

Customer Review Sentiment Classifier

Goal:

Create an NLP model that automatically categorizes reviews into positive, neutral or negative sentiments.

Problem Statement:

Businesses struggle to efficiently interpret large amounts of unstructured customer feedback. Manual methods are slow and inconsistent, creating a need for an automated sentiment classifier that can reliably categorize reviews and support faster decision-making.

Context:

Customer reviews hold important feedback, but their volume makes manual analysis impractical. This project uses NLP techniques on the Amazon Reviews dataset to automatically classify sentiments and support data-driven decisions.

Criteria for Success:

- Accurate classification of reviews into positive, neutral, and negative ($\geq 85\%$ accuracy).
- Balanced performance across classes (macro F1-score ≥ 0.80).
- Efficient processing of large review volumes.
- Clear, actionable insights for stakeholders.

Scope of Solution Space

- Analyze Amazon customer reviews and classify them as positive, neutral, or negative.
- Include text preprocessing, feature extraction, and model training.
- Provide sentiment labels and summary insights to support business decisions.

Constraints

- Dataset may contain missing, noisy, or imbalanced reviews.
- Limited time and computational resources restrict extensive model training.
- The model may not fully capture sarcasm, slang, or ambiguous sentiment.
- Trained models may not generalize perfectly to reviews from other platforms.

Stakeholders:

- Product Managers
- Customer Success & Support Teams
- Business Analysts & Data Analysts
- Marketing Teams
- Operations & Quality Assurance Teams
- Executive Leadership
- End Customers
- Data Science / Engineering Team

Data Sources

- Amazon US Customer Reviews Dataset
<https://www.kaggle.com/datasets/cynthiarempel/amazon-us-customer-reviews-dataset/data>

Solution Approach (Brief Outline)

1. Data Acquisition & Labeling
 - Use the Amazon US Customer Reviews dataset from Kaggle.
 - Convert star ratings into sentiment classes (positive, neutral, negative).
2. Exploratory Data Analysis
 - Examine class distribution, review lengths, noise levels, and linguistic patterns.
 - Identify missing values, duplicates, and non-English text.
3. Text Preprocessing
 - Clean and normalize review text (lowercase, remove special characters, handle formatting).
 - Tokenize text for feature extraction.
4. Feature Engineering & Modeling
 - Extract features using TF-IDF and/or sentence embeddings.
 - Train multiple models ranging from classical ML (Logistic Regression) to transformer-based models (e.g., DistilBERT).
 - Address class imbalance using weighting or sampling techniques.
5. Model Evaluation & Selection
 - Evaluate using accuracy, precision, recall, and macro F1-score.
 - Select the model that best meets the project's success criteria.
6. Insight Generation
 - Summarize sentiment distributions and identify common themes in negative reviews.
 - Provide business insights that support product and service improvements.

Deliverables

1. Complete Codebase
 - Well-documented scripts/notebooks for preprocessing, modeling, evaluation, and inference.
2. Trained Sentiment Classification Model
 - Optimized final model with all necessary files (vectorizer/tokenizer).
3. Technical Report / Paper
 - Covers methodology, dataset overview, modeling steps, results, limitations, and future work.
4. Stakeholder Slide Deck
 - Communicates goals, approach, model performance, and key business insights..
5. Sentiment Insight Summary
 - Highlights trends, common complaint themes, and actionable recommendations..