Sentiment Classification in Movie Reviews

An Approach Using Subjectivity Filtering

Daniel Pomerantz

daniel.pomerantz@mail.mcgill.ca

McGill University

Problem

- **Problem:** Determine whether a given document is a positive or negative document
- **Motivation:** Customer feedback, blogs, editorials...
- **Our domain:** Movie Reviews
- **Approach:** Using machine learning, based largely on work of Bo Pang and Lillian Lee of Cornell

Problem Formulation

- **Given**: A set of training documents with known labels
- **Given**: A (set of) test documents with unknown labels
- **Run**: A machine learning classification algorithm
- **Output**: Whether the test documents are positive or negative reviews (binary)

Issues to consider

- What machine learning algorithm should we run?
- How do we want to represent our document?
- Can we isolate important parts of the document?
- ...

Classifiers Used

- We use Naive Bayes and SVM
- Recall that in Naive Bayes, we say that
-

$$P(C|D) = \frac{P(C)P(D|C)}{P(D)}$$

- We assume all features of our data are *conditionally independent*, given the class.

Feature Selection

- Choose how we want to store our document.
- Choose which features we want to consider
- We use unigrams only
- Out of these unigrams, we would like to eliminiate some

Feature selection

- Some words occur infrequently.
- Result: Susceptible to noise.
- Some words occur too often (e.g. the)
- Result: No useful information
- Determine this using information gain

Examples

- The two most positive words are:
- "excellent" "perfectly"
- The two most negative words are:
- "bad" "worst"
- Some useless words are:
- baseball, bartkowiak, michael, jackson, machine, learned

Running: Naive Bayes

- $$P(C|D) = \frac{P(C)P(D|C)}{P(D)}$$
- We use *maximum likelihood estimates*

MLE Estimates

- $$P(C) = \frac{\#\ documents\ in\ class\ C}{total\ \#\ of\ documents}$$

- $$P(D|C) = \prod_{f \in features} \frac{\#times\ f\ occurs\ in\ class\ c}{size\ of\ class}$$

Subjectivity Filter

- This process works fairly well, but has some problems
- An idea to improve classification is to preprocess data
- Try to filter out objective sentences
- This can be done via Naive Bayes as well

Subjectivity Filter

- Idea is same as with document analysis
- Only difference is in this case we look at individual sentences instead of documents
- Look at every sentence separately
- Determine whether words occur more often in subjective sentences or objective sentences

Example: Subjectivity Filter

- Subjective:
- I don’t think anyone needs to be briefed on jack the ripper.
- Overall, the film doesn’t stick because it doesn’t entertain
- Objective:
- two teen couples go to a church party, drink and then drive
- The film, however, is all good.

Final algorithm: Training

- Train a subjectivity detector.
- For every (labeled) document, run every sentence through the subjectivity detector
- Only keep sentences that are considered subjective
- Look at the (filtered) documents and determine which words are most useful to store

Final algorithm: Query

- Given a query document, apply the subjectivity filter
- Look at the filtered document
- Determine using Naive Bayes (or SVM) whether it is positive or negative

Results:

Table 1: A Comparison of the Algorithms:

Algorithm	Freq?	Regular	Subjective Filter
Naive Bayes	no	82.15 %	85.65 %
Naive Bayes	yes	79.90 %	84.90 %
SVM	no	87.3 %	85.85 %
SVM	yes	82.05 %	83.75 %

Future work

- Digrams, punctuation, etc.
- Learning from unlabeled data
- Recommending movies based on this
- Questions?