CIS/STA 9665: Assignment 5

Applied Natural Language Processing

Due Date: 11:59 pm Nov 20

Guidelines:

- Use Python as a programming language and finish this assignment in Jupyter Notebook
- Work is to be done individually for this assignment
- Students handing in similar work will both receive a grade of 0 and will face disciplinary actions.

Chapter 6. Learning to Classify Text

1. Using Naïve Bayes classifier described in this chapter, and any features you can think of, build the best name gender classifier you can. Begin by splitting the Names Corpus into three subsets: 500 words for the test set, 500 words for the dev-test set, and the remaining 6900 words for the training set. Then, starting with the example name gender classifier, make incremental improvements. Use the dev-test set to check your progress. Once you are satisfied with your classifier, check its final performance on the test set.

2. Using the movie review document classifier discussed in Chapter 6- Section 1.3 (constructing a list of the 2500 most frequent words as features and use the first 150 documents as the test dataset) , generate a list of the 10 features that the classifier finds to be most informative. Can you explain why these particular features are informative? Do you find any of them surprising?

3. Select one of the classification tasks described in this chapter, such as name gender detection, document classification, part-of-speech tagging, or dialog act classification. Using the same training and test data, and the same feature extractor, build three classifiers for the task: a decision tree, a naive Bayes classifier, and a Maximum Entropy classifier. Compare the performance of the three classifiers on your selected task.

4. Identify the NPS Chat Corpus, which was demonstrated in Chapter 2, consists of over 15,000 posts from instant messaging sessions. These posts have all been labeled with one of 15 dialogue act types, such as "Statement," "Emotion," "ynQuestion", and "Continuer." We can therefore use this data to build a classifier that can identify the dialogue act types for new instant messaging posts. Build a simple feature extractor that checks what words the post contains. Construct the training and testing data by applying the feature extractor to each post and create a Naïve Bayes classifier. Please print the accuracy of this

classifier. We use the first 15,000 messages from these instant messages as our dataset and use 8% data as our test data.

5. Given the following confusion matrix, please calculate: a) Accuracy Rate; b) Precision; c) Recall; d) F-Measure

	No	Yes
No	104	33
Yes	13	50

Chapter 7. Extracting Information from Text

6. Write a tag pattern to match noun phrases containing plural head nouns in the following sentence: "Many researchers discussed this project for two weeks." Try to do this by generalizing the tag pattern that handled singular noun phrases too. Please 1) pos-tag this sentence 2) write a tag pattern (i.e. grammar); 3) use RegexpParser to parse the sentence and 4) print out the result containing NP (noun phrases).

7. Write a tag pattern to cover noun phrases that contain gerunds, e.g. "the/DT receiving/VBG end/NN", "assistant/NN managing/VBG editor/NN". Add these patterns to the grammar, one per line. Test your work using some tagged sentences of your own devising.

8. Use the Brown Corpus and the cascaded chunkers that has patterns for noun phrases, prepositional phrases, verb phrases, and clauses to print out all the verb phrases in the Brown corpus.

9. The bigram chunker scores about 90% accuracy. Study its errors and try to work out why it doesn't get 100% accuracy. Experiment with trigram chunking. Are you able to improve the performance any more?

10. Explore the Brown Corpus to print out all the FACILITIES (one of the commonly used types of name entities).

What to Submit

a. Use Python as a programming language and finish this assignment in Jupyter Notebook
b. I have created an ipynb file with questions. Please add your code and answers in this ipynb file
c. After completion, please save your finalized ipynb file as a PDF file
d. Submit both **PDF file** and **ipynb file** to Blackboard
e. Please answers questions clearly, concisely, and completely. To answer some questions, the code is not sufficient. You should complement your answers in words by using **comments (#)**
f. The assignment will be graded on the correctness of the answers, comprehensiveness of the analysis, clarity of results' presentation and neatness of the report.