Sentiment-Based Product Clustering Using K-Means, Hierarchical, and DBSCAN Algorithms

M.H.M Arif Billah Chowdhury Ruddro
Department of Computer Science
American International University-Bangladesh
Dhaka, Bangladesh
21-45066-2@student.aiub.edu

Hozayfah R. Karim
Department of Computer Science
American International University-Bangladesh
Dhaka, Bangladesh
22-46939-1@student.aiub.edu

Zannatul Adon
Department of Computer Science
American International University-Bangladesh
Dhaka, Bangladesh
20-42796-1@student.aiub.edu

Rafayet Zaman Abir
Department of Computer Science
American International University-Bangladesh
Dhaka, Bangladesh
21-45791-3@student.aiub.edu

***Abstract*— In the era of e-commerce, online product reviews are a rich source of customer feedback, yet their large volume makes it challenging for businesses to extract meaningful insights. This project addresses the problem of grouping products based on the sentiment of their reviews to provide actionable information for sellers and customers. We collected product review data from e-commerce platforms and applied text preprocessing steps. A TF-IDF matrix was generated to convert textual data into numerical features, followed by clustering using K-means, Hierarchical, and DBSCAN algorithms. Principal Component Analysis (PCA) was employed to reduce feature dimensions for effective visualization. Our approach enables identification of clusters representing positive, negative, and mixed sentiments, offering businesses a clearer understanding of customer perceptions. Compared to simple sentiment scoring methods, the combination of TF-IDF and clustering provides a more granular and interpretable grouping of products based on review content.**

Keywords—e-commerce, product reviews, sentiment analysis, clustering, tf-idf, pca.

I. Introduction

Online shopping platforms such as Amazon and Daraz allow customers to share their experiences through product reviews. These reviews are important because they provide insights into customer satisfaction, highlight product strengths and weaknesses, and influence future buyers. However, the massive number of reviews makes it very difficult to analyze them manually.

The problem we focus on in this project is how to automatically group products based on the sentiment of their reviews. By clustering reviews into categories such as positive, negative, and mixed, businesses can quickly understand customer opinions and improve their services, while customers can make better purchase decisions.

To solve this, we applied text preprocessing steps such as tokenization, stopword removal, and stemming to clean the review data. The cleaned text was then transformed into numerical features using Term Frequency–Inverse Document Frequency (TF-IDF). We applied three clustering algorithms—K-means, Hierarchical, and DBSCAN—to group the reviews and used Principal Component Analysis (PCA) to reduce the dimensions for visualization. This approach provides an efficient way to analyze large-scale customer feedback and uncover meaningful patterns.

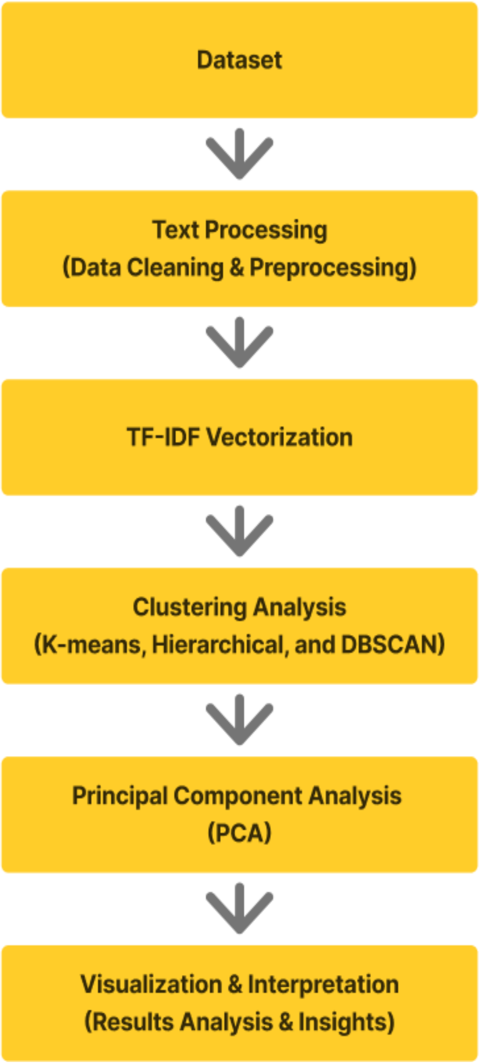
II. Literature Review

Previous studies have addressed the challenge of analyzing large-scale product review data to discover "hot spots" or key topics of discussion [1]. One such approach, based on the MapReduce framework, first uses the Vector Space Model (VSM) with TF-IDF to vectorize the text data. It then applies the Canopy algorithm to identify the initial cluster centers, which solves the problem of not knowing the number of clusters beforehand. Finally, the K-Means algorithm is used for the actual clustering analysis. This algorithm demonstrates high accuracy and parallel efficiency, making it well-suited for processing large-scale data. A key advantage is its ability to handle massive data efficiently due to the parallel processing of the MapReduce framework. However, it is noted that the Canopy algorithm on its own can only perform a "rough clustering" and is not accurate enough for final results.

Other research addresses the problem of segmenting customer opinions from online reviews to provide businesses with critical insights into consumer preferences and behaviors [2]. These studies aim to group customers who share similar opinions based on their reviews. One study proposed a hierarchical clustering method for segmenting reviews. The process involves collecting review data and then applying hierarchical clustering algorithms to group the reviews, using distance measures to evaluate the similarity between them. When compared to a manual, human-driven method of segmentation, this approach achieved an accuracy of approximately 83.6%. The primary advantage of this approach is its ability to provide valuable insights into customer opinions, which enables businesses to better tailor their marketing strategies and improve customer satisfaction. The use of real-world online reviews as the primary data source provides authentic feedback that more accurately reflects customer sentiments and opinions. The study does not explicitly state disadvantages, but the reliance on a manually validated dataset for accuracy measurement can be a time-consuming process.

III. Methodology

Our approach followed a structured pipeline to cluster the given dataset effectively. The methodology is divided into several steps, starting from text preprocessing to visualization of the clustering results. A flow diagram of the process is shown below.


Dataset
Text Processing
(Data Cleaning & Preprocessing)
TF-IDF Vectorization
Clustering Analysis
(K-means, Hierarchical, and DBSCAN)
Principal Component Analysis
(PCA)
Visualization & Interpretation
(Results Analysis & Insights)

A. Dataset Collection

We used a dataset that contains 40,000 product reviews. Each review contains the following fields: productId, Title, reviews, and Score. The dataset was read into R using the readxl package.

B. Text Preprocessing

Raw reviews were processed to prepare them for analysis. The main steps included:

1. Lowercasing and cleaning: All text was converted to lowercase, contractions were expanded (e.g., "can't" → "cannot"), HTML tags and special characters were removed, and extra spaces were eliminated.
2. Tokenization: Reviews were split into individual words using tokenize_words().
3. Lemmatization and stemming: Tokens were reduced to their root forms using lemmatize_words() and wordStem().
4. Stopword removal: Common English stopwords and domain-specific words (e.g., "product", "amazon") were removed.
5. Reconstruction: Clean tokens were pasted back into complete sentences for further analysis.

These steps ensured that the text was consistent and noise-free, which improved the quality of feature extraction and clustering.

C. Sentiment Labeling

We assigned each review a sentiment category based on its numeric score:

- Score 1–2 → Bad
- Score 3 → Neutral
- Score 4–5 → Good

Each sentiment was also converted into a numeric value (-1 for Bad, 0 for Neutral, 1 for Good) to serve as a numeric feature in clustering.

D. TF-IDF Feature Extraction

We applied Term Frequency–Inverse Document Frequency (TF-IDF) to extract meaningful words from each review. Steps included:

1. Tokenizing clean reviews into individual words.
2. Removing stopwords.
3. Counting word occurrences per product.
4. Calculating TF-IDF scores using bind_tf_idf().
5. Selecting top words per product for exploratory visualization.

Additionally, Latent Semantic Analysis (LSA) was performed on the TF-IDF matrix to reduce dimensionality while preserving important semantic information. The resulting LSA features were combined with numeric features (Score and sentiment) to create a final feature matrix for clustering.

E. Clustering

We applied three clustering algorithms to group reviews by similarity:

1. K-means Clustering

   - Scaled features were clustered into K groups.
   - The number of clusters, K, was set to 3, corresponding to the three sentiment categories (Bad, Neutral, and Good).
   - Cluster labels were assigned to each review.
2. Hierarchical Clustering
   - A distance matrix was computed using Euclidean distance.
   - Ward's method was used to create a dendrogram.
   - The tree was cut at K clusters to assign cluster labels.
3. DBSCAN (Density-Based Spatial Clustering)
   - Identified dense regions of reviews while marking sparse points as noise.
   - Parameters eps and minPts were determined heuristically.
   - Outliers were labeled as Noise.

These three algorithms allowed us to compare cluster formation across different methods.

F. PCA for Visualization

To visualize the high-dimensional feature space, Principal Component Analysis (PCA) was applied:

- Features were projected into two principal components.
- Cluster assignments from K-means, Hierarchical, and DBSCAN were plotted in 2D scatterplots.
- Color-coding enabled visual interpretation of cluster separation and overlap.

G. Cluster Interpretation

Finally, we interpreted clusters based on:

1. Top TF-IDF words per cluster: Revealed the most representative words in each cluster, helping identify the characteristics of positive, negative, or mixed reviews.
2. Sentiment distribution per cluster: Bar plots showed the proportion of positive, neutral, and negative reviews in each cluster.

This analysis provided insights into product performance and customer feedback patterns.

IV. Implementation

The system was implemented in the R programming environment (version 4.x) using RStudio as the IDE. Several text mining, machine learning, and visualization libraries were used:

- dplyr, ggplot2, patchwork → Data wrangling and visualization
- tidytext, tokenizers, textstem, SnowballC, textclean, hunspell → Text preprocessing (tokenization, lemmatization, stemming, contraction replacement, and cleaning)
- text2vec, irlba, Matrix → TF-IDF vectorization and dimensionality reduction
- cluster, dbscan → Clustering algorithms (K-means, Hierarchical, DBSCAN)
- readxl → Reading Excel dataset
- scales → Color mapping for visualizations

Dataset Information:

- File name: ids_final_dataset_sample_group_06.xlsx
- Size: ~40,000 customer reviews

Preprocessing Parameters:

- Text converted to lowercase
- HTML tags, numbers, and special characters removed
- Stopwords removed (default + custom words like *product, amazon, review*)
- Lemmatization + stemming applied
- Final clean tokens joined back into sentences

TF-IDF Parameters:

- Tokens extracted using unnest_tokens()
- Word counts aggregated per product
- TF-IDF applied to highlight unique keywords

Sentiment Analysis Parameters:

- Rule-based sentiment classification based on review scores:
   - 1–2 = Negative (Bad)
   - 3 = Neutral
   - 4–5 = Positive (Good)

Clustering Parameters:

- K-means: Number of clusters $k = 3$ (Bad, Neutral, Good), nstart = 50, iter.max = 300
- Hierarchical Clustering: Distance metric = Euclidean, Linkage = Ward's method, clusters cut into $k = 3$
- DBSCAN: eps chosen from 90th percentile of kNN distances, minPts = max(5, round(ncol(features_scaled)/10))
- PCA: Used for dimensionality reduction to 2 components for visualization of clusters

V. Result Analysis

The performance of sentiment-based product clustering was evaluated through both visual and quantitative analysis. This section presents the results obtained from applying K-means, Hierarchical, and DBSCAN clustering algorithms to a dataset of 40,000 product reviews. A preliminary analysis of the dataset's sentiment, based on review scores, showed a clear dominance of positive ("Good") reviews, with a much smaller proportion of negative ("Bad") and neutral reviews.

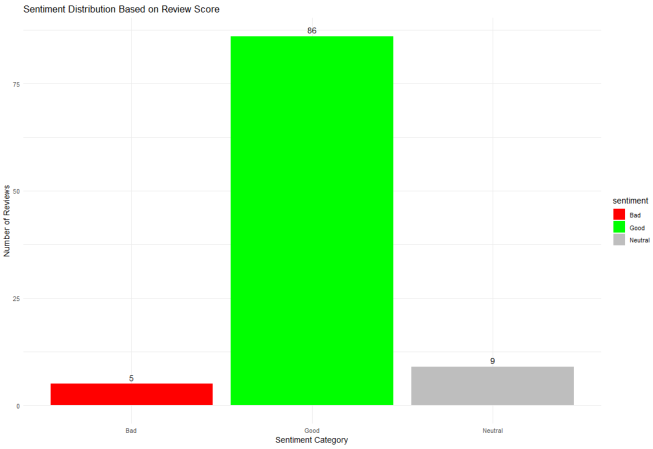


Figure 1: Sentiment Distribution Based on Review Score

The TF-IDF analysis provided insights into the topics of discussion. As shown in Figure 2, top keywords for various product IDs (e.g., B00000DMER, B00000GBQL, B00000IS3K, etc.) revealed distinct themes, such as toys, electronics, and cosmetics. This confirms that TF-IDF effectively captured domain-specific keywords tied to product categories.

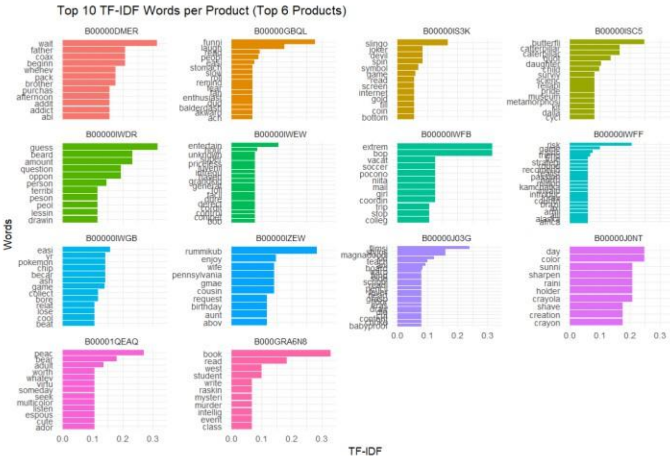


Figure 2: Top 10 TF-IDF Words per Product (Top 6 Products)

A. K-Means Clustering:

K-means clustering was performed with a predetermined number of clusters,

k=3. The PCA visualization shows that the algorithm successfully partitioned the data into distinct clusters, with clear separation in the reduced 2D space. The visual separation of these clusters is evident in the PCA plot.

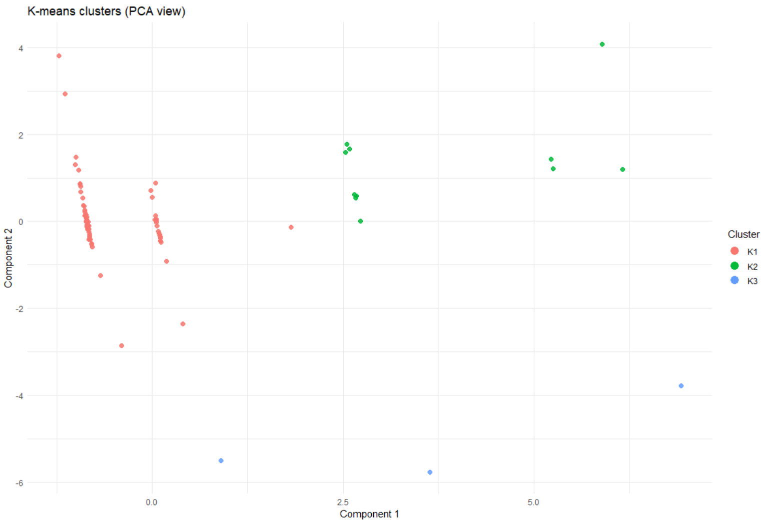


Figure 3: PCA Visualization of K-means Clusters

The sentiment distribution within the K-means clusters was highly distinct. While most clusters were dominated by 'Good' reviews, K2 and K3 showed higher proportions of 'Bad' and 'Neutral' reviews.

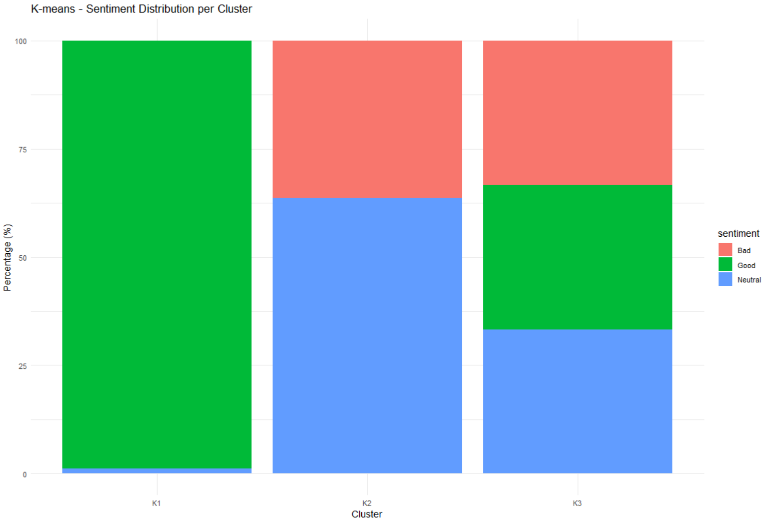


Figure 4: Sentiment Distribution per Cluster (K-means)

The top TF-IDF keywords associated with the K-means clusters provided meaningful insights[4]. For example, words like "game," "play," and "age" pointed to toy-related products, while keywords such as "skin," "lotion," and "moistur" corresponded to cosmetic items.

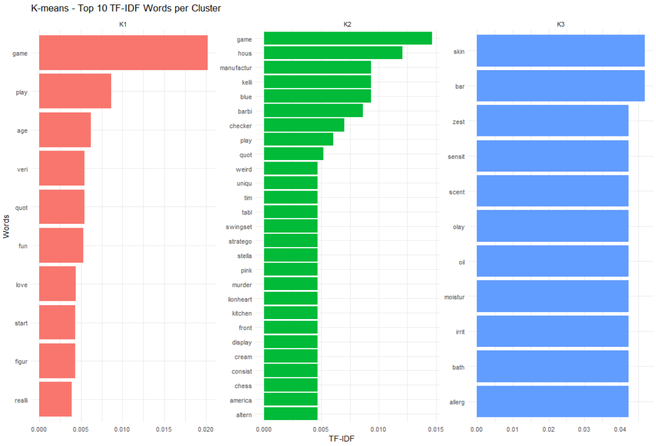


Figure 5: K-means Top 10 Tf-IDF Words per Cluster

B. *Hierarchical Clustering:*

The hierarchical clusters exhibited less separation compared to K-means, with significant overlaps in the PCA plot.

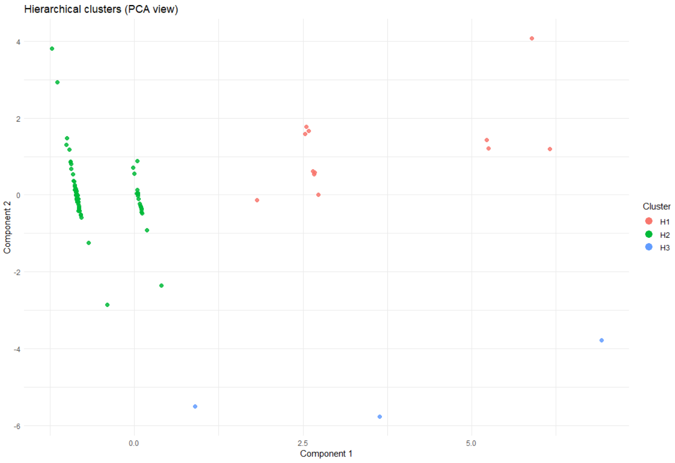


Figure 6: PCA Visualization of Hierarchical Clusters

The sentiment distribution showed a dominance of 'Good' reviews across most clusters, though H1 and H3 contained noticeable negative and neutral proportions. This suggests that while hierarchical clustering is less distinct, it still captures meaningful differences in sentiment.

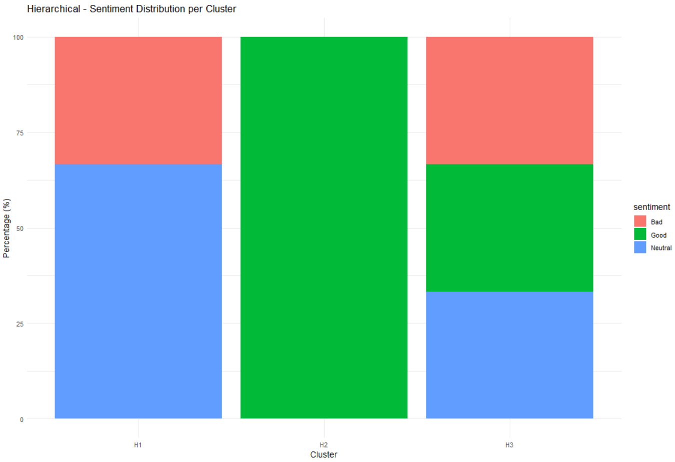


Figure 7: Sentiment Distribution per Cluster (Hierarchical)

The TF-IDF keywords for hierarchical clustering showed similar topical themes, with H1 and H2 focused on "game" and "play", while H3 concentrated on "skin" and "bar".

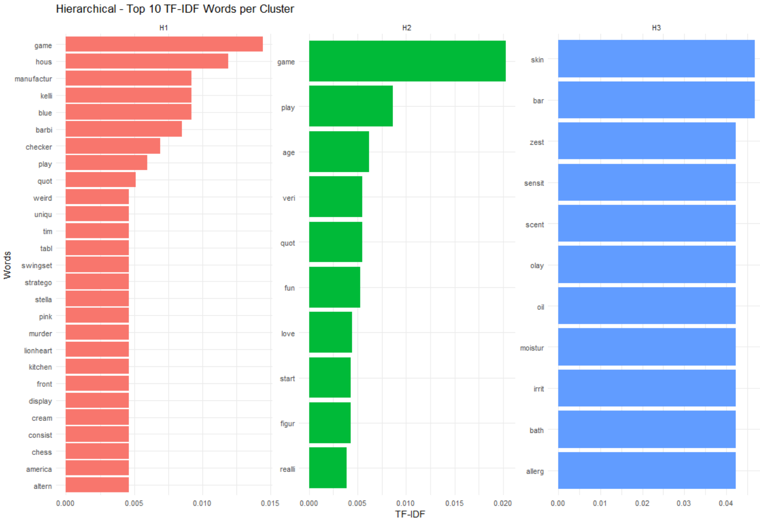


Figure 8: Hierarchical Top 10 TF-IDF Words per Cluster

C. *DBSCAN Clustering:*

DBSCAN identified one main cluster along with a set of noise points. The PCA visualization showed less distinct separation compared to K-means.

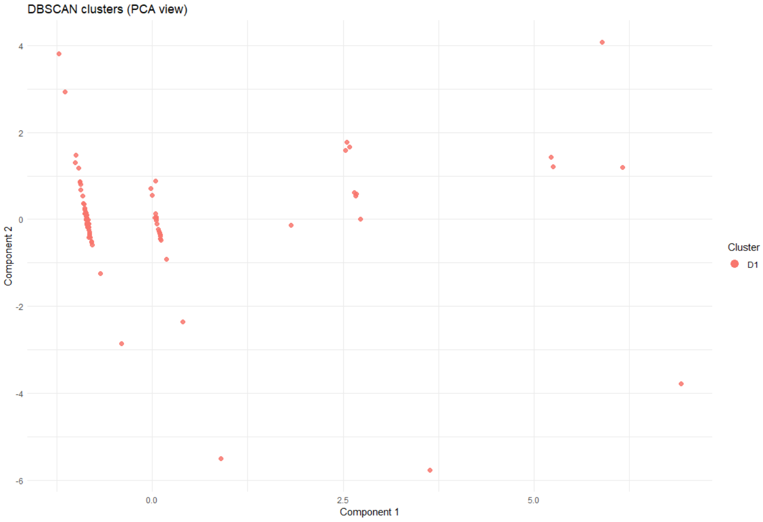


Figure 9: PCA Visualization of DBSCAN Clusters

The sentiment distribution indicates that most reviews within the dense clusters were "Good", while noise points contained a larger proportion of "Bad" reviews. This shows that DBSCAN was effective at isolating outliers. The top TF-IDF keywords for DBSCAN are difficult to interpret due to the visualization scale.

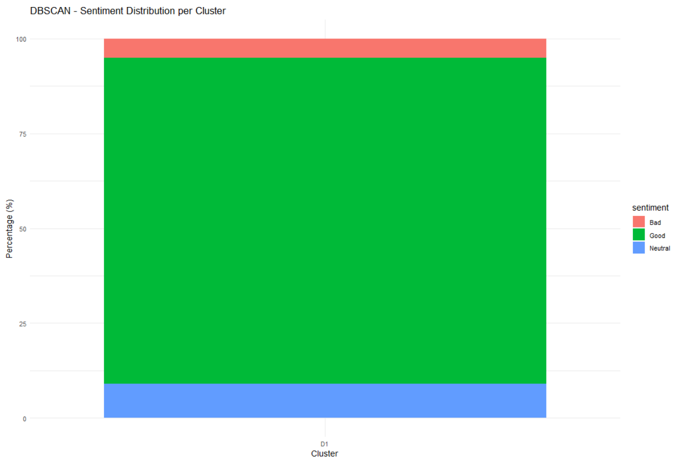


Figure 10: Sentiment Distribution per Cluster (DBSCAN)

D. Comparative Analysis

A direct comparison of the three clustering methods revealed that

K-means provided the most distinct and interpretable clusters. Its clear separation in the PCA visualization and meaningful sentiment variations allowed for straightforward and actionable insights into the product reviews. Hierarchical clustering, while revealing nested structures, suffered from cluster overlaps, limiting its interpretability. DBSCAN was effective for detecting noise and outliers but did not form strongly separable clusters in dense regions.

Overall, K-means emerged as the best-performing method for this dataset, while DBSCAN added value by identifying anomalous reviews.

VI. Conclusion

In this work, a sentiment-based clustering system was designed and implemented. The dataset was preprocessed through several steps including tokenization and lemmatization. The reviews were then classified into three sentiment categories (positive, negative, and neutral) using a rule-based approach.

Feature extraction was carried out using the TF-IDF method, and dimensionality reduction was applied using PCA. Clustering was then performed with algorithms such as K-means and DBSCAN. The results were visualized using PCA plots, and the clusters were analyzed to identify similarities and differences among reviews.

It was observed that positive reviews were grouped more distinctly, while negative and neutral reviews showed partial overlap. The performance of clustering was demonstrated through diagrams and figures, which confirmed that the preprocessing and feature extraction steps were effective.

The work was carried out in the R programming environment, and multiple libraries were used to support preprocessing, sentiment analysis, and clustering. Overall, the project was successfully completed, and it was shown that combining sentiment analysis with clustering techniques could provide useful insights into product reviews.

Future improvements could be made by applying deep learning–based models such as BERT or LSTMs to enhance sentiment detection and clustering accuracy.

References

[1] [1] H. Su, Q. Liu, and C. Mu, “Research on Product Reviews Hot Spot Discovery Algorithm Based on Mapreduce,” *IEEE Access*, vol. 8, pp. 106067-106079, Jun. 2020.

[2] [2] H. T. T. Nguyen, H. D. T. Lan, T. N. Thi, H. K. T. Thu, and Q. P. Dinh, “Customer opinion segmentation by using the Hierarchical clustering method with the online reviews data,” in *2025 10th International Conference on Information and Network Technologies (ICINT)*, 2025.