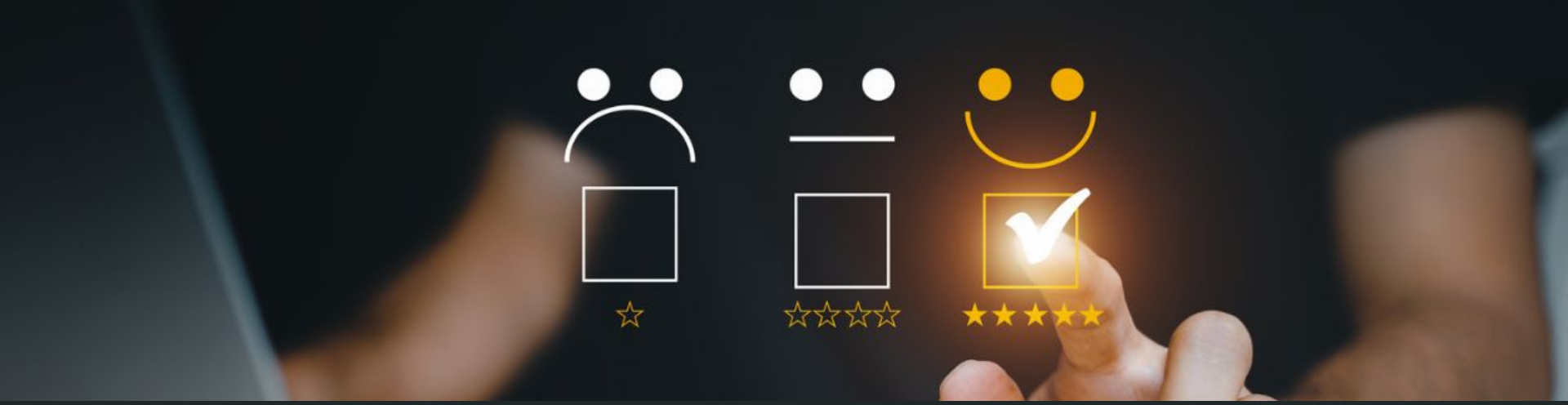


Sentiment Analysis for Amazon Product Reviews

Capstone Three - Final Presentation

Springboard Data Science Career Track

Tuba Cakir Kranda

08/06/2024

Introduction & Problem Statement

Problem Explanation:

- Customer reviews on e-commerce platforms like Amazon are crucial for understanding customer satisfaction and product performance.
- The goal is to analyze these reviews to classify the sentiment as positive, negative, or neutral, providing valuable insights for businesses.

Context: Sentiment analysis helps companies to gauge customer opinions, improve products, and enhance customer service.

Success Metrics: The success of the sentiment analysis model is measured by its accuracy, precision, recall, and F1 score.

Stakeholders: E-commerce companies, product managers, marketing teams, and customer service departments.

Constraints: Handling a large volume of unstructured data, ensuring accurate classification across diverse product categories.

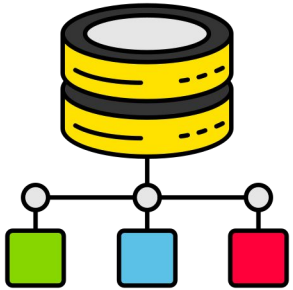
Project Formulation & Dataset

Data Science Problem:

- Formulate the task of sentiment analysis as a classification problem where each review is labeled as positive, negative, or neutral.

Dataset Description:

- The dataset consists of Amazon product reviews, including fields like review text, star rating, and other metadata.
- Data Source Link: [Amazon Reviews'23]
- Collected by: University of California San Diego (UCSD) McAuley Lab
- Data Source Origin: Amazon website (1996 - 2023)
- Language: English

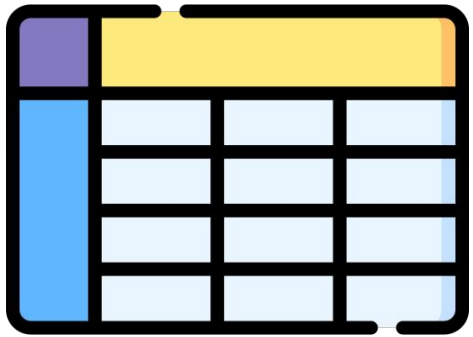


Dataset Comparing to Previous Versions

Year	#Review	#User	#Item	#R_Token	#M_Token	#Domain	Timespan
2013	34.69M	6.64M	2.44M	5.91B	–	28	Jun'96 - Mar'13
2014	82.83M	21.13M	9.86M	9.16B	4.14B	24	May'96 - Jul'14
2018	233.10M	43.53M	15.17M	15.73B	7.99B	29	May'96 - Oct'18
2023	**571.54M**	**54.51M**	**48.19M**	**30.14B**	**30.78B**	**33**	**May'96 - Sep'23**

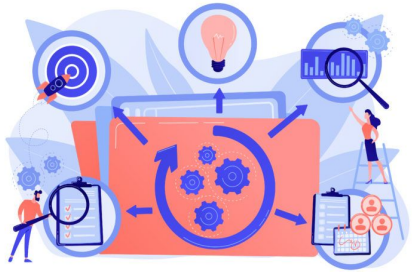
Data Model

- **rating**: Float value from 1.0 to 5.0 representing the product rating
- **title**: String containing the title of the user review
- **text**: String containing the full text body of the user review
- **images**: List of images posted by users after receiving the product, with different size options (small, medium, large)
- **asin**: String ID of the specific product
- **parent_asin**: String parent ID of the product (products with different colors, styles, sizes often share a parent ID)
- **user_id**: String ID of the reviewer
- **timestamp**: Integer representing the review time in unix format
- **verified_purchase**: Boolean indicating if the user's purchase was verified
- **helpful_vote**: Integer count of helpful votes for the review



Methodology

- **Data Sampling and Formatting**
  - Ensure balanced, clean, smaller sized (1M) dataset for models
- **Data Wrangling**
  - Ensure data cleanliness and completeness.
- **Exploratory Data Analysis (EDA)**
  - Explore distributions, correlations, and trends.
- **Pre-processing and Training Data Preparation**
  - Encode variables and scale features.
- **Modeling and Benchmarking**
  - Train and evaluate both statistical ML models and LLMs.
- **Key Findings and Recommendations**
  - Interpret model performance and feature importance.



Data Sampling and Formatting

- **Tool**: PySpark for parallel processing
- **Features Generated**: 'product_category', 'has_image'
- **Feature Dropped**: 'images' (image URLs)
- **Data Sampling**:
  - Balanced samples across ratings and product categories
  - Target: ~5882 samples per rating per category
- **Data Cleaning**:
  - Dropped incorrectly formatted records
  - Removed records with unexpected data types
- **Output**:
  - Cleaned, merged dataset saved as Parquet file
  - Total: ~1 million reviews

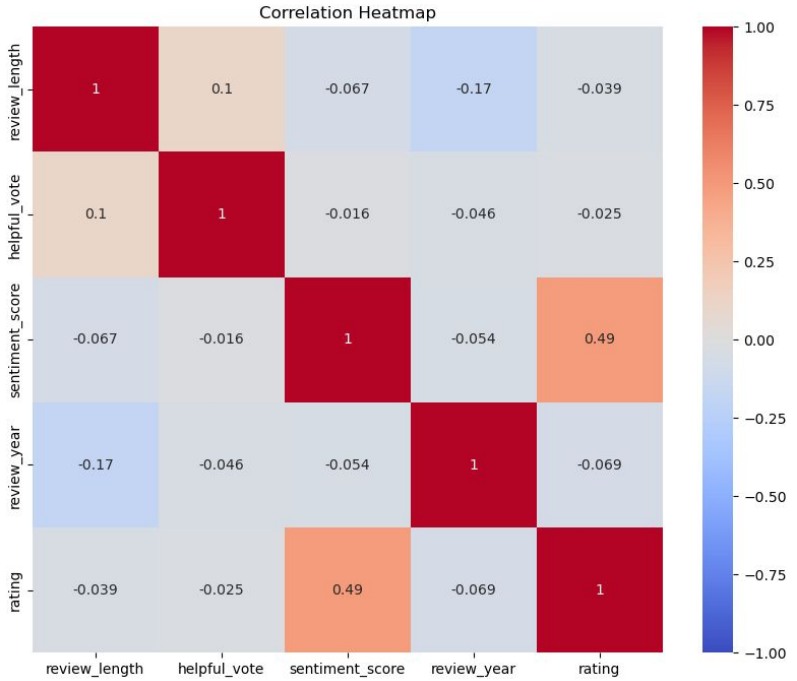
Data Wrangling

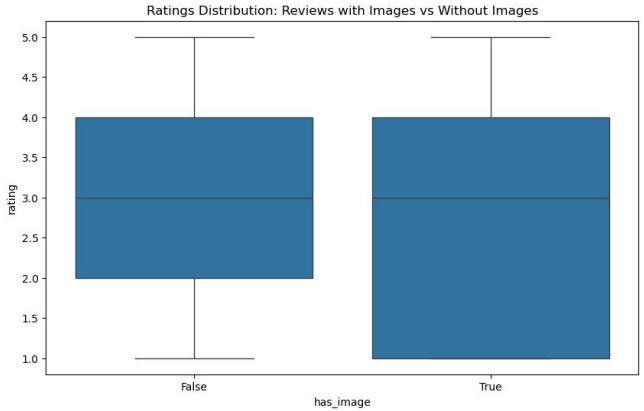
- **Discovery**: Explored and understood the collected raw data. Identified data sources, assessed data quality, and gained insights into the structure and format of the data. Balanced ratings across product categories resulted in a mean rating of 3.01 with a standard deviation of 1.42.
- **Structuring**: Organized and formatted the raw data to facilitate efficient analysis by handling missing values and converting data types. For example, the timestamp column originally in UNIX Time format was converted into a human-readable datetime format for more efficient exploratory data analysis.
- **Cleaning**: Addressed inconsistencies, errors, and outliers within the dataset by removing inaccurate data, addressing anomalies, standardizing inputs, and removing irrelevant data. Specifically, removed 1035 records with empty review fields, resulting in a final dataset size of 951,769 reviews.
- **Enriching**: Enhanced the dataset with additional information to provide more context or depth. For example, added ‘review_year’ and ‘review_month’ columns from the timestamp for more efficient EDA. Also used the sentiment analysis function of the TextBlob library to enrich the data with a new ‘sentiment_score’ column.

Key Takeaways: The cleaned and preprocessed dataset is ready for analysis, ensuring consistency and reliability for further steps.

Exploratory Data Analysis (EDA) - Part 1

- **Correlation Heatmap:** Shows correlations between various features to identify significant relationships.
- **Ratings Distribution:** Reviews with Images vs. Without Images: Boxplot illustrating how ratings differ between reviews with images and those without.
- **Descriptive Statistics:** Summary statistics such as mean, median, standard deviation, and quartiles.
- **Feature Importance:** High positive correlation (0.49) between 'sentiment_score' and ratings.





Exploratory Data Analysis (EDA) - Part II

Word Clouds

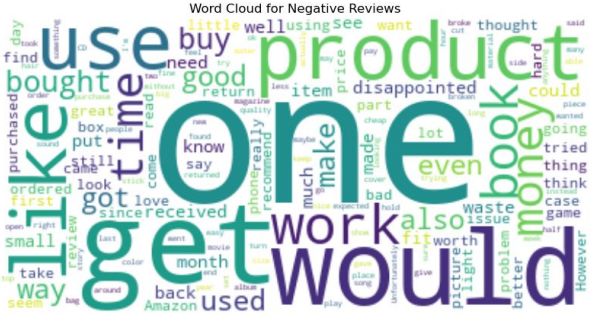
- Visualizing the most common words in positive and negative reviews.

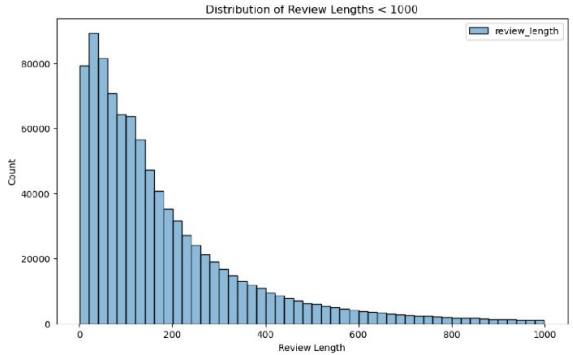
Review Length Distribution

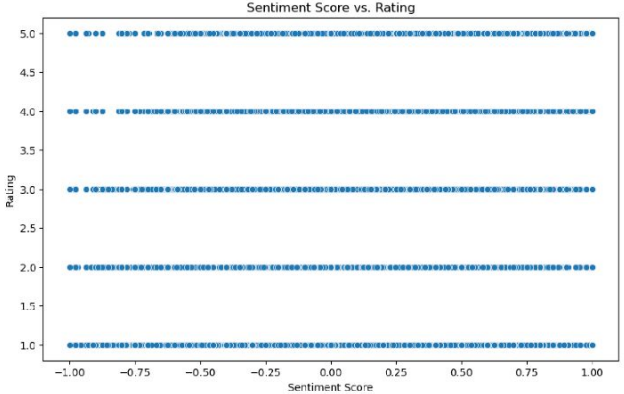
- Review length distribution shows most reviews are short, with a weak correlation to rating.

Sentiment score vs Rating

- limitations of basic sentiment analysis techniques







Data Pre-processing

- One-hot encoding for categorical variables
- Reviews are tokenized and vectorized
  - TfidfVectorizer, numeric_transformer and categorical_transformer
  - Hugging Face tokenizers (bert-base-uncased and distilbert-base-uncased)
- Feature scaling with StandardScaler for numeric variables
- 80% training, 20% testing split
- Target column (labels) generated from rating column starts

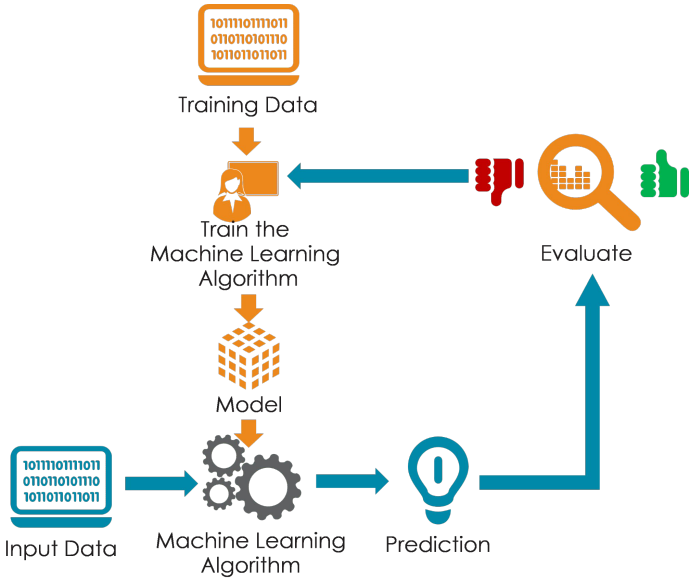
Rating Star	Sentiment Mapping	Encoded Numeric Value
1 or 2	Negative	0
3	Neutral	1
4 or 5	Positive	2

Modeling and Benchmarking

- **Models**
  - Logistic Regression Modeling with 761K Reviews,
  - Random Forest Modeling with 761K Reviews,
  - XGBoost Modeling with 200K Reviews,
  - Pre-trained BERT Finetuned with 200K Reviews LLM Model,
  - Pre-trained DistilBERT Finetuned with 400K Reviews LLM Model,
  - Pre-trained DistilBERT Finetuned with 761K Reviews LLM Model

- **Hyperparameter tuning**
  - GridSearchCV for statistical models
  - Different batch sizes for LLM Finetuned models

- **Evaluation Metrics**
  - Accuracy in average
  - Precision, Recall and F1-score in average
  - Precision, Recall and F1-score per class type
  - Each evaluated with 190-200K unseen evaluation datasets



Best Performing Model in Accuracy and Efficiency

- Pre-trained DistilBERT Finetuned with 761K Reviews LLM Model
- Accuracy: 82%
- Recall: 0.83 (Negative), 0.51 (Neutral), 0.91 (Positive)
- F1-Score: 0.86 (Negative), 0.55 (Neutral) and 0.91 (Positive)

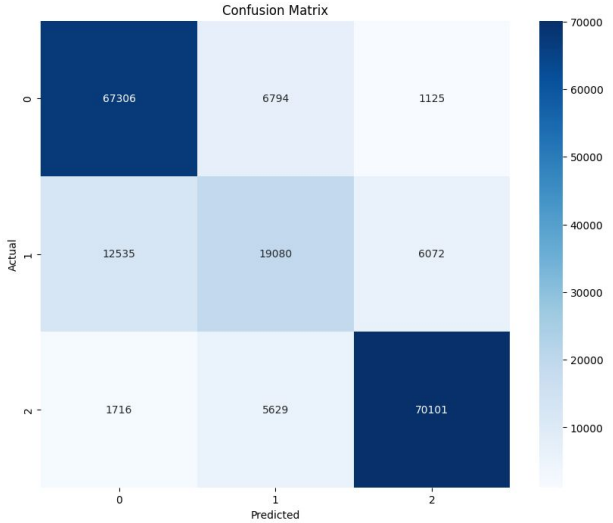
		Negative			Neutral			Positive		
Model	**Acc**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**
Logistic Reg. 761K	0.69	0.69	0.82	0.75	0.45	0.17	0.25	0.74	0.83	0.78
Random For. 761K	0.68	0.66	0.84	0.74	0.49	0.08	0.13	0.72	0.82	0.77
XGBoost Model 200K	0.70	0.70	0.83	0.76	0.48	0.16	0.29	0.75	0.82	0.79
BERT Finetun. 200K	0.80	0.82	0.86	0.84	0.55	0.49	0.52	0.88	0.89	0.89
DistilBERT Finetun. 400K	0.78	0.78	0.88	0.83	0.52	0.39	0.45	0.88	0.87	0.87
DistilBERT Finetun. 761K	**0.82**	**0.83**	**0.89**	**0.86**	**0.61**	**0.51**	**0.55**	**0.91**	**0.91**	**0.91**

Model Requirements and Best Model Arguments

Model	Hardware	Training Time	Model Size
Logistic Reg. 761K	Intel i7 16-Core CPU	~4 Minutes	237KB
Random For. 761K	Intel i7 16-Core CPU	~1.0 Hour	8.9GB
XGBoost Model 200K	Intel i7 16-Core CPU	~2.4 Hours	1752KB
BERT Finetun. 200K	Nvidia L4 24GB GPU	~9.5 Hours	418MB
DistilBERT Finetun. 400K	Nvidia A100 48GB GPU	~4.2 Hours	256MB
DistilBERT Finetun. 761K	Nvidia A100 48GB GPU	~7.0 Hours	256MB

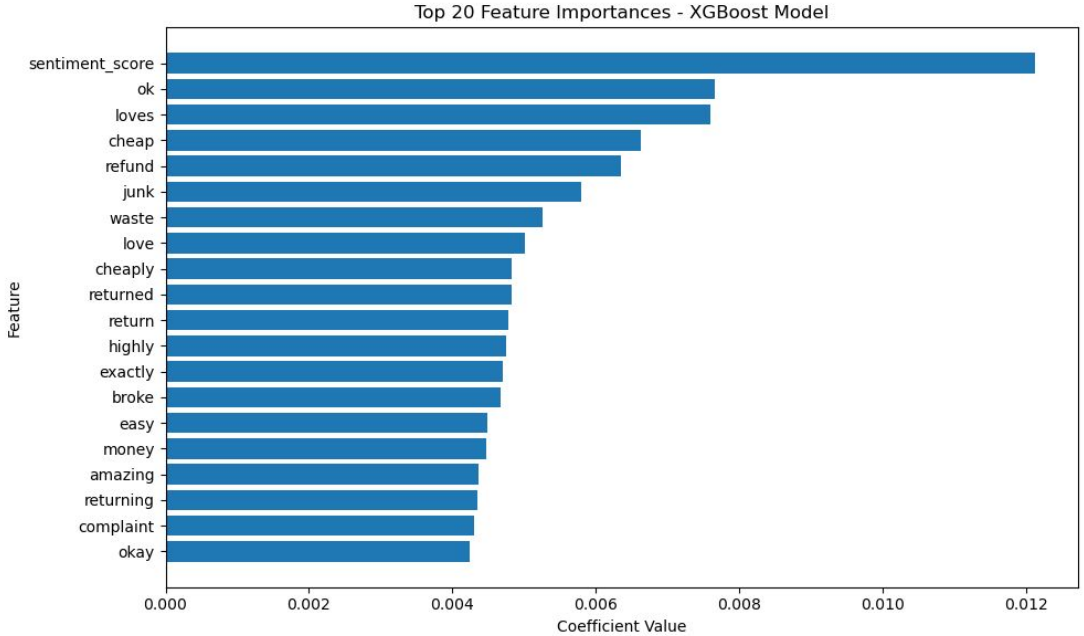
Distibert Model Training Arguments

- output_dir='./results',
- num_train_epochs=3,
- per_device_train_batch_size=100,
- per_device_eval_batch_size=100,
- warmup_steps=500,
- weight_decay=0.01,
- logging_dir='./logs',
- logging_steps=10,
- evaluation_strategy="epoch",
- save_strategy="epoch",
- load_best_model_at_end=True



Feature Importance for Best Statistical ML Model

- The Top-20 features accommodated by the best statistical ML model, XGBoost



Modeling Analysis & Key Findings

Model Performance:

- The DistilBERT model finetuned with 761K reviews achieved the highest accuracy (82%) and was the most effective across recall and F1-score metrics.
- Statistical models like Logistic Regression, Random Forest, and XGBoost performed well but fell short compared to the LLM models in accuracy and F1-scores.

Evaluation Metrics:

- For the DistilBERT 761K model:
  - Recall: 0.83 (Negative), 0.51 (Neutral), 0.91 (Positive)
  - F1-Score: 0.86 (Negative), 0.55 (Neutral), 0.91 (Positive)
- The BERT 200K model showed high precision and recall but had lower F1-scores for Neutral reviews.

Feature Importance:

- XGBoost highlighted key features influencing sentiment classification. These features were critical in identifying patterns and improving model accuracy.

Resource Requirements:

- DistilBERT models required significantly less time and hardware compared to BERT, making them more efficient for large-scale datasets.

Training Time and Hardware:

- Logistic Regression and Random Forest were quicker to train but less effective compared to LLMs.
- DistilBERT models, while requiring more advanced hardware (Nvidia A100 GPU), offered better performance, justifying the additional resources.

Hyperparameter Tuning:

- GridSearchCV was used for statistical models to optimize performance.
- For LLMs, different batch sizes were tested to balance training efficiency and model accuracy.

Tools and Technologies

- PySpark: Efficient data processing and analysis.
- Google Colab: Collaborative coding environment.
- Hugging Face: Advanced NLP models for sentiment analysis.

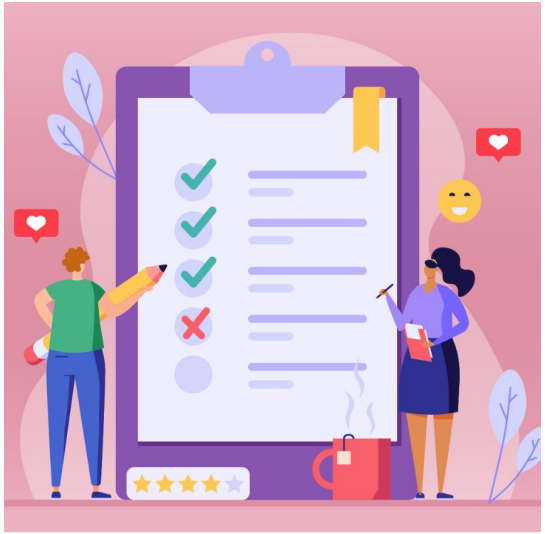
Recommendations

For Model:

- **Adopt DistilBERT:** Leverage DistilBERT for its effective balance between accuracy and computational efficiency. While BERT offers superior performance, DistilBERT meets project needs while conserving resources.
- **Transition to Advanced NLP Techniques:** Implement BERT to enhance sentiment classification, benefiting from its superior contextual understanding compared to traditional models.
- **Scale Data for Better Performance:** Expand the dataset to improve model generalization and better capture the nuances of sentiment analysis.

For Clients:

- **Customer Service:** Use sentiment analysis to prioritize responses to negative reviews.
- **Product Development:** Identify common issues in negative reviews for product improvements.
- **Marketing:** Tailor marketing strategies based on sentiment trends.



Practical Considerations & Future Work

Practical Considerations:

- **Resource Management:** Advanced models like BERT require more computational resources than DistilBERT. Ensure hardware can handle these needs.
- **Neutral Sentiments:** Use models with better contextual understanding to improve classification of neutral sentiments.
- **Feature Learning:** Rely on advanced models like BERT that learn features automatically, reducing manual engineering.

Future Work:

- **Explore More Models:** Benchmark models such as T5, GPT, and XLNet to enhance accuracy.
- **Expand Data Sources:** Gather reviews from multiple platforms beyond Amazon for a richer dataset.
- **Improve Neutral Sentiment Classification:** Develop methods to better classify neutral sentiments using advanced NLP techniques.
- **Investigate New Features:** Look into new features, such as subscription information, to boost model performance.

Conclusion

Summary: This project demonstrates the significant advantages of advanced NLP models like DistilBERT over traditional statistical models for sentiment analysis.

Key findings include:

- **Model Performance:** Advanced models outperform statistical models, particularly in classifying neutral and positive sentiments.
- **Resource Efficiency:** DistilBERT strikes a good balance between performance and computational requirements, making it a practical choice for this project.
- **Challenges:** Statistical models struggle with neutral sentiment classification, highlighting the need for more advanced techniques.

Final Thoughts:

The insights gained from this project underscore the importance of adopting sophisticated NLP methods for accurate sentiment analysis. Moving forward, leveraging models like BERT and expanding the dataset will enhance performance further. This approach not only improves classification accuracy but also provides a more nuanced understanding of customer sentiments, offering valuable information for product and service improvements.

Dataset: Amazon Reviews'23

GitHub Repository: Link