Sentiment Analysis amazon reviews

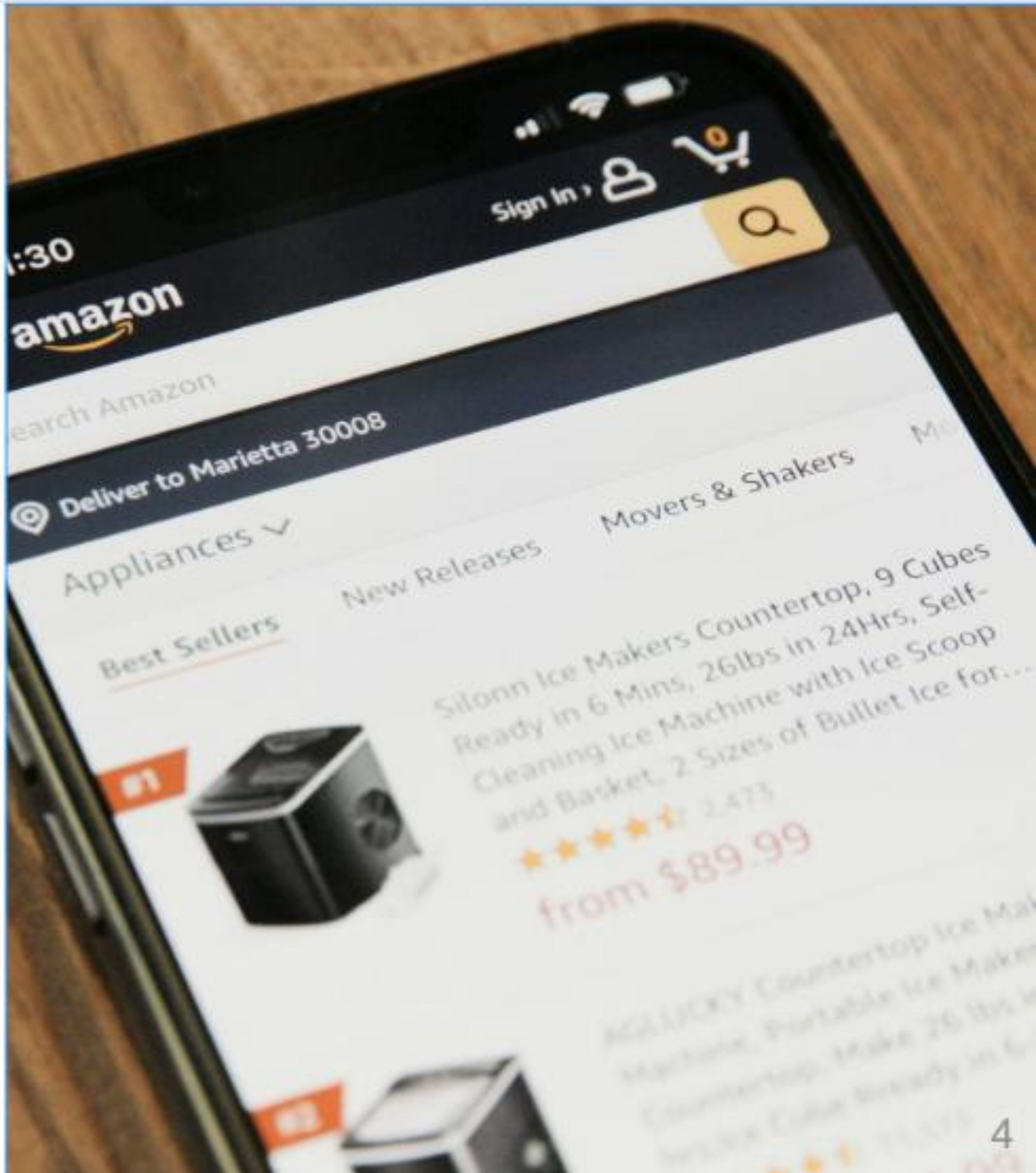
Team Members

1- Shahd Hesham
2- Ranwa Gamal
3- Mona Sherif
4- Basmala Salama
5- Menna Ramadan

problem statement

Customer reviews play a critical role in shaping buying decisions. However, understanding these reviews is challenging due to varying user behaviors and sentiments. Our project aims to analyze these patterns to support fair and accurate product recommendations

Dataset Description



We used the **Amazon Reviews'23 dataset** collected by the McAuley Lab, which contains real customer reviews from the Amazon platform.
Key Columns:

- rating — (float64) → Numerical rating score given by the customer.
- text — (object) → The written review text.
- verified_purchase — (bool) → Indicates whether the review comes from a verified purchase.
- helpful_vote → Number of users who marked the review as helpful.

-This dataset provides valuable insights into customer opinions, behaviors, and the factors that influence product trust and recommendation.

Key Insights from Basic EDA

Sentiment Distribution:

- The majority of reviews are positive, followed by negative and neutral, showing a clear lean toward customer satisfaction.

Helpful Votes Distribution:

- Most reviews receive a small number of helpful votes, but some reviews are highly voted, which highlights their influence on other buyers.

Labeling Sentiment:

A custom sentiment label was generated based on the rating:

- Ratings 4-5 = Positive
- Rating 3 = Neutral
- Ratings 1-2 = Negative

Data Preprocessing & Feature Engineering

1. Data Cleaning:

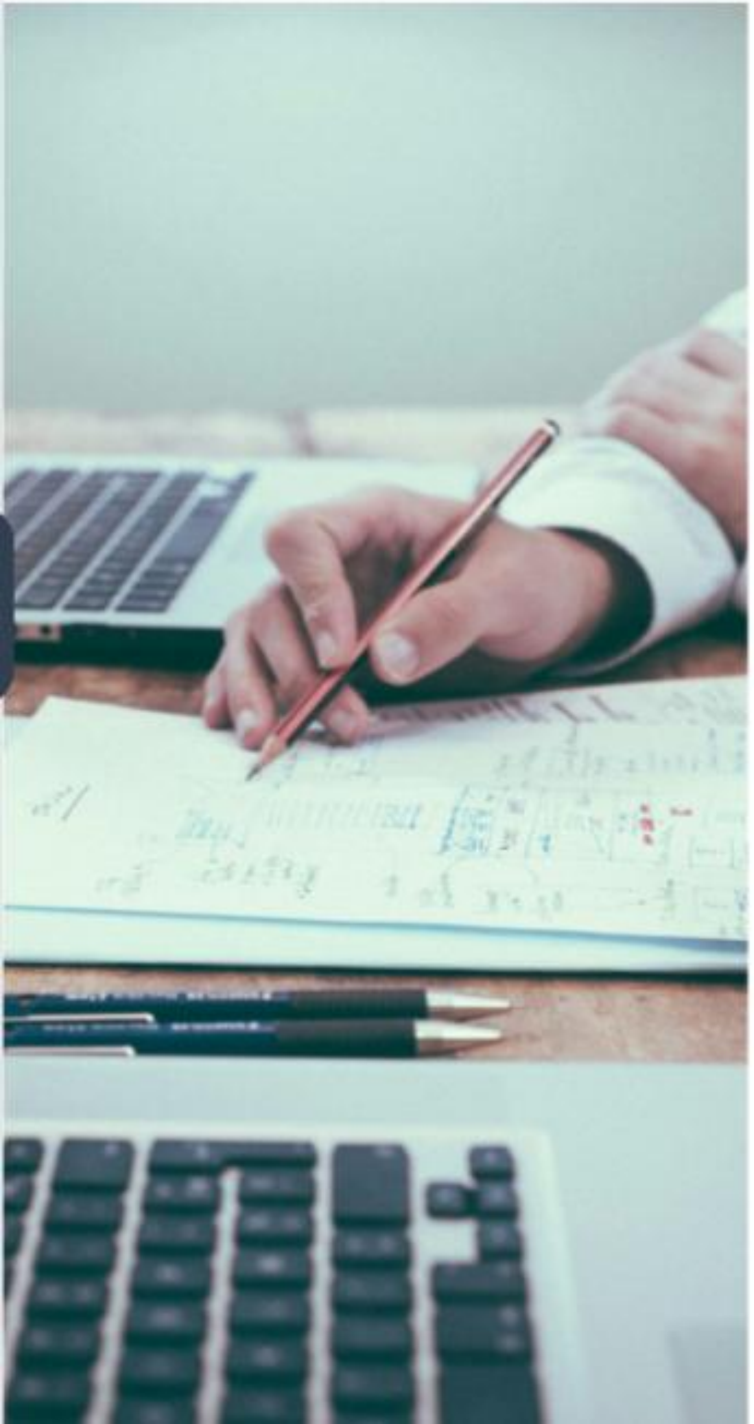
The dataset contained many incomplete rows
we removed them to ensure clean and reliable data for analysis.

2. Data Balancing (undersampling):

- The data was imbalanced (positive much higher than neutral and negative).
- We used undersampling to balance the three categories.
- Outcome: 12470 for each category.

3. Text Cleaning & Preparation:

- Converted all text to lowercase.
- Removed punctuation & stop words.
- Expanded contractions (e.g., I'm → I am).
- Applied lemmatization to get words in their base form



3- Label Encoding:

Converted sentiments into numeric format:

- negative → 0
- neutral → 1
- positive → 2

4.Feature Extraction:

Used two methods to convert text into numerical features:

TF-IDF : Assigns weight to words based on frequency.

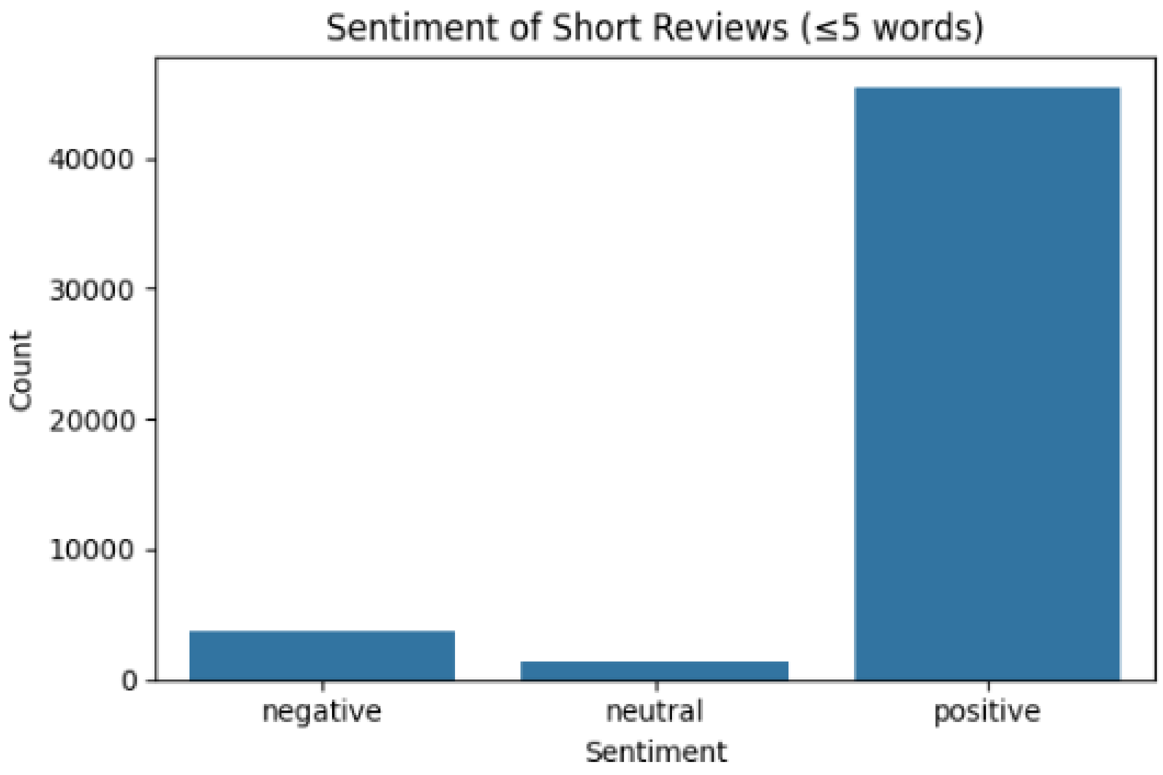
CountVectorizer:Counts word frequencies.

5. New Feature Creation:

Sentiment: Based on the review rating (Positive, Neutral, Negative).

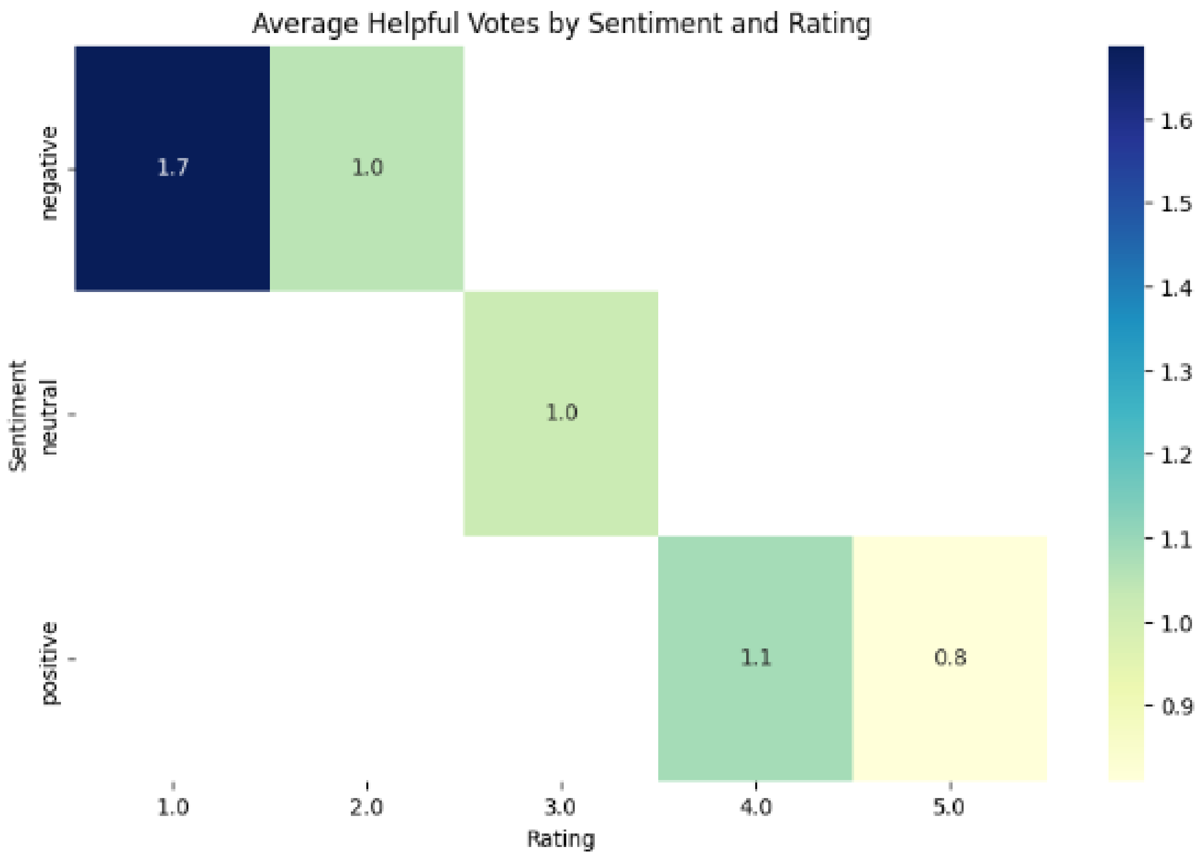
Review Length: Measured the number of words in each review.

Key Question & Insight

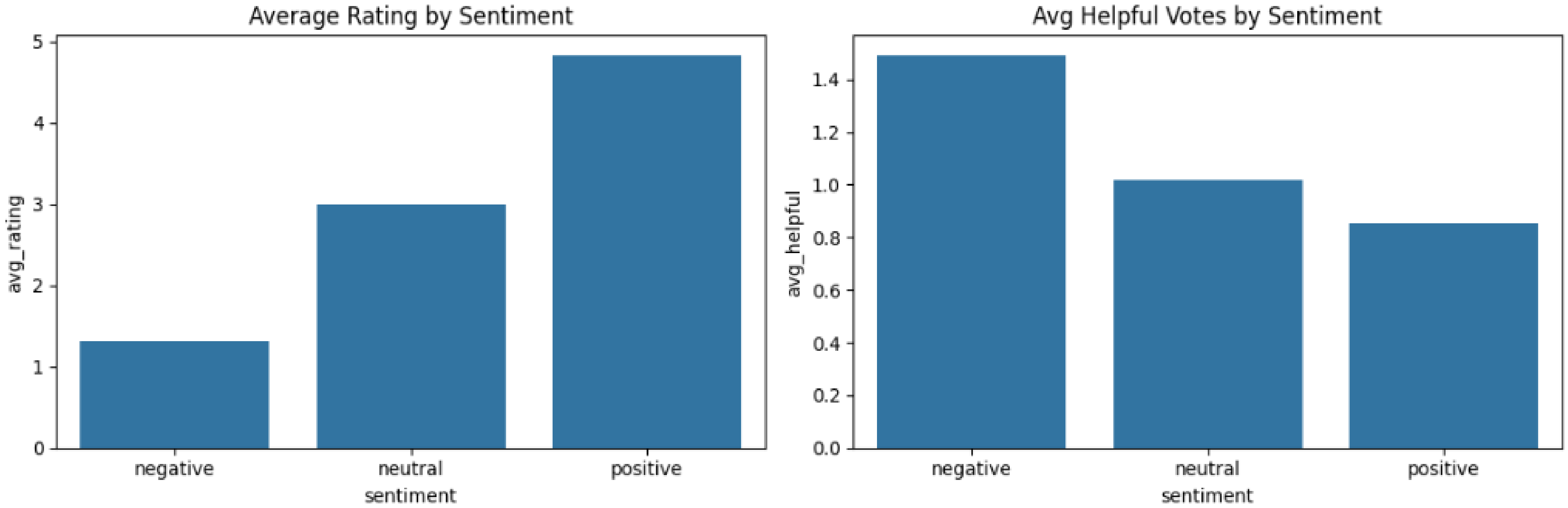


Insight:

Most short reviews are positive This shows that when customers are happy, they often leave quick and simple feedback



insights :*Negative reviews, particularly those with a 1-star rating, tend to be considered the most helpful.
***the helpfulness of positive reviews appears to decrease as the rating increases.**
***Neutral reviews receive a consistent, moderate level of helpful votes across the mid-range ratings.**



Insight: first graph: here is a strong positive correlation between review sentiment and star ratings.
second graph :The correlation between review sentiment and helpful votes shows ainverse relationship

Overview of Model Experimentation

1. **Problem Definition**:
The goal is to classify customer reviews into three sentiment categories: Positive, Neutral, and Negative.

2. **Model Selection & Experimentation**
We experimented with different machine learning models:

1-Logistic Regression (Baseline Model):
Chosen for its simplicity and interpretability. Achieved strong accuracy and high performance on the positive class.

2-Random Forest:
Applied to explore performance improvements. Showed high accuracy overall, but struggled to predict neutral sentiment effectively.

3-XGBoost:
A gradient boosting technique. Delivered the highest accuracy, particularly for the positive class, but still weak on the neutral class.

1-Model Comparison - Final Model vs. Baseline

Model	Accuracy	Macro F1	Negative F1	Neutral F1	Positive F
Logistic Regression	0.76	0.76	0.73	0.71	**0.85**
Random Forest	0.75	0.75	**0.74**	0.71	0.80
XGBoost	0.67	0.68	0.66	0.62	0.76
Naive Bayes	0.71	0.71	0.71	0.63	0.78
SVM	**0.76**	**0.77**	0.73	**0.71**	**0.85**

2. Key insights:

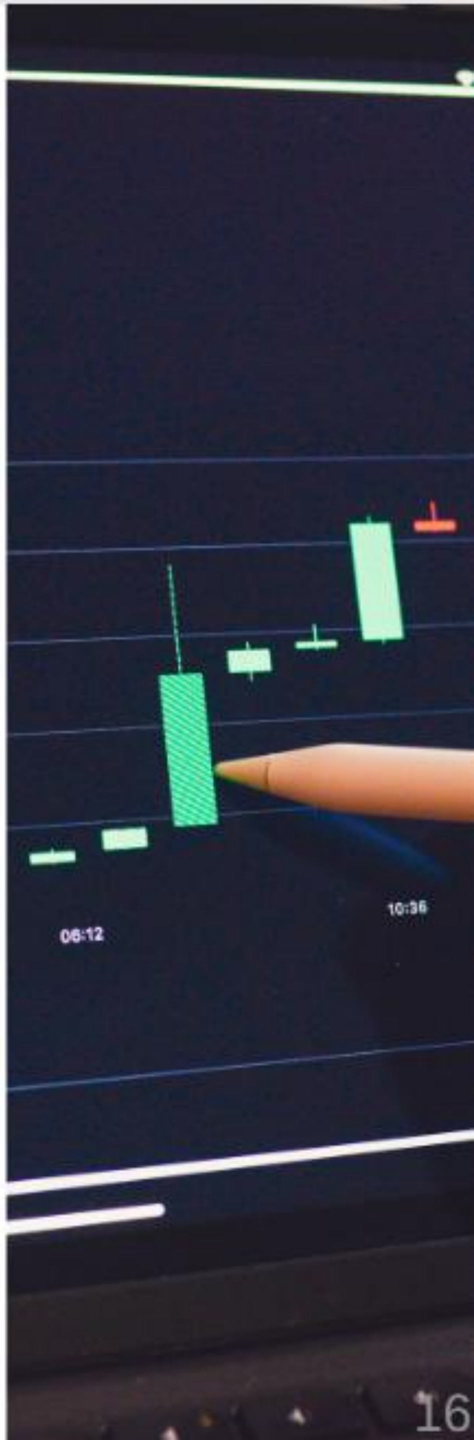
Logistic Regression, used as the **baseline model**, achieved **high accuracy (76%)** and strong, balanced performance across all sentiment classes.

SVM slightly **outperformed it in macro F1**, especially on positive and neutral sentiments.

Random Forest performed **best on negative sentiment,** while Naive Bayes showed moderate results.

XGBoost is **underperformed**, particularly on neutral sentiment

- **Neutral sentiment remains the most challenging across models.**



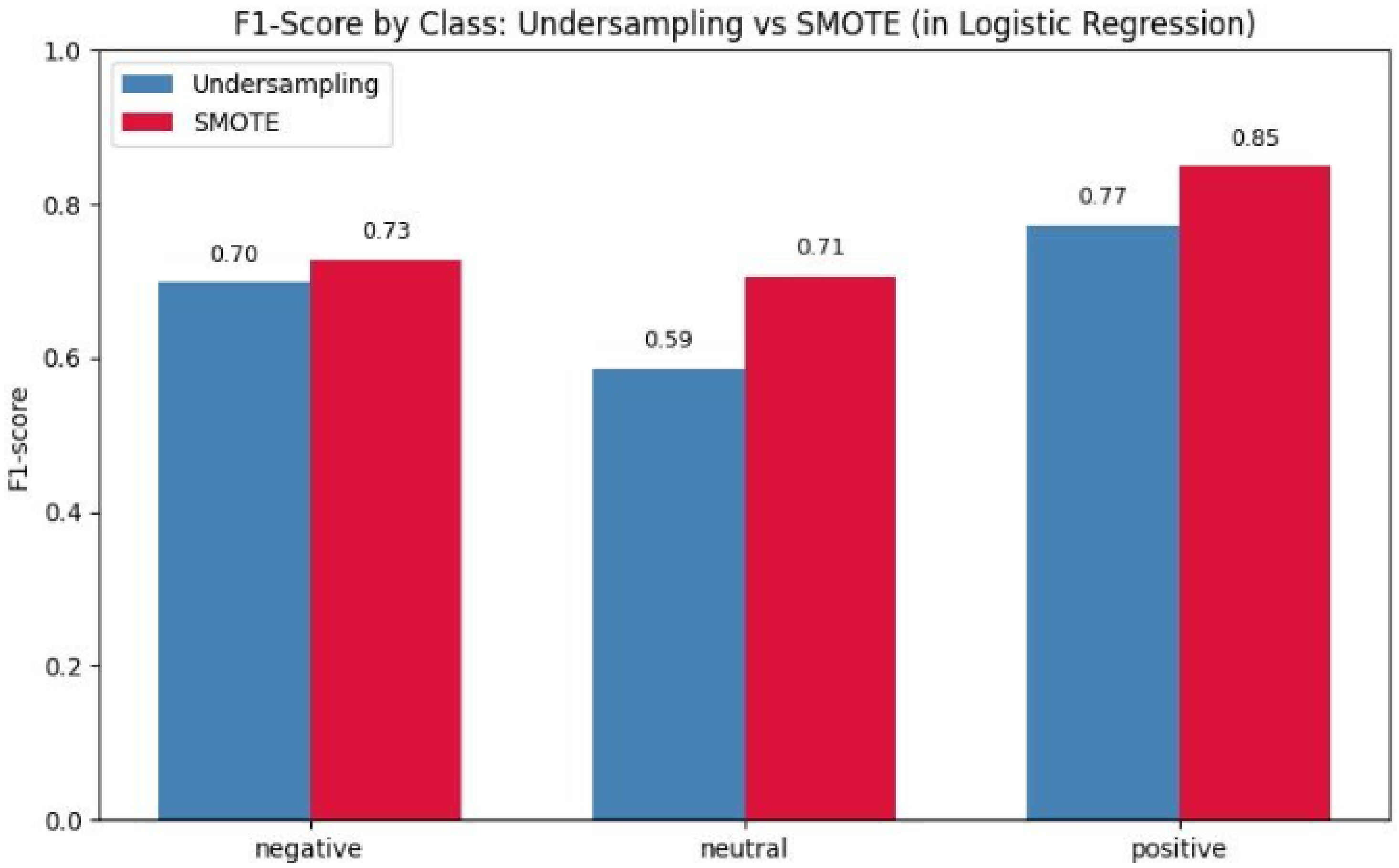
SMOTE vs undersampling

SMOTE improves F1-scores across all sentiment classes

Neutral class shows the largest gain: 0.59 → 0.71

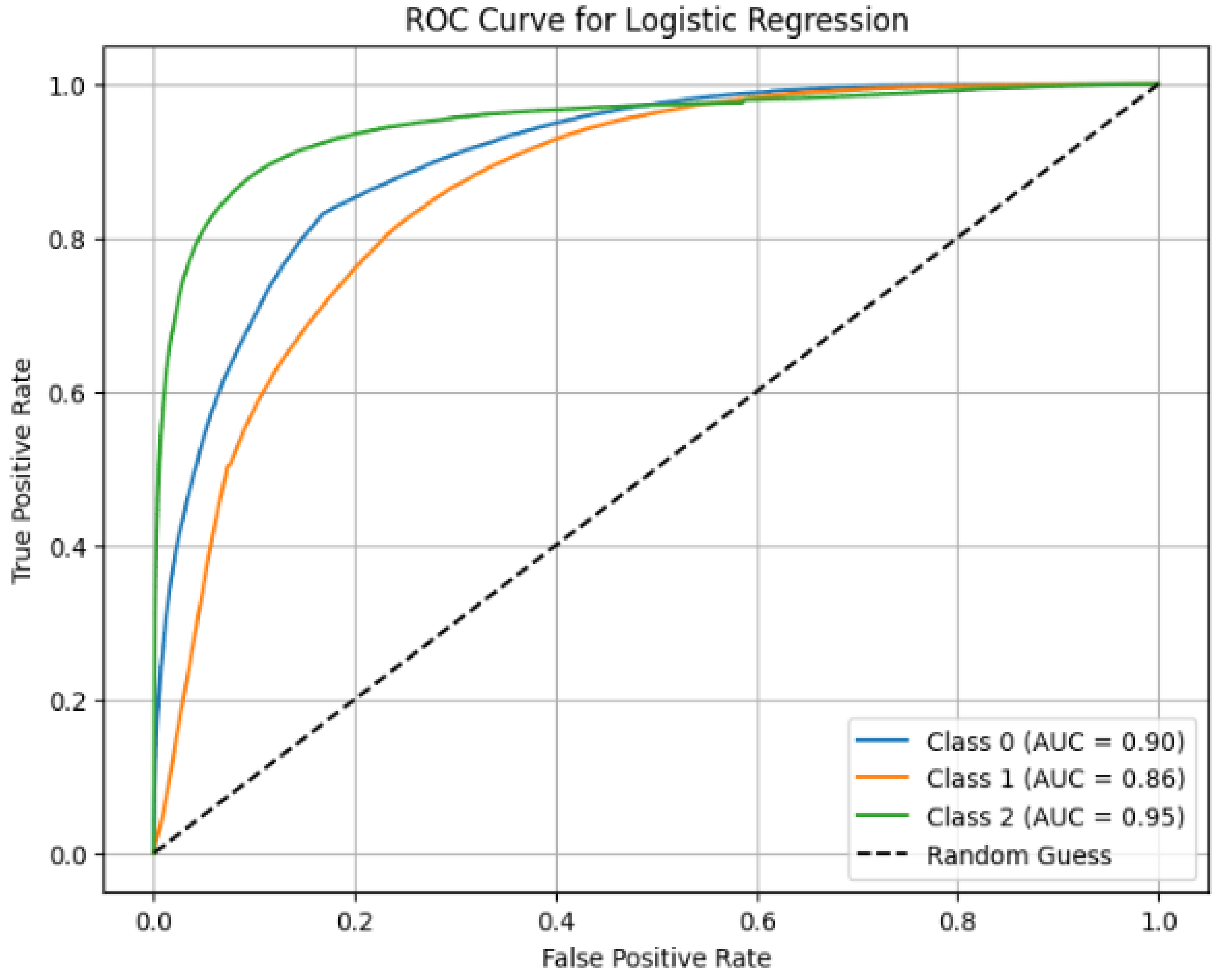
Positive class also benefits: 0.77 → 0.85

-Confirms undersampling removed too much valuable data
-SMOTE provided better class balance without data loss



ROC Curve Evaluation for Sentiment Analysis Model for logistic regression

- Class 0 (Negative): AUC = 0.90 → Strong detection of negative sentiment.
- Class 1 (Neutral): AUC = 0.86 → Moderate performance; neutral texts are harder to distinguish.
- Class 2 (Positive): AUC = 0.95 → Near-perfect identification of positive sentiment.
- All classes significantly outperform random guessing (AUC = 0.5), validating model reliability."



the Final Model and Business Relevance

1. Final Model: Logistic Regression:

- **Balanced Performance**: Logistic Regression performed consistently across all sentiment classes.
- **Best for Neutral Class:** Logistic Regression had the best performance for neutral sentiment, which is typically harder to classify.
- **Robust and Interpretable**: Logistic Regression is a simple, interpretable model that provides clear insight into customer sentiment, making it suitable for business applications.

2-Business Relevance:

Practical Application: The model's balance ensures that all sentiment classes are accurately classified, which is essential for understanding customer feedback in a real-world setting.

Scalability: Logistic Regression is computationally efficient and can handle larger datasets as the business grows.

Decision-Making: By accurately classifying sentiment, businesses can tailor their strategies, improve customer service, and enhance product offerings based on customer feedback.

Thank You!