ASEE AMERICAN SOCIETY FOR ENGINEERING EDUCATION

Paper ID #36217

Recommendation Engine using Adamic Adar Measure

Mr. Sourabh Dadapure, Sacred Heart University

I'm a Software engineer at a tech startup. I'm also an experienced full-stack developer and a data analyst. Apart from academics, I'm a public speaker, investor, mentor, and explorer. I'm also an avid reader and a guitar enthusiast, my favorite being "Atomic Habits" by James Clear. Having built technological advancements throughout my life, I also have a strong motivation to keep expanding my knowledge and share that knowledge to empower others

RECOMMENDATION ENGINE USING ADAMIC ADAR MEASURE TO MAKE COURSE ENROLLMENT EASIER

ABSTRACT

In recent years, recommendation engines have gained a lot of success on many online giant commerce and entertainment platforms.

Recommending similar products that users will like based on the user's past behavior is a challenging problem especially because of the unpredictable nature of people's likes and dislikes. It also involves a guess about the future based on something that the user has never seen which makes it that much harder to predict given people's tastes change all the time. What we can do is try to estimate those values as best as we can using the Adamic Adar measure by creating nodes and finding similarities between those nodes. Unlike most of the existing recommendation systems that use either collaborative filtering or content-based filtering to generate recommendations, this paper explores a slightly different approach by creating node pairs consisting of common neighbors but with a lower degree and calculating the Adamic Adar Coefficient of those two nodes. Adamic Adar Coefficient is a measure that is used to calculate the closeness of two nodes based on their common neighbor. This paper describes a recommendation engine built to predict similar items when a user is browsing an eCommerce, music, or movie platform based on the user's behavior. It takes in the item's features such as description, price, title, ratings, etc., and creates nodes for each word to find commonalities between those nodes. The focus of this paper is to help graduate and Ph.D. students pick a course that they will like based on features from their previous courses and their likes using the recommendation engine. It generates nodes with the highest Adamic

Adar Coefficient which will result in the courses that are close in characteristics to the currently viewed course by a student.

Adamic Adar Coefficient:

- If I and J are two nodes, the Adamic Adar Coefficient of I and J would be calculated as

  - $$AdamicAdar(i, j) = \sum_{ku \in N(i) \cap N(j)} \frac{1}{log(N(k))}$$

    Whereas N(node) is a function that returns the set of neighboring nodes

PROBLEM STATEMENT

There has been a dramatic shift in higher education in the last decade towards online education. As a result, online courses are now a core feature of most colleges and universities (Larreadmendy-Joerns & Leinhardt, 2006; Layne, Boston & Ice, 2013; Sutton & Nora, 2008). With this shift in the increased usage of online courses, the size and complexity of some problems are growing exponentially. One such problem is difficulty in choosing the right courses when it comes to online learning. Difficulties are common and often cannot be avoided during learning. But one of the most challenging obstacles students run into before even beginning their

journey is selecting the right courses or a course for the academic year/semester. Enrollment choice in online education plays a major role in students' success. There are a lot of well-documented studies that show that online retention rates of courses are lower than face-to-face retention rates (Hachey, Wladis & Wladis, 2014) and there are many factors that affect online retention:

1. Having too many options
2. Selecting a course just because it has high credits
3. Rushing the decision because the deadline to enroll is approaching

All these factors add more pressure on students, thus resulting in bad decisions and ending up not doing so well in that course or dropping out. For the online experience in choosing courses to be responsive and easier, there has to be a recommendation engine in place that will solve the confusion around selecting courses and increase the retention rate. The system needs to take into account all the students' activities and generate recommendations based on their browsing history of the course catalog. This system works best for graduate and Ph.D. students more than for undergraduate students; undergraduate course curriculums give very little flexibility for the students to choose courses of their choosing, but graduate and Ph.D. students are able to take advantage of selecting courses to their liking and this system will help them with that. Those students can take the output of the system and then discuss the next steps with their academic advisors, resulting in more productive conversations with course recommendations closer to their field of interest.

Data sources for the problem statement:

This section discusses the research done by utilizing data provided by the Office of Institutional Research from a large, urban community college in the Northeast region of the United States (Hachey, Wladis & Wladis, 2014). Enrolling approximately 23,500 students in degree programs each year, the college meets the requirements to be deemed a "large institution" (Allen & Seaman 2010). Data consists of 122-course sections out of which half of them were taught online and half of them were face-to-face.

Significant tests were performed using the Bonferroni procedure and z-scores to determine whether different types of courses in the sample had significantly different retention rates (Hachey, Wladis & Wladis, 2014).

The sample data was divided into these categories:

1. Face-to-face courses
2. Online course
3. Lower level
4. Upper level
5. Electives
6. Required courses
7. Major courses

Table 1**:** Retention rates for students in each category of course type (Hachey, Wladis & Wladis, 2014)

	Retention	n	z	p	Z-score comparisons
Face-to-face	81.01%	1107	5.46	<0.0001	Face-to-face vs online
Online	70.6%	887	n/a	n/a	n/a
Lower level	69.3%	1092	-8.16	<0.0001	Lower level vs upper level
Upper level	84.9%	902	n/a	n/a	n/a
Electives	67.7%	449	-2.79	ns	Electives vs major
Required courses	74.8%	980	-5.21	<0.0001	Required vs major
Major courses	86.0%	565	6.98	<0.0001	Electives vs major

ns- the result is not statistically significant, p - values in bold are statistically significant, and z - score to determine retention rate of the course

It is very clear from *Table 1* that online retention is lower than face-to-face retention. Since online education is growing rapidly, students are going to incline towards registering for courses on their own, resulting in more dropouts or performing poorly in class because usually, academic advisors are responsible for helping students to enroll in courses each semester. The academic advisory process is known as “the process in which advisor and advisee enter a dynamic relationship respectful of the student's concerns” (T . O'Banion, 1994). Nothing can replace

in-person academic advising but since there is a good chance that students are going to shift towards selecting courses on their own, there is a need for a solution to enhance the user experience of online course registration and bring it as close to in-person academic advising as possible.

SOLUTION:

Registration is an important step for all students. The main purpose of this research is to develop a recommendation model to make the user experience around selecting courses for their prospective semester easier for graduate and Ph.D. students.

The solution of recommending similar courses or courses that a student likes contains two parts. The first part is generating recommendations similar to the course that a student is looking at. In this part, the recommendation engine will take the course's description into consideration and generate 5 or more courses similar to the course that a student had clicked on similar to the movie/e-Commerce recommendations that we see in our daily lives. The second part consists of generating recommendations of courses by taking students' browsing history of the course catalog into account and generating a list of courses that are similar to or close to the courses that students' looked at previously.

METHODOLOGY:

Adamic Adar measure:

- It is a measure to compute the closeness of nodes based on their shared neighbors
- X and Y are two nodes
- N(one-node) is a function that returns the set of adjacent nodes to one-node
    - Adamic Adar (x,y) = Σ uεN(x)∩N(y) $1/\ log(N(u))$c

The dataset used to prove the concept of similar course recommendations is gathered from the course catalog at Berkley, 2018 provided by data.world. This data set contains 305,669 courses with its name, description, the field of study, area of study, general area, and which semester it's being taught (fall, winter, spring, and summer).

Step1:

Load the data and clean up the data if there are any null values

Pseudocode:

1. Load the data into a data frame
2. Check for null values in the dataset and replace null values with an empty string

Output:

Table 2: Cleaned data

	Name	Description	Fall	Winter	Spring	Summer	Field	Area	GenArea
0	Officer Basic Military Training	Study of world military systems and basic lead...	False	False	False	False	Aerospace Studies	Aeronautical Engineering	Engineering
1	Drugs, Health, and Society	Two hours of lecture and one hour of discussio...	False	False	True	False	Public Health	Health Sciences	Professional
2	Policy, Planning , and Evaluation of Health Pr...	Three hours of lecture/discussion per week. T...	True	False	True	False	Public Health	Health Sciences	Professional
3	Cognitive Science C1 Molecularand Cell Biology...	The course will survey the field of the human...	True	False	True	False	Public Health	Health Sciences	Professional
4	Officer Advanced Military Training	Four weeks advanced officer training conducted...	False	False	False	False	Aerospace Studies	Aeronautical Engineering	Engineering

Step 2:

Convert the raw description into a matrix of TF-IDF(Term Frequency - Inverse Document Frequency)

TF-IDF is a measure used for extracting core words (i.e., keywords) from documents, calculating similar degrees among documents, deciding search ranking, and so on (S. Kim & J. Gil 2019). Term frequency in TF-IDF is the frequency of a word in a given document of documents, which means the words with higher TF have importance in a given document, and IDF(Inverse Document Frequency) is a measure of words that are rare in a document, it is basically the opposite of Term Frequency. TF-IDF is the product of TF and IDF

Pseudocode:

1. Calculate TF(Term Frequency) of the description field of all the courses
2. Calculate IDF(Inverse Document Frequency) of the description field of all the course
3. Calculate TF-IDF by multiplying TF and IDF

   TF - IDF = Term Frequency * Inverse Document Frequency

Step 3:

Create a graph by adding name, description, area of study, the field of interest as nodes and with edges as course description (relationship between course and description), course area (

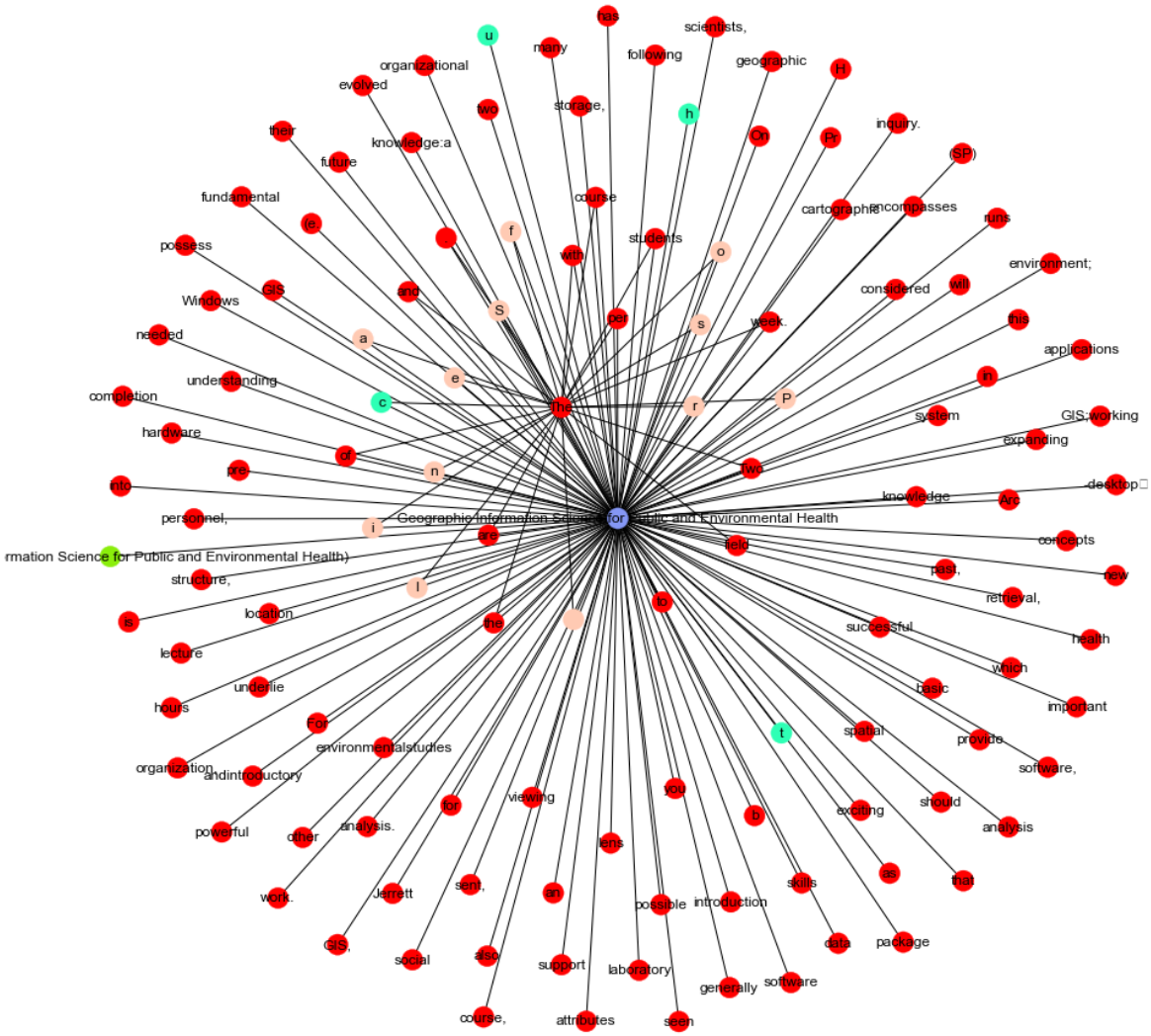
relationship between course and area), course field (relationship between course and field of the course)

Pseudocode:

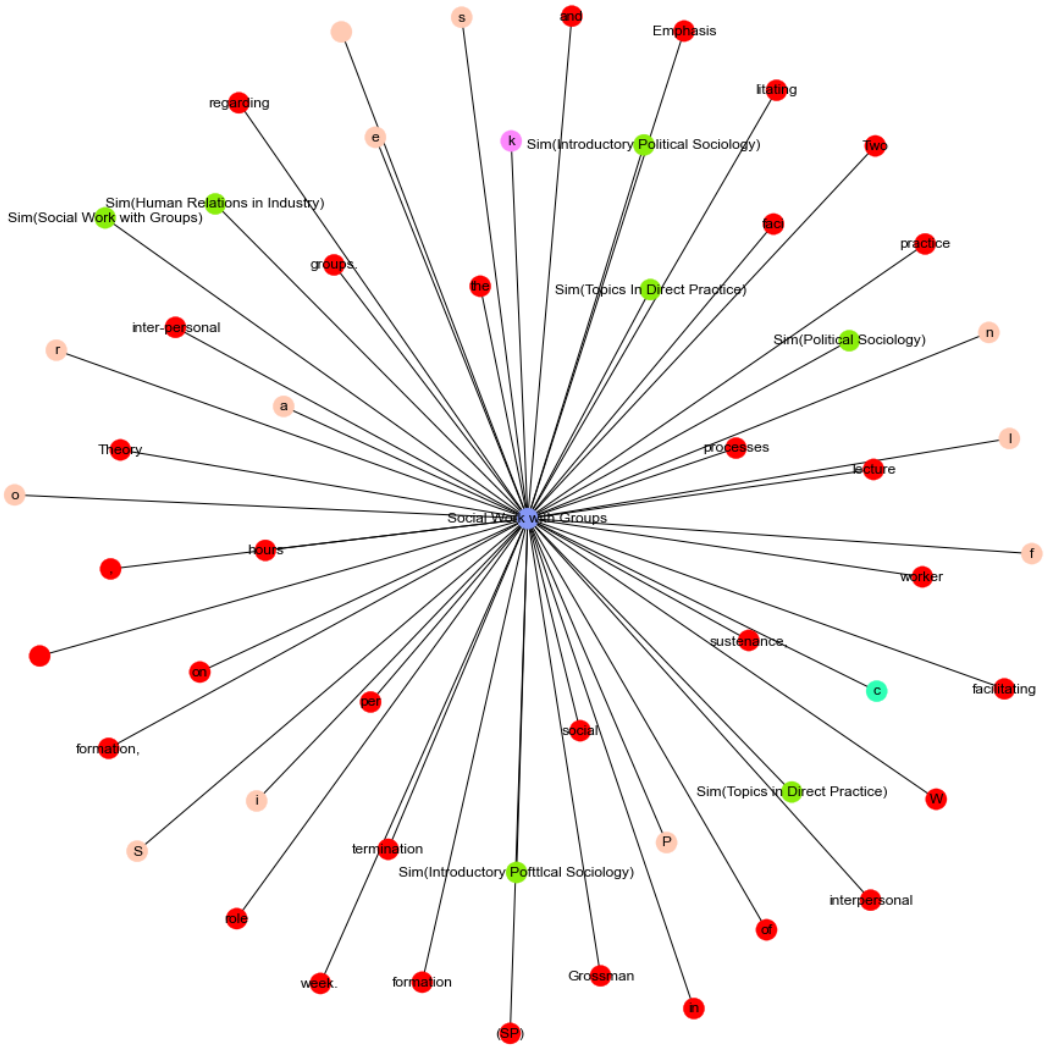
1. Graph by iterating through the dataset and for each row:
    a. Add names as a node
    b. Description as a node
    c. Area of the course as a node
    d. Field of the study as a node
    e. Course area as an edge to the graph
    f. Course field as an edge
    g. Course description as another edge
2. To see a sample node, enter a course name and it will output all of its nodes and edges and similar courses related to it

Examples:

Graph 1: Edges of the node “Geographic Information Science for Public and Environmental Health” course



Graph 2: Edges of the node “Social Work with Groups” course



Step 4: Recommender function

The recommender function explores all the neighbors of the target course, for each neighbor it discovers the course that shares a common node with the target course and generates the Adamic

Adar coefficient of that particular node. Finally, it calculates the Adamic Adar coefficient of these similar courses and returns the top 5 courses that have the highest Adamic Adar coefficient to the target course

Pseudocode:

1. For each node, get all the neighboring nodes and calculate the logarithm degree and 1 over the logarithmic degree of that node based on the number of nodes. For example, if a node has 3 neighbors its logarithmic value becomes log(3) which is 0.3010, and 1 over logarithmic value becomes 1 / log (3) which is equal to 3.32222
2. Store all the 1 / log(i,j) values of each node
3. Consider all the node pairs and capture the common nodes between these two node pairs
4. Adamic Adar coefficient is calculated by adding the 1/log(i,j) values of each common node within a given node pair
5. Return the top 5 nodes with the highest Adamic Adar coefficient

Results:

Example input 1:

When a student clicks on the “Introduction to Architectural Administration” course to get more details, the recommender function gets executed and generates 5 courses that are similar to the “Introduction to Architectural Administration” course under the similar courses section after the description of the course on the course catalog webpage

Example output 1:

Figure 1: Courses similar to “Introduction to Architectural Administration”

```
Recommendation for 'Introduction to Architectural Administration'
Introduction to the Practice of Architecture            9.709946
Special Topics: Social and Cultural Bases of Design     6.354438
Twerm-eth Slavic Literary Criticism                     6.011179
Three 1 H-hour lectures per week                        5.587591
Elementary Polish                                       5.232692
```

Example input 2: When a student clicks on the “Individual Study for Doctoral Students” course to get more details, the recommender function gets executed and generates 5 courses that are similar to the “Individual Study for Doctoral Students” course under the similar courses section after the description of the course on the course catalog webpage

Example output 2:

Figure 2: Courses similar to “Individual Study for Doctoral Students”

```
Recommendation for 'Architectural Administration'
Architectural Practice                  6.421803
Specifications                          6.058170
Acoustics in Architectural Design       4.759905
Social and Cultural Factors             3.494051
Exploration Toward a Theory of Form     3.261488
```

Conclusions:

Course selection as an administrative task is a tedious process that students face every semester. During the enrollment period, students need help in selecting courses online and a lot of support to make the right choices because their course selection is directly correlated to their performance. In this paper, a solution has been presented to make it easier for graduate and Ph.D. students to enroll in the right courses by recommending courses similar to the course that they are looking at when browsing the course catalog. With these recommendations, students can generate a list and be more prepared before talking to their advisors.

Future work:

In the future, this recommendation system is going to be more advanced and it's going to generate recommendations based on students' interests, goals, and history of previous semesters' course choices. Recommender systems help companies like Amazon, Netflix, and e-Commerce websites to increase their customer base and revenue; there is a huge potential to use the same to help students (expanding from graduate and Ph.D. to all students) for example aiding them in selecting food options on campus, dorm/apartments, and other aspects of a student's college life beyond courses.

References

R. Meteren and M. Someren (2000). *Using Content-Based Filtering for Recommendation.* Retrieved from http://users.ics.forth.gr/~potamias/mlnia/paper_6.pdf

I. Allen & J. Seaman. (2010). *Class Differences: Online Education in the United States*. Retrieved from https://files.eric.ed.gov/fulltext/ED529952.pdf

M. Al-Sarem (2017). *Solving Course Selection Problem by a Combination of Correlation Analysis and Analytic Hierarchy Process.* Retrieved from http://ijece.iaescore.com/index.php/IJECE/article/view/8722

S. Kim & J. Gil (2019). *Research paper classification systems based on TF-IDF and LDA schemes.* Retrieved from https://hcis-journal.springeropen.com/articles/10.1186/s13673-019-0192-7#:~:text=2)

Hachey, Wladis & Wladis, (2014). *The Role of Enrollment Choice in Online Education: Course Selection Rationale and Course Difficulty as Factors Affecting Retention.* Retrieved from https://files.eric.ed.gov/fulltext/EJ1043163.pdf