Assignment 1:

POS Tagging

Implement a POS tagger in Python using Hidden Markov Model

Input and output:

- Dataset: **Brown corpus (tagset = "universal")**
- Output: Accuracy (5-fold cross-validation), confusion matrix, per POS accuracy
- Create a document which reports the following for your implementation
    1. Draw confusion matrix.
    2. Report per POS accuracy (accuracy for each tag).
    3. Observe the strength and weaknesses of the model with respect to each POS tag.
    4. Perform detailed error analysis.
    5. Write a short paragraph on your learning.
- NOTE
    1. Use 5-fold cross-validation for reporting all accuracy values
    2. HMM should be implemented from scratch

Dataset:

- Brown corpus (Available in NLTK library) (http://www.nltk.org/nltk_data/)

Submission Instructions:

- The assignment is to be submitted in groups of 3 (Same group for every assignment and project)
- The submission link will be created on moodle to submit the assignment
- Only one person from the group with the lowest id is supposed to make the submission
- The name of the folder should be <id1_id2_id3>_Assignment1.zip
    - The uncompressed folder should contain one folder named HMM, readme and a report in pdf format <id1_id2_id3_Assignment1>.pdf)
    - The HMM folder should contain the code files.
    - The readme should contain details about the tools, versions, pre-requisites if any, and how to run the code for reproducing results.
    - The report should contain all things mentioned in the problem statement.

        - Accuracies, Per POS accuracies, confusion matrix, error analysis, strengths, and weaknesses of the model with respect to particular POS tags, and a short paragraph on your learning.

Deadline

- No-Hard deadline (Continuous Evaluation). First Evaluation date will be announced soon.

Important Note:
We shall check the submitted code for plagiarism. Please be aware that copying code from Git or from different teams is considered a serious violation.

References

- https://www.nltk.org/book/ch05.html

FAQs:

1) To count the number of times a tag/token occurs in the corpus, can we use the nltk library?

   It should be implemented from scratch.

2) To generate all bigrams, can we use nltk?

   It should be implemented from scratch.

3) How to deal with unseen words in POS Tagging using HMM?

   For handling words which are not in the training corpus, you can explore various smoothing techniques.

   Reference: Smoothing from Manning and Schutz, "Foundations of Statistical Natural Language Processing", Page 199 (general) and Page 354 (POS specific).

4) Can we preprocess the data?

   You are free to choose preprocessing steps like lower casing based on your intuition of what will work better. Specify clearly in the report about the preprocessing steps.