COMP 550 Programming Assignment 2 Report

Analysis of Models:

- Most Frequent Sense Baseline achieves relatively high accuracy, making it a strong and simple baseline. This suggests that a significant portion of words in the test set adheres to their most common sense, reflecting the "one sense per discourse" or "one sense per collocation" phenomena.
- NLTK's Lesk Algorithm shows the lowest performance among all methods. This traditional approach may struggle due to its reliance solely on dictionary definitions and example sentences without considering broader contextual cues.
- Extended Lesk Algorithm improves upon the basic Lesk by incorporating more context such as hypernyms and hyponyms, which enhances its ability to disambiguate based on broader lexical semantics. The improvement in accuracy in the test set is indicative of its effectiveness over the standard Lesk algorithm.
- BERT-based WSD uses deep contextual embeddings which theoretically should capture nuanced meanings. However, its moderate performance might be due to challenges such as embedding noise or the complexity of mapping deep embeddings directly to word senses without fine-tuning.
- Yarowsky's Algorithm, while innovative, performed better on the dev set than on the test set, suggesting that its unsupervised learning approach might be sensitive to the initial conditions or require further refinement to generalize better across different sets of data. Feature extraction is applied around the ambiguous words, where we use them to compute sense scores during the labeling, and a bootstrapping loop is applied where we iteratively label unlabelled instances based on features occurrences, and then update them checking for convergence based on the ratio of new labels to total instances.

WSD Model Performance Results		
Method	**Dev Set Accuracy**	**Test Set Accuracy**
Most Frequent Sense Baseline	0.5619	0.5421
NLTK's Lesk Algorithm	0.2216	0.2862
Extended Lesk Algorithm	0.4948	0.5269
BERT-based WSD	0.3557	0.4131
Yarowsky's Algorithm	0.4433	0.3641

Model Training and Selection:

- **Training**: For BERT and potentially other models, training on the development set with hyperparameter tuning can optimize performance before testing. For unsupervised methods like Yarowsky, the choice of seeds and features is critical.
- **Model Selection**: Should be based on performance metrics on the development set. Additionally, qualitative analysis (reviewing specific instances where models fail) can provide insights into why certain methods struggle and how they might be improved.

Evaluation Concerns:

- **Data Sparsity**: Some senses might appear infrequently, leading to models performing well on frequent senses but poorly on rare ones. Stratified sampling or sense-balancing techniques during training could mitigate this.
- **Lexical Focus**: If evaluation centers on specific lexical items known to be challenging or ambiguous, targeted adjustments in model architecture or training data might be necessary.

Challenges and Improvements:

- **NLTK's Lesk & Extended Lesk**: Could benefit from enhanced feature engineering, such as including POS tags, dependency relations, and more sophisticated NLP techniques to enrich context understanding.
- **BERT-based WSD**: Might improve through fine-tuning on a task-specific corpus or by integrating sense embeddings directly during training, allowing the model to learn disambiguation implicitly.
- **Yarowsky's Algorithm**: Its performance is highly dependent on initial seeds and iterative labeling quality. Introducing a confidence measure for labeling and expanding the diversity of context features could enhance its robustness and accuracy.
- **Training:** For the BERT-based WSD, incorporating a phase of fine-tuning on a WSD-specific dataset could significantly enhance its capability to understand and disambiguate based on training examples.
- **Feature Engineering:** For Yarowsky's Algorithm, exploring different sets of linguistic features such as syntactic dependencies, more advanced collocation features, or even embedding-based features could improve its predictive power.
- **Hybrid Approaches:** Combining the strengths of rule-based methods (like Extended Lesk) with learning-based methods (like BERT or a custom machine learning classifier) might yield better results than using any single approach.

Conclusion:

The evaluation illustrates the strengths and weaknesses of each WSD approach. While the most frequent sense baseline provides a tough benchmark due to its simplicity and effectiveness, more complex models show potential but require careful tuning and possibly more sophisticated NLP techniques to fully realize their capabilities. Future work should focus on improving the integration of contextual understanding and the use of advanced model training techniques to enhance the accuracy and reliability of WSD systems.