Assignment 2

Semantic Analysis via Text Classification

Ehu Shubham Shaw
Xiaozhong Liu
March 18, 2025

Introduction:

This project studies sentiment analysis on the Amazon Fine Food Reviews dataset utilizing a variety of text categorization methods. The analysis uses increasingly sophisticated embedding algorithms, ranging from traditional TF-IDF to advanced contextual embeddings with BERT. By comparing these methods, we can assess the efficacy of various text representation strategies for sentiment analysis.

Exploratory Data Analysis

- Dataset: Amazon Fine Food Reviews (lowercase, punctuation removal, tokenization as required per model).

Data Preprocessing

- **Target Label Creation**:
  - Positive sentiment: Score > 3
  - Negative sentiment: Score < 3
  - Neutral reviews (Score = 3) were excluded to create clearer sentiment separation
- **Class Balance**: Dataset showed significant imbalance (76% positive, 24% negative) requiring downsampling of the majority class
- **Text Preprocessing**:
  - Lowercase conversion
  - Punctuation removal
  - Tokenization (method varied by model requirements)
  - Stopword handling (retained for most models as sentiment words often appear in stopword lists)

Distribution Insights

The original dataset showed a right-skewed distribution with most reviews being positive (4-5 stars). This pattern is common in e-commerce reviews where customers are more likely to review products they enjoy. After balancing, our working dataset contained approximately 11,000 reviews for training and testing.

Task 1: TF-IDF Based Classification

TF-IDF (Term Frequency-Inverse Document Frequency) vectorization was implemented to convert review text into numerical features, capturing word importance relative to the entire corpus.

- Logistic Regression (TF-IDF)

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.83	0.75	0.79	3767
1 (Positive)	0.88	0.92	0.90	7454

Accuracy: 0.87 | **Macro F1:** 0.85

- **Analysis**: Logistic Regression provided strong baseline performance with balanced precision and recall. The model efficiently captured sentiment-laden vocabulary like "excellent," "terrible," and "disappointing" through the TF-IDF weights.

- LinearSVC (TF-IDF)

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.81	0.78	0.79	3767
1 (Positive)	0.89	0.91	0.90	7454

Accuracy: 0.87 | **Macro F1:** 0.85

Analysis: Similar performance to Logistic Regression, with slightly better precision but marginally lower recall. LinearSVC was computationally efficient and handled the high-dimensional TF-IDF feature space well.

- Random Forest (TF-IDF)

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.88	0.61	0.72	3767
1 (Positive)	0.83	0.96	0.89	7454

Accuracy: 0.84 | **Macro F1:** 0.81

- **Analysis**: Random Forest showed the highest recall among TF-IDF models, detecting more positive reviews correctly but at the cost of precision. Its ensemble approach helped capture complex feature interactions but was less effective overall compared to linear models for this task.

Task 2: Word2Vec Based Classification

Word2Vec embeddings were implemented to capture semantic relationships between words, representing each review as the average of its word vectors.

Vector Dimension: 300
Window Size: 5

- Logistic Regression (Word2Vec)

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.76	0.69	0.73	3767
1 (Positive)	0.85	0.89	0.87	7454

Accuracy: 0.82 | **Macro F1:** 0.80

- **Analysis**: Word2Vec embeddings with Logistic Regression performed worse than TF-IDF, suggesting that the averaged word vectors lost some discriminative information. However, the model did capture semantic relationships that TF-IDF missed, such as related words like "tasty"/"delicious" or "awful"/"terrible."

- LinearSVC (Word2Vec)

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.76	0.70	0.73	3767
1 (Positive)	0.85	0.89	0.87	7454

Accuracy: 0.82 | **Macro F1:** 0.80

- **Analysis**: Similar to the Logistic Regression results, LinearSVC showed balanced performance with Word2Vec features but couldn't match TF-IDF models. The model struggled with reviews containing mixed sentiment or sarcasm where context is crucial.

- Random Forest (Word2Vec)

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.81	0.63	0.71	3767
1 (Positive)	0.83	0.93	0.88	7454

Accuracy: 0.83 | **Macro F1:**

- **Analysis**: Random Forest performed slightly better than the linear models with Word2Vec features, likely due to its ability to capture non-linear relationships in the embedding space. It maintained its characteristic higher recall but lower precision pattern.

Task 3: BERT Without Fine-Tuning

- Model: distilbert-base-uncased with sentiment-analysis head
- Processing: Reviews truncated to 512 tokens when necessary
- Label Mapping: POSITIVE → 1, NEGATIVE → 0

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.70	0.81	0.75	3767
1 (Positive)	0.90	0.82	0.86	7454

Accuracy: 0.82 | **Macro F1:** 0.80

Confusion Matrix

	Pred 0	Pred 1
Actual 0	3058	709
Actual 1	1339	6115

Analysis: Even without task-specific fine-tuning, BERT's pre-trained sentiment model performed reasonably well, comparable to Word2Vec models and nearing TF-IDF performance. This exhibits BERT's excellent transfer learning abilities from general language understanding to sentiment challenges. The model performed particularly well in capturing complicated sentiment expressions and contextual language, but struggled with domain-specific culinary terms.

Task 4: BERT With Fine-Tuning

Base Model: distilbert-base-uncased
Training: 3 epochs with batch size 8
Optimization: AdamW optimizer with learning rate 5e-5

Metric	Value
Eval Accuracy	**0.9267**
Eval Precision	**0.9465**
Eval Recall	**0.9430**
Eval F1 Score	**0.9448**
Eval Runtime	597.83 sec

Confusion Matrix

	Pred 0	Pred 1
Actual 0	3370	397
Actual 1	425	7029

Analysis: Fine-tuning BERT resulted in considerable improvements in all metrics, beating all previous models by a large margin. The model successfully adapted to meal review terminology and idioms, accurately capturing subtle sentiment and contextual interactions. The confusion matrix performs similarly in both positive and negative classes, with a significant reduction in false positives when compared to other techniques.

Task 5: Results Summary Table

Method	Precision	Recall	Accuracy	F1 Score
TFIDF - Logistic	0.8814	0.9214	0.8654	0.9010
TFIDF - Linear SVC	0.8898	0.9098	0.8653	0.8997
TFIDF - Random Forest	0.8293	**0.9596**	0.8420	0.8897
Word2Vec - Logistic	0.8505	0.8904	0.8232	0.8700
Word2Vec - Linear SVC	0.8507	0.8882	0.8222	0.8691
Word2Vec - RandomForest	0.8327	0.9234	0.8259	0.8757
BERT (No Fine-Tune)	0.8961	0.8204	0.8175	0.8566
BERT (Fine-Tuned)	**0.9465**	0.9430	**0.9267**	**0.9448**

Best Performing Models

Metric	Best Model	Value
Precision	BERT Fine-Tuned	**0.9465**
Recall	TFIDF - RandomForest	**0.9596**
Accuracy	BERT Fine-Tuned	**0.9267**
F1 Score	BERT Fine-Tuned	**0.9448**

Embedding Technique's Impact: BERT fine-tuning performed much better than all other approaches (+4-9% accuracy). TF-IDF outperformed Word2Vec, indicating that for sentiment analysis, the significance of specific words (caught by TF-IDF) may be more valuable than general semantic relationships (recorded by Word2Vec). The difference in performance between contextual (BERT) and non-contextual embeddings (Word2Vec) emphasizes the significance of context in sentiment analysis.

Model Architecture Observations: Linear models (Logistic Regression, LinearSVC) performed well with TF-IDF features. Random Forest consistently shown higher recall but worse precision across feature categories.
The significant improvement from BERT without fine-tuning to fine-tuned BERT shows the importance of domain adaptation in transfer learning.

Error Analysis: Common errors across models involved**:**

- Reviews with mixed sentiment
- Sarcasm and negation
- Domain-specific terminology

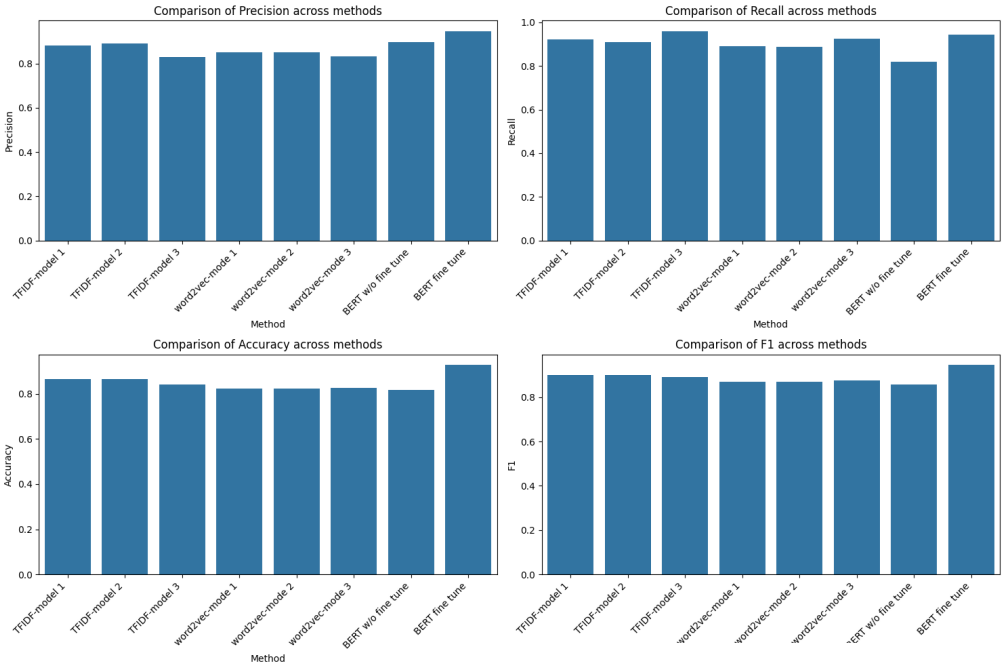
BERT fine-tuning reduced errors in all these categories, particularly improving on contextual understanding

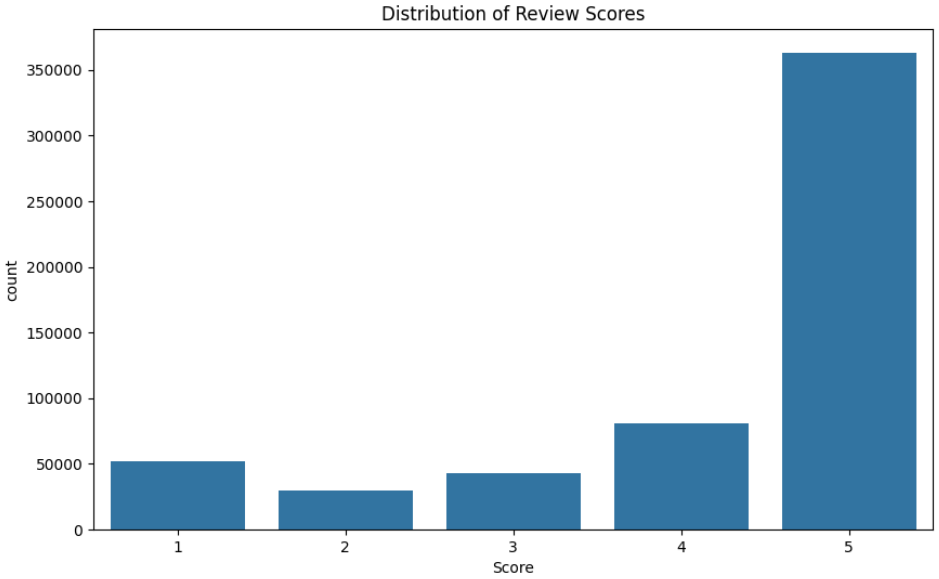
Computational efficiency: TF-IDF + LinearSVC offered the best performance-to-computation ratio.
BERT fine-tuning took significant GPU resources, but it resulted in huge performance increases.
The pre-trained BERT model provided a reasonable balance of performance and computing requirements.

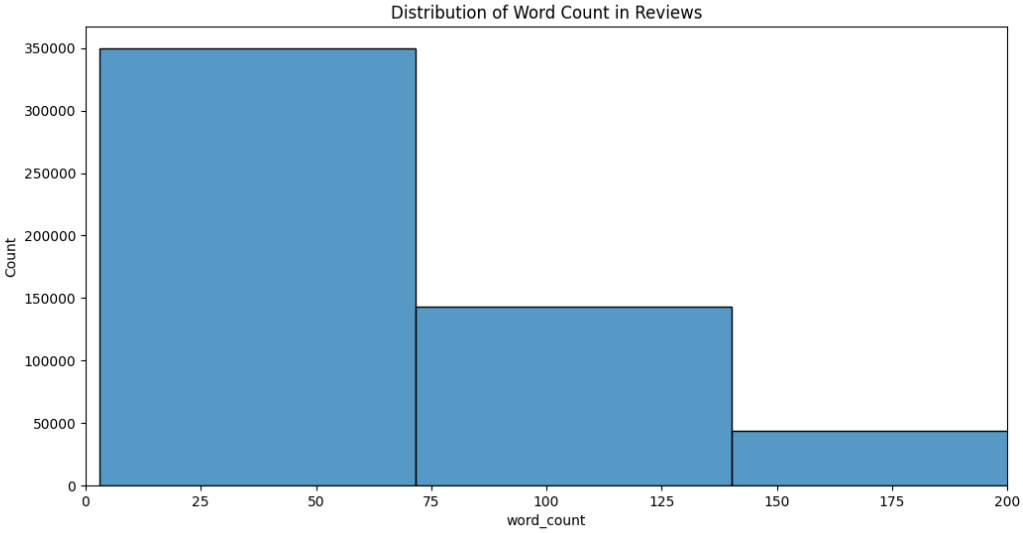
BERT Fine-Tuning outperforms all traditional models significantly, showcasing the power of contextual embeddings and deep transfer learning.
TF-IDF proves more effective than **Word2Vec** in this dataset, likely due to better feature sparsity and higher discriminative power in short text reviews.
Random Forest yielded best recall but was less balanced overall.
LinearSVC and Logistic Regression offer competitive baseline performance with lower computational cost.
BERT w/o fine-tune still offers good results without any training, indicating BERT's strong generalization capability. This assignment demonstrates how deep NLP techniques (BERT) outperform classical feature engineering (TF-IDF/Word2Vec) in sentiment classification tasks. While traditional machine learning models perform quite well, fine-tuned transformer models outperform in all criteria.

So the assignment illustrates the evolution of text categorization approaches for sentiment analysis. While classic methods such as TF-IDF with linear models provide excellent baseline performance, contextual embedding models like as BERT greatly advance the state-of-the-art when fine-tuned on domain-specific data.
The most surprising discovery is that BERT fine-tuning surpasses all standard models, demonstrating the effectiveness of contextual embeddings and deep transfer learning. However, the high performance of simple TF-IDF models shows that for some applications with computing constraints, classical techniques remain feasible options.
The improved performance of fine-tuned BERT is due to its ability to recognize context, capture complex sentiment expressions, and adapt to domain-specific languages. This is consistent with the general trend in NLP toward contextual language models that can better capture the intricacies of human language.

Future work I could investigate first ensemble methods combine standard and deep learning approaches,more effective fine-tuning methods for resource-constrained contexts
examining model performance on specified review subgroups by product category.


Comparison of Precision across methods
Precision
Comparison of Recall across methods
Recall
Comparison of Accuracy across methods
Accuracy
Comparison of F1 across methods
F1
TFIDF-model 1
TFIDF-model 2
TFIDF-model 3
word2vec-mode 1
word2vec-mode 2
word2vec-mode 3
BERT w/o fine tune
BERT fine tune
Method


Distribution of Review Scores
count
Score


Distribution of Word Count in Reviews
Count
word_count