Semantic Analysis Assignment

You may use any programming language or annotation tool(s) to complete this assignment. Please submit the following documents:

1. A PDF report answering the questions mentioned below.
2. Your code and README file in a compressed archive.

Part I: Word Sense Disambiguation (WSD)

This part focuses on identifying the correct sense of words in context using lexical resources and a disambiguation algorithm using computational techniques discussed in class. Dataset WSD to build and evaluate a simple WSD system.

1. **Preprocessing and POS Tagging (5 marks)** Perform sentence- and word-level tokenisation on the assigned portion of the dataset. Use any POS tagger to label each token.

2. **Listing Possible Senses (5 marks)** For each content word in the first 10 sentences in the training set (noun, verb, adjective, adverb), list all possible meanings (senses) from *WordNet*. Each sense should include its definition (gloss) and example usage. Store these senses in a structured format such as a dictionary or JSON file.

3. **Most Frequent Sense Baseline (5 marks)** For each training instance, extract the most frequent sense of the target word from wordnet. Compute Precision, Recall and F1 scores for this baseline.

4. **Simplified Lesk Algorithm (10 marks)** Implement the Simplified Lesk Algorithm **from scratch**. For each target word marked in the training set, select the sense whose wordnet gloss has the maximum word overlap with the context window (sentence or local neighborhood). Provide an illustrative example showing one word, its candidate glosses, and how overlap is computed. Compare precision, recall and F1 scores of the Lesk Algorithm with the most frequent sense baseline.

5. **Improved Lesk Algorithm (10 marks)** Extend your system to improve performance us two or more of the following enhancements to the data in the training set:

   - Lemmatization and stopword removal.

- POS-based filtering (only compare senses with matching POS).
- Including glosses of hypernyms/hyponyms in overlap computation.
- Expanding the context window to include neighboring sentences.

Compare your enhanced version's results with the basic Lesk algorithm and briefly analyze the improvements observed (if any). Also compare precision, recall and F1 scores of the Lesk Algorithm with the most frequent sense baseline.

6. **Evaluation and Analysis (5 marks)** Select ten ambiguous words from your dataset and manually evaluate the correctness of predicted senses against WordNet's intended sense. Discuss which linguistic or knowledge-based features were most helpful in resolving ambiguity, and note the key limitations of the Lesk algorithm for real-world disambiguation.

7. **Extra Credit: Bayes WSD system (15 marks)** Implement a Naive Bayes WSD system using features extracted from the training set and compare the performance of this system with the earlier Lesk Algorithms you had implemented on the test set.

Part II: Semantic Role Labeling (SRL)

This part focuses on identifying predicate–argument structures using PropBank and FrameNet frameworks. Use your assigned sentences from the SRL for annotation and analysis.

1. **Annotation Process (10 marks)** Identify the main verbs (predicates) in each sentence, annotate their arguments (e.g., *Agent*, *Theme*, *Goal*), and map these roles to the corresponding PropBank or FrameNet frames.

2. **Annotation Tools (5 marks)** Use any annotation tool such as **BRAT**, **ELAN**, or **WebAnno**. Mention the tool used and briefly describe the configuration process in your README file.

3. **Quality Assurance (5 marks)** Manually review all annotations for consistency. Resolve any ambiguous or unclear instances by consulting PropBank and FrameNet documentation.

4. **Analysis and Discussion (10 marks)**

   - Describe patterns observed in argument structures across English and the chosen Indian language.

- Discuss linguistic or structural challenges encountered and note interesting cross-lingual variations.

Submission format: `rollno_a6.tar`

References

- Jurafsky, D., & Martin, J. H. (2009). *Speech and Language Processing.* Pearson Education.

- Palmer, M., Gildea, D., & Kingsbury, P. (2005). *The Proposition Bank: An Annotated Corpus of Semantic Roles.* Computational Linguistics, 31(1), 71–106.

- Baker, C. F., Fillmore, C. J., & Lowe, J. B. (1998). *The Berkeley FrameNet Project.* In Proceedings of ACL.

- Fillmore, C. J., Johnson, C. R., & Petruck, M. R. L. (2003). *Background to FrameNet.* International Journal of Lexicography, 16(3), 235–250.