Analysis Report on Malicious URL Detection Models

Performance Analysis:

1. **XGBoost Model:**
   - XGBoost demonstrated the highest accuracy (96%) among the three models.
   - Achieved excellent precision, recall, and F1-scores for all classes, with a particularly strong performance in the benign and defacement categories.
   - Struggled slightly with class 3 (malware), likely due to overlapping features with other classes.
2. **LSTM Model:**
   - The LSTM model achieved moderate performance with an accuracy of 89%.
   - Exhibited lower recall and F1-scores compared to XGBoost, especially in the 3 (malware) and 4 (spam) classes.
   - Sequence learning helped capture patterns in URL structures but suffered from class imbalances, resulting in reduced performance for underrepresented classes.
3. **BERT (LLM-Based Model):**
   - BERT achieved an accuracy of 72.91%, significantly lower than both XGBoost and LSTM.
   - The pre-trained model was less effective in this domain, possibly due to limited fine-tuning on URL-specific data.
   - Computationally expensive and struggled to generalize across all classes, especially the minority ones.

Model Comparison:

Model	Accuracy	Macro F1-Score	Strengths	Weaknesses
XGBoost	96%	0.95	High accuracy, interpretable, handles imbalanced data well	Slightly lower recall for class 3
LSTM	89%	0.81	Captures sequential patterns effectively	Overfits and struggles with minority classes

Model	Accuracy	Macro F1-Score	Strengths	Weaknesses
BERT	72.91%	-	Leverages contextual embeddings	Requires significant fine-tuning, less effective for URLs

Challenges:

1. **Class Imbalance:**
   - Undersampling helped to balance the dataset but led to a loss of data diversity. The minority classes (2, 3, and 4) were particularly challenging for both LSTM and BERT.
   - XGBoost managed imbalance better due to its inherent handling of weighted losses.
2. **Resource Intensity:**
   - The BERT model required significant computational resources for fine-tuning and inference.
   - LSTM training was time-intensive due to the sequential nature of computations.
3. **Feature Relevance:**
   - URL structures have unique characteristics that traditional NLP models like BERT may not capture effectively without specific pretraining.

Proposed Improvements:

1. **Data Augmentation:**
   - For better generalization, consider synthetic data generation tailored to minority classes to increase representation without sacrificing diversity.
2. **Regularization for LSTM:**
   - Add dropout layers and tune hyperparameters to mitigate overfitting and improve generalization.
3. **Domain-Specific Embeddings:**
   - Fine-tune BERT on a URL-specific corpus to improve its understanding of patterns unique to URLs.
4. **Hybrid Approaches:**

- Combine the strengths of XGBoost and deep learning models (e.g., use XGBoost on features extracted by LSTM or BERT).

Conclusion:

XGBoost emerged as the best-performing model due to its high accuracy, robust handling of imbalanced data, and interpretability. LSTM showed moderate performance but struggled with minority classes and overfitting. BERT underperformed due to a lack of domain-specific fine-tuning and challenges in generalizing to URL data. Future efforts should focus on optimizing LSTM and BERT models for the URL domain while leveraging hybrid approaches to improve performance further.