**Classifying and Clustering Courses at UC Berkeley**

INFO 290 Final Project

May 16, 2013

Joanna Hsu & Vanessa McAfee

**Introduction**

Choosing courses at UC Berkeley can be daunting. This world-class university offers about 5,000 different courses spanning a wide range of topics in about 150 different departments. Given both the interdisciplinary nature of many departments and the fact that subject matter expertise often transcends department boundaries, knowing what courses are available on a certain topic or group of topics is a nontrivial task. The goal of our project was to explore the Berkeley course data set and address the following questions: Given a course description, can we predict what department the course belongs to? Can we determine which departments and courses are most similar to each other based on their course descriptions? To address the first question, we built a Naïve Bayes classifier that achieved 39% accuracy. Classification accuracies were higher in larger departments. To address the second question, we used hierarchical agglomerative clustering to find similar departments and courses.

**Obtaining, Exploring, Cleaning Course Data**

We decided to build our own courses data set using the Berkeley [online catalog search engine](http://general-catalog.berkeley.edu/catalog/gcc_search_menu)). Search results were copied and pasted into a text file, and then the text file was parsed into a dictionary using regular expressions to identify the course code, title, description, and department.

After building the data set, we used MapReduce to generate some general statistics about the set and also looked through many of the course descriptions manually to gain an understanding. We had some prior knowledge about the set from our own experiences searching for courses at Berkeley. We knew that some course descriptions were generic, some courses are not included in the course catalog (such as most of the INFO 290 courses, which are considered special topics courses). Also courses with course numbers greater than 300 are used for graduate teaching courses and courses for doctoral students. We decided to remove courses that were above the 300 level at first. We also removed exact duplicate courses. This left us with a data set of 5777 courses. Later, when doing the unsupervised learning we decided to also removed the (90s) courses because many of them contained generic descriptions that we thought would affect the departmental clustering. This resulted in a dataset of 3926 courses.

The number of courses listed in each department ranges from 1 to 212; the department with the most courses was for Public Health. Interestingly the School of Public Health does not list courses by subject or program area, so courses in Epidemiology and Health Management and Policy are listed under the same code. Other Schools have courses listed by their specific program such as the College of Engineering, which has codes for each engineering discipline. This fact could influence the accuracy of our classifier which tried to predict which department a course belongs to. The average number of courses per department was 37, with a median of 26 and a standard deviation of 38. The average number of words in a course description was 50 words with a median of 46 and standard deviation of 31. The longest course description was 188 words. Some courses had as few as one word and about 330 courses had 10 or fewer words. Some courses with such few words failed to explain what the course was about, such as “Research.” But some short descriptions do have some information about the course.

**Course Classification**

We developed a Naïve Bayes classifier (code: Bayes\_classifier v2.py) to classify a course to a department based on its course description. We saved 10% of our data to use as a test set, and used the remaining 90% as a training set. To ensure that we could test the accuracy of our classifier on courses of all departments, we used stratified random sampling to ensure that every department was represented in both the test and training sets.

Macintosh HD:Users:joannahsu:Documents:Courses:projectCourse:accuracy.pdf

The classifier ranged in success from 0% to 77% across departments. The classifier performed best with the largest department, Public Health, classifying 17 out of 22 test courses correctly. With one exception, departments with more than 100 courses had at least 40% classification accuracy. However, because test courses from small and medium-sized departments could be classified either very accurately or very inaccurately, there was no linear correlation between number of courses in a department and classification accuracy. Average classification accuracy also did not vary between types of departments (liberal arts, languages, interdisciplinary, math and engineering, science). Many of the classification errors made sense given course descriptions. For example, INFO 207 was likely classified as an MBA course because of the words “cost”, “accounting”, and “budgeting” in its course description. Perhaps some departments had low accuracy because they do not have many discipline-specific words, such as City Planning, whose courses in the training set were classified into MBA (Business), Public Health, and Education departments.

In one experiment, we reduced the data set to also exclude courses ending in -90 courses (seminars). With this reduced data set, the classifier achieves 44.4% accuracy, an increase from the original model.

**Department and Course Clustering**

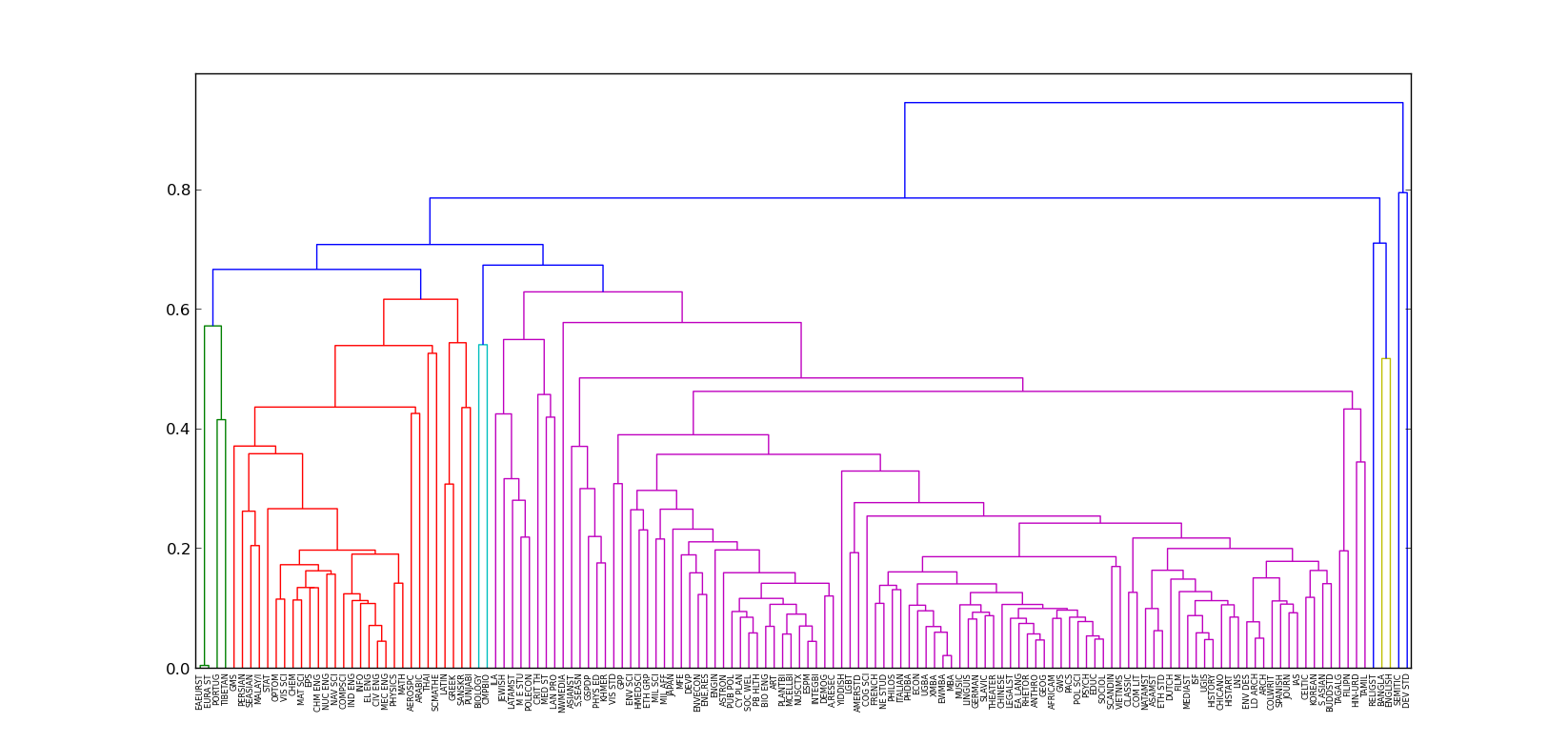
We wanted to determine which departments and courses were most similar based on the words provided in the descriptions. To find similar departments, the text from each description grouped based on the department. To do the clustering we used the hierarchical clustering SciPy library. For both clustering procedures, a term-frequency matrix is built using the [textmining Python package](https://pypi.python.org/pypi/textmining/1.0). Next, we built a linkage matrix. Using the following code:

linkage = hierarchy.linkage(feature\_vector[1::], method='complete', metric='cosine')

We used the complete clustering method and the cosine distance for the distance. We generated a dendrogram from the results of the linkage matrix, which shows department clustering.

The dendrogram reveals which departments clustered together first. Some of the first departments to cluster together are East European Studies (EAEURST) and Eurasian Studies (EURA ST), Civil Engineering (CIV ENG) and Mechanical Engineering (MECH ENG), Environmental Science, Policy and Management (ESPM) and Integrative Biology (INTEGBI), Evening and Weekend Master's in Business (EWMBA) and Master’s in Business Administration (MBA), Social Welfare (SOC WEL) and Public Health (PB HLTH),

Plant and Microbial Biology (PLANTBI) and Molecular and Cell Biology (MCELLBIO), Anthropology (ANTHRO) and Geography (GEOG), and Sociology (SOCIOL), Education (EDUC), and Psychology (PSYCH). We also observed that most of the engineering and language departments are clustered close together.



In order to cluster individual courses we built a term frequency matrix where each row represented a course and then built a linkage matrix list of clusters using SciPy. The complete method was used again along with cosine distance using the following code:

linkage = hierarchy.linkage(feature\_vectors[1::], method ='complete',metric ='cosine')

clusters = hierarchy.fcluster(linkage, threshold, criterion='distance')

One of the challenges with using the clustering the data set was determining what threshold to use and how many clusters to generate. For this project, a threshold of .9 was used to generate 615 clusters of “similar” courses. Further work could involve humans looking through clusters to rate them to see if the courses in the clusters really seem like reasonable recommendations. For now, we have generated a dictionary that contains the courses and a list of similar courses. The average number of courses in a cluster is 5 courses, with the largest cluster having 43 courses.

**Conclusions and Potential for Further Work**

We learned that it is not easy to predict the department a course belongs to. This reflects the fact that similar topics and concepts are often covered in multiple departments, and that departments themselves span wide ranges of topics. One way to further improve the classifier would be to use term frequency and inverse document frequency to teach the classifier to weigh certain words more. Another way to improve the classifier would be to allow courses to be classified to more than one department if conditional probabilities between the top departments are very close.

We found that agglomerative clustering was a good method for exploring the data set. Using this method we discovered which departments were most similar and also found clusters of courses. Future work could include validating similar courses and using this information to improve the clusters. Also, supplementing the data set with “special topics” courses or information from a course website or syllabus would likely improve the accuracy.