Journal of
Theoretical and Applied Electronic Commerce Research

MDPI

Article

A Multi-Branch Framework for Multidimensional Online Review Analytics

Fuangfar Pensiri [1], Pattarapol Pornsirirung [2], Natthakit Khonthong [2], Wisit Poonsawat [2], Suchada Chomjan [1], Keun Ho Ryu [3] and Porawat Visutsak [2,*]

[1] Department of Computer Science and Information Technology, Faculty of Science at Sriracha, Kasetsart University, Sriracha Campus, Chonburi 20230, Thailand; fuangfar.p@ku.ac.th (F.P.); suchada.chom@ku.th (S.C.)
[2] Department of Computer and Information Science, Faculty of Applied Science, King Mongkut's University of Technology North Bangkok, Bangkok 10800, Thailand; s6704062810011@kmutnb.ac.th (P.P.); s6804062810061@kmutnb.ac.th (N.K.); s6804062810044@kmutnb.ac.th (W.P.)
[3] Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam; khryu@tdtu.edu.vn
* Correspondence: porawatv@kmutnb.ac.th

Abstract

Online product reviews combine unstructured narratives, star ratings, and helpfulness votes, but these signals answer different managerial questions and require different validation strategies. This study presents a reproducible multi-branch analytical framework for the Amazon Fine Food Reviews dataset (568,454 records; October 1999–October 2012). The framework includes exploratory text mining, LDA topic modeling, rating-derived polarity classification, structured-feature score prediction, and rule-based rating-helpfulness segmentation. To address sampling bias and leakage, the revised experiments use reproducible random sampling, product-grouped holdout sets, and feature estimation only on training data. LDA models with K = 3–8 were compared on a random sample of 20,000 reviews across three seeds; K = 3 showed the best held-out perplexity and the strongest topic stability among the tested solutions. For rating-derived polarity, class-weighted TF-IDF Logistic Regression achieved 94.55% accuracy, 94.04% balanced accuracy, 90.47% macro-F1, 93.29% negative recall, and 93.27% negative-class PR-AUC on an independent product-grouped test set (n = 104,282). A regression tree using structural review features outperformed a mean predictor but remained modest (MAE = 0.884, RMSE = 1.194, R^2 = 0.175), limiting its use to exploratory analysis. Helpfulness analyses treated the 270,052 reviews with no votes as Unrated rather than NotHelpful. Sensitivity analysis across thresholds of 0.50–0.80 and minimum-vote requirements of 1, 2, 5, and 10 showed that the estimated Helpful share depends materially on both choices. The revised evidence therefore supports descriptive screening and hypothesis generation, not causal claims or autonomous managerial decisions. The contribution is a transparent comparison of complementary analytical branches under leakage-aware evaluation, together with explicit construct, sampling, and external-validity limits. The findings show why class-sensitive metrics, topic stability checks, independent test sets, and separate treatment of unexposed reviews are necessary before review analytics can inform practice.

Keywords: online product reviews; text analytics; sentiment analysis; topic modeling; logistic regression; review-score prediction; review helpfulness; rule-based segmentation

Check for updates

Academic Editors: Joonheui Bae and Min Jung Cho

Received: 15 August 2026
Revised: 4 October 2026
Accepted: 4 October 2026
Published: 7 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Online product reviews have become an important source of customer feedback and market intelligence because they contain both numerical ratings and detailed descriptions of customers' experiences. Previous research has demonstrated that the content, sentiment, and numerical characteristics of online reviews can influence product sales, purchase decisions, and perceptions of product quality [1–4]. Nevertheless, the rapid accumulation of reviews across e-commerce platforms creates substantial challenges for businesses. The large volume of data makes manual examination costly and impractical. In contrast, the unstructured nature of review text makes it difficult to consistently identify customer needs, emerging complaints, and changes in product perception. Businesses must therefore move beyond collecting ratings and reviews and transform them into predictive, actionable information. Predicting customer sentiment or review scores, for example, may enable organizations to identify potential dissatisfaction earlier, prioritize product-related issues, and allocate resources to the aspects of products and services that most affect customer experience and business performance [1–4].

An effective review analytics mechanism is therefore needed to integrate textual content, numerical ratings, temporal information, and review-helpfulness indicators to support business decision-making. Text mining of online reviews can reveal the dimensions that shape customer satisfaction and guide improvements in competitiveness and service quality [5–7]. The resulting analysis can help businesses identify product or service attributes customers appreciate or complain about, detect negative reviews that require timely attention, and monitor topics that appear frequently in customer discussions [5–8]. In addition, the increasing volume and redundancy of online reviews can cause information overload, making it necessary to identify, rank, or separate helpful reviews from less informative feedback [9–11]. These capabilities allow organizations to use customer-generated information not only for descriptive reporting but also for improving products, services, complaint-handling strategies, customer communication, and managerial responses [7–11].

To address these needs, this study organizes five established methods as a multi-branch analytical framework rather than claiming a new end-to-end model. Review text, ratings, and helpfulness votes provide different signals for analysis [1,9–11]. The rating-derived polarity task gives particular attention to negative-class recall because missing a negative review may be more consequential for the proposed screening use case. The technical contribution is therefore methodological transparency: the same review population is examined through lexical, thematic, predictive, and descriptive lenses, while the validity limits of each output are kept explicit.

2. Materials and Methods

RQ1. Which lexical and latent thematic patterns are reproducible across random samples and model specifications?

RQ2. How accurately can review text recover rating-derived polarity when class imbalance, grouped holdout evaluation, and leakage control are applied?

RQ3. How much out-of-sample rating variation can structural review features explain relative to a mean-prediction baseline?

RQ4. How sensitive are rating-helpfulness segments to the helpfulness threshold, minimum vote count, and treatment of reviews with no votes?

This study employed a multi-branch analytical framework comprising five complementary methods distributed across three workflows. The branches share a common dataset and managerial objective, but they use task-specific representations and are evaluated independently. The framework is not presented as a single end-to-end model.

https://doi.org/10.3390/jtaer21100351

2.1. Data Source

The study used the public Amazon Fine Food Reviews dataset (Reviews.csv). It contains 568,454 review records posted from 8 October 1999 to 26 October 2012 and ten original variables [12]. A reproducibility audit found 831 exact duplicate records under the combined key UserId, ProductId, Score, Time, Summary, and Text; we removed these duplicates for predictive analyses, leaving 567,623 records. Because repeated text and shared products can inflate random-split performance, predictive holdouts were grouped by ProductId.

1. Score: numerical product rating from 1 to 5 stars;
2. Summary: short review title or summary;
3. Text: complete review text;
4. HelpfulnessNumerator: number of users who considered the review helpful;
5. HelpfulnessDenominator: total number of helpfulness votes;
6. Time: review-posting time represented as a Unix timestamp.

Additional identification variables included Id, ProductId, UserId, and ProfileName. We retained these identifiers during data import but removed them when they were not analytically relevant. Online reviews combine unstructured textual feedback with structured information such as ratings, timestamps, and helpfulness votes. This combination permits the analysis of customer language, latent themes, expressed sentiment, expected ratings, and the perceived usefulness of individual reviews [13–16].

2.2. Analytical Environment

The original analyses were implemented in MATLAB R2024b. The supplied trainRegressionModel.m file specifies a regression tree fitted with fitrtree, MinLeafSize = 36, Surrogate = 'off', and five-fold cross-validation. The revision independently reproduced the evaluation pipeline in Python 3.12 with pandas 2.2.3 and scikit-learn 1.8.0, using random seed 20260919. All vocabulary construction and TF-IDF weighting were fitted on training data only. We provide the complete scripts, split logic, model settings, and result tables in Data Availability Statement section.

The overall design contained:

a. Workflow A: Text Analysis and Sentiment Classification;
b. Workflow B: Review Score Prediction;
c. Workflow C: Rule-Based Review Segmentation.

The workflows are complementary analytical branches rather than a sequential predictive pipeline. Outputs are compared at the interpretation stage. Rating-derived polarity scores are not passed into rating prediction because both originate from the same star-rating target and such integration would create circularity or target leakage. Future work may integrate topic proportions or independently derived text embeddings using nested, out-of-fold estimation.

2.3. Data Preparation

The dataset was first read into a MATLAB table. We removed variables not required for a particular workflow. Missing-value handling was workflow-specific:

1. Workflow A removed observations with missing values in Score, Summary, or Text;
2. Workflow C removed observations with missing Score values and replaced missing helpfulness counts with zero;
3. The available implementation did not explicitly document the missing-value handling procedure for Workflow B.

Workflow-specific preprocessing is reasonable because text mining, regression, and segmentation require different data representations.

2.4. Overview of the Multi-Branch Framework

The framework combined five methods:

a. Exploratory Text Mining;
b. LDA Topic Modeling;
c. TF-IDF-Based Logistic Regression for Sentiment Classification;
d. Regression-Based Review Score Prediction;
e. Rule-Based Review Segmentation.

The first two are primarily NLP and exploratory methods. The third and fourth are supervised machine-learning methods, whereas the fifth uses predefined analytical rules. Prior work has separately examined visualization of review sentiment for e-commerce business intelligence, product-specific helpfulness prediction, interpretable helpfulness determinants, cumulative helpfulness prediction, neural helpfulness modeling, review credibility, and topic-assisted rating prediction. The present study does not treat these distinct tasks as one integrated end-to-end model; its workflows share a dataset and reporting layer but are trained and interpreted separately.

2.5. Workflow A: Text Analysis and Sentiment Classification

2.5.1. Text Corpus Construction and Preprocessing

Workflow A selected the Score, Summary, and Text variables. Records containing missing values were removed. The summary and the full review text were concatenated into a new document variable:

$$Doci = Summary_i + Text_i$$

Combining these fields preserved both the concise expression in the review title and the detailed explanation in the review body.

We did not select the revised corpus by taking the first N records. Descriptive counts used the full eligible dataset. Computationally intensive LDA used a reproducible simple random sample of 20,000 reviews (seed 20260919), with 80% used for model fitting and 20% held out for perplexity evaluation. The polarity classifier used all eligible non-neutral reviews after exact-record deduplication and a product-grouped 80/20 holdout.

a. tokenization;
b. conversion to lowercase;
c. punctuation removal;
d. stop-word removal;
e. removal of words with two or fewer characters;
f. lemmatization.

Text preprocessing standardizes the text and creates a consistent vocabulary for Bag-of-Words, topic modeling, and supervised classification [13,15,16]. Studies of text-based sentiment classification show that feature representation and model choice can affect performance [15–18].

Workflow A comprises three parallel analytical methods: exploratory text mining, LDA topic modeling, and TF-IDF-based Logistic Regression. The revised analysis compares LDA models and applies ProductId-grouped holdout evaluation for rating-derived polarity classification, as illustrated in Figure 1.

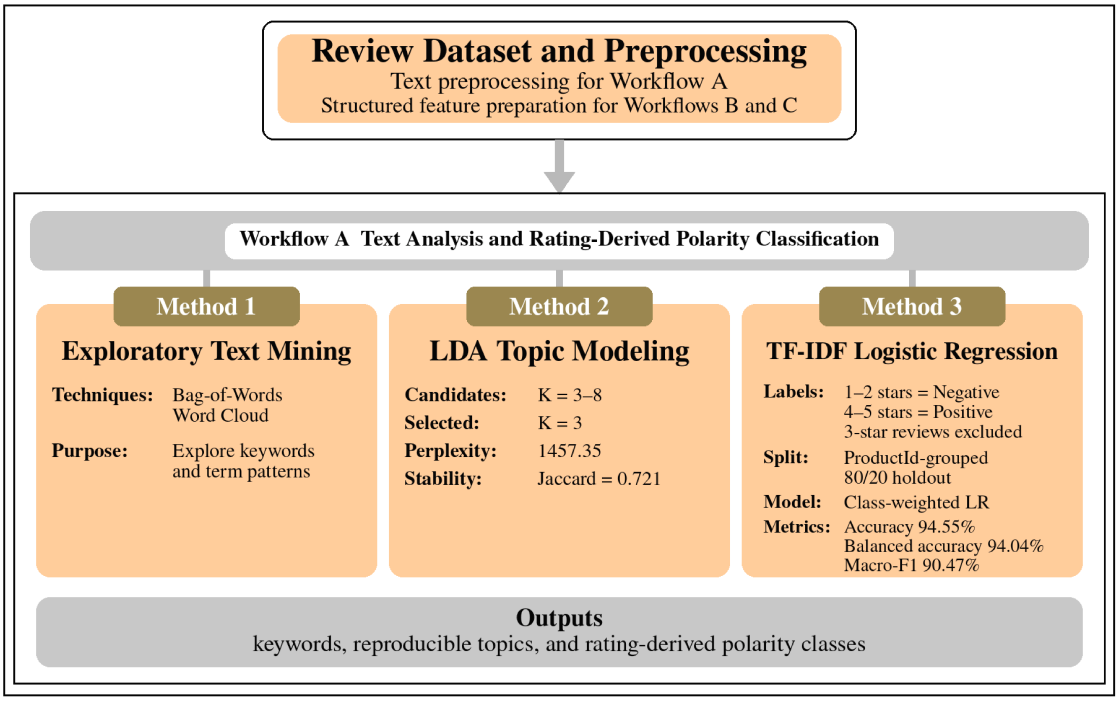


Figure 1. Workflow A: Text Analysis and Rating-Derived Polarity Classification.

2.5.2. Method 1: Exploratory Text Mining

Exploratory Text Mining identified frequently occurring words and broad lexical patterns in the review corpus. This method provided an initial descriptive view of the language customers used before formal topic or predictive analysis.

1. Bag-of-Words Representation

We created a Bag-of-Words model from the cleaned, tokenized documents. In this representation, each review was represented by how many times each vocabulary term appeared in that document. Word order and sentence structure were not modeled.

Let d_i denote review i and let w_j denote vocabulary term j. The document-term matrix contained:

$$x_{ij} = \text{count of } w_j \text{ in } d_i$$

Terms occurring in fewer than five documents were removed, and documents that became empty after vocabulary pruning were excluded. Exploratory term summaries were generated from the reproducible simple random sample of 20,000 reviews (seed = 20260919) used for the topic-model comparison.

Bag-of-Words provides directly interpretable term-frequency information, although it does not retain semantic context or word order. Count-based features are used here for exploratory term analysis; TF-IDF features are used separately for the supervised polarity classifier [19].

2. Word Cloud Visualization

We generated a word cloud from the Bag-of-Words representation of the reproducible random sample of 20,000 reviews. Word size represented the relative frequency of each term within this sample.

Separate word clouds were also constructed for:

- Positive reviews with scores of 4–5 stars;
- Negative reviews with scores of 1–2 stars.

This positive–negative comparison did not include three-star reviews. The word clouds provided a visual summary but were not treated as inferential results. Because high frequency does not necessarily imply high business importance, support word-cloud interpretations with term frequencies, TF-IDF weights, or topic-model probabilities.

3. Output

Method 1 produced: an overall vocabulary:

- Document-term frequencies;
- An overall word cloud;
- Positive-review and negative-review word clouds;
- An initial set of candidate customer concerns and preferred product attributes.

Exploratory text mining of review data has been applied to reveal satisfaction dimensions, frequently discussed attributes, and customer issues.

2.5.3. Method 2: LDA Topic Modeling

LDA Topic Modeling was used to discover latent themes in the reviews. Whereas Method 1 identified individually frequent terms, Method 2 modeled groups of words that tended to co-occur and interpreted those groups as topics.

Topic models are unsupervised probabilistic methods designed to identify hidden thematic structures within document collections. Researchers have applied them to online reviews to extract customer-experience dimensions, product attributes, satisfaction factors, and topic-linked polarity [20–22]. Topic-coherence research also shows that model assessment depends on corpus representation and evaluation choices [23].

1. Data Subset

LDA model selection used a reproducible random sample of 20,000 preprocessed reviews. Candidate topic counts K = 3, 4, 5, 6, 7, and 8 were fitted under three random seeds (20260919–20260921). We compared models using held-out perplexity and the stability of the 15 highest-probability words, measured by optimal topic matching and mean Jaccard similarity across seeds.

We constructed a new Bag-of-Words representation from this subset. We removed terms occurring fewer than five times, then removed empty documents.

2. Topic Model Specification

The candidate number of topics ranged from K = 3 to K = 8.

$$K = 3$$

We fitted the model using MATLAB's fitlda function with the stochastic approximate variational Bayes solver.

Conceptually, LDA assumes that:

- Every document consists of a mixture of topics;
- Every topic consists of a probability distribution over words.

For review d, the topic-mixture vector can be expressed as:

$$\theta_d = (\theta_{d1}, \theta_{d2}, \ldots, \theta_{dK})$$

where θ_{dk} represents the estimated proportion of topic in document d.

For topic k, the word-probability vector is:

$$\phi_k = (\phi_{k1}, \phi_{k2}, \ldots, \phi_{kV})(\phi_{k1}, \phi_{k2}, \ldots, \phi_{kV})$$

where V is the vocabulary size and ϕ_{kv} is the probability of a vocabulary term v under topic k.

3. Topic Interpretation

We extracted the ten highest-probability words for each topic. We generated topic-specific word clouds. The topic labels were not assigned automatically by LDA. The researchers must interpret them after examining:

- The highest-probability terms;
- Representative reviews with high topic proportions;
- Coherence among the words;
- Distinctions from other topics.

Previous review-mining studies have demonstrated that LDA can reveal interpretable dimensions of tourist satisfaction, service experiences, and recurring customer concerns [20–22]. However, topic quality depends on preprocessing, sample composition, vocabulary filtering, the selected number of topics, and the evaluation measure [20–23].

4. Selection of K

We selected the number of topics empirically. Across the tested values, K = 3 produced the lowest mean held-out perplexity and the highest mean top-word stability (Jaccard = 0.721). Because automated topic scores do not guarantee semantic interpretability, topic labels remain provisional and are not used to support strong managerial claims.

5. Output

Method 2 produced:

- A preferred three-topic solution and candidate solutions for K = 4–8;
- Top words associated with each topic;
- Topic-specific word clouds;
- Document-topic membership proportions.

2.5.4. Method 3: TF-IDF-Based Logistic Regression for Rating-Derived Polarity Classification

The third method classified reviews as positive or negative using review text. The approach combined TF-IDF feature weighting with Logistic Regression.

Classical supervised sentiment classifiers can provide transparent text features and efficient baselines. This study uses TF-IDF with Logistic Regression as a conventional rating-derived polarity baseline, drawing on prior studies of review sentiment classification [15–18].

1. Label Construction

Sentiment labels were derived from review scores:

$$Y_i = \begin{cases} \text{Negative}, & Score_i \in \{1,2\} \\ \text{Positive}, & Score_i \in \{4,5\} \end{cases}$$

We excluded reviews with a score of 3 because we treated them as neutral or ambiguous. This is therefore a rating-derived sentiment classification. The labels were not independently annotated from the semantic content of the reviews. A potential limitation is that textual polarity and star rating may occasionally be inconsistent.

2. TF-IDF Feature Construction

ProductId split records before text feature construction. The TF-IDF vocabulary (unigrams and bigrams; min_df = 5; max_df = 0.98; maximum 60,000 features; sublinear term

frequency) was fitted only on the training partition and then applied unchanged to the test partition. This order prevents test documents from influencing vocabulary selection or inverse-document-frequency estimates.

A common TF-IDF formulation is:

$$\text{tfidf}(w,d) = \text{tf}(w,d) \times \log\left(\frac{N}{df(w)}\right)$$

where

- $\text{tf}(w,d)$ is the frequency of term w in document d;
- $df(w)$ is the number of documents containing w;
- N is the number of documents.

TF-IDF reduces the influence of terms common across documents while assigning greater weight to terms that distinguish documents. Here, the resulting text features are inputs to Logistic Regression; performance is evaluated on the independent product-grouped test set.

3. Train–Test Partition

The labeled data were divided using an 80/20 holdout:

- 80% training data;
- 20% testing data.

The eligible data were divided into product-grouped training (n = 420,764) and independent test (n = 104,282) partitions using seed 20260919. This design prevents reviews of the same product from appearing in both partitions. Exact duplicate records were removed before splitting. Both unweighted and class-weighted Logistic Regression models were evaluated against a majority-class baseline.

4. Logistic Regression Model

A linear classifier was trained using a logistic learner and the limited-memory Broyden–Fletcher–Goldfarb–Shanno optimizer.

For a binary outcome, the estimated probability of a positive review was:

$$P(Y_i = \text{Positive} \mid x_i) = \frac{1}{1 + \exp\left[-(\beta_0 + \beta^\top x_i)\right]}$$

where x_i is the TF-IDF vector of review i.

The predicted class was obtained from the fitted probability or decision score.

5. Evaluation

We evaluated the revised classifier on the independent ProductId-grouped test set. Because the test data were imbalanced, overall accuracy was interpreted together with balanced accuracy, macro-F1, ROC-AUC, negative-class PR-AUC, class-specific recall, and the confusion matrix.

However, accuracy should not be the only reported measure because the supplied result contains substantially more positive than negative test instances.

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

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

$$F1 = 2\left(\frac{Precision \times Recall}{Precision + Recall}\right)$$

Evaluation included accuracy, balanced accuracy, macro-F1, ROC-AUC, negative-class PR-AUC, class-specific recall, confusion matrices, and 95% bootstrap confidence intervals for balanced accuracy and macro-F1. Because labels were created from star ratings, the task is described consistently as rating-derived polarity classification, not independently annotated sentiment recognition.

6. Output

Method 3 produced:

- Positive or negative labels for unseen reviews;
- A confusion matrix;
- Overall accuracy;
- A trained sentiment-classification model.

2.6. Workflow B: Review Score Prediction

The regression tree predicts review scores from structural review features using ProductId-grouped training, validation, and test partitions. Performance is evaluated against a mean-prediction baseline using MAE, RMSE, and R^2. The revised workflow is shown in Figure 2.

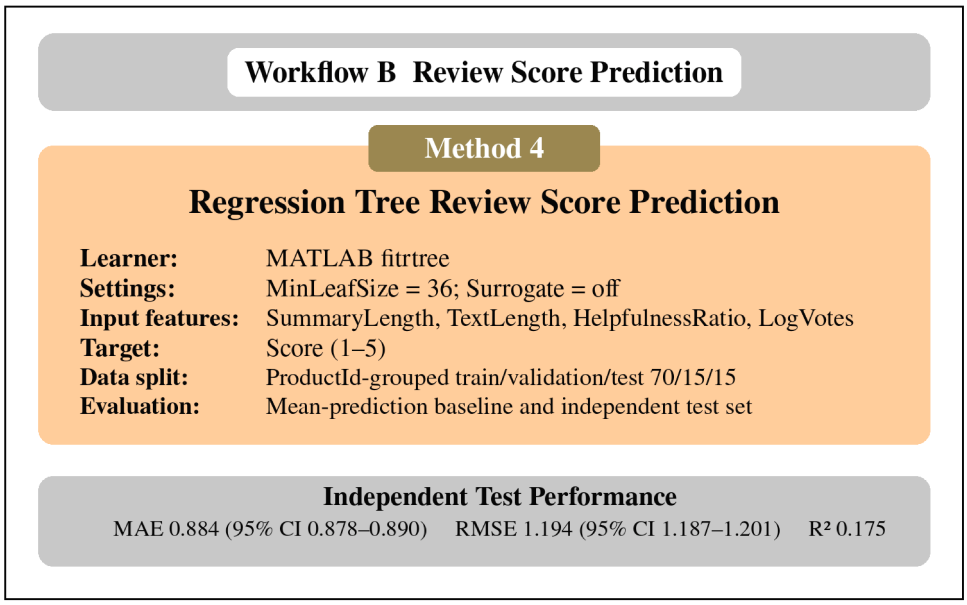


Figure 2. Workflow B: Review Score Prediction.

Method 4: Regression-Based Review Score Prediction

We conducted a preliminary regression-based score-prediction experiment using structured review attributes. Earlier work has predicted review ratings from aspect-level sentiment or topic information [24,25]; the present experiment instead tests how much variation can be explained by structural features alone.

1. Data Preparation

The Id, UserId, and ProfileName variables were removed. ProductId was retained solely for constructing the grouped training, validation, and test partitions and was excluded from the regression predictor set.

2. Feature Construction

Two length-based variables were created:

$$SummaryLength_i = \text{number of characters in Summary}_i$$
$$TextLength_i = \text{number of characters in Text}_i$$

The original text fields were then removed from this workflow. Consequently, Method 4 used text length but not its semantic content. The helpfulness ratio was calculated as:

$$HelpfulnessRatio_i = \frac{HelpfulnessNumerator_i}{\max(HelpfulnessDenominator_i, 1)}$$

To represent vote exposure while reducing the influence of highly skewed vote counts, we defined LogVotes as:

$$LogVotes_i = \log(1 + HelpfulnessDenominator_i)$$

where HelpfulnessDenominator represents the total number of helpfulness votes received by review i.

We bounded the denominator below by 1 to avoid division by zero. Under this rule, reviews receiving no votes were assigned a ratio of zero.

Review length, rating, linguistic features, and contextual information have been examined in studies of review helpfulness and credibility [26–32]. For the regression workflow, zero-vote reviews were assigned a structural HelpfulnessRatio value of zero to permit numerical modeling. For the segmentation workflow, however, these reviews were classified separately as Unrated because no observed helpfulness judgment was available.

3. Predictors and Target

The predictor set was:

$$X = \{SummaryLength, TextLength, HelpfulnessRatio, LogVotes\}$$

The target variable was:

$$Y = Score \in \{1, 2, 3, 4, 5\}$$

The score was treated as a numerical response, allowing the regression model to produce decimal predictions such as 4.61.

4. Model Training

The code called:

```
[trainedModel, validationRMSE] = trainRegressionModel(ReviewData);
```

Inspection of trainRegressionModel.m confirmed that the learner is a regression tree fitted by fitrtree with MinLeafSize = 36 and Surrogate = 'off'. The exported function uses SummaryLength, TextLength, and HelpfulnessRatio and reports five-fold cross-validation RMSE. In the revised independent evaluation, LogVotes was added to distinguish vote exposure from the helpfulness ratio, and ProductId-grouped train/validation/test partitions of approximately 70/15/15 were used.

The revised analysis explicitly reports the hyperparameters, validation scheme, feature definitions, model-selection criterion, random seed, grouped data split, and evaluation procedure. Feature normalization was not applied because the selected regression tree is invariant to monotonic rescaling of individual predictors.

5. Prediction and Error Calculation

The trained model generated predicted scores:

$$\hat{y}_i = f(x_i)$$

Prediction error was calculated as:

$$Diff_i = y_i - \hat{y}_i$$

and absolute error as:

$$AbsDiff_i = \mid y_i - \hat{y}_i \mid$$

The intended evaluation measures included validation RMSE and mean absolute error:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \mid y_i - \hat{y}_i \mid$$

We removed the workspace-name collision that shadowed MATLAB's mean function. We calculated MAE, RMSE, and R^2 on the independent test set, with 95% bootstrap confidence intervals for MAE and RMSE. A training-set mean predictor served as the baseline.

6. Evaluation Limitation

The revised evaluation does not score the training data as if it were unseen. Model selection is confined to the training/validation partitions, and final metrics are reported once on the product-grouped test partition.

7. Interpretation Limitation

Because this method uses only review length, helpfulness ratio, and vote exposure, it cannot directly distinguish positive from negative language. A long negative review may have structural features similar to a long positive review. This explains why some 1-star observations in the example were predicted near 4–5 stars.

Previous research has used aspect-level sentiment and topic membership in review-rating prediction [24,25].

8. Output

Method 4 produced test-set predictions and complete MAE, RMSE, R^2, and uncertainty estimates under the grouped evaluation design.

2.7. Workflow C: Review Segmentation

Workflow C classifies reviews by rating group and helpfulness status, with zero-vote reviews treated separately as Unrated. The resulting nine groups are analyzed using cross-tabulation and sensitivity analysis across helpfulness thresholds and minimum-vote requirements, as summarized in Figure 3.

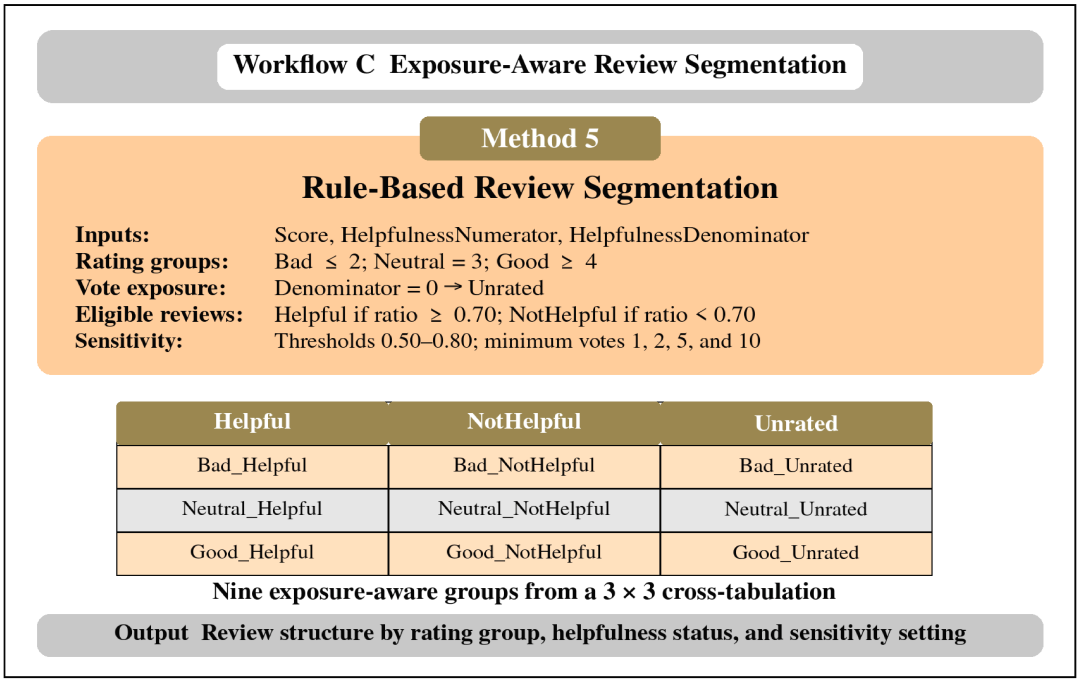


Figure 3. Workflow C: Exposure-Aware Review Segmentation.

Method 5: Rule-Based Review Segmentation

Method 5 summarized the structure of the review population by jointly considering:

- Rating-derived sentiment;
- Review helpfulness.

The groups were formed using predefined thresholds rather than an unsupervised clustering algorithm; therefore, the procedure was classified as rule-based review segmentation.

1. Sentiment Segmentation

Each review was assigned one of three sentiment categories based on its score:

$$Sentiment_i = \begin{cases} Bad, & Score_i \leq 2 \\ Neutral, & Score_i = 3 \\ Good, & Score_i \geq 4 \end{cases}$$

These categories are based on ratings and should not be confused with the text-derived sentiment predictions produced by Method 3.

2. Helpfulness Segmentation

The helpfulness ratio was calculated as:

$$H_i = \frac{HelpfulnessNumerator_i}{\max(H, elpfulnessDenominator_i 1)}$$

For reviews with at least one vote, we retained 0.70 as a descriptive reference threshold rather than a universal cutoff. Reviews with zero votes were assigned to a separate Unrated category. Sensitivity analysis varied the threshold (0.50, 0.60, 0.70, and 0.80) and minimum vote requirement (1, 2, 5, and 10). Review age is acknowledged as an imperfect proxy for exposure because page position and impressions are unavailable.

$$HelpfulLabel_i = \begin{cases} Helpful, & H_i \geq 0.70 \\ NotHelpful, & H_i < 0.70 \end{cases}$$

Prior work has studied review helpfulness using textual and contextual features and models of voting behavior [26–32]. In this study, the helpful-vote ratio is calculated only for reviews with at least one vote. We predefined a 0.70 threshold as an analytical cutoff to distinguish Helpful from NotHelpful reviews and did not treat it as a universally established threshold.

3. Nine Exposure-Aware Groups

The rating and exposure-aware helpfulness labels were cross-classified into nine descriptive categories, including a separate Unrated status for each rating group.

Conceptually, the cross-classification can be expressed as:

$$Group_i = Sentiment_i \times HelpfulLabel_i$$

This segmentation makes it possible to distinguish, for example:

- Negative reviews that users nevertheless found informative;
- Highly rated reviews that received strong helpfulness support;
- Neutral reviews with limited or substantial perceived usefulness.

4. Descriptive Analysis

A 3 × 3 cross-tabulation was computed:

$$C_{s,h} = \text{number of reviews in sentiment } s \text{ and helpfulness category } h$$

Within-sentiment proportions were calculated as:

$$P_{s,h} = \frac{C_{s,h}}{\sum_h C_{s,h}}$$

The results were visualized using:

- A 3 × 3 table of counts;
- A 100% stacked bar chart of within-sentiment proportions;
- Sensitivity tables for threshold and minimum-vote choices.

Because rating categories are constructed from Score, no score-by-segment boxplot is interpreted as independent evidence.

5. Treatment of Reviews with No Votes

The revised method does not assign a helpfulness ratio to reviews with HelpfulnessDenominator = 0. These 270,052 records (47.51% of the dataset) are labeled Unrated and excluded from ratio-based Helpful versus NotHelpful comparisons.

The earlier rule combined low helpfulness ratios with absence of votes; the revision separates these situations.

6. Output

Method 5 produced:

- Nine exposure-aware descriptive categories;
- Group counts and percentages;
- A sentiment-by-helpfulness cross-tabulation;
- Threshold and minimum-vote sensitivity summaries;
- An overall profile of the review population.

2.8. Relationship Among the Five Methods

1. Methods 1–3 comprise three complementary analyses within Workflow A:

$$\text{Text preprocessing} \rightarrow \text{Exploratory terms} \rightarrow \text{Latent topics} \rightarrow \text{Sentiment classes}$$

2. Method 4 constitutes a separate supervised regression workflow:

$$\text{Structured review features} \rightarrow \text{Predicted score}$$

3. Method 5 constitutes a descriptive rule-based workflow:

$$\text{Score} + \text{HelpfulnessRatio} \rightarrow \text{Review segments}$$

Thus, the five methods share a dataset, theoretical motivation, and reporting layer, but they remain separate analytical branches. The article consistently uses the term "multi-branch framework" and does not claim that the analytical branches form a single integrated learner. This scope improves interpretability and prevents rating-derived labels from leaking into rating prediction.

3. Results

The revised experiments used the full eligible data for classification, regression, and segmentation, and a reproducible random sample of 20,000 reviews for LDA model selection. All predictive results below are from independent ProductId-grouped test partitions.

3.1. Exploratory Text-Mining Results

Exploratory term summaries were recomputed on the reproducible 20,000-review random sample used for topic-model comparison. Because word clouds are illustrative rather than inferential, the revision does not treat vocabulary size or font size as evidence of prevalence in the full population.

The overall Word Cloud indicated that frequently occurring words included good, like, taste, great, product, flavor, food, coffee, buy, and price. These terms suggest that customers commonly discussed taste and flavor, general product quality, purchasing decisions, beverages, and value-related considerations.

Separate Word Clouds were also generated for positive reviews rated 4–5 stars and negative reviews rated 1–2 stars. The positive-review Word Cloud prominently contained favorable terms such as good, great, love, and delicious. The negative-review Word Cloud included unfavorable terms such as bad and disappointing, although neutral or favorable words such as good, like, taste, and product also appeared.

This overlap demonstrates that the presence of an individual word does not necessarily determine the sentiment of an entire review. For example, a negative review may state that a product tasted good but arrived damaged, was overpriced, or did not match its description. Therefore, Word Clouds are useful for exploratory interpretation but should not be treated as a complete sentiment-analysis method.

Objective assessment: The objective of identifying frequently discussed terms and broad customer concerns was achieved. The results revealed recurring product-related vocabulary and provided an interpretable overview of how customers described their experiences.

3.2. Topic-Modeling Results

We evaluated candidate LDA solutions from K = 3 to K = 8 on a held-out subset under three random seeds. Mean held-out perplexity was lowest for K = 3 (1457.35) and

increased for larger solutions; K = 3 also achieved the highest top-word stability (mean Jaccard = 0.721), compared with 0.628 for K = 4 and 0.586 for K = 5. The earlier fixed K = 5 solution was therefore not retained as the preferred model.

The stability analysis indicates that increasing K produced less reproducible top-word sets. The preferred K = 3 result is treated as a broad exploratory partition rather than a definitive taxonomy. Representative-review inspection and independent human labeling remain necessary before assigning business-facing topic names.

The comparison resolves the arbitrary-K concern but does not by itself establish semantic validity. Consequently, LDA answers RQ1 at an exploratory level, and no causal or prescriptive managerial conclusion is based solely on its topics.

A cautious interpretation is that broad themes concern product taste, food types, purchasing experience, price, and customer preference. The preferred K = 3 solution is exploratory; specific topic labels require inspection of representative reviews and independent human validation.

Objective assessment: The objective of identifying latent review topics was partially achieved. The model produced topic-word distributions and document-topic proportions, but stronger topic separation is needed before the results can be translated into reliable managerial categories.

We removed the former fixed-K topic figures because they no longer reflect the selected specification. Detailed K-by-seed results are provided in the Data Availability Statement section.

Figure 4 shows the candidate LDA models evaluated across three random seeds. The K = 3 model achieved the lowest mean held-out perplexity and the highest mean topic-word stability and was therefore selected for the revised analysis. Error bars represent ±1 standard deviation across random seeds.

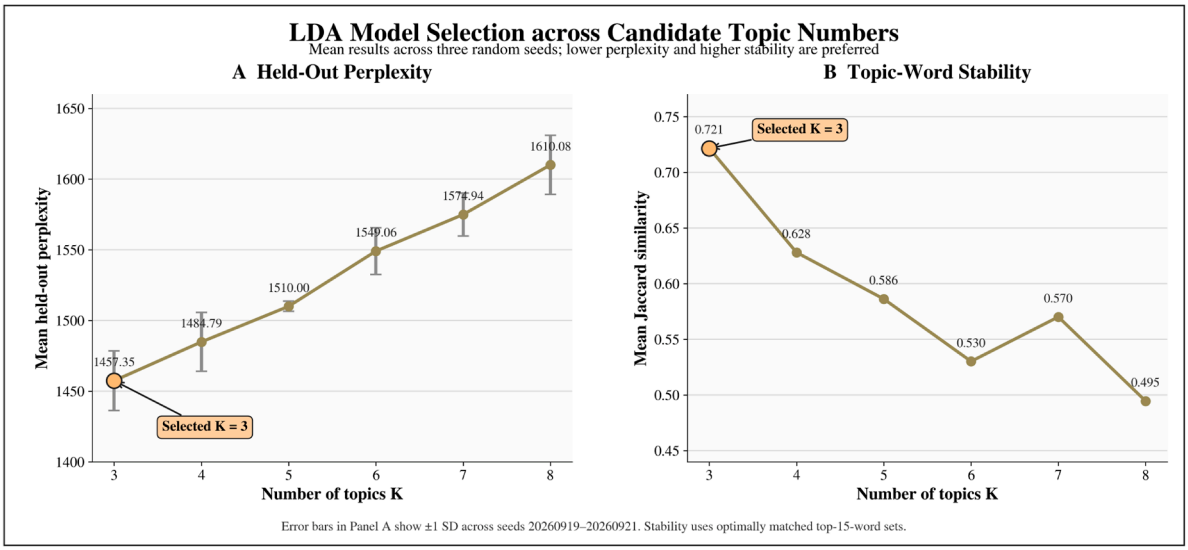


Figure 4. LDA Model Selection across Candidate Topic Numbers.

3.3. Rating-Derived Polarity Classification Results

The revised rating-derived polarity experiment used product-grouped holdout evaluation. The unweighted model achieved 95.66% accuracy, 88.91% balanced accuracy, 91.25% macro-F1, and 79.10% negative recall. Class weighting increased negative recall to 93.29% and balanced accuracy to 94.04%, while overall accuracy decreased modestly to 94.55%. The majority-class baseline achieved 84.41% accuracy but only 50.00% balanced accuracy and 45.77% macro-F1.

For the selected class-weighted model, the negative/positive confusion matrix was [[15,165, 1090], [4591, 83,436]]. ROC-AUC was 98.35%, and negative-class PR-AUC was 93.27%. The 95% bootstrap interval was 93.83–94.22% for balanced accuracy and 90.26–90.70% for macro-F1. These measures show that accuracy alone would materially understate the imbalance problem and the benefit of cost-sensitive learning.

The class-weighted model is preferable when missing a negative review carries greater operational cost. Nevertheless, its labels are rating-derived and therefore measure recovery of star-rating polarity from text, not independent psychological sentiment.

Objective assessment: RQ2 was answered under leakage-aware, product-grouped evaluation. The classifier substantially outperformed the majority baseline, but construct validity remains limited by rating-derived labels.

Table 1 reports the primary performance metrics for the revised product-grouped evaluation. The previous confusion-matrix figure was removed because the complete revised evaluation is reported in the table and accompanying text.

Table 1. Rating-Derived Polarity Classification Performance.

Metric	Selected Model	95% CI or Note
Accuracy	94.55%	Class-weighted LR
Balanced accuracy	94.04%	93.83–94.22%
Macro-F1	90.47%	90.26–90.70%
Negative recall	93.29%	Unweighted: 79.10%
Negative PR-AUC	93.27%	Majority accuracy: 84.41%

3.4. Review-Score Prediction Results

The revised regression workflow used SummaryLength, TextLength, HelpfulnessRatio (zero only for the structural feature calculation), and LogVotes. ProductId-grouped train, validation, and test partitions contained 402,224, 82,939, and 82,460 records, respectively.

- Summary length;
- Full-text length;
- Helpfulness Ratio.
- Log-transformed Vote Count.

The mean-prediction baseline produced MAE = 1.046, RMSE = 1.314, and R^2 approximately 0.000. The regression tree (MinLeafSize = 36) improved performance to MAE = 0.884 (95% CI 0.878–0.890), RMSE = 1.194 (95% CI 1.187–1.201), and R^2 = 0.175 on the independent test set (Table 2).

Table 2. Score Prediction Results.

Model	Test Metrics	Interpretation
Mean predictor	MAE 1.046; RMSE 1.314; R^2~0	Baseline
Regression tree	MAE 0.884; RMSE 1.194; R^2 0.175	Modest improvement

The positive but modest R^2 indicates that structural review features explain only a limited share of rating variation. The model should not be used to replace observed ratings or infer customer satisfaction at the individual-review level.

The remaining error is consistent with the absence of semantic text features and the concentration of ratings at four and five stars. Future work could test integrating topic proportions; rating-derived polarity probabilities must be generated strictly out of fold, and even then their circular relationship to the score target must be disclosed.

We corrected the MAE calculation bug by removing the workspace variable named mean and evaluating the final model on the independent test set.

The regression algorithm, hyperparameters, split sizes, baseline, and uncertainty intervals are now reported in full.

Objective assessment: RQ3 was answered. The regression tree improved on a mean predictor, but the low R^2 supports only a cautious, exploratory interpretation.

3.5. Rule-Based Review-Segmentation Results

The segmentation workflow divided reviews according to rating-derived sentiment and Helpfulness Ratio.

Sentiment was defined as:

- Bad: 28.97%/43.15%/27.88%;
- Neutral: 29.07%/27.72%/43.21%;
- Good: 39.44%/9.01%/51.55%.

Helpfulness status was defined only for reviews with at least one vote:

- Helpful: Helpfulness Ratio of at least 0.70; NotHelpful: ratio below 0.70; Unrated: no helpfulness votes.

The revised cross-tabulation used three exposure-aware statuses. At the 0.70 threshold, Bad reviews comprised 23,767 Helpful, 35,400 NotHelpful, and 22,870 Unrated records; Neutral reviews comprised 12,397, 11,820, and 18,423; and Good reviews comprised 175,043, 39,975, and 228,759, respectively (Table 3).

Table 3. Rule-Based Review-Segmentation Results.

Rating Group	Helpful	Not Helpful	Unrated
Bad	23,767	35,400	22,870
Neutral	12,397	11,820	18,423
Good	175,043	39,975	228,759
Total	211,207	87,195	270,052

Among voted reviews, the Helpful share changed materially with the analytical choices:

- At minimum one vote: 83.23% (threshold 0.50), 75.46% (0.60), 70.80% (0.70), and 67.62% (0.80).
- At minimum five votes: 81.74% (0.50), 77.11% (0.60), 69.56% (0.70), and 63.70% (0.80).
- At minimum ten votes: 81.91% (0.50), 77.69% (0.60), 72.73% (0.70), and 65.69% (0.80).

Because 47.51% of all reviews had no helpfulness votes, treating them as NotHelpful would conflate lack of exposure with an unfavorable evaluation. Segment sizes are therefore descriptive and conditional on exposure.

At the 0.70 reference threshold, within-category percentages for Helpful/NotHelpful/Unrated were:

- Bad: 23,767/82,037 $\approx$ 29.0%.
- Neutral: 12,397/42,640 $\approx$ 29.1%.
- Good: 175,043/443,777 $\approx$ 39.4.

The association between rating category and observed helpfulness does not establish causality. Page position, visibility, product popularity, and review age may affect vote exposure but are not fully observed in the dataset.

RQ4 was answered by explicitly separating Unrated reviews and reporting threshold/minimum-vote sensitivity. The 0.70 result is retained only as one reference specification.

The crosstab output mapping was corrected. Revised tables use verified category order and labels; we removed the erroneous heatmap and legend.

Objective assessment: The objective of describing the review population according to sentiment and helpfulness was achieved at a descriptive level. However, the method should be presented as rule-based segmentation rather than unsupervised clustering.

4. Discussion

4.1. Complementary Value of the Multi-Branch Framework

The contribution is not a new predictive architecture. It is a transparent, reproducible comparison of complementary branches that expose different aspects of review data and their limits of validity. Lexical and topic analyses describe content; rating-derived polarity classification quantifies how well rating valence can be recovered from text; structural regression tests whether nonsemantic attributes explain ratings; and exposure-aware segmentation summarizes observed helpfulness patterns.

The branches are intentionally not fused through rating-derived polarity because doing so would reintroduce the score target into score prediction. Future integration should use genuinely independent text representations, nested out-of-fold estimation, and ablation tests demonstrating value beyond each standalone branch.

4.2. Interpretation of the Text-Mining Results

The exploratory text-mining results showed that frequently occurring terms included taste, flavor, good, product, price, and coffee. These words suggest that customers commonly focused on product taste, flavor, general quality, value, and beverage-related characteristics. The findings therefore provide an initial overview of the product attributes and experiences that customers frequently mentioned in their reviews.

Nevertheless, Word Clouds reflect only word frequency and do not fully capture the context in which the words were used. The same word may occur in both positive and negative reviews. For example, a customer may describe a product as having a good taste but still complain about its price, packaging, or delivery. Therefore, interpret word cloud findings alongside sentiment classification and topic modeling, rather than using them as standalone evidence of customer satisfaction or dissatisfaction.

4.3. Practical Implications

The evidence supports human-in-the-loop screening. The class-weighted model may surface potentially negative reviews for staff review, and topic summaries may support exploratory monitoring. These outputs should not trigger automated customer or product decisions without contemporary external validation.

Helpfulness segments are conditional on receiving votes. Unrated reviews must remain separate, and conclusions should be checked across thresholds and minimum-vote rules. The Bad-Helpful segment can be inspected as a candidate queue, but the present data do not show that prioritizing it improves business outcomes.

Accordingly, managerial implications are limited to descriptive prioritization and hypothesis generation. Claims about product development, complaint resolution, sales, or customer relationships require prospective validation and outcome data not available here.

5. Conclusions

This paper presents a multi-branch framework for review analytics. It revises the evaluation to address sampling bias, data leakage, class imbalance, arbitrary topic selection,

regression validation, and no-vote handling. The class-weighted rating-derived polarity model achieved 94.04% balanced accuracy and 93.29% negative recall on a product-grouped test set. K = 3 was the most stable of the tested LDA solutions. The regression tree improved on a mean baseline but explained only 17.45% of test-set variance. Helpfulness conclusions changed with threshold and vote-count rules, and 47.51% of reviews were Unrated.

The dataset ends in October 2012, so external validity for current platforms, interfaces, recommendation systems, and consumer language is unproven. Future research should preregister tests on newer platforms and product categories; use independently annotated sentiment; compare LDA [20,23] with BERTopic [33], using BERT or Sentence-BERT embeddings [34,35] where appropriate under the same grouped splits; validate topic labels with human coders; model vote exposure; and evaluate whether review prioritization improves real managerial outcomes. Interpret the present findings as a reproducible benchmark on a historical dataset.

Author Contributions: Conceptualization, P.P., N.K., W.P. and P.V.; methodology, P.P., N.K., W.P. and P.V.; software, P.P., N.K. and W.P.; validation, F.P., S.C., P.P., N.K., W.P. and P.V.; formal analysis, F.P. and P.V.; investigation, F.P., P.V. and K.H.R.; resources, F.P., S.C., P.P., N.K. and W.P.; data curation, P.P., N.K., W.P.; writing—original draft preparation, F.P.; writing—review and editing, F.P. and P.V.; visualization, F.P., S.C., P.P., N.K. and W.P.; supervision, P.V.; project administration, F.P., S.C., P.V. and K.H.R.; funding acquisition, F.P. and P.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository. See link https://drive.google.com/drive/folders/1YVXMT2Irt-rcZ3-IyFr0h9gsDSjgGvQ_?usp=sharing (accessed on 15 August 2026).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, X.; Wu, C.; Mai, F. The effect of online reviews on product sales: A joint sentiment-topic analysis. *Inf. Manag.* **2019**, *56*, 172–184. [CrossRef]
2. Ireland, R.; Liu, A. Application of data analytics for product design: Sentiment analysis of online product reviews. *CIRP J. Manuf. Sci. Technol.* **2018**, *23*, 128–144. [CrossRef]
3. Eslami, S.P.; Ghasemaghaei, M. Effects of online review positiveness and review score inconsistency on sales: A comparison by product involvement. *J. Retail. Consum. Serv.* **2018**, *45*, 74–80. [CrossRef]
4. Shihab, M.R.; Putri, A.P. Negative online reviews of popular products: Understanding the effects of review proportion and quality on consumers' attitude and intention to buy. *Electron. Commer. Res.* **2019**, *19*, 159–187. [CrossRef]
5. Lucini, F.R.; Tonetto, L.M.; Fogliatto, F.S.; Anzanello, M.J. Text mining approach to explore dimensions of airline customer satisfaction using online customer reviews. *J. Air Transp. Manag.* **2020**, *83*, 101760. [CrossRef]
6. Wu, J.-J.; Chang, S.-T. Exploring customer sentiment regarding online retail services: A topic-based approach. *J. Retail. Consum. Serv.* **2020**, *55*, 102145. [CrossRef]
7. Xu, X. Examining an asymmetric effect between online customer reviews emphasis and overall satisfaction determinants. *J. Bus. Res.* **2020**, *106*, 196–210. [CrossRef]
8. Zhao, Y.; Wen, L.; Feng, X.; Li, R.; Lin, X. How managerial responses to online reviews affect customer satisfaction: An empirical study based on additional reviews. *J. Retail. Consum. Serv.* **2020**, *57*, 102205. [CrossRef]
9. Zhang, J.; Weng, Z.; Ni, N. Helping consumers to overcome information overload with a diversified online review subset. *Front. Bus. Res. China* **2019**, *13*, 15. [CrossRef]
10. Wang, J.-N.; Du, J.; Chiu, Y.-L. Can online user reviews be more helpful? Evaluating and improving ranking approaches. *Inf. Manag.* **2020**, *57*, 103281. [CrossRef]

11. Mitra, S.; Jenamani, M. Helpfulness of online consumer reviews: A multi-perspective approach. *Inf. Process. Manag.* **2021**, *58*, 102538. [CrossRef]
12. Amazon Fine Food Reviews. SNAP: Stanford Network Analysis Project. Available online: https://snap.stanford.edu/data/web-FineFoods.html (accessed on 25 September 2026).
13. Mäntylä, M.V.; Graziotin, D.; Kuutila, M. The evolution of sentiment analysis—A review of research topics, venues, and top cited papers. *Comput. Sci. Rev.* **2018**, *27*, 16–32. [CrossRef]
14. Rambocas, M.; Pacheco, B.G. Online sentiment analysis in marketing research: A review. *J. Res. Interact. Mark.* **2018**, *12*, 146–163. [CrossRef]
15. Zablith, F.; Osman, I.H. ReviewModus: Text classification and sentiment prediction of unstructured reviews using a hybrid combination of machine learning and evaluation models. *Appl. Math. Model.* **2019**, *71*, 569–583. [CrossRef]
16. Bansal, B.; Srivastava, S. Sentiment classification of online consumer reviews using word vector representations. *Procedia Comput. Sci.* **2018**, *132*, 1147–1153. [CrossRef]
17. Alantri, H.J.; Currim, I.S.; Deng, Y.; Singh, S. An empirical comparison of machine learning methods for text-based sentiment analysis of online consumer reviews. *Int. J. Res. Mark.* **2022**, *39*, 1–19. [CrossRef]
18. Yucel, A.; Dag, A.; Oztekin, A.; Carpenter, M. A novel text analytic methodology for classification of product and service reviews. *J. Bus. Res.* **2022**, *151*, 287–297. [CrossRef]
19. Salton, G.; Buckley, C. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.* **1988**, *24*, 513–523. [CrossRef]
20. Guo, Y.; Barnes, S.J.; Jia, Q. Mining meaning from online ratings and reviews: Tourist satisfaction analysis using latent Dirichlet allocation. *Tour. Manag.* **2017**, *59*, 467–483. [CrossRef]
21. Kirilenko, A.P.; Stepchenkova, S.O.; Dai, X. Automated topic modeling of tourist reviews: Does the Anna Karenina principle apply? *Tour. Manag.* **2021**, *83*, 104241. [CrossRef]
22. Hong, M.; Wang, H. Research on customer opinion summarization using topic mining and deep neural network. *Math. Comput. Simul.* **2021**, *185*, 88–114. [CrossRef]
23. Syed, S.; Spruit, M. Full-text or abstract? Examining topic coherence scores using latent Dirichlet allocation. In Proceedings of the 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Tokyo, Japan, 19–21 October 2017; pp. 165–174. [CrossRef]
24. Qiu, J.; Lin, Z.; Li, Y. Leveraging sentiment analysis at the aspects level to predict ratings of reviews. *Inf. Sci.* **2018**, *451–452*, 295–309. [CrossRef]
25. Yang, N.; Korfiatis, N.; Zissis, D.; Spanaki, K. Incorporating topic membership in review rating prediction from unstructured data: A gradient boosting approach. *Ann. Oper. Res.* **2024**, *339*, 631–662. [CrossRef]
26. Wu, C.; Mai, F.; Li, X. The effect of content depth and deviation on online review helpfulness: Evidence from a double-hurdle model. *Inf. Manag.* **2021**, *58*, 103408. [CrossRef]
27. Huang, G.; Liang, H. Uncovering the effects of textual features on trustworthiness of online consumer reviews: A computational-experimental approach. *J. Bus. Res.* **2021**, *126*, 1–11. [CrossRef]
28. Du, J.; Zheng, L.; He, J.; Rong, J.; Wang, H.; Zhang, Y. An interactive network for end-to-end review helpfulness modeling. *Data Sci. Eng.* **2020**, *5*, 261–279. [CrossRef]
29. Pooja, K.; Upadhyaya, P. What makes an online review credible? A systematic review of the literature and future research directions. *Manag. Rev. Q.* **2024**, *74*, 627–659. [CrossRef]
30. Qin, J.; Zheng, P.; Wang, X. Comprehensive helpfulness of online reviews: A dynamic strategy for ranking reviews by intrinsic and extrinsic helpfulness. *Decis. Support Syst.* **2022**, *163*, 113859. [CrossRef]
31. Sengo Furtado, F.; Reutterer, T.; Schröder, N. The carrot and the stick in online reviews: Determinants of un-/helpfulness voting choices. *J. Bus. Econ.* **2022**, *92*, 565–590. [CrossRef]
32. Saumya, S.; Roy, P.K.; Singh, J.P. Review of helpfulness prediction on e-commerce websites: A comprehensive survey. *Eng. Appl. Artif. Intell.* **2023**, *126*, 107075. [CrossRef]
33. Grootendorst, M. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv* **2022**, arXiv:2203.05794. [CrossRef]
34. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. [CrossRef]
35. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Proceedings of the EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; pp. 3982–3992. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.