Naïve Bayes Classifiers

Christina Wallin, Period 3

Computer Systems Research Lab 2008-2009

Abstract

One part of computational linguistics is the classification and comparison of texts into classes. This comparison could be author classification, spam filtering, or in the case of this project, the classification of texts from different genres of news stories. The goal of the project is to be able to classify a text as belonging to one of two classes based on the actual words in the text. This method of classification is called a naive Bayes classifier. In this age of huge amounts of data available online, a classifier which could discriminate between two types of news could be increasingly useful.

Background

The naive Bayes classification is a relatively simple method for classifying texts based on the false assumption that all of the variables, in this case words in the documents, are independent of each other. Even though this assumption is false, this project is done to achieve fundamental understanding concerning the effectiveness of the naive Bayes as compared to other methods, and to find a way of improving upon the performance of this classifier. However, it is also done to try to provide a way for news stories from different genres to be classified. In this age with huge amounts of data popping up on the internet every day, an effective way to sift through the texts with classification will be essential.

There have been many studies which have used the naive Bayesian classification method. One paper which puts together information from previous studies is "Idiot's Bayes--Not So Stupid After All?" by David Hand and Keming Yu, which using theoretical and real data situations shows that the Naive Bayes is not a horrible method because of its false assumption that all of the variables (in my case, occurrences of words) are independent. This paper is a review of past uses of naive Bayes and the conclusions of those researchers and a theoretical treatise as to why the naive Bayes is effective.

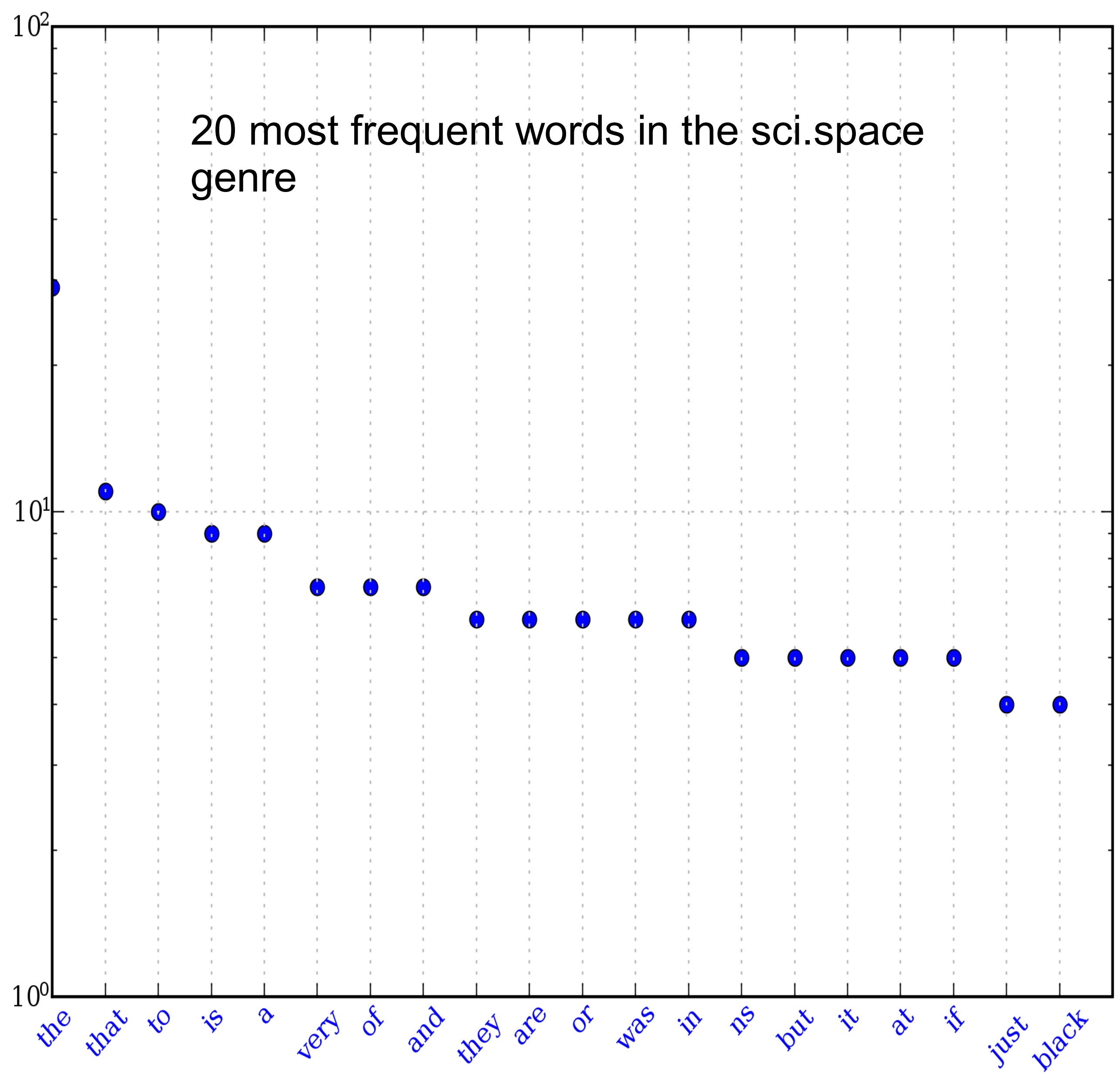
Expected Results

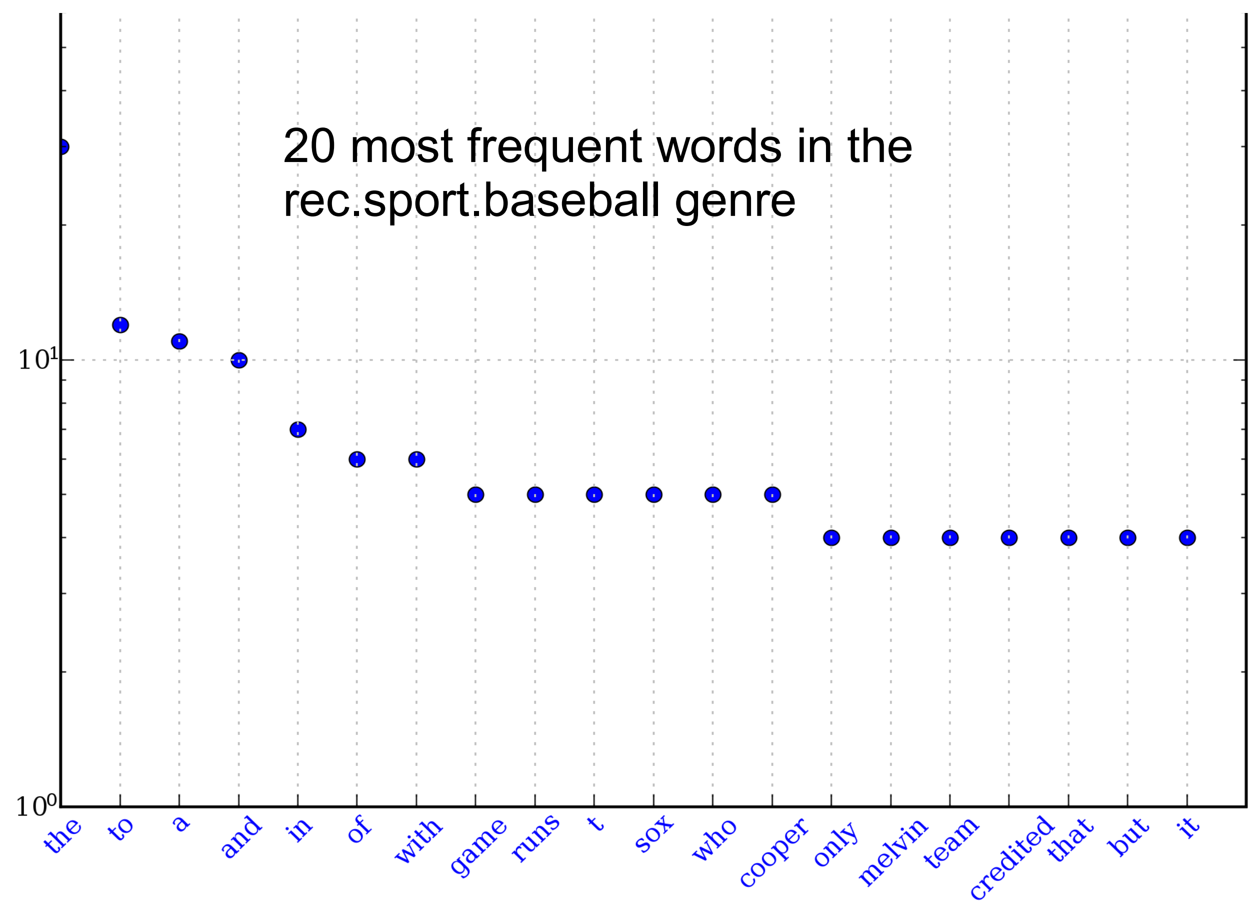
I have not yet finished, but my expected results are a percentage of the test cases which the naïve Bayes classifier correctly classifies, and the comparison of that percentage to that of a Bayes classifier which also uses a Porter stemmer to group words with the same stem together as the “same word.”

Methodology

The programming language used for these manipulations is Python. The database which I will be using to classify texts is the 20 Newsgroups database. There are 20 different genres of news stories, and it is divided into training and testing sets. The training and testing sets are separated in time by a little, and so it is more realistic.

For the first quarter version, I have focused on researching how to make a naive Bayes classifier and creating the first section of it. It reads in a text and parses the string to remove all punctuation and capitalization. Then, it creates a dictionary of words occurring in that text and their frequencies. This step is in anticipation of the next part of the program, which is to create a probability vector (PFX) for each class. The program has been tested by using small data sets and checking manually whether or not the frequencies were correct. The portion which used regular expressions to remove punctuation was tested by printing out the parsed strings and seeing if the regular expression did the correct operation.


20 most frequent words in the sci.space genre
10^2
10^1
10^0
the
that
to
is
a
very
of
and
they
are
or
was
in
ns
but
it
at
if
just
black


20 most frequent words in the rec.sport.baseball genre
10^1
10^0
the
to
a
and
in
of
with
game
runs
t
sox
who
cooper
only
melvin
team
credited
that
but
it