Introduction

The goal of this project is to create a recommender system to
recommend courses to currently enrolled students for the
upcoming semester. The system will recommend courses to
students based on courses that students most similar to them in
prior years have taken in their next corresponding semester.

Motivation

As students rush to schedule classes for the next semester, they
are often overwhelmed with the number of options they have.
Additionally, some classes that they want to take might fill up
and students are left with little time to change their opinion. If
courses were recommended to students, they already have a list
of top courses that will be appropriate to their interests and
major and will no longer be in a situation of doubt. Another
issue at Rutgers is buses tend to be crowded during some times
over others. With this recommender system, the scheduling
office can predict what courses students can take and adjust
classroom locations and times accordingly to free up buses and
classroom congestion.

Design - Algorithm

To recommend courses, we decided to apply user-based
collaborative filtering which is a technique to predict the rating
of an item based on the ratings of most similar other users who
rated other items. In terms of courses and students, we decided
to use a weighted average of common courses and grades
among those courses as the rating of a student. We then use the
Pearson correlation coefficient to calculate the correlation
among each current student and all the historic students using
the rating of common courses taken. Since there are a great
amount of historic students and current students, this becomes
difficult to process with traditional data processing applications. Thus we implement the algorithm with big data
techniques such as Hadoop MapReduce.

Conclusion

We built a recommender system using collaborative filtering and the
Pearson correlation coefficient. Since this method required large
processing and space, it quickly overwhelmed conventional data
processing techniques. Therefore, we utilized HBase and
MapReduce to perform the large computation required and used Pig
and Spring/Hibernate to format and filter the data to transfer between
different parts of the system. The recommendations themselves
proved to be more successful in some students than others. Students
with stricter curriculums had very good recommenda
ations whereas
students with more relaxed choices of study tended to have weaker
but broader range of recommendations. Furthermore, the more
courses students took, the better the recommendations were. These
results were expected since neighbors were found using common
courses and neighbors within a stricter curriculum also had to follow
the same courses. Since this project is also part of ESS, future work
will be done to improve the algorithm.