

CSE472 : Natural Language Processing

Assignment 3 & 4

Tokenization and language modeling

Marks : 70 + 90

Deadline : 14 September 2019

1. Download following raw data from (common for both tasks)
 - a. <https://drive.google.com/open?id=1wAKArhgiYAseLsSrSVwS62ppBme3WfwB>
 - i. corpus1.txt
 - ii. corpus2.txt
 - iii. corpus3.txt
 - iv. corpus4.txt

PART 1: Tokenization

2. Write a tokenizer (in any programming language) which can do basic tokenization and following (do not use any existing tokenization library)
[implementation 30 marks]
 - a. Word tokenizer
 - b. Punt tokenizer (-,,. etc)
 - c. Email tokenizer
 - d. Url tokenizer
 - e. Number/Currency tokenizer
 - f. Name tokenizer , i.e. John M.
 - g. Hastag tokenizer
 - h. Mention tokenizer (@john)
3. Evaluation parameter (score): **[30 marks]**
 - a. Apply tokenizer on corpus3.txt and corpus4.txt (will compare them with existing tokenized text)
4. Submission Details
 - a. Code : To be uploaded on moodle with README
 - b. tokenized text (to be uploaded on google-drive and url must be given in README)
 - c. Zipf graph for corpus1.txt and corpus2.txt, give analysis for 10001 to 11000 ranked words for each corpus in report.
 - d. README
 - i. Name
 - ii. Roll No:
 - iii. Tokenized text url :
 - iv. How to run : python tokenizer.py corpus1.txt
 1. Tokenized text output must only be standard out on terminal

PART 2: Language Models

5. Use corpus1.txt and corpus2.txt as training data for LM
6. Write a code to create an N-Gram Model (N is parameter) **[implementation 30 marks]**
7. Write a code to calculate perplexity, apply kneser ney smoothing **[implementation 30 marks]**
8. Create language models for following parameters **[20 marks]**
 - a. corpus1.txt
 - i. **LM1**: tokenization + 4-gramLM + smoothing + interpolation
 - ii. **LM2**: tokenization + 6-gramLM + smoothing + interpolation
 - b. corpus2.txt
 - i. **LM3**: tokenization + 4-gramLM + smoothing + interpolation
 - ii. **LM4**: tokenization + 6-gramLM + smoothing + interpolation
 - c. Calculate perplexity score for each sentence of corpus3.txt and corpus4.txt for each of the above models and also get average perplexity score/corpus/LM
 - d. Generate sentences from conditional language models for corpus2.txt (compare results with unigram, bigram, trigram , 5-gram LMs)
 - e. Plot and compare all above LMs .
9. Submission Details
 - a. Code : To be uploaded on moodle with README
 - b. All LMs : (to be uploaded on google-drive and url must be given in README)
 - c. Perplexity scores for each LMs on corpus3.txt and corpus4.txt (8 files)
 - i. Format :
 1. Sentence TAB perplexity-score, at the end , average score
 - ii. Naming must be :
 1. roll_number-LM1-corpus3-perplexity.txt,
 - roll_number-LM1-corpus4-perplexity.txt,
 - roll_number-LM2-corpus3-perplexity.txt, etc
 - d. README
 - i. Name
 - ii. Roll No:
 - iii. LM url :
 - iv. How to run :
 1. How to create LM
 2. How to get perplexity on trained LM model

Note:

1. For both parts, submit a report on your observations from the outputs. **[20 marks]**