Financial Sentiment Analysis Report

1. Introduction

The primary goal of this project was to classify financial sentiments into negative, neutral, and positive classes using machine learning techniques. The dataset provided had a significant class imbalance, which affected the model's accuracy and generalization ability. Various methods were employed to improve the model's performance while addressing this imbalance.

2. Dataset Overview

Total Samples: 4306
Class Distribution:
- Neutral: 2879 (66.9%)
- Positive: 1363 (31.6%)
- Negative: 604 (14.0%)

Challenges:
- Imbalance Problem: The neutral class dominated the dataset, causing the model to predict this class more often and leading to poor recall for the minority classes (positive and negative).

3. Base Model

Architecture:
- Embedding Layer: Converts text into dense numerical vectors.
- Convolutional Layer: Captures local patterns in text sequences.
- LSTM Layer: Captures sequential dependencies and context in the text.
- Fully Connected Layer: Classifies into one of three sentiment classes (negative, neutral, positive).

Results from Base Model:
- Training Accuracy: High (80-90%)
- Validation Accuracy: Lower (60-65%)

Observations:
- High training accuracy indicated overfitting.
- Poor recall for minority classes (negative and positive).

4. Methods Used to Improve Accuracy

4.1. Data Preprocessing

- Text Cleaning: Removal of punctuation, numbers, and conversion to lowercase.
- Tokenization: Splitting text into words and mapping them to numerical indices.
- Padding: Ensured uniform input sequence length.

4.2. Class Imbalance Solutions

4.2.1. Class Weighting

Adjusted loss contributions based on class frequencies.
Neutral: 0.561
Positive: 1.185
Negative: 2.674
Result: Improved loss convergence but failed to improve minority class recall significantly.

4.2.2. Oversampling

Applied SMOTEENN (Synthetic Minority Oversampling + Edited Nearest Neighbors) to create a balanced dataset.
Result: Improved recall and F1-scores for minority classes.

4.2.3. Undersampling

Reduced neutral class samples to balance the dataset.
Trade-off: Risk of losing valuable information in the majority class.

4.3. Alternative Loss Functions

Focal Loss can help focus the training process on hard-to-classify minority samples.
Parameters: alpha=0.25, gamma=2.0
Result: Increased minority class recall while maintaining overall accuracy.

4.4. Model Regularization

- Batch Normalization: Added after convolutional and LSTM layers to stabilize learning. Improved generalization.
- Dropout: Increased dropout rates to reduce overfitting.

4.5. Hyperparameter Tuning

Tested different configurations:
- Embedding Size: 50, 100, 200
- LSTM Units: 64, 128, 256
- Convolutional Filters: 32, 64, 128

Result: The simplest model (embed_size=50, conv_filters=32, lstm_units=64) performed best, likely due to the small dataset size.

5. Results Comparison

Model	Embedding Size	Conv Filters	LSTM Units	Dropout	Accuracy	Precision (Macro)	Recall (Macro)	F1-Score (Macro)
Base	50	32	64	0.2	0.62	0.72	0.35	0.45
Weighted	50	32	64	0.2	0.65	0.75	0.42	0.50
Oversampled	50	32	64	0.3	0.68	0.78	0.54	0.60

Focal Loss	50	32	64	0.3	0.69	0.80	0.58	0.62

6. Technical Insights

- Imbalanced Data Impacts Recall: Models without addressing class imbalance heavily predict the majority (neutral) class.
- Regularization is Crucial: Overfitting was mitigated using dropout and batch normalization.
- Focal Loss is Effective: Improves minority class recall significantly by penalizing the easy-to-predict majority class.
- Simpler Models Work Better: Overly complex models tend to overfit small datasets.

7. Final Recommendations

- Use SMOTEENN for Balancing: Balances the dataset effectively without losing majority class information.
- Focal Loss: Should be employed for tasks with severe class imbalance.
- Simplify Architectures: Focus on lightweight models for small datasets.
- Evaluate Using F1-Score: Use F1-Score and recall as primary metrics for imbalanced classification problems.

8. Conclusion

This project demonstrated the challenges of financial sentiment classification with an imbalanced dataset. Through methods like oversampling, focal loss, and regularization, we improved model performance, particularly for minority classes. Future work could explore transformer-based models (e.g., BERT) for further improvements.