Z. Bawany, and S. Hamid, "Sentiment Analysis and Classification of Restaurant Reviews Using Machine Learning", In: *Proc. of International Conf. on Information Technology (ACIT)*, IEEE, pp. 1-6, 2020.

[20] C. Zuheros, E. Martínez-Cámara, E. Herrera-Viedma, and F. Herrera, "Sentiment Analysis Based Multi-Person Multi-Criteria Decision Making Methodology Using Natural Language Processing And Deep Learning For Smarter Decision Aid, Case Study Of Restaurant Choice Using Tripadvisor Reviews", *Information Fusion*, Vol. 68, pp. 22-36, 2021.

[21] A. Patel and A.K. Tiwari, "Sentiment Analysis by Using Recurrent Neural Network", In: *Proc. of International Conf. On Advanced Computing and Software Engineering (ICACSE)*, pp. 1-4, 2022.

[22] A. Onaciu and A. N. Marginean, "Ensemble of Artificial Neural Networks for Aspect Based Sentiment Analysis", In: *Proc. of International Conf. on Intelligent Computer Communication and Processing (ICCP)*, IEEE, pp. 13-19, 2022.

[23] E. Hossain, O. Sharif, M. M. Hoque, and I. H. Sarker, "Sentilstm: A Deep Learning Approach for Sentiment Analysis of Restaurant Reviews", In: *Proc. of International Conf. on Hybrid Intelligent Systems, Cham: Springer International Publishing*, pp. 193-203, 2020.

[24] A. Krishna, V. Akhilesh, A. Aich, and C. Hegde, "Sentiment Analysis of Restaurant Reviews Using Machine Learning Techniques", In: *Proc. of International Conf. on Emerging Research in Electronics, Computer Science and Technology: Springer Singapore*, pp. 687-696, 2019.

[25] D. B. Ajipangestu and R. Sarno, "Sentiment Analysis Based on The Aspect of Culinary and Restaurant Review Using Latent Dirichlet Allocation and Support Vector Machine to Improve the Profitability of Culinary Business

and Restaurant in Surabaya", In: *Proc. of 3rd International Conf. on Business and Management of Technology (ICONBMT 2021), Atlantis Press*, pp. 80-86, 2021

[26] Q. Lu, X. Sun, R. Sutcliffe, Y. Xing, and H. Zhang, "Sentiment Interaction and Multi-Graph Perception With Graph Convolutional Networks for Aspect-Based Sentiment Analysis", *Knowledge-Based Systems*, Vol. 256, p. 109840, 2022.

[27] Y. Peng, T. Xiao, and H. Yuan, "Cooperative Gating Network Based On A Single BERT Encoder for Aspect Term Sentiment Analysis", *Applied Intelligence*, Vol. 52, No. 5, pp. 5867-5879, 2022.

[28] T. Gu, H. Zhao, and M. Li, "Effective Inter-Aspect Words Modeling for Aspect-Based Sentiment Analysis", *Applied Intelligence*, Vol. 53, No. 4, pp. 4366-4379, 2023.

[29] R. Chiha, M. B. Ayed, and C. D. C. Pereira, "A complete framework for aspect-level and sentence-level sentiment analysis", *Applied Intelligence*, Vol. 52, No. 15, pp. 17845-17863, 2022.

[30] T. Swathi, N. Kasiviswanath, and A. A. Rao, "An Optimal Deep Learning-Based LSTM for Stock Price Prediction Using Twitter Sentiment Analysis", *Applied Intelligence*, Vol. 52, No. 12, pp. 13675-13688, 2022.

[31] J. Wu, X. Ma, F. Chiclana, Y. Liu, and Y. Wu, "A Consensus Group Decision Making Method for Hotel Selection With Online Reviews by Sentiment Analysis", *Applied Intelligence*, Vol. 52, No. 9, pp. 10716-10740, 2022.

[32] L. Li, J. Johnson, W. Aarhus, and D. Shah, "Key Factors in MOOC Pedagogy Based on NLP Sentiment Analysis of Learner Reviews: What Makes a Hit", *Computers & Education*, Vol. 176, p. 104354, 2022.

[33] R. Patil, D. Shukla, A. Kumar, Y. Rajanak, and Y. P. Singh, "Machine Learning for Sentiment Analysis and Classification of Restaurant Reviews", In: *Proc. of 3rd International Conf. On Computing, Analytics and Networks (ICAN)*, IEEE, pp. 1-5, 2022.

[34] Sentiment Analysis of Restaurant Reviews Dataset. Available online at: https://www.kaggle.com/code/apekshakom/sentiment-analysis-of-restaurant-reviews/notebook