C S 479/579: Text Mining & NLP

Assignment 2: Naïve Bayes and Sentiment Analysis

1 Objective

In this assignment, you will have a chance to use sklearn library (https://scikit-learn.org/stable/) to train Multinomial Naïve Bayes and binarized Naïve Bayes for sentiment analysis.

2 Task 1: Data preprocessing

We will use 3 datasets from https://archive.ics.uci.edu/ml/datasets/Sentiment+Labelled+Sentences. The attributes are text sentences, extracted from reviews of products, movies, and restaurants. Each dataset contains sentences labeled with positive or negative sentiment. The three datasets come from three different websites/fields:

- imdb.com (imdb_labelled.txt)
- amazon.com (amazon_cells_labelled.txt)
- yelp.com (yelp_labelled.txt)

For each website, there exist 500 positive and 500 negative sentences.

Use CountVectorizer (https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html) in sklearn to perform the following:

1. Remove punctuation
2. Remove stopwords
3. For each dataset, split it into training set and test set (20%). You can use sklearn.cross_validation.train_test_split (https://scikit-learn.org/0.16/modules/generated/sklearn.cross_validation.train_test_split.html) to do this. Submit the following files: imdb_train.txt, imdb_test.txt, amazon_train.txt, amazon_test.txt, yelp_train.txt, yelp_test.txt.
4. Build the bag of word representations (document-term matrix of word counts) on training sets. For each dataset, use the same CountVectorizer object fitted on the training set to transform the test set into the bag of word representation.

3 Task 2: Train Multinomial Naïve Bayes

Train Multinomial Naïve Bayes on the three training sets. Please use Laplace smoothing (add-1 smoothing). You can use sklearn.naive_bayes.MultinomialNB (https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html).

4 Task 3: Train Binary Multinominal Naïve Bayes

Train Binary Multinomial Naïve Bayes on the three training sets. Please use Laplace smoothing (add-1 smoothing).
For this task, you should clip our word counts at 1. Then you can use sklearn.naive_bayes.MultinomialNB (https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html) on that new data.

Don't use BernoulliNB (https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.BernoulliNB.html#sklearn.naive_bayes.BernoulliNB) as it is different from Binary Multinominal Naïve Bayes.

5 Task 4: Evaluation

Test each model on the three test sets and report the accuracy. You can use sklearn.metrics.accuracy_score (https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html) to compute the accuracy. Which model (Multinomial Naïve Bayes or Binarized Naïve Bayes) has better accuracy on test sets?

6 Submission Instructions

Files to submit:

1. Data and results: imdb_train.txt, imdb_test.txt, amazon_train.txt, amazon_test.txt, yelp_train.txt, yelp_test.txt.
2. Source code: all your Python files, or Python notebooks.

Zip all your files to a zip file named assignment2.zip and upload it to Canvas.

7 Tutorials

These are useful tutorials that can help you complete this assignment.

- https://www.ritchieng.com/machine-learning-multinomial-naive-bayes-vectorization/
- Section 6.2.3 in https://scikit-learn.org/stable/modules/feature_extraction.html#text-feature-extraction
- Usage examples below each class/function in sklearn
- https://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html