Stephanie Fuccio
Engl 520
Fall 2016
Assignment #4, Question 2

LightSIDE Analysis

This analysis will use the LightSIDE Researcher's Workbench to compare the Naive Bayes and Logistic Regression classifiers with sentiment sentences extracted from movie reviews, ultimately trying to identify the most accurate classifier by comparing numerous models with the features available. The sentiment.sentences.csv training file available in the LightSIDE documentation folder was used for this analysis. This training data consisted of sentences from numerous movie reviews along with their human tagged "pos" or "neg" sentiment code. "Pos" tags were given to sentences that were positive in meaning, such as "you will likely prefer to keep on watching" and "neg" tags were given to review sentences with a negative meaning such as, "a major waste….generic." A brief summary of all of the model data can be found in Appendix A. Lastly, it should be noted that unless specified, the default selections in LightSIDE were chosen in each part of this analysis.

As suggested for novices of Machine Learning in the LightSIDE instruction manual, the workflow for this analysis was as follows: extract features > build models > compare models. The other three tabs were not utilized for this paper. For this analysis I chose to run the same 7 models with Naive Bayes and Logistic Regression classifiers from within LightSIDE. Of the 7 models, 4 were run with unigrams, 2 with bigrams and 1 with trigrams. This was decided due to the newness of the user and the advice of the creators of LightSIDE who recommended that bag of words, the default feature that uses unigrams, be used for initial modeling. Also, since sentiment tagging was the focus, it was decided *not* to utilize numeric features such as 'count occurrence' or 'line length', and instead focus on 'part of speech (POS)' and 'include punctuation'. 'Part of speech' was chosen due to the heavy adjective and adverb usage expected in emotionally laden positive and negative movie reviews. Likewise, punctuation in this informal genre can help identify where sentiment language would appear in the data set("!" near strong emotional words being an example. It is also important to mention that 'track feature hit location' was chosen for all of the 7 models because I hoped to eventually do the extensive error analysis detailed in the instruction manual. However, results of an extensive error analysis are outside the scope of this brief comparative classifier analysis, both in space and time.

Looking now at the "Comparison Chart "in Appendix A, accuracy comparisons can be made. Of the 7 models run for each classifier, Naive Bayes was far more accurate overall than the Logistic Regression classifier was. If we exclude the lowest accuracy score for both classifiers, model 1 with trigrams, than the Naive Bayes classifier range is from 71% to 78%, whereas the Logistic Regression classifier has a slightly but noticeable lower accuracy range of 70% to 76%. If accuracy were a sole determiner for model selection, then based on these 7 models, the Naive Bayes classifier would be preferred for sentiment datasets, as can be seen in the LightSIDE "Compare Model" screenshot in Appendix B. Also, if one were to choose a model of the 7 tested in this analysis for the sentiment movie review analysis context, model 4 would be the clear winner. The Naive Bayes accuracy score for that model, which included the bigram, words/POS pairs, include punctuation, and track feature hit location features, received a score of 78%, compared to the 76% from the Logistic Regression classifier. Why might this model have gotten the highest accuracy score? As mentioned earlier, it is logical that bigrams would elicit a higher percentage of correctly tagged sentiment Ngrams due to the propensity of adjectives and adverb to provide rich emotional clues to the reviewer's positive or negative tone in a review. Is it possible that the machine ‘learned’ how rich these WORD-ADJ and WORD-ADV bigrams were with emotionally laden meaning that it started to search specifically for these POS tags instead of moving through the reviews bigram by bigram? It is possible that since this model has the most tokens of all 7 models, that this could play a significant role in this high accuracy rating, since in statistics the more data once has to analyze the closer the context becomes to a real outcome and the more reliable the calculations are.

However, as we know, accuracy is *not* the sole determiner in selecting a language model: precision and recall also play a large role. Whereas accuracy utilizes true positive and true negatives along with their false counterparts, prevision focuses on true and false positives only. What fraction of the data was labeled as positive was done correctly? is the question that precision answers. Referring again to Appendix A it is interesting to note that the least accurate model, model 1, received the highest precision score: 51%. This could be true because it has the least amount of tokens, at only 2326. Could it be that less work to do means less potential mistakes? However, even though model 1 had the highest precision score, the others were not far behind. The precision score range was a narrow one, from 49.1% to this high of 51%. This, it appears that the precision would not be the strongest factor, in this analysis, to decide on the best language model for this movie review sentiment data.

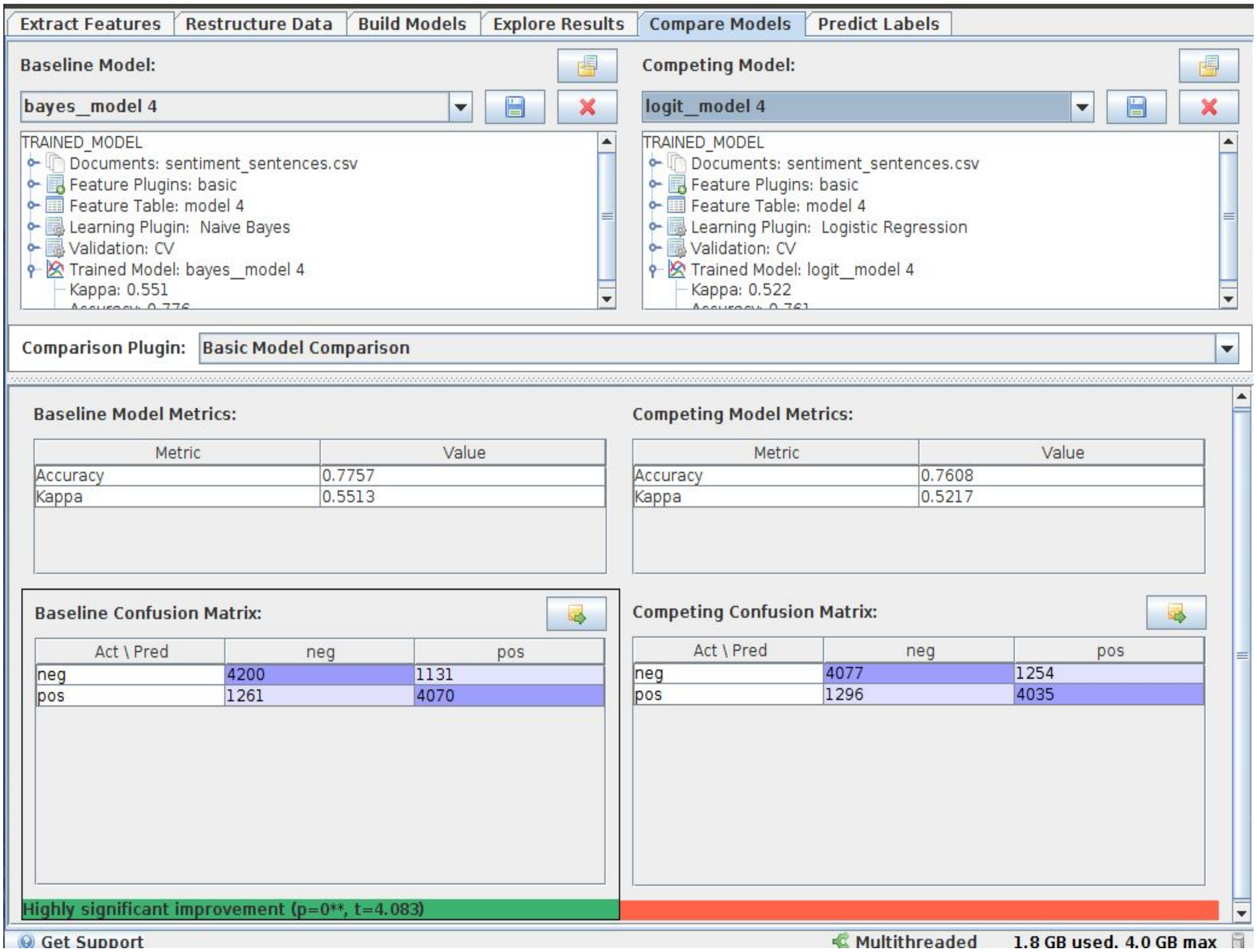
Moving on to the third major factor in identifying an ideal language model, recall, does provide comparative data useful in making a decision. Recall focuses on the positive examples and what fraction of these the classifier picked up on. Looking at the data in Appendix A once again it is clear to see that the vast majority of the 7 models produced extremely low recall scores, ranging from .1% to .36%. However, there is one outlier, model 5, where the recall is as high as 50%. This still seems like a rather low recall number *but* compared to the other 6 options, it is much, much higher.

But before we select 1 of the 7 models a side-by-side comparison is in order. Using the “Compare Models” tab in LightSIDE we can see some surprising results, only 2 of the 7 models were labeled as “highly significant” : model 6 and 7, with 71% and 77% accuracy respectively, similar precision scores and very low recall scores. This label is a clear reminder that it is (at minimum) a balance of accuracy, precision, and recall that determine the overall desirability of a language model. With all of this in mind, model 5 seems to be the best one from these seven choices, in this movie review genre, with this exact data set. Having said that, it is important to keep in mind what the LightSIDE creators reminded of us of in the instructor’s manual, that it is best to use a test set *in the same genre* as your training set to get the most out of this tool. If you mix apples and oranges you may end up with a delicious smoothie, but that will not help you build a useful language model.

Appendix A: Classifier Comparison Chart

Model	Model Features	# of tokens	Naive Bayes (NB)	Logistic Regression (LR)	Compare Models	Average precision (PR)/ Average Recall (R)
1	-trigrams -include punctuation -track feature hit location	2326	**62%** **(.6213)**	**61%** **(.611)**	NB signif improvement: p=.015, t=-2.428	**PR=.510** RI=.0010
2	-unigram* -include punctuation -track feature hit location	4485	77% (.775)	76% (.7587)	NB sig imp: p=0, t=5.644	PR= .492 R= .0038
3	-words/POS pairs -include punctuation -track feature hit location	4605	77% (.7697)	76% (.7579)	NB sig imp: p=.001, t=3.426	PR= .496 R= .0036
4	-bigrams -words/POS pairs -include punctuation -track feature hit location	**10414**	**78%** **(.7757)**	**76%** **(.7608)**	NB sig imp: p=0, t=4.083	PR= .494 R=.0025
5	-bigrams -normalized N-gram counts -include punctuation -track feature hit location	5809	71% (.7088)	70% (.6956)	**NB** **highly sig diff:** **p=0, t=3.547**	PR= .496 **R= .496**
6	-unigram* -word/POS pairs -include punctuation -track feature hit location	9090	77% (.7737)	76% (.7567)	**NB** **highly signif imp:** **p=0, t=4.753**	PR= .491 R= .0017
7	-unigram* -include punctuation -POS bigrams -track feature hit location	5334	76% (.7562)	75% (.7504)	Insignif imp: p=.143, t=1.466	PR= .495 R= .0069

Appendix B: Comparison of Model 4: Highest Accuracy for Naive Bayes & Logistic Regression



Baseline Model: bayes_model 4

TRAINED_MODEL
- Documents: sentiment_sentences.csv
- Feature Plugins: basic
- Feature Table: model 4
- Learning Plugin: Naive Bayes
- Validation: CV
- Trained Model: bayes_model 4
  - Kappa: 0.551
  - Accuracy: 0.776

Competing Model: logit_model 4

TRAINED_MODEL
- Documents: sentiment_sentences.csv
- Feature Plugins: basic
- Feature Table: model 4
- Learning Plugin: Logistic Regression
- Validation: CV
- Trained Model: logit_model 4
  - Kappa: 0.522
  - Accuracy: 0.761

Comparison Plugin: Basic Model Comparison

Baseline Model Metrics:

Metric	Value
Accuracy	0.7757
Kappa	0.5513

Competing Model Metrics:

Metric	Value
Accuracy	0.7608
Kappa	0.5217

Baseline Confusion Matrix:

Act \ Pred	neg	pos
neg	4200	1131
pos	1261	4070

Competing Confusion Matrix:

Act \ Pred	neg	pos
neg	4077	1254
pos	1296	4035

Highly significant improvement (p=0**, t=4.083)

1.8 GB used, 4.0 GB max