NLP Final Project
Fall 2013, Due Thursday, December 12

For the final project, everyone is required to do some sentiment classification and then choose one of the other three types of projects: annotation, sentiment classification experiments and implementation. You may also propose your own project and you may work in groups.

Required Part 0. Sentiment Classification

For this part, you are to do at least one experiment in the NLTK on the movie review data in which you compare the baseline performance of using just word features with some modified or additional features that you implement.

You will need to write a Python feature function to generate the features that you choose. One idea is to use a stop word list to restrict the words in the word feature sets; other ideas are to vary the representation of the subjectivity lexicon features or to use positive and negative emotion words from LIWC.

Note that you will first want to develop your feature function until it works. Then to do the experiment, define a random raining and test set; define the regular word features, run the classifier and get accuracy; and then define your new features, run the classifier and get accuracy. The important thing is the two classifier runs are made on the same training and test sets so that you can compare the accuracy.

Include a description of your experiments in the final project report, together with the feature function code that you wrote and the accuracies of the two classifications.

Now choose one of the following 3 options.

Option I. Annotation and Analysis of Data

For this task, you will annotate data in a corpus and compare your annotations with other annotators. Two corpora that are available are described here. You may also collect your own data for annotation, with the permission of the instructor.

A. Twitter Health Care Opinion Corpus

For another research project, a number of tweets have been downloaded from Twitter that were tweeted on the topic of the Obama Health Care plan. For each tweet, we would like to have a label that says whether the opinion of the tweet is Positive, Negative or Neutral towards the plan. There are also some tweets not on the topic of health care reform and these can be labeled Irrelevant. Tweets are stored on spreadsheets in groups of approximately 100. Each tweet set will have two spread sheets; one will not have opinion labels on each tweet from previous annotators, and one will. In this corpus, each tweet will have two label columns; the first column is the opinion label that you ultimately decide on, but if you find it difficult to decide between 2 labels, you may put a second label in the next column.

For each person, the process is to read the definition of how to annotate. Then choose one set of unlabeled tweets and assign labels as best you can, using the second label column if it is hard to decide. Now compare your labels with the labeled version of the data and count how many of yours agree with the previous annotator. Compute the percentage agreement (the number of tweet labels that agree in their first label divided by the number of tweets).

Now continue and annotate a second sheet of unlabeled tweets and compare your results with the previous annotator. Did your percentage agreement improve?

To complete this project, write a report that describes what you did and what your agreement percentages were. Include in your report an example of tweet of each label that was easy and an example of one that was hard to decide. Discuss why the hard ones were difficult.

B. Occupy Sandy corpus

This project will be to annotate tweets for a research project analyzing how volunteers organize themselves to respond to a crisis through social media. This project is working with a collection of tweets from the hashtag #OccupySandy during the two weeks after Hurricane Sandy in December 2012.

This annotation task will be primarily a group effort, and you should expect to meet with me for annotation discussion sessions for about 6 hours in the last week of class and in the exam week, depending on the schedules of the people participating. The task will be defined and a set of labels described that will annotate each tweet with one or more labels that identify Purpose. There may also be a secondary classification label. You will be asked to annotate some number of tweets and then in the sessions, we will compare our results and train everyone to annotate uniformly. We will compare inter-annotator agreements.

Examples

Message	Purpose
Volunteers needed tomorrow from 274 Garfield sign up here: #SandyVolunteer All areas affected by #Sandy starting 2 feel short on volunteers. Reminder: help is still needed!	Request for Help (volunteers)
Come to the community-wide #OccupySandy meeting tonight 7:30pm at Jacobi! We'll come together to share needs and work toward solutions.	Notification
LES needs diapers, baby food, batteries/flashlights, water, PBJ+bread, blankets, fruit+veg. 638 E 6th bet B and C. #SandyAid	Request for Help (Stuff)

Option 2. Processing and Classification of Sentiment or other Data

For this task, you should choose to work on classification of a data set. If you do enough experiments on this task, then you do not also have to do the required part 0 of the Final Project, as that will be replaced by the extra experiments that you do in this option.

Based on the data and the task, decide what level of NLP processing is desirable and carry out any NLP processing needed, e.g. you may want to run special purpose Tweet POS tagging. Read the data into the NLTK (either by reading the file, or using a PlainCorpusReader) and write Python/NLTK that defines features.

Produce the features in the notation of the NLTK and use one of their classifiers to train and test a classifier on the data, or produce the features as a csv file and use Weka to train and test a classifier.

Available Data:
Twitter data annotated with positive, negative and neutral opinions towards health care reform.
Twitter data annotated with general sentiment from Sentiment 140
Email data separated into Spam and Ham directories for spam detection
(See http://blog.nerdery.com/2013/03/playing-in-the-sandbox-building-a-spam-detector-with-python/ for a description of this data and using it to make a spam detection classifier.)

Subjectivity dataset from Pang and Lee contains 5000 each objective and subjective sentences.
Product debate corpus and Political debate corpus from Wiebe at the MPQA data web site.
(Others are available)

Available Resources:
Twitter tokenizer and POS tagger from Motif.
Lexical resources: the Subjectivity lexicon from Wiebe, the LIWC dictionary from Pennebaker and the ANEW dictionaries from Florida.
Stanford POS tagger, named entity recognizer and parser(s).

For Tweet processing, the paper by Kouloumpis, Wilson, and Moore, "Twitter Sentiment Analysis: The Good, the Bad and the OMG!", ICWSM 2011 is recommended to read for ideas, both for processing and for sentiment features.

To complete this project, carry out at least several experiment where you use two different sets of features and compare the results. For example, you may take the unigram word features as a baseline and see if the features you designed improve the accuracy of the classification. Write a report that describes the data processing, the features and the classification experiment(s). As one of your experiments, you may instead compare results from different classifier algorithms in Weka.

Option 3. Programming Projects

Write a program to process text and discover lexical chains. We will work from the paper:

Barzilay and Elhadad, "Using Lexical Chains for Text Summarization", 1999. We will identify a subset of their algorithm that is reasonable to implement in NLTK using WordNet. More details will be forthcoming.

Write a Python program with a window interface that allows a user to specify a file or directory of files to process. The program should use the Stanford Named Entity Recognizer to process the text. There is a Python interface to the Stanford NER by Dat Hoang at https://github.com/dat/pyner. After the NER processes the text, the python program should make and display most frequent words, pruned by a stop word list, and most frequent named entities for the categories Person, Organization and Location. More details will be forthcoming.

The programs should be well-documented in a report that is handed in with the code.

What to Hand In

If you are working in a group, you should choose a task for each person. If you do annotation, hand in the annotation data. Every group should hand in a report with the description of all that you did and the discussion of the results. As usual, submit these documents to the Blackboard system by the end of the day on the due date.