Analyzing CULPA data

STAT 4249 Final Project - Team 20

Chih-Kai Chang cc3527
Jan Pieter Leibbrandt jpl2148
Yi Jiang yj2306
Tingyu Gong tg2468
Mingyun Guan mg3419

Introduction

For our project we decided to apply text analysis skills to derive information from a dataset from CULPA, a website that allows students to review professors from past classes and to track other's opinions on their potential classes for the next semester.

Using CULPA, students can write a review and comment on professor, course's content, and its workload. Other students have the option of agreeing or disagreeing with a particular review, and deciding whether a review is funny. Students also nominate professors for silver or gold nuggets, which are given to professors by CULPA based on their reviews.

Objective

For each review, we retrieved the actual textual contents of the review, and workload, the numbers of "agree"/ "disagree"/ "funny review" received by the review, and the nuggets assigned to each professor.

We split the projects into two parts: analysis and prediction. For analysis we looked at the data, trying to characterize behavior across departments in terms of word use, workload declaration. The first predictive objective of this project is to predict whether or not a professor has a nugget and its type based on other variables with a main focus on review and workload. The second predictive goal is to predict whether the review is regarded as funny.

Data Description

We got data from CULPA in JSON format. There are two files in the data, one with professors, and one with reviews. Rjson package in R Studio is applied to convert the JSON objects to R objects. In total there are 3000 professors and 21000 reviews. The entire datasets can be found at: http://www.columbia.edu/~zjn2101/culpadump.zip and on their website, http://www.culpa.info/

We also requested data on departments and schools from CULPA, so we could look at behavior across different departments, and they obliged by sending us additional data which was very kind! Unfortunately no data was available on schools (so we could not distinguish between graduate and undergraduate classes).

As for data pre-processing, we used the tm package to build a corpus from character vectors of reviews. Then we applied a number of transformations, including changing letters to lower case, removing punctuations, removing white space, and removing stopwords. The remaining terms then became effective as the textual representation. After generating a document-term matrix from clean corpus, tasks like classification can be applied directly to it.

Preliminary Analysis

I. Characteristic Word Use Across Departments

To get an idea of the kind of data we were dealing with we did some visualizations of the text data.The first manner of analysis was creating frequency tables for the reviews, looking at what words were used most frequently. We used a package called wordcloud to then visualize these frequencies - the package displays the words in random positions with their sizes relative to their frequency. We used this technique throughout the rest of the project to visualize the word choices.

The most interesting application of this was to look at the individual departments and look at their particular patterns, which were very revealing. To get words specific to one department, we created the frequency tables for all of the departments together as well as for each department individually. Then we took the 220 most common words from all reviews, and ignored them in the tables for individual departments. This way, only the words specific to each department remained in their lists. A subset of the images is presented below (more can be found at http://www.columbia.edu/~jpl2148/CULPAstuff.html).

The 'wordcloud' for the economics department is pictured below. We can identify two popular economics professors (Gulati and Musatti), as well as descriptive words like "concepts," "principles," "textbook." These are all terms used more by reviews for the econ department than by the reviews in general. Another inference we can make is that 'math' is central to economics major, as it is the only specific subject mentioned.

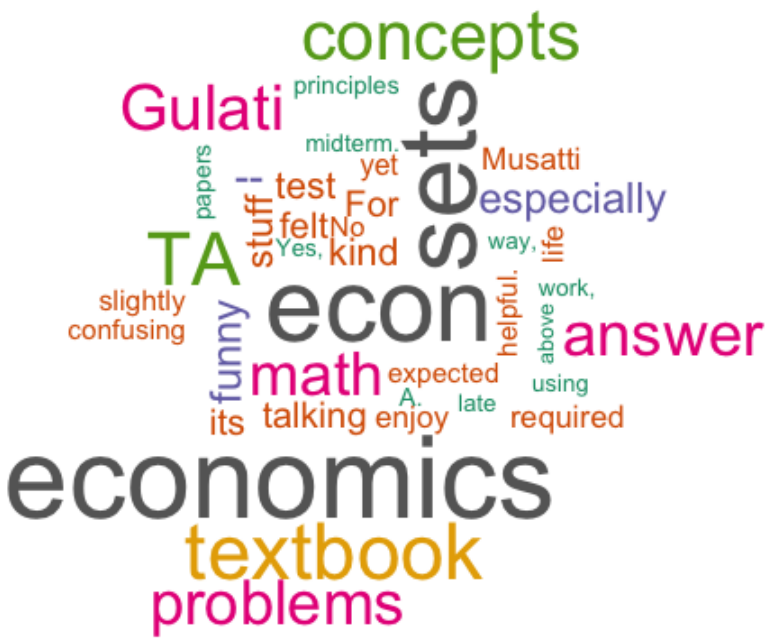

Figure 1. Wordcloud for the Economics Department

Below are four other wordclouds made like this for other departments. The department name usually is the

most frequent word, appearing the largest in the image. Reviews in the scientific departments tend to have words like “exams,” “textbook,” “lecture” and “problems,” while non-scientific subjects have more descriptive words like “interesting,” “difficult,” “comments,” “listen.” Reviews for the philosophy department were the only ones containing significant profanity.

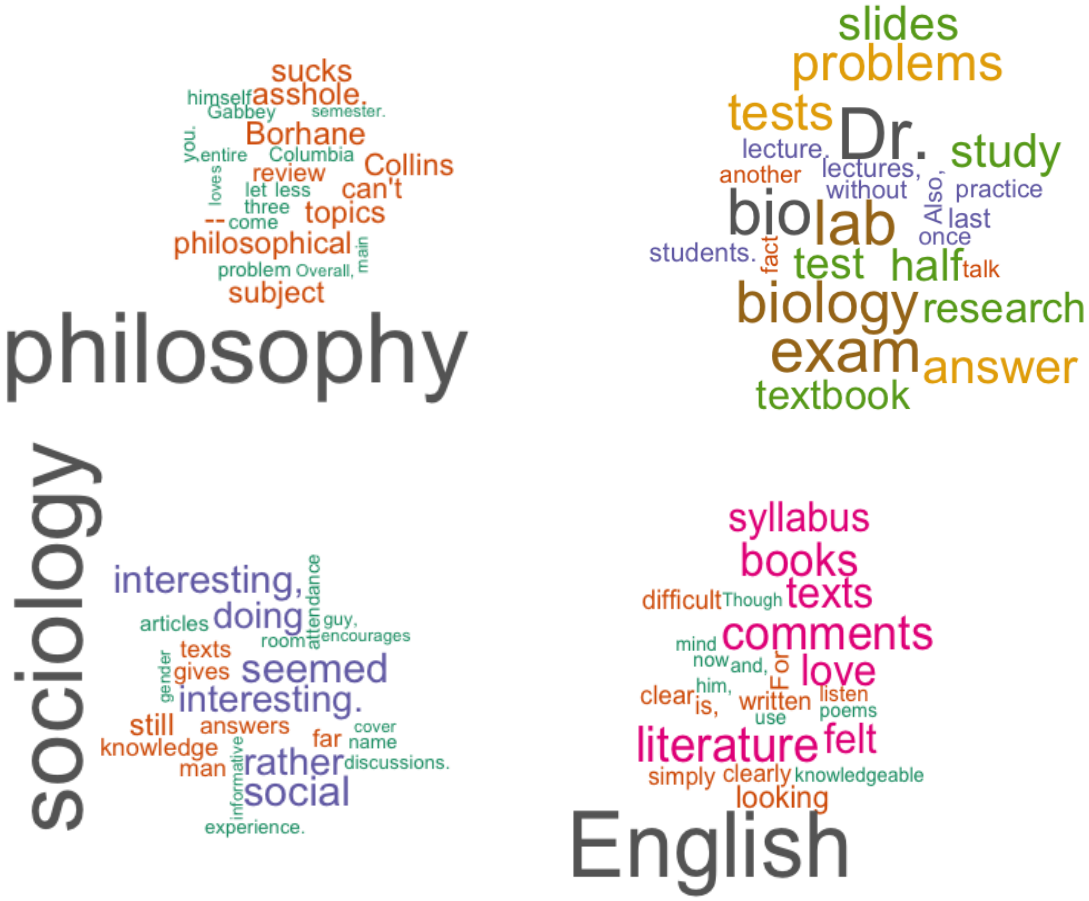

Figure 2. Wordcloud for Selected Departments

II. Nugget Proportions across Departments

We also looked at what departments have the highest proportion of professors with gold nuggets. This would illustrate what departments are popular, and have better professors. Subjects above the diagonal, like philosophy and economics have a high proportion of professors with nuggets, whereas subjects below, like the political science and psychology have fewer. This graph also shows that the mathematics department has the highest number of professors with gold nuggets. Here is the link for the interactive graph: http://yiginger.github.io/Myd3/Bubble.html. The static graph is here:

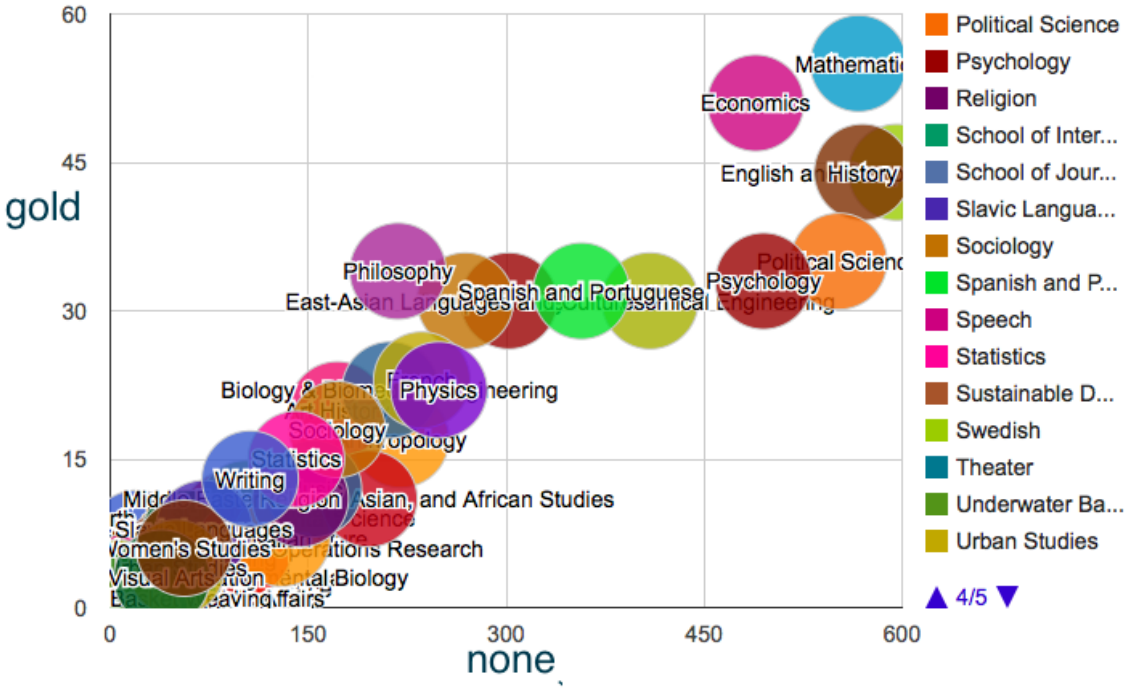

Figure 3. Gold Nuggets vs. None-Nuggets in Different Departments

III. Workload Assessment Using Regular Expressions

Another interesting aspect of the data that is not captured by this text-mining technique is the workload column, in which students can express how tough the class is. After looking at the data we found that a lot of the workload fields contained phrases like “2 papers”, or “1 midterm” referring to the number of papers and midterms in that particular class. We decided to capture this information by creating a regular expression and parsing it in each review. This way if somebody said “1 midterm” and “2 papers” in a review, a 1 would be added to the ‘midterm’ column and a 2 to the ‘papers’ column.

Below is a plot showing average number of papers mentioned in reviews for the departments. The first thing we notice is that in mathematics, physics and computer science there are no papers. Classes in language, music and core departments are all between 2 and 3, except portuguese, in which people write more than 3 papers on average.

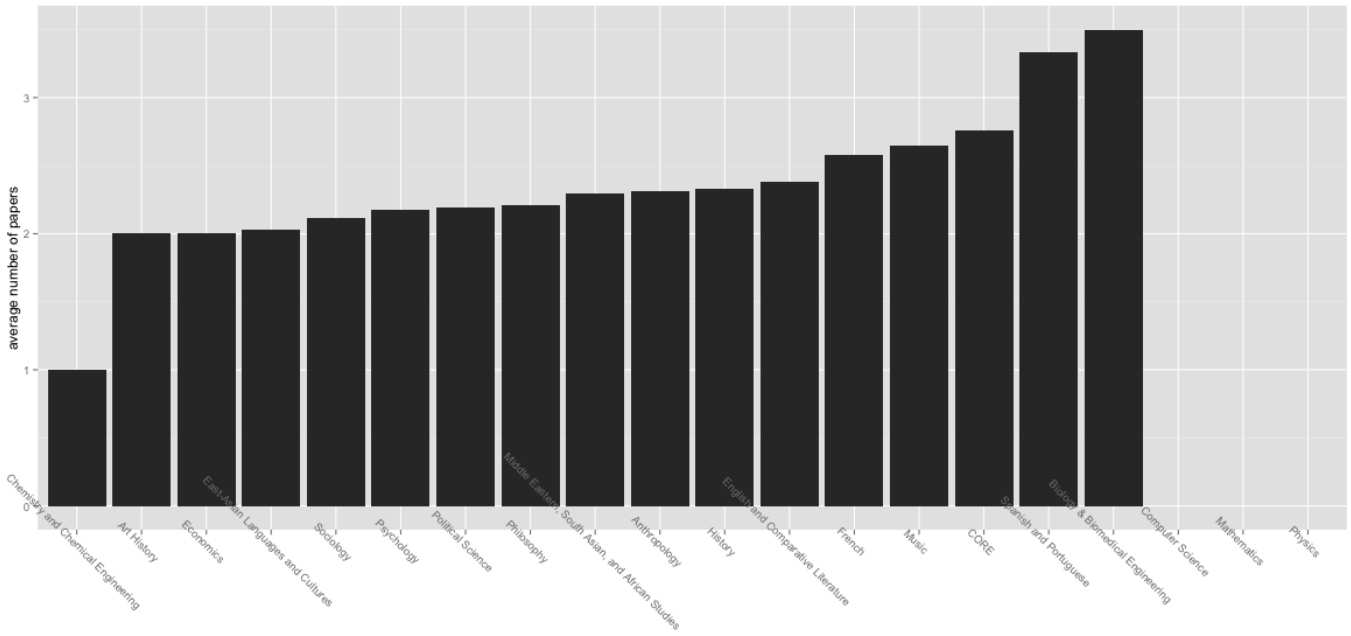

Figure 4. Average Number of Papers in Different Departments

We can repeat this process for midterms to see how many midterms classes have on average. In this plot we notice that scientific subjects like engineering, mathematics and physics have more midterms than classes like history and languages.

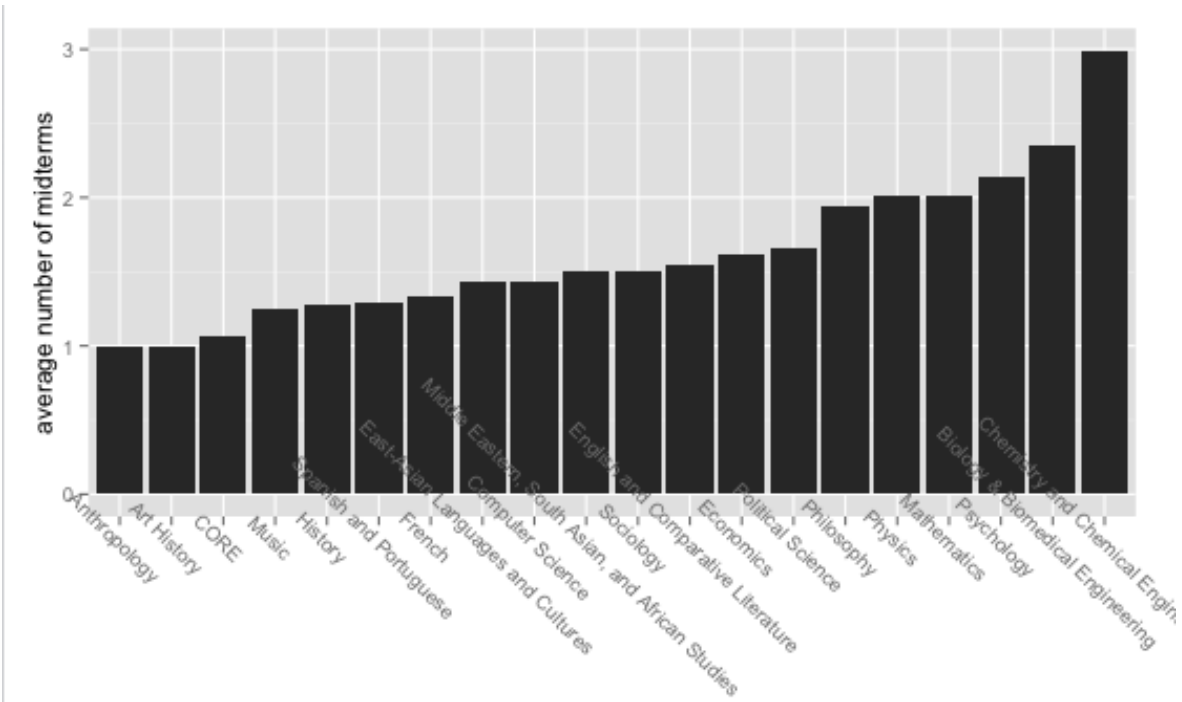

Figure 5. Average Number of Midterms in Different Departments

Correlation Between Workload and Nuggets

Something else that we were interested in was to see if workload was related with nugget status of the professor. To do this, we had to differentiate between classes with light and heavy workloads. We applied clustering to the different workload reviews, and this allowed us to form two groups. Using these and the frequency analysis techniques employed earlier, we built two dictionaries. We chose words with frequencies higher than 50 to be included in the dictionaries. We also performed some manual deletions to words we didn't believe would influence results like "you're" and other stopwords. The results are as follows:

Table1 : Workload Dictionaries

light workload	"easy","short","essay","fair","discussion","light","good","worth","standard", "fairly","nottoo","manageable","interesting","little","reasonable","optional","notbad", "lowest","fine","nothing","least","doable","tough","moderate","fun","nomidterm","helpful", "normal","small","medium","nocumulative","easily","decent","simple","regular","nofinal", "nograded","notgraded","nothard","notmuch","shorter","notnecessary","nohomework", "managable","notdifficult","brief","notvery","notreally","not impossible","worthwhile"
heavy workload	"final","midterm","papers","reading","sets","weekly","readings","midterms","pages","essays" ,"quizzes","assignments","homework","exams","lot","hard","long","read","tests","bad","reser ch","presentation","difficult","lots","writing","project","heavy","exam","bad","material","2paper s","topic","assignment","extra","quiz","review","lab","2midterms","oral","written","homeworks ","test","exercises","compositions","3papers","multiple","report","reports","many","postings", "articles","recitation","challenging","practice","cumulative","analysis","projects","concert","of en","mandatory","presentations","drafts","texts","posts","annoying","big","quizes","impossibl e","biweekly","seminar","3midterms","assigns","finals","memorize","harshly","tons","1midter m","consuming","article","journal","comments","graded","grades","labs","reviews","midtermfi nal","post","museum","discussion","passage","discussed","film","large","intense","boring","h uge","2page","3essays","2essays","timeconsuming","poems","rewrite","posting","4essays", "poem","3tests","stupid","harsh","rewrites","movies","conversation","noteasy","proposal","un necessary","composition","poetry","translation","novels","2exams","experiment","hws","heav ily","literature","materials","worst"

As we can see, the clustering worked well to differentiate between classes with light and heavy workloads. In the light workload group we see words like "easy," "short" and "light" whereas the other group contains words describing exams, and "hard," "long," and "lot." It seems that we managed to correctly distinguish easy and difficult classes using clustering.

Similar to the twitter example in the class, we assigned a score to each review by counting the number of occurrences of "heavy workload" and "light workload" words in a workload comment, using the dictionaries we just built. Then to compare professors with and without nuggets, we compared the densities for the different groups. The results of this are shown below.

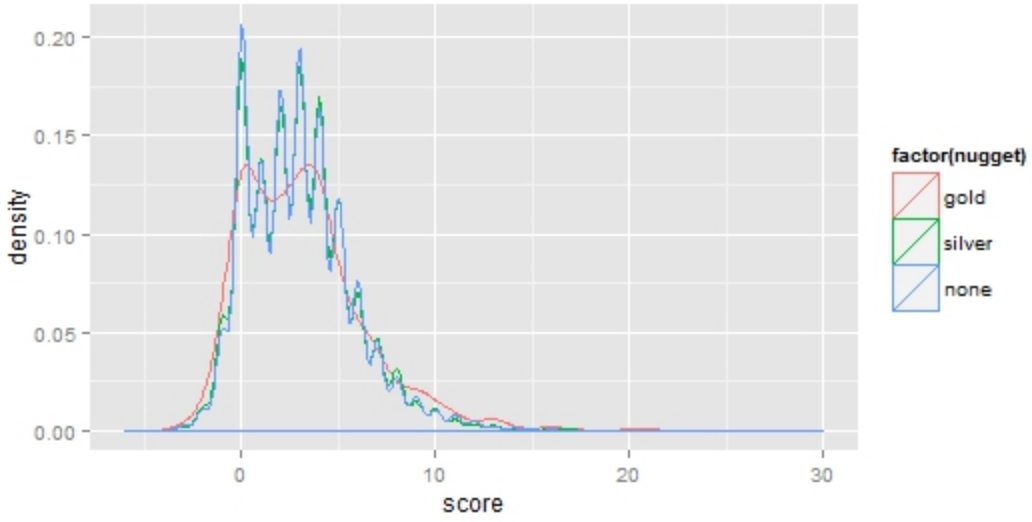

Figure 6. Workload density for professors with and without nuggets

Predictive Analysis

I. Predict whether a professor has a nugget and its type

Data Cleanup

First, to get a better understanding of the words used in student reviews, we create an ordered frequency table of all the words used in the reviews. We soon discover that the top used words are school related words such as ‘professor’ and ‘class.’ As these words do not tell us whether the review is positive or negative, we classify the first 250 most frequently used words as common words. We disregard these words from our dictionary.

An attempted approach is to remove common words between reviews of nugget professors and no-nugget professors. However, the amount of common words between reviews is very large. When these are removed, the dictionary is left with professor’s names, course names, and department names, which is not very helpful.

Predicting Nugget

We split up the available data into two sets, one for training and one for testing. We create two dictionaries from the training data categorized by nugget or none. The basic approach is to see how well the words used in the reviews are contained in the nugget dictionary. We give a score to each professor based on the following system. If a word belongs to the nugget dictionary, then we increase the score by 1. If a word belongs to the none-nugget dictionary, we decrease by 1. If a word belongs to both, we do nothing.

Using the developed score, we tried to predict whether or not a professor has a nugget. Using the mean of score as the cutoff, we find that our model is 42% accurate. Taking a closer look at the distribution of the score, we discovered why this is the case.

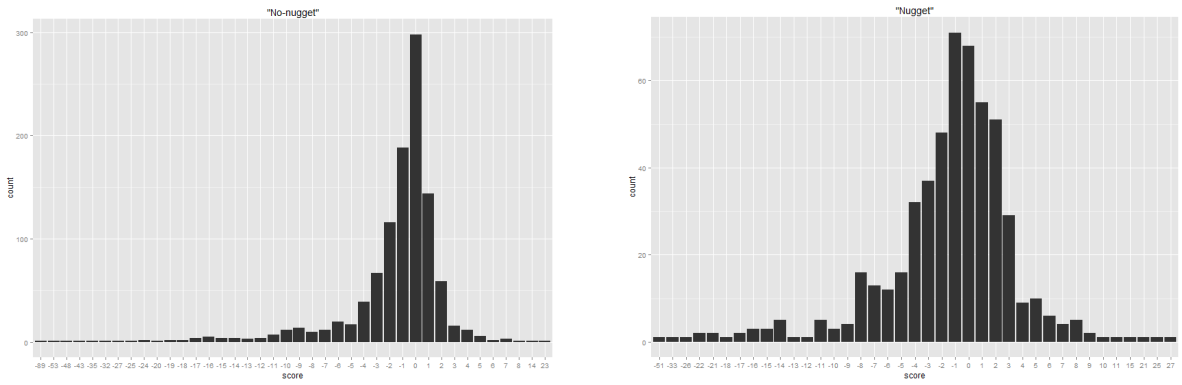

Figure 7. Score Distributions from None-Nugget and Nugget Dictionaries

Both score distributions are roughly the same with a mean and median at -1.2. With such a close distribution, our model cannot accurately predict the existence of nugget from the review. As it turns out, frequency of words has nothing to do with determining the existence of nugget for a professor. The main problem with this model is that we do not capture the content of the review by simply examining the words individually. The content of a review changes based on groups of words. A simple negation of an adjective changes the meaning and is ignored by our model.

It seems that word-choice is not related with professor quality; this is similar to what we found with relationship between workload and nuggets. The distributions are similar for professors with and without nuggets.

II. Predict whether a review is considered funny

Data Cleanup

The corpus is split into two sets, a training set (50% of the initial corpus) and a test set (50%).
We employed **Naïve Bayes Classifier** to decide whether a particular review is funny.
First, we classified reviews into 2 categories, i.e. not funny, funny. depending on the number of “funny review” they received. Then we defined 0 as not funny, larger than 0 as funny.

Method

Step 1. Generate the Document-Term-Matrix according to all the reviews.
Step 2. Seperate the Document-Term-Matrix into two parts, one containing matrix of all the reviews with the number of fun equals to 0, and the other is the rest part of the matrix.
Step 3. Respectively seperate the two part of the Document-Term-Matrix into training data and test data.
Step 4. Getting the log frequency of each words in training data of “fun” reviews and training data of “not fun“ reviews plus the log prior frequency of “fun” reviews and “not fun” reviews.
Step 5. Calculate for each reviews in test data that which class it is more likely to belong to.

Model Assessment

In the “not fun” review test data set, the probability of being labeled as “not fun” is 90.4% but the prior probability of being labeled as “not fun” is 83.7%. In the “fun” review test dataset, the probability of being labeled as “fun” is 22.0% but the prior probability of being labeled as “fun” is 16.3%. So in this sense, the

classifier is slightly better than randomly guess classifier.Besides,we calculated the several indicators of whether the classifier is useful. Here is the result:

Table2 : Contingency Table of Predict Result

	false positive	false negative	true positive	true negative
train	0.04761905	0.0258567	0.952381	0.9741433
test	0.6904943	0.1442577	0.3095057	0.8557423

The true positive in test data is not good. And we can see that this model seems to have some overfitting, because the error in training data is much better than that in test data. We also try to add the number of un for each reviews as the weight, but the result doesn't improve too much. Another reason for the bad result is that the word frequency might not be a good indicator of whether a review is fun because we can see the words frequency in fun reviews and not fun reviews do not have great differences. We list the most popular words in both dataset below:

Most common words in funny and non-funny reviews:
Funny: class, professor, will, really, students, one, take, course, just, like
Non-funny: class, really, professor, students, will, one, take, course, can, just

From these lists we can see that the differences are not significant at all. The distribution of words frequency in "funny " reviews and "not funny" appear to be very similar.

Conclusion

In conclusion, word choice is not as important as we thought in rating professors and determining whether a review is funny or not. The Naive Bayes algorithm assumes that the probability of presence of two words are independent but this is not the case here, which could be a reason why this method doesn't work well for predicting. Also, it seems that word choice is not important here, it is content and style that really matter!
We did learn some other interesting things from this data however:

- we found that people in different departments use very different vocabularies to describe their classes.

- we developed a model that distinguishes between easy and a difficult classes based on their CULPA reviews. We used this to demonstrate that difficulty of a class is not related to the popularity of the professor, which is an interesting result.

- we used regular expressions to see which departments assign the most papers and midterms so we could compare departments, with interesting results.

- we compared nugget proportions across departments, finding the most popular departments in terms of professors.