ELEG 6913: Machine Learning for Big Data
Fall 2016
Department of Electrical and Computer Engineering
Prairie View A&M University

Project 3

(a) Sentiment analysis refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source materials. Sentiment analysis is widely applied to reviews and social media for a variety of applications, ranging from marketing to customer service. The aim of this project is to construct a sentiment classifier to classify reviews from internet into two emotion categories: positive and negative. Text data for the project is available at https://github.com/BruceDong/Resources-for-Projects-of-Machine-Learning/tree/master/Datasets/sentiment.

(b) Extract sentiment word features with the emotion lexicon http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm and corresponding Part-of-speech (POS) features with Stanford Part-Of-Speech Tagger http://nlp.stanford.edu/software/tagger.shtml on dataset 1 https://github.com/BruceDong/Resources-for-Projects-of-Machine-Learning/tree/master/Datasets/sentiment/dataset1.

(c) Represent the text data as feature data with the sentiment word and POS features.

(d) Divide the feature data into 5 parts: 4 parts as training data and 1 part as testing data.

(e) Choose a classification algorithm from logical regression, navie-bayes, neural network, and support vector machine, and train it on the training data to construct a sentiment classifier.

(f) Evaluate the sentiment classifier on the testing data with evaluation metrics, namely precision, recall, and F-score.

(g) Repeat everything in (b) plus extracting features of sentiment word phrases and corresponding POS features.

(h) Repeat everything in (c-f) with the same classification algorithm. Compare your results with (f).

(i) Repeat everything in (b-g) with the same classification algorithm on dataset 2.

(j) Summarize your observations and conclusions.

Bonus Section:

If you can implement "incremental navie-bayes classifier"[1] with C++ as the classification algorithm for this project, you will obtain extra 20% scores. Requirements:

(1) Implementing incremental flexible frequency discretization (IFFD)

(2) Comparing your results with those conducted by navie-bayes classifier[2]

Submit your programs and supporting documents as one zip file in ecourses by December 6, 2016.

---

[1] http://www.csse.monash.edu/~webb/Files/LuYangWebb06.pdf

[2] https://en.wikipedia.org/wiki/Naive_Bayes_classifier