CS563 - NLP

ASSIGNMENT-I : Part-of-Speech (PoS) tagging using HMM and Recurrent Neural Network

(Read all the instructions carefully and adhere to them)

Date : 30-01-2021 **Deadline:** February 08, 2021

Instructions:

1. Markings will be based on the correctness and soundness of the outputs.
2. Marks will be deducted in case of plagiarism.
3. Proper indentation and appropriate comments (if necessary) are mandatory.
4. You should zip all the required files and name the zip file as: <roll_no>_assignment_<#>.zip , eg. 1501cs11_assignment_01.zip.
5. Upload your assignment (the zip file) in the following link: https://www.dropbox.com/request/w5PQuhRtWIMSFvFtFVVh

For any queries regarding this assignment contact:

Zishan Ahmad (zeeman.zishan@gmail.com) or **Deeksha varshney** (deeksha.varshney2695@gmail.com)

Problem Statement: Part-of-Speech (PoS) tagging assigns grammatical categories to every token in a sentence. In this assignment, you have to develop a PoS tagger using Hidden Markov Model (HMM) and Recurrent Neural Network.

Dataset: WSJ (Wall Street Journal)

Number of PoS tags: 46

List of tags : 'MD', 'TO', 'WP', 'WP$', '.', 'PRP', 'PDT', '#', 'POS', 'VBN', '-RRB-', 'DT', '"', ':', 'EX', 'RP', 'RBR', '-NONE-', 'UH', 'VBZ', 'VBG', '$', 'RBS', 'JJR', 'IN', ',', 'VBD', 'LS', 'JJS', 'WRB', 'VBP', '-LRB-', 'NNP', 'NNS', 'PRP$', 'JJ', 'CC', 'FW', 'CD', 'VB', 'NN', 'NNPS', 'SYM', 'WDT', '``', 'RB'

Link to download the dataset:
https://drive.google.com/file/d/1GnH_RD087pyyMwJr4JoQSsDv-q9TrUB0/view?usp=sharing

- **Hidden Markov Model (HMM)**

  You have to implement HMM on your own. Do not use any existing libraries. Consider a bigram HMM model. Calculate the Emission and Transition Probability matrices. Use Viterbi decoding to obtain the best PoS sequence.

- **Recurrent Neural Network:**
  - You may consider the following details for the implementation.
    - Input Vec(Wi): The word embeddings will be the input to the model. You can use the Word2Vec or GLOVE embedding.
    - Link → Word2vec: http://vectors.nlpl.eu/repository/20/5.zip or https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit?usp=sharing
    - Link→ Glove: http://nlp.stanford.edu/data/glove.840B.300d.zip
    - Output (Ti): Sequence of POS tags.
  - You may use any deep learning libraries such as TensorFlow, PyTorch, Keras etc. for the implementation. Use 300 dimensions for word embeddings.

Evaluation:

1. Perform 3-fold cross-validation.
2. Compute the overall accuracy for each of HMM and RNN models for each of the 3 folds.
3. Show the class-wise accuracy of the best-performing fold (i.e. out of the 3-folds, you have to show for the fold that shows the best performance).