Naive Bayes

Classifying Movie Reviews HW 6 Team 6 - April 22, 2012

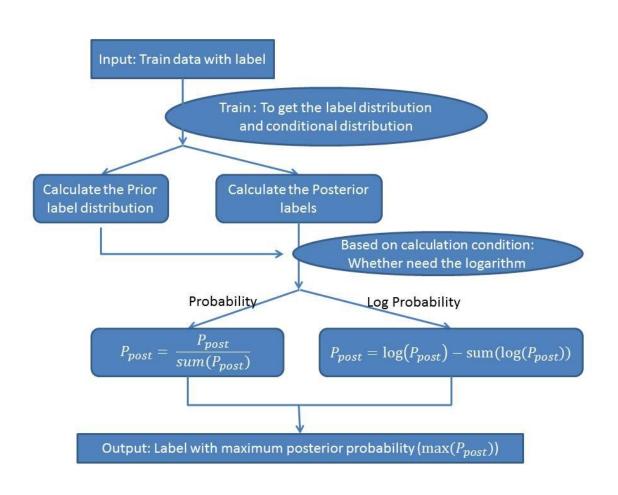
Model specification

- P(review = pos| hasword("brilliant")) -- posterior= P(hasword("brilliant")|review = pos)*P(review = pos)
- P(hasword("brilliant")|review = pos) -- likelihood = P(hasword("brilliant"), review = pos) / P(review = pos)
 - = count(hasword("brilliant"), review = pos) / count(review = pos)
- Assume each of the F features (e.g. hasword("brilliant")) are conditionally independent from one another
 - P(review = pos |all features) proportional to:
 P(review = pos)*P(feature1|review = pos)*P(feature2|review = pos)*...*P(featureF|review = pos)
- Classify review as either pos or neg by choosing the label with the highest conditional probability
 - Optionally consider log prob and smoothing for estimation

posTagger

- Input documents and optionally a chosen p-o-s tag
- Process the documents
 - Tokenize documents to words
 - Tag words using nltk package
- Return a list of tagged words, which in conjunction with getTopWords, will provide us with the proper features to use

Naives Bayes classifier



- Input training data coupled with the list of relevant features.
 Optionally choose to include a smoothing parameter and using log density for estimation
- Use trained classifier to classify test data and assign classes (positive or negative)

Overall procedure

Training data Train classifier **Extract features** Train posTagger: tag words from the first 50 Split training **NaiveBayes** data between reviews based on p-o-s filters (adjectives, classifier with and without verbs) positive and (features) with getTopWords: get top 10% of tagged negative and without 0.5 ratings words smoothing Recombine set of tagged words from positive and negative sets document features: convert tagged words to produce feature lists for each review **Test classifier** Clean training data to form list of *(review,* [features]) to pass into classifier Classify test data Calculate error metrics

Results - Using adjectives

	Basic			Smooth (0.	5)	
<u>Trial</u>	<u>TP</u>	<u>FP</u>	<u>SPEC</u>	<u>TP</u>	<u>FP</u>	<u>SPEC</u>
1	77.8%	46.7%	53.3%	77.1%	45.9%	54.1%
2	74.8%	39.4%	60.6%	74.8%	39.0%	61.0%
9	66.2%	34.4%	65.6%	67.5%	33.5%	66.5%
10	62.9%	24.0%	76.0%	62.1%	23.4%	76.6%
Average	71.5%	38.8%	61.2%	71.6%	38.3%	61.7%

148 FEATURES

Results - Using adjectives

Top features	<u>PosNegRatio</u>	Bottom features	PosNeg Ratio
'accessible'	7.26	'bite'	0.19
'breathtaking'	6.57	'worst'	0.22
'ambitious'	5.18	'biblical'	0.26
'additional'	5.18	'bitchy'	0.26
'annual'	5.18	'awful'	0.30

Note: PosNegRatio = Probability of feature in positive reviews / Probability of feature in negative reviews

Results - Including verbs

	Basic			Smooth (0.5	5)	
<u>Trial</u>	<u>TP</u>	<u>FP</u>	<u>SPEC</u>	<u>TP</u>	<u>FP</u>	<u>SPEC</u>
1	68.2%	39.0%	61.0%	68.6%	38.6%	61.4%
2	81.1%	54.0%	46.0%	81.1%	53.8%	46.2%
9	73.2%	46.1%	53.9%	73.4%	46.1%	53.9%
10	68.8%	44.1%	55.9%	68.8%	43.5%	56.5%
Average	68.7%	44.3%	55.7%	68.8%	44.1%	55.9%

116 FEATURES

Results - Including verbs

<u>Top features</u>	PosNeg Ratio	Bottom features	PosNeg Ratio
'disturbing'	3.05	'dressed'	0.30
'opened'	2.96	'dumb'	0.36
'minor'	2.50	'pull'	0.51
'compelling'	2.20	'guess'	0.53
'loose'	1.88	'replaced'	0.56

Note: PosNegRatio = Probability of feature in positive reviews / Probability of feature in negative reviews

Conclusion

- Adding verbs did not improve the classification of reviews
 - May consider more manual/selective process for choosing features among different p-o-s types
 - e.g. nouns like "Oscar", "Academy"
- Looking at the top 10 percent of adjectives used was a natural and basic yet effective approach.