Approach

1. Traditional Machine Learning

- **Label Mapping**: Ratings were grouped into three sentiment classes:
    - Negative: 1–3
    - Neutral: 4
    - Positive: 5
- **Preprocessing**:
    - Lowercasing, punctuation/special character removal, whitespace cleanup
    - Lemmatization
    - Feature extraction using TfidfVectorizer followed by dimensionality reduction via TruncatedSVD
- **Modeling**:
    - Dealt with class imbalance using SMOTE
    - Logistic Regression selected via grid search and cross-validation
    - Optimal hyperparameters: C=10, max_iter=1000

2. Transformer-Based Modeling (Hugging Face)

- **Preprocessing**: Same text cleaning steps as traditional ML
- **Tokenization**: Used AutoTokenizer.from_pretrained("distilbert-base-uncased")
- **Data Preparation**: Constructed DataLoader with input_ids, attention_mask, and labels
- **Model Selection**:
    - Compared BERT, RoBERTa, and DistilBERT
    - Chose **DistilBERT** for its balance of speed and performance (~97% of BERT's accuracy with 60% faster inference)
    - Dealt with the class imbalance by integrating weights in the loss function:

```python
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")

label_count = {i: 0 for i in range(3)}
for ii, am, labels in train_loader:
    batch_count = Counter(labels.tolist())
    for i in range(3):
        label_count[i] += batch_count[i]

total = sum(label_count.values())

weights = torch.tensor([
    total / label_count[i] for i in range(len(label_count))
], dtype=torch.float)

criterion = torch.nn.CrossEntropyLoss(weight = weights.to(device))
```

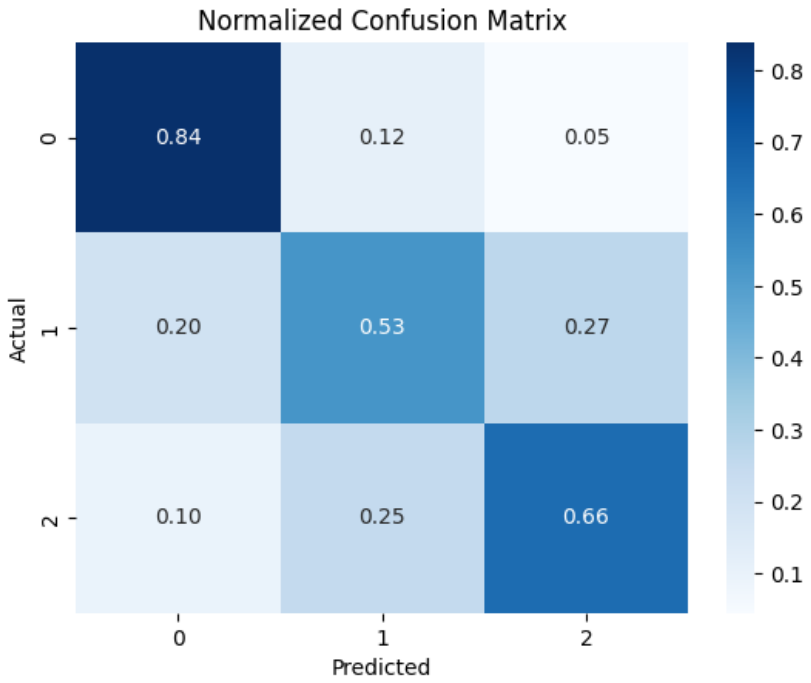
Results

Traditional ML

- **Model**: Logistic Regression
- **Evaluation**:
    - Classification report

	precision	recall	f1-score	support
0	0.7374	0.8382	0.7846	6782
1	0.5932	0.5310	0.5604	6783
2	0.6771	0.6554	0.6661	6782
accuracy			0.6749	20347
macro avg	0.6692	0.6749	0.6703	20347
weighted avg	0.6692	0.6749	0.6703	20347

    - confusion matrix



Transformer Model

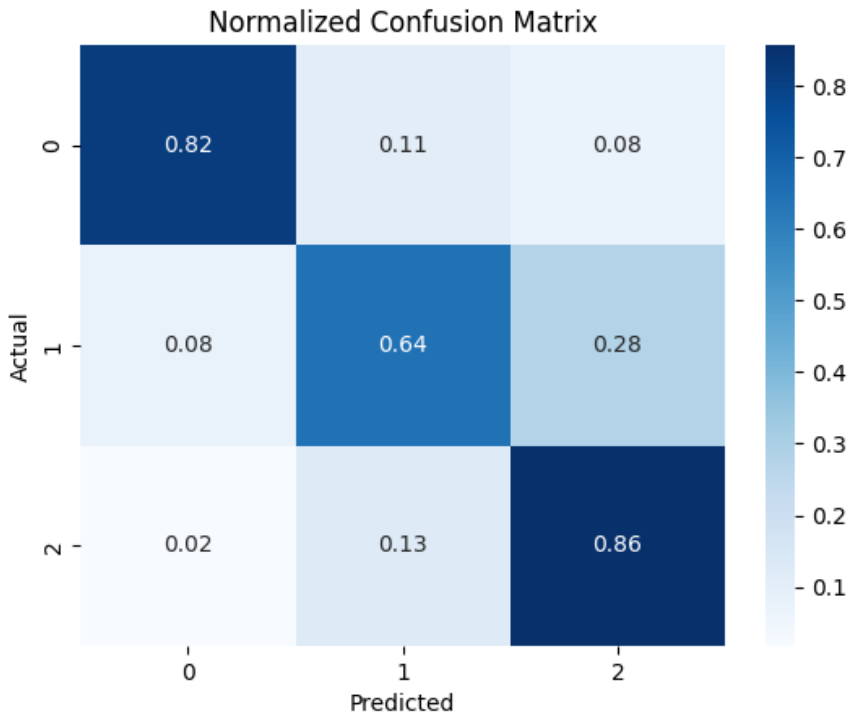
- **Model**: Fine-tuned DistilBERT
  - Optimizer: AdamW
  - Learning rate = 2e-5
  - num_training_steps = len(train_loader) * 3
- **Evaluation**:
  - classification report:

```
Test Set Classification Report:
              precision    recall  f1-score   support

           0     0.6847    0.8159    0.7445       793
           1     0.6071    0.6439    0.6249      2241
           2     0.8949    0.8568    0.8754      6774

    accuracy                         0.8049      9808
   macro avg     0.7289    0.7722    0.7483      9808
weighted avg     0.8121    0.8049    0.8076      9808
```

  - confusion matrix:



- **Deployment**: Model deployed on Hugging Face hub (https://huggingface.co/spaces/yang181614/customer-review-distilbert)

Analysis

- **Traditional ML**:
    - Efficient and interpretable
    - Performance may be limited by feature engineering and class imbalance
- **Transformer Approach**:
    - Superior contextual understanding and generalization
    - DistilBERT offered a practical trade-off between accuracy and speed
    - Hugging Face integration streamlined tokenization, training, and deployment
- Problems identifies:
    - Split train, validation and test data first, then apply SMOTE only on training data.
    - Logistic regression:
        - By default, the regularization is L2.
        - C is the inverse of the regularization strength, a higher C means less regularization, allowing the model to fit the training data more closely.