CSE556: Natural Language Processing

Assignment 04

Deadline: 24 Nov, 11:59:59 PM

Instructions:

1. The assignment can be submitted in a group of a maximum of two members. However, it is not mandatory to do assignments in a group.
2. Language allowed: Python
3. You are allowed to use libraries for data processing.
4. For Plagiarism, institute policies will be followed
5. You need to submit README.pdf, Code files (it should include both .py files and .ipynb files), and Output.pdf.
6. Mention methodology, preprocessing steps, and assumptions you may have in README.pdf.
7. You are advised to prepare a well-documented code file.
8. Submit code, readme, and output files in ZIP format with the following name: A4_rollno.zip
9. Make sure to use Pickle or any other library to save all your trained models. There will not be enough time during the demo to retrain your model. This is a strict requirement. Use this link to understand more about how to use Pickle.
10. Use classroom discussion for any doubt.

NOTE: If the assignment is attempted in a group, then, kindly mention the contribution of each member in the report.

Multilingual extension of BingLiu lexicon

DATA SET: You are requested to use the attached files for the given task. The english.txt and hindi.txt contain the English and Hindi mono-lingual corpus respectively. English-hindi-dictionary.txt contains the English-Hindi bilingual dictionary. BingLiu.csv contains Bing Liu lexicon.

Given the BingLiu lexicon, an English-Hindi bilingual dictionary, and two mono-lingual (English and Hindi) corpus, extend the Bing-Liu lexicon for multi-lingual support.

1. Using the bilingual dictionary, create an English-Hindi version of the BingLiu lexicon. Call it L1.
2. **Train** two monolingual (word2vec and glove) word embedding models, i.e. one for English and another for Hindi. No pre-trained models are allowed.
   You can follow these links for word2vec and glove.
   https://radimrehurek.com/gensim/models/word2vec.html

https://radimrehurek.com/gensim/auto_examples/tutorials/run_word2vec.html
https://github.com/stanfordnlp/GloVe#train-word-vectors-on-a-new-corpus
https://github.com/stanfordnlp/GloVe/tree/master/src

3. For each pair of English-Hindi (e.g., good → अच्छा) words in L1, find out 5 closest words each in the word-embedding space, i.e., five closest words of 'good' in the English word embedding model and five closest words of 'अच्छा' in Hindi word embedding.
   [Note: You can use the predefined distance function of word2vec/glove for the closest words or you can implement any of your own.]
4. If a word in English closest list can be mapped to a word in the Hindi closest list using the bilingual dictionary, then add this pair to list L1.
5. Repeat steps 3-4 until no new words are added to L1.
6. Report the extended lexicon in the Output.pdf.