

Predicting Student Success in Courses via Collaborative Filtering

Ali Cakmak¹

Accepted : 11/03/2017 Published: 30/03/2017

Abstract: Based on their skills and interests, students' success in courses may differ greatly. Predicting student success in courses before they take them may be important. For instance, students may choose elective courses that they are likely to pass with good grades. Besides, instructors may have an idea about the expected success of students in a class, and may restructure the course organization accordingly. In this paper, we propose a collaborative filtering-based method to estimate the future course grades of students. Besides, we further enhance the standard collaborative filtering by incorporating automated outlier elimination and GPA-based similarity filtering. We evaluate the proposed technique on a real dataset of course grades. The results indicate that we can estimate the student course grades with an average error rate of 0.26, and the proposed enhancements improve the error value by 16%.

Keywords: Collaborative Filtering, Educational Data Mining, Student Success Estimation, Outlier Elimination

1. Introduction

In colleges, students elect courses based on a variety of motivations. Some may choose a course because they are interested in the covered material, while some others may take a course just because s/he likes the way the course instructor teaches. More often, college students tend to choose courses in which they expect to get good grades, whenever they are given chance in the form elective courses. Hence, guiding students about their expected future course performance may allow them to make informed choices. In addition, such guidance may be invaluable for academic advisors and instructors as well. Academic advisors may suggest certain courses to their students in a personalized manner. Moreover, instructors may design the content of a course according to the academic level of the current set of students in the class at the very beginning of the semester, before it gets late in the middle of the semester. In addition, potential struggling and low-performing students may be identified at an early stage, and these students may be offered additional help in the form of extra tutorial and practice sessions. Hence, providing automated means of estimating future course grades of students may greatly enhance the learning experience of students at colleges.

Students' expected grade perception for a future course is usually shaped by (i) what they hear from other fellow students about the course, as well as (ii) how well they did in similar courses in the past. Such a reasoning scheme is actually the basis of what is called *collaborative filtering* [1]. Collaborative filtering is a machine learning technique that is usually employed in recommender systems where users are recommended movies to watch, songs to listen, books to read, etc. based on their past ratings of movies, songs, books, etc. The two main steps of collaborative filtering are (a) locating similar users, and (b) computing a weighted sum of ratings for each item that will be recommended to a user.

In this paper, we employ an enhanced collaborative filtering-based method to predict the future course grades of students. Here, we consider students' past course grades as the "ratings" that they give to the corresponding courses. Hence, the underlying assumption we make is that if a student gets an A from a course, we consider that the student liked the course a lot. Similarly, a grade F is considered to indicate that the student did not like the course at all. By comparing the past grades of students, we find "similar" students. In order to compute similarity, we employ different measures such as Pearson correlation [2], Euclidean distance [3], and Jaccard measure [4]. We also employ two different flavors of collaborative filtering, namely, user- and item-based [1]. Item-based collaborative filtering is usually preferred for large and sparse databases (such as shopping web sites like Amazon.com) due to its efficiency.

As part of this study, we further enhance the collaborative filtering in the context of course grade problem. More specifically, our proposed enhancements are two-folds: (i) among the courses that a student has already taken, we compute the set of outlier courses, and prevent them from contributing to the computation of the student's future grades, (ii) we compute a running average of grades for a student, and make sure that the student is not getting recommendations from students with significantly higher or lower GPAs than herself.

We evaluate the proposed techniques on a real data set of around 55,000 course grades obtained from Istanbul Sehir University, where student names and ids are obfuscated. Our results show that the proposed methods predict student success in future courses with an average error rate of 0.26, and the proposed enhancements improve the average error rate by 16%. We show that the user-based collaborative filtering provides slightly better accuracy values (6%) than the item-based collaborative filtering. However, in terms of the running time performance, item-based collaborative filtering outperforms user-based collaborative filtering by several folds. Among different similarity measures, our experimental results show that Euclidean distance performs better than the other similarity measures.

We implemented a standalone and web-based proof-of-concept

¹ Department of Computer Science, Istanbul Sehir University,
Kusbakisi Cad. No: 27, 34662, Uskudar, Istanbul, Turkey
Email: alicakmak@sehir.edu.tr

versions of the above-described future course grade prediction system. It will soon be made available to Istanbul Sehir University community in the first stage, and may later be offered to the students at other universities as well in the second stage.

The rest of the paper is organized as follows. In the next section, we discuss the related work. Section 3 describes our methodology. In Section 4, we evaluate the proposed techniques on real-life course grade dataset. Section 5 concludes with pointers for future work.

2. Related Work

The general field of educational data mining for predicting student success has drawn considerable attention from the research community. [5, 6, 7] provide recent surveys of different approaches in the field.

Luo et al. [8] proposes collecting comments from students after each lecture, and analyzing these comments automatically to predict student grades. They convert comments of students into word vectors and apply a neural network-based technique to estimate the student success. Although their results are promising, collecting comments from students after each class do not seem practical and scalable especially for large classes.

Zacharis [9] analyzes the online learning management system's usage logs to predict the student success for a course. To this end, they identified 29 online activities, and track students for these activities. Using stepwise multivariate regression, they have shown that activities such as reading and posting messages, quiz attributes, etc. are the most indicative student success factors.

Elbadrawy et al. [10] construct course specific regression models to predict future course grades, and compare it to matrix factorization-based models as baseline. They show that the best method achieves 0.51 error rate on the average, which is a lot higher than the error rate that our study achieves.

Geiser and Santelices [11] challenge the general assumption that average high school grades might not be reliable for college admission decision. On a real-life data set, through a correlation study, they show that high school grade point average is the best predictor not only for freshman courses, but also for the whole four-year courses at college.

Huang and Fang [12] compare four different mathematical models such as the multiple linear regression model, the multilayer perception network model, etc. to predict the final exam grades of students in a particular engineering course. As input, they use past grades from math courses, students' GPA, and three midterm results that are taken as part of the same course before the final exam during the semester.

Dekker et al. [13] aim to predict student drop-out based on the first semester grades of Electrical Engineering students. They show that explanatory techniques such as decision trees achieve reasonable performance, while pointing out most indicative factors in students' dropping out process.

Kabakchieva [14] casts the student course success prediction problem as that of a classification. In addition to past grades, they also employ other factors, e.g., admission scores, as features. They compare five different classification methods which provide around 52-56% accuracy.

Loll and Pinkwart [15] employ collaborative filtering to score student assignment solutions based on peer evaluation, and show that it is as effective as manual assessment.

Finally, similar to our work, Ray and Sharma [16] propose using collaborative filtering to predict student course grades. Our work

differs from [16] in the following aspects:

- i. In [16], the authors only focus on a relatively narrower problem of estimating student success in elective courses. On the other hand, in this paper, we study the more general problem of predicting all future course grades as soon as a student completes just the first semester at college.
- ii. We extend the collaborative filtering with new techniques specific to this problem, such as, outlier detection and elimination, and grade point average-based filtering.
- iii. We study the effect of student seniority on prediction accuracy by providing a semester-based evaluation.

3. Proposed Methodology

We consider the problem of student grade prediction for future courses as a recommendation problem, and use an extended version collaborative filtering to solve it.

Def'n (Recommendation Problem): Given a database D of people, their ratings for a number of items (movie, product, book, etc.), a person P from D , and a number N , the recommendation problem is to compute the top- N items that person P would most likely be interested in along with their estimated ratings.

We translate the above general recommendation problem definition into the *student course grade estimation problem* by doing the following mapping: users \rightarrow students, items \rightarrow courses, and item ratings \rightarrow courses grades.

Collaborative filtering has two subtypes: (i) user-based collaborative filtering, and (ii) item-based collaborative filtering. We next summarize these two types of collaborative filtering techniques, and then present our extensions to the collaborative filtering.

3.1. User-based Collaborative Filtering

Assume that the recommendation will be done for a given user P from a ratings database D . The user-based collaborative filtering has two main steps:

1. *Locating users similar to P:* In this step, pairwise similarities between P and every other user in D are computed. To this end, the algorithm is oblivious to the employed similarity computation method. Hence, any similarity scheme may be plugged in here. In this work, we exploit three commonly used similarity measures, namely, Pearson Correlation [2], Euclidean Distance [3], and Jaccard Measure [4].
2. *Computing Recommendations:* For each item that is not rated by user P , a weighted average score is computed. This score represents the estimated rating of P for that item. Each user U in D contributes to this score in proportion to his/her similarity to P . More specifically, for a particular item I , let the rating of user U for I be $rating(U, I)$, and the similarity between U and P be $sim(U, P)$. Then, the contribution of U to the score computed for I is $sim(U, P) * rating(U, I)$. After summing up the contribution of each user in D , the resulting score is normalized by the total similarity of users in D to P , i.e., $\sum_{U \in D \text{ and } P \neq U} sim(P, U)$. Mathematically, the final estimated rating is expressed as in Equation 1.

$$rating(P, I) = \frac{\sum_{I \in D, U \in D \text{ and } P \neq U} sim(P, U) * rating(U, I)}{\sum_{U \in D \text{ and } P \neq U} sim(P, U)} \quad (1)$$

We give an example.

Table 1. Course Grade Estimation for a Student, Ahmet, using User-based Collaborative Filtering

<i>Student</i>	<i>Similarity</i>	<i>ENGR 211</i>	<i>Sim * ENGR 211</i>	<i>CS 340</i>	<i>Sim * CS 340</i>	<i>EECS 468</i>	<i>Sim * EECS 468</i>
Seda	0.92	A (4.0)	3.68	D (1.0)	0.92	B (3.0)	2.76
Osman	0.45	B+ (3.3)	1.49	A (4.0)	1.80	B+ (3.3)	1.49
Erdem	0.64	C (2.0)	1.28	-		B- (2.7)	1.73
Hasan	0.82	A- (3.7)	3.03	C (2.0)	1.64	A (4.0)	3.28
Esma	0.35	F (0.0)	0	B- (2.7)	0.95	-	
Total Score			9.48		5.305		9.25
Total Similarity			3.18		2.54		2.83
Total Score/Total Similarity			2.98 (B)		2.09 (C)		3.27 (B+)

Example 1: Consider the sample case illustrated in Table 1. Assume that we would like to estimate the future grades of a student, called Ahmet, for courses ENGR 211, CS 340, and EECS 468. Further assume that there are five other students in our database with their grades for these courses (if they have taken them). Column 1 in Table 1 lists these students. Column 2 contains the pairwise similarity score between Ahmet and each student in the database. Columns 3, 5, and 7 list the grades of each student for courses ENGR 211, CS 340, and EECS 468, respectively. Note that this example uses the grading scheme out of 4 expressed as letter grades where each letter grade is converted to its corresponding number value (i.e., A: 4.0, A-: 3.7, B+: 3.3, B: 3.0, B-: 2.7, C+: 2.3, C: 2.0, C-: 1.7, D+: 1.3, D: 1.0, D-: 0.7, F: 0). Column 4 lists $sim(Student, Ahmet) * rating(Student, ENGR 211)$ where rating is mapped to each student's course grade. Column 6 and 8 contain similar information for courses CS 340 and EECS 468, respectively. Three rows at the bottom of Table 1 list the summation of the above scores, summation of user similarities, and the normalized scores as in Eq. 1 (i.e., predicted grade), respectively. Please note that not all students in the database have taken all of these three courses. For instance, Erdem has not taken CS 340, and Esma has not taken EECS 468 yet. Therefore, the corresponding cells are empty. In addition, if a student has not taken a course, since s/he does not contribute to the prediction of the grade for that course, his/her similarity to Ahmet is not included in the "Total Similarity" value for that course. For instance, since Esma has not taken EECS 468, her similarity value to Ahmet (i.e., 0.35) is not included in the "Total Similarity" row under the last column. Finally, the estimated numeric grades may be mapped to the closest letter grade.

3.2. Item-based Collaborative Filtering

In item-based collaborative filtering, the underlying assumption is that the database is quite large and sparse, and changes happen to this database at slow rates relative to its size. As the name suggests, in item-based collaborative filtering, the focus is on items (i.e., courses), rather than users (i.e., students). Hence, instead of locating similar users, this time, similar items are computed, and recommendation is done based on these item similarity values. Item-based collaborative filtering has two main steps too, as summarized next:

1. *Locating similar items:* In this step, for each pair of items (I_i ,

I_k), a similarity score $sim(I_i, I_k)$ is computed. Again, in this step, any similarity metric may be used. In this work, we use the same set of similarity measures as in user-based collaborative filtering. The computed pairwise item similarities are stored in a matrix where rows and columns are items, and cells store the corresponding similarity values.

2. *Computing Recommendations:* For each item that is already rated by the target user P in D, the list of similar items with their similarities are obtained from the item similarity matrix which is computed in Step 1. Let this set be T. Then, for each item that is not rated by P, a weighted average score is computed. Each item I in T contributes to this score in proportion to its similarity to items that are rated by P. More specifically, let I_1 be an item that P already rated with score $rating(P, I_1)$. Let I_2 be an item that P has not rated yet. Then, the contribution of I_1 to the score computed for I_2 is $sim(I_1, I_2) * rating(P, I_1)$. After summing up the contribution of each item to the overall score, the resulting value is normalized by the total similarity of items, i.e., $\sum_{I_1 \text{ rated by } P} sim(I_1, I_2)$. Mathematically, the final estimated rating is expressed as in Equation 2.

$$rating(P, I_2) = \frac{\sum_{I_1 \text{ rated by } P} sim(I_1, I_2) * rating(P, I_1)}{\sum_{I_1 \text{ rated by } P} sim(I_1, I_2)} \quad (2)$$

We give an example.

Example 2: As in Example 1, assume that we would like to estimate Ahmet's future grades, but this time using item-based collaborative filtering. Table 2 illustrates the steps to this end. Note that, in Table 2, no users (i.e., students) are involved at all. Column 1 lists the courses that Ahmet has already taken, for which grades are listed in column 2. Columns 3, 5, and 7 list the similarity of ENGR 211, CS 340, and EECS 468, respectively, to each course that Ahmet has taken. Columns 4, 6, and 8 list the grade of each course taken by Ahmet multiplied by their similarity to ENGR 211, CS 340, and EECS 468, respectively. Bottom two rows list the raw total and normalized total scores as in Eq. 2 (i.e., predicted grade), respectively.

3.3. Extensions to Collaborative Filtering

In this work, we extend the collaborative filtering in the following dimensions:

Table 2. Course Grade Estimation for a Student, Ahmet, using Item-based Collaborative Filtering

Course	Grade	ENGR 211	Grade * ENGR 211	CS 340	Grade * CS 340	EECS 468	Grade * EECS 468
MATH 101	A (4.0)	0.55	2.20	0.40	1.60	0.05	0.08
ENGR 251	B+ (3.3)	0.20	0.66	0.10	0.33	0.30	0.10
CS 360	C (2.0)	0.80	1.60	0.95	1.90	0.45	0.86
Total Score		1.55	4.46	1.45	3.83	0.80	1.03
Total Score/Total Similarity			2.88		2.64		1.29

i. *Elimination of Outliers*: Usually, the grades that students get in different courses are consistent and not very distant from each other. However, occasionally, a student may get unexpectedly low or high grades in comparison to his/her other grades. We call such inconsistent grades “outliers”. Including outliers in the collaborative filtering steps may lower the accuracy, as the future course grades are expected to be consistent with the majority of past grades. Hence, to alleviate this issue, in collaborative filtering process, we introduce two substeps: (i) locating outliers, and (ii) eliminating them from the scoring process that is described in the previous section. In order to locate the outliers, we assume that student grades follow normal distribution [17]. That is, we compute the mean and standard deviation of the currently known grades in a student’s transcript, and eliminate those grades that are two standard deviation away from the mean. Mathematically, we only employ those grades that satisfy Equation 3 for scoring purposes.

$$mean - 2 * std \leq grade \leq mean + 2 * std \quad (3)$$

Here, the drawback is that this approach does not fully accommodate outliers in future course grades. However, by definition, outliers are expected to happen rarely. Therefore, we focus on the majority of courses. In the experimental evaluation section, we show that this assumption indeed works in practice.

ii. *Disallowing High GPA Differences*: In real life, students usually get advice about courses that they plan to take from students alike. That is, a poorly performing student do not feel comfortable with the advices of a star student, as s/he often thinks that such a student has extra skills that s/he himself/herself does not have. Hence, such advices would not apply to him. Similarly, academically brilliant students usually do not take advices from poorly performing students, as they may think that poorly performing students may not take courses serious or they may not have the self-discipline to study properly. Hence, often unconsciously, students talk to (or, even choose friends from) other students who are academically not very different from themselves. In order to reflect this social phenomenon in our methodology, we make sure that future grade prediction for a student is performed only based on grades of other students with similar academic performance. We use GPA as the proxy indicator of students’ academic performance, and disallow considerable GPA differences between students in the recommendation process. That is, given two students s_1 and s_2 , we make sure that

$$|GPA(s_1) - GPA(s_2)| \leq \delta \quad (4)$$

where δ is a threshold whose value is experimentally learnt

from the data.

Other than the above enhancements, we also incorporate one additional enhancement as follows: If a course C is similar to another course with grade F in the training data (i.e., a student’s past course grades), then we set C’s estimated grade to F as well. This enhancement provides very slight improvement in accuracy. Hence, we omit it from the discussion in the rest of the paper.

4. Results and Discussion

In this section, we quantitatively evaluate the proposed techniques from different perspectives. We first discuss the dataset and the evaluation metrics, and then discuss individual experimental results with their implications.

4.1. Dataset

For experimental evaluation, we use a real student course grade dataset that is obtained from Istanbul Sehir University. We are given the dataset after all student personal information (e.g., name, id, nationality, class, etc.) are removed. The raw dataset contains 55,475 rows of course grades spanning between the years of 2010 – 2015. Several courses are designed to be pass/fail courses, that is, there is no letter grade for such courses, and students either pass or fail. There are also several courses with incomplete grades. We eliminated all these rows (6,128 course grade records in total). Furthermore, in order to make sure that there is at least one semester of data for training and one semester of data for testing, we eliminated students with less than 10 courses completed (i.e., 606 additional rows eliminated). After all these filtering steps, the final dataset contains 48,741 rows of course grades that belong to 2,524 distinct students.

4.2. Metrics

In order to evaluate the accuracy, we employ *mean absolute error* (MAE) metric [18]. Formally, MAE is defined as follows.

$$MAE = \frac{|actual_grade - estimated_grade|}{actual_grade} \quad (5)$$

Note that for courses where the actual grade is F (i.e., 0), Equation 5 may not be used. Therefore, for courses with grade F, we compute a separate error metric as follows.

$$F_{error} = average(estimated_grade) \quad (6)$$

For comparing different approaches, we combine MAE and F_{error} in a weighted manner to come up with a single value as the basis of comparison. More specifically, we do the following:

- In an experiment, we choose the run with the poorest result as

the baseline. Then, we mark MAE of this run as $MAE_{baseline}$, and F_{error} of the same run as $F_{error-baseline}$.

- Then, for all the runs of the same experiment, we compute the following ratios:

$$MAE_{ratio} = MAE / MAE_{baseline} \quad (7)$$

$$F_{error-ratio} = F_{error} / F_{error-baseline} \quad (8)$$

- Finally, for each run, we combine these ratios in a weighted manner. That is, in our data set, approximately, 10% of course grades are F, and 90% of grades are none-F. Hence, we combine these ratios based on their frequency in the dataset as follows:

$$Combined_{error} = 0.9 * MAE_{ratio} + 0.1 * F_{error-ratio} \quad (9)$$

4.3. Experiments

In this section, we present major experimental results and the associated observations that we made. The proposed techniques are implemented in Python 2.7.9, and experiments are run on an iMac machine with 3.2 GHz i5 CPU and 16 GB memory. For all runs, we employ Euclidean distance similarity metric and the user-based approach unless noted otherwise, since they provide the best $combined_{error}$ as shown in sections 4.3.4 and 4.3.5.

4.3.1. Learning GPA Difference Threshold (δ)

For the second enhancement discussed in the previous section, a threshold value (i.e., δ) needs to be determined to enforce Equation 4. We learn the value of delta threshold experimentally from the data. More specifically, we experiment with different values of delta, and choose the one that provides the best accuracy value. Figures 1, 2, and 3 show MAE , F_{error} , and $Combined_{error}$ values, respectively, for different values of delta.

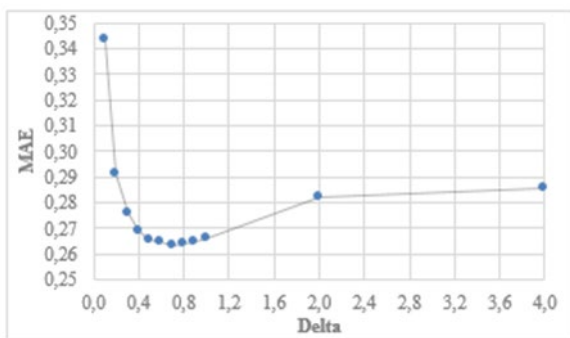


Figure 1. MAE vs. delta (δ)

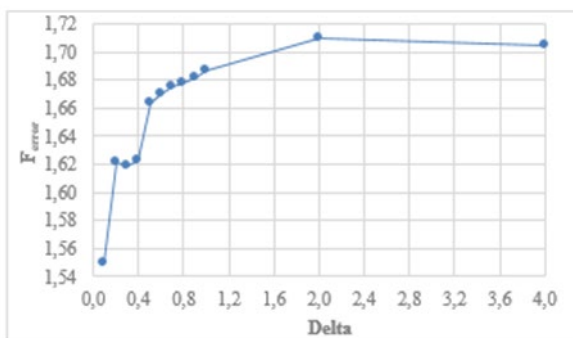


Figure 2. F_{error} vs. delta (δ)

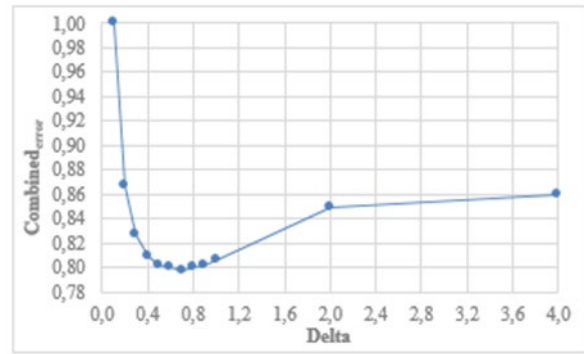


Figure 3. $Combined_{error}$ vs. delta (δ)

Observation 1: MAE is the lowest when delta = 0.7, while F_{error} reaches its minimum value when delta = 0.1. $Combined_{error}$ is minimum when delta = 0.7.

Since we target to minimize the $combined_{error}$, we use delta = 0.7 for the rest of the experiments

4.3.2. Semester-based Evaluation

In this section, we perform a semester-based evaluation. That is, we simulate the college life of a student starting from the first semester. More specifically, starting from $N=1$, we use course grades from first N -semesters to predict the course grades for the remaining future semesters (N is 8 at maximum, as there are 8 semester in the current dataset). We repeat the same experiment while incrementing N by 1 in each experiment up until 7 which represents using first 7 semesters to predict the course grades for the last semester (i.e., semester 8). Figure 4 shows MAE, F_{error} , and $Combined_{error}$ values for different values of N .

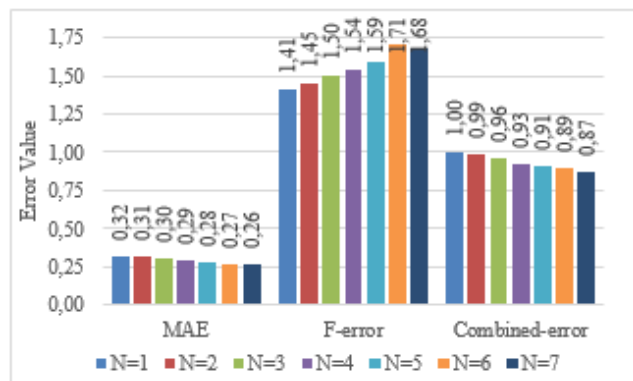


Figure 4. Change of MAE, F_{error} , and $Combined_{error}$ when using first- N semesters to predict the remaining future semesters

Observation 2: As more past course grades for a student become available, in general, mean absolute error and $combined_{error}$ decrease steadily. F_{error} gradually increases, as it becomes more and more difficult to predict courses with F grades owing to the increased diversity of grades over the time in a student's currently known grades.

In all other experiments, we provide results assuming that $N=7$, where we have the most past course grades to predict future course grades.

4.3.3. Accuracy Contribution of Enhancements, and Comparison with the State of the Art

In this section, we compare our approach to that of Ray and Sharma [16], which is referred as RS in short. RS approach represents the standard collaborative filtering. For comparison, we first run the

experiment with standard collaborative filtering (labelled as “RS”),

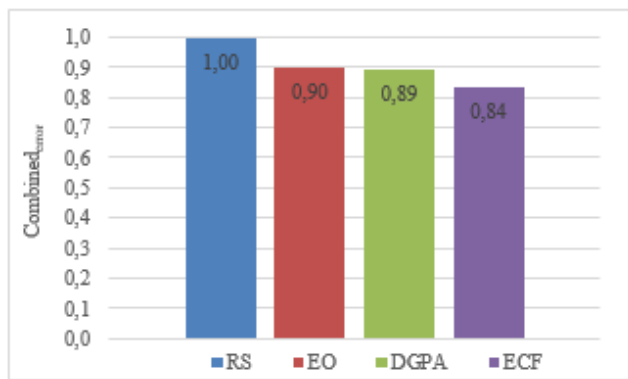


Figure 5. Comparison of enhancements to the baseline

then repeated the experiment by turning on enhancement 1 only (labelled as “EO”), enhancement 2 only (labelled as “DGPA”), and both enhancements 1 and 2 (labelled as “ECF”). In each case, we compute combined_{error} as in Equations 5, 6, and 9. Figure 5 shows the resulting Combined_{error} values, and Figure 6 shows individual MAE and F_{error} values.

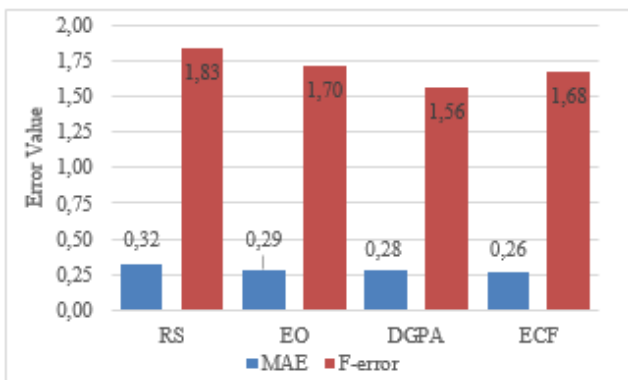


Figure 6. MAE and F_{error} for enhancements

Observation 3: Outlier elimination enhancement (EO) by itself improves combined_{error} by 10% in comparison to RS approach.

Observation 4: Disallowing considerable GPA difference enhancement (DGPA) by itself improves combined_{error} by 11%.

Observation 5: When both enhancements combined, together, they improve combined_{error} by 16% in comparison to RS approach.

The above observations show that the proposed enhancements make a significant difference over baseline by themselves. In addition, when they are combined, the overall improvement is better than using any of the enhancements alone. This shows that the proposed enhancements are complimentary to each other covering different aspects.

4.3.4. User-based vs. Item-based Collaborative Filtering

In this section, we perform a comparative study on user- and item-based collaborative filtering approaches. We first perform an accuracy-based comparison. Figures 7 and 8 compare MAE, F_{error}, and Combined_{error}, respectively. Figure 9 provides the running time measurements for both approaches.

Observation 6: User-based collaborative filtering provides somewhat better error values (6% improvement in Combined_{error}). The main gain here comes from considerable improvement in F_{error} (13% improvement). There is also a slight improvement in MAE (2% improvement) that further contributes to Combined_{error} as

well.

Observation 7: Item-based collaborative filtering runs 5 to 10 times faster than user-based collaborative filtering.

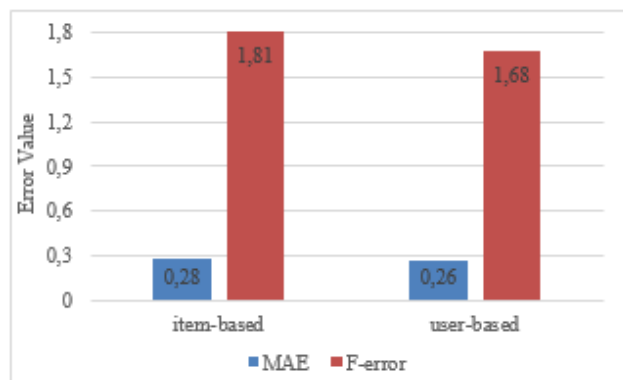


Figure 7. MAE and F_{error} for user- and item-based approaches

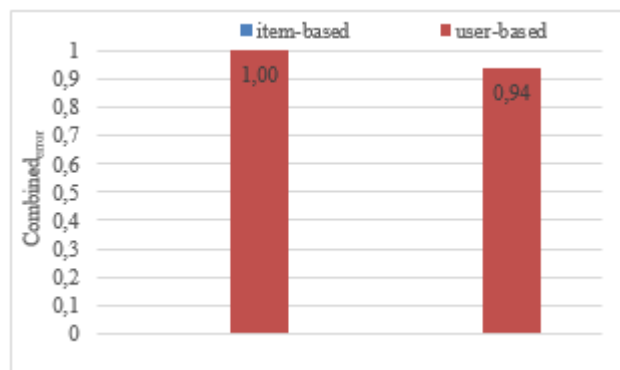


Figure 8. Combined_{error} for user- and item-based approaches

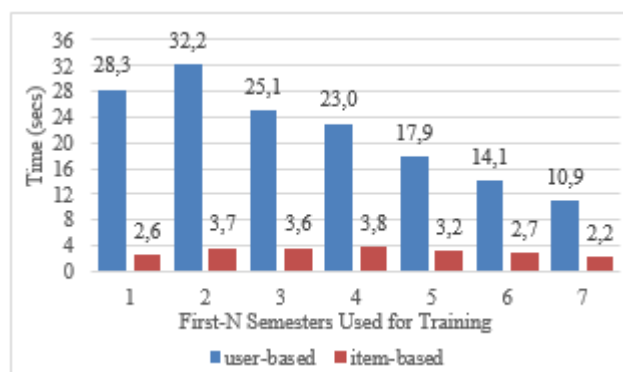


Figure 9. Running time comparison of user- and item-based approaches

The above observations are consistent with the expectations. First, in terms of error rate, item-based collaborative filtering performs nearly well as user-based collaborative filtering with slight decrease in accuracy. In return for this small compromise in accuracy, item-based approach provides dramatic improvement in running time. This is mostly due to the fact that item-based approach computes item similarity matrix once, and reuses it for the rest of the runs.

4.3.5. Comparison of Similarity Measures

In this section, we compare the accuracy values for three different similarity measures, namely, Euclidean distance, Pearson

correlation, and Jaccard measure. Figure 10 compares combined_{error} values, and Figure 11 shows individual MAE and F_{error} values.

Observation 8: Euclidean distance provides the best overall accuracy value (in terms of combined_{error}), while Jaccard distance has the worst accuracy.

Observation 9: In terms of predicting F-grades, Jaccard measure achieves the least error amount.

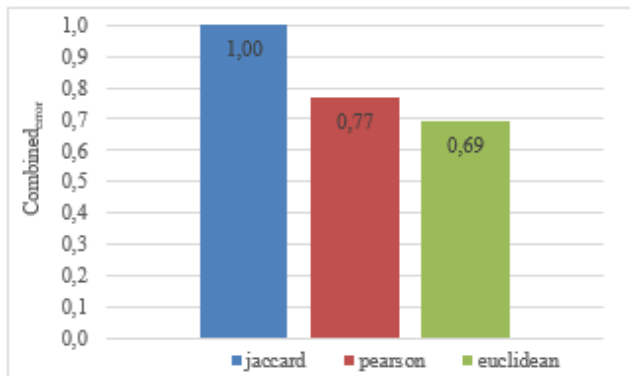


Figure 10. Combined_{error} comparison of similarity measures



Figure 11. MAE and F_{error} for individual similarity measures

Since Jaccard measure is the most simplistic measure among the three compared measures, it is expected that the overall error value (i.e., combined_{error}) is the highest for it. This is mostly because it cannot capture the intrinsic relationship patterns between past and future course grades. The fact that Jaccard provides the lowest error value for F-grades may at first seem inconsistent with the previous observation. However, that is not the case. The underlying reason is that since Jaccard measure only cares about the ratio of commonly taken courses by two students to the union of courses taken by these students individually, in many cases, the resulting similarity values are very small. Since similarity values are used as coefficient during estimated grade computation, the majority of the resulting grades are close to zero (i.e., F). Hence, it more accurately estimates F-grades, while suffering during the estimation of higher-valued grades for the same reason.

5. Conclusion and Future Work

Estimating student grades in future courses may be invaluable for three important use cases: (i) providing guidance for students so that they can make informed choices and planning regarding the elective courses as well as the order and timing of the mandatory courses, (ii) enabling instructors to tailor their class organization

and content according to the academic level of the audience in a particular class, and (iii) helping instructors identify struggling students earlier so that they may arrange extra help for those students in the form of out-of-classroom activities, tutorials, peer-help, etc. In this paper, we propose an extended collaborative filtering approach, and evaluate it on a real course grade dataset. We show that our approach may estimate future student grades with improved accuracy in comparison to standard collaborative filtering approach.

As part of our future work, we plan to combine collaborative filtering with clustering and classification approaches. More specifically, we plan to identify student groups with similar academic achievement capacities, and within each group, apply adaptive estimation techniques. We are also currently in the process of integrating these methods into a web-based tool that will be used by Istanbul Sehir University community to make more informed decisions during course registration period. It will be also available to the faculty members to get information on the academic level of a particular class based on the average predicted grade for the students in that class.

Acknowledgements

We would like to thank Istanbul Sehir University for providing the dataset exploited in this work.

References

- [1] Ricci, F., Rokach, L., & Shapira, B. (2011). *Introduction to recommender systems handbook* (pp. 1-35). Springer US.
- [2] Benesty, J., Chen, J., Huang, Y., & Cohen, I. (2009). Pearson correlation coefficient. In *Noise reduction in speech processing* (pp. 1-4). Springer Berlin Heidelberg.
- [3] Gower, J. C. (1985). Properties of Euclidean and non-Euclidean distance matrices. *Linear Algebra and its Applications*, 67, 81-97.
- [4] Cha, S. H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *City*, 1(2), 1.
- [5] Sukhija, K., Jindal, M., & Aggarwal, N. (2015). The recent state of educational data mining: A survey and future visions. In *MOOCs, Innovation and Technology in Education (MITE), 2015 IEEE 3rd International Conference on* (pp. 354-359). IEEE.
- [6] Verma, K., Singh, A., & Verma, P. (2016). A Review on Predicting Student Performance Using Data Mining Method. *Futuristic Trends in Engineering, Science, Humanities, and Technology (FTESHT-16)*, pp. 124 - 129.
- [7] Sullare, V. A., Thakur, R. S., & Mishra, B. (2016). Analysis of Student Performance Using Mining Technique: A Review. *Artificial Intelligent Systems and Machine Learning*, 8(3), 94-97.
- [8] Luo, J., Sorour, S. E., Goda, K., & Mine, T. (2015). Predicting Student Grade Based on Free-Style Comments Using Word2Vec and ANN by Considering Prediction Results Obtained in Consecutive Lessons. *International Educational Data Mining Society*.
- [9] Luo, J., Sorour, S. E., Goda, K., & Mine, T. (2015).

Predicting Student Grade Based on Free-Style Comments Using Word2Vec and ANN by Considering Prediction Results Obtained in Consecutive Lessons. *International Educational Data Mining Society*.

- [10] Elbadrawy, A., Polyzou, A., Ren, Z., Sweeney, M., Karypis, G., & Rangwala, H. (2016). Predicting Student Performance Using Personalized Analytics. *Computer*, 49(4), 61-69.
- [11] Geiser, S., & Santelices, M. V. (2007). Validity of High-School Grades in Predicting Student Success beyond the Freshman Year: High-School Record vs. Standardized Tests as Indicators of Four-Year College Outcomes. Research & Occasional Paper Series: CSHE. 6.07. *Center for studies in higher education*.
- [12] Huang, S., & Fang, N. (2013). Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models. *Computers & Education*, 61, 133-145.
- [13] Dekker, G., Pechenizkiy, M., & Vleeshouwers, J. (2009). Predicting students drop out: A case study. In *Educational Data Mining 2009*.
- [14] Kabakchieva, D. (2013). Predicting student performance by using data mining methods for classification. *Cybernetics and information technologies*, 13(1), 61-72.
- [15] Loll, F., & Pinkwart, N. (2009). Using collaborative filtering algorithms as eLearning tools. In *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on* (pp. 1-10). IEEE.
- [16] Ray, S., & Sharma, A. (2011). A collaborative filtering based approach for recommending elective courses. In *International Conference on Information Intelligence, Systems, Technology and Management* (pp. 330-339). Springer Berlin Heidelberg.
- [17] Walpole, R. E., Myers, R. H., Myers, S. L., & Ye, K. (1993). *Probability and statistics for engineers and scientists* (Vol. 5). New York: Macmillan.
- [18] Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, 30(1), 79-82.