University of Waterloo
Department of Management Sciences
MSCI 446: Introduction to Machine Learning
Winter 2022

PROJECT REPORT

Meenakshi Andoorveedu (20837987)
Nayeema Nonta (20837920)

Table of Contents

I. Introduction

Business Problems:

As learning styles evolve, it makes it imperative for Professors to adapt and improve the quality of their course and teaching delivery. A popular platform where students provide feedback for Professors is ratemyprofessors.com. Our team has decided to analyze the reviews from students by exploring the comments made by students as well as the tags/categories they selected to describe the Professor. In our business problems we want to identify what makes a "Good Professor" versus a "Bad Professor" based on student perception and experiences.

Supervised Learning:

The business problem we are trying to solve is helping professor's understand and adapt their course delivery to better reflect student expectations. Using the comments from student reviews, we plan on identifying the words associated with highly rated professors and the words related to lowly rated professors. Logistic Regression and Naive Bayes are the two algorithms that will be used to develop these predictions.

Unsupervised Learning:

The second business problem we are trying to solve is identifying key characteristics that constitute a "Good Professor" versus a "Bad Professor". We would like to determine associations and relationships between attributes of a professor to better understand students needs in a course.

Motivation:

These business problems are interesting because it highlights what students find most valuable in a Professor and how certain course structures and delivery are preferred within the student community. Making use of the knowledge learned from these busines problems will enable a higher standard of education, interest, and growth relationship between students and professors.

Supervised Learning

Text mining is a powerful form of sentiment analysis that enables us to understand subjective information about a Professor through the student's comments. This is a good machine learning problem because there are a complex set of factors that determine if a Professor is perceived as "Good" or "Bad" which can not be easily predicted. It is also an interesting machine learning problem since the highly occurring words outputted from the algorithm can help develop relationships between the Professor's ratings.

Unsupervised Learning

Using the Apriori Algorithm to identify association rules between professors rated highly and rated lowly helps identify what students truly value most in professors. This is an interesting machine learning problem because given a certain support and confidence threshold, we can see

what are the related types of personality traits and course delivery that tends to be common for Good professors and how that differs from Bad professors.

II. Data Description & Feature Engineering

The source of the data is Mendeley Data (He 2020), published on March 4, 2020. The dataset has a sample of 20,000 rows, where each row constitutes a review written by a student for a specific Professor. There are 51 columns that encompass general information about the professor, information about the student, comments, ratings, and different characteristics of the professor. The first five sample rows are shown in figures (Appendix A, Figure 1, Figure 2, Figure 3). In our dataset, there are 1413 unique instances of Professors, excluding nulls and missing data.

Supervised Learning

The supervised learning tasks, which will be expanded on later in this paper, used the 'student_star' and 'comments' columns. In order to clean the data and pre-process it for the supervised learning tasks, the first step was to remove any rows in the dataset which were 'null' for the 'comments' column. This reduced the rows in the original dataset from 20000 to 19993. Next, rows which have comments consisting of two or less words were removed from the database. This decision was made to remove comments which have no meaning (e.g., "no comment", "NO COMMENT!!", etc.). This would increase the overall accuracy for the supervised learning models since these comments are common between high and low rated professors. This reduced the row count of the dataset to 18582. Lastly, all 'student_star' ratings were converted to type *integer*. This was done to reduce the number of possible outcomes for the predicted ratings to five (i.e., student ratings of 1,2,3,4, and 5), and remove values in between (e.g., 1.5, 2.5, etc.) to avoid model overfitting and increase prediction accuracy.

The supervised learning tasks consist of both binary and multinomial classification. The final row count for the multinomial dataset was 18582 as mentioned previously. For the binary classification dataset, 'student_star' ratings of 4 and 5 were assigned a Boolean value of 1, ratings of 1 and 2 were assigned a Boolean value of 0. Any rows where 'student_star' was equal to 3, were dropped from the database. This was done to simplify the dataset and split it into binary class labels for the models to predict. The final row count for the binary classification dataset was 16221 rows.

Unsupervised Learning

The variable 'star_rating' is the average rating of the Professor based on all students who have left a review. Since some professors were left 30 reviews while others may have been left 5, all duplicate rows of the same instance of the professor were removed from the dataset in order to increase the accuracy of the model.

The variable 'tag_professor' are the pre-set characteristics of the Professor as perceived by students. Null values in this column were dropped from the dataset. This was critical as the tags are transformed into the 20 Boolean variables; 1 if the professor embodies the characteristic, 0 for the absence of this characteristic. If there are nulls/no entries for a particular professor in the 'tag_professor' column, all 20 Boolean variables of the Professor's characteristics are recorded

as 0. This impacts the model's accuracy as 0 would incorrectly overinflate the absence of characteristics. Thus, the model would not be able to precisely develop relationships between the Professor's characteristics based on the class variables of "Good" or "Bad".

The data was separated into two unsupervised learning tasks. The variable 'star_rating' was used to split the data where ratings above 4.0 would be used to learn associations of "Good" Professors ('goodProfs' dataset), and ratings under 2.9 would be used to identify relationships between "Bad" Professors ('badProfs' dataset). Ratings of 3 were removed for the unsupervised learning tasks as they did not provide a strong opinion of the overall teaching quality of a professor.

The dataset was further cleaned such that the 2 new dataframes were created. The first dataframe was a subset of the new 'goodProfs' dataset. It only holds the columns of the 20 Boolean variables. Similarly, the second dataframe was a subset of the new 'badProfs' dataset with the 20 Boolean variables columns.

Exploratory Data Analysis

To study the distribution of ratings and visualize the correlation among 'student_star' ratings with other attributes of a professor, four histograms were created.

The first graph, from the binary classification dataset, depicts the relationship of a professor's 'student_star' rating with their 'year_since_first_review' (Appendix A, Figure 4). From this graph we can see that there is no strong correlation between good reviews and bad reviews based on how long it's been since a professor was first reviewed on the site.

The next graph depicts the data from the supervised multinomial classification dataset and shows the distribution of star ratings 1 to 5 based on the perceived race of the professor (Appendix A, Figure 5). This graph shows that the distribution of ratings between the races Hispanic, White, and Black are almost identical. Therefore, even though there are a greater number of white professors in the dataset, it is evident that there is no clear racial bias between the race of the professor and their 'student_star' rating. The most frequent rating is 5.

The third graph illustrates the relationship between a professor's difficulty index ('diff_index') and if they have a good or bad review (Appendix A, Figure 6). Here, there is a clear relationship that professors with average and low difficulty index tend to have more good reviews, whereas those with higher difficulty index tend to have bad reviews.

The fourth graph describes the frequency of the 20 Professor characteristics variables as described in (Appendix A, Figure 7). It can be seen that among all Professors, tough grader was the most common characteristic. Some of the least commonly occurring characteristics was pop quizzes, test heavy, and extra credit.

III. Supervised Learning

The goal of the supervised learning tasks was to perform sentiment analysis and predict if a professor has a high or low rating based on the words in the comments, as well as to determine

which words are highly correlated with each class label. The type of supervised learning being done is classification, including binary and multinomial classification.

In the supervised learning models, the comments were converted into TF-IDF (term frequency-inverse document frequency) vectors and were the independent variables (features) for each model. The text in the 'comments' column was cleaned by removing punctuation and making all the words lowercase. Any 'stop words' (i.e., commonly repeated words like 'the', 'in', etc.) were also removed. Then, the words in the text were lemmatized to reduce each word to their root word.

In order to determine how accurately and precisely the model predicted the outcomes, four methods of model evaluation were used: confusion matrix, normalized confusion matrix, classification report, and k-fold cross validation. K-fold cross validation was used to check for model overfitting. Due to the stochastic nature of the learning algorithms, for each runtime, there is a different train/test split, as well as different accuracy rates for the same training data when a model is run several times. If the k-fold cross validation accuracy was greater than the model accuracy from the classification report, it would reveal model overfitting. However, in general, k-fold the accuracy rates were extremely close to classification reports, and the difference was negligible.

Learning Task 1

Model Description

Logistic Regression Model: Is_Good_Professor (Binary Classifier)
Dependent Variables: 'student_star' = 1,0

- 0 is mapped to actual star rating of 1,2
- 1 is mapped to actual star rating of 4,5

Independent Variables: TF-IDF vectors

Purpose

The purpose of this experiment is to build a model, Is_Good_Professor, to accurately predict if a professor has "good" ('student_star' = 1) or "bad" ('student_star' = 0) reviews based on the TF-IDF vectors extracted from the 'comments' column.

A secondary model, Is_Bad_Professor, with the flipped dataset where "bad" reviews had 'student_star' = 1 and "good" reviews had 'student_star' = 0 was analyzed as part of this learning task. This was done to see if the same words which have the lowest correlation with Is_Good_Professor, also have the highest correlation with Is_Bad_Professor.

Hypothesis

Our hypothesis is that the model will have a high accuracy rate of at least >80%, since the training dataset uses a great deal of generalization with the removal of 'student_star' ratings of 3

and splitting the ratings into binary outcomes of 0 and 1. We also believe the words which have the lowest correlation with Is_Good_Professor will also have the highest correlation with Is_Bad_Professor.

Model Evaluation

As seen from the confusion matrices for the Is_Good_Professor model (Appendix B, Figure 1, Figure 2) and Is_Bad_Professor model (Appendix B, Figure 3, Figure 4), both models predicted the majority of the labels correctly. The classification reports (Appendix B, Table 2, Table 3) for the models show that each model had similar accuracy levels, with Is_Good_Professor and Is_Bad_Professor having an overall accuracy rate of 87% and 88% respectively. We can also see that the recall rates for the bad professors were 72% and in comparison, to 94% for the good professors, this is because bad professors have less rows of data to learn from. The k-fold cross validation results (Appendix B, Table 4, Table 5) show that the average accuracy over 10, 25, 50, and 100-fold are very close and are between 87 and 88 percent.

Results

The top 10 words which have the highest and lowest correlation with Is_Good_Professor and Is_Bad_Professor can be found in the results table (Appendix B, Table 1). From the results, it can be seen that the word with the highest correlation from each model is also the word with the lowest correlation in the other model. Explicitly, ‘great’ has the highest correlation with Is_Good_Professor and the lowest correlation with Is_Bad_Professor, while ‘worst’ has the highest correlation with Is_Bad_Professor and lowest with Is_Good_Professor.

However, the top 10 words which are the most correlated with Is_Good_Professor are not identical to the top 10 words least correlated with Is_Bad_Professor. However, 8 out of 10 words that are most correlated with Is_Good_Professor are also least correlated with Is_Bad_Professor. These 8 words are: 'love', 'fun', 'excellent', 'wonderful', 'easy', 'amazing', 'awesome', 'best', 'great'. The similarity of the results between the two models is due to the fact that they use the same dataset but flipped. The slight differences are due to the learning algorithm using different train/test splits and randomly predicting the values.

Learning Task 2

Model Description

Logistic Regression Model: Professor_Star_Rating (Multinomial Classifier)
Dependent Variables: ‘student_star’ = 1,2,3,4,5
Independent Variables: TF-IDF vectors

Purpose
The purpose of this experiment is to build a model, Professor_Star_Rating, to accurately predict a professor’s ‘student_star’ rating based on the TF-IDF vectors extracted from the ‘comments’

column. This learning task aims to build a model that provides more information on the student's sentiment than Is_Good_Professor.

Hypothesis

Our hypothesis is that the model will have a lower accuracy rate than the binary classifier Is_Good_Professor, since the training dataset uses less generalization than the binary training dataset such that there are more possible outcomes.

Model Evaluation

As seen from the confusion matrices (Appendix B, Figure 5, Figure 6), the most accurately predicted labels were 1 and 5, whereas ratings 2, 3, and 4 had low accuracy. The classification report (Appendix B, Table 7) shows that class labels 1 and 5 had the highest precision score of 61% and 57% respectively. The class labels 2, 3, and 4, had precision scores of 34%, 32%, and 38% respectively. The overall accuracy of the model was 51%. One of the downsides of using so many class labels is model overfitting. To check for overfitting the k-fold validation report (Appendix B, Table 8) reveals that the model has 50 to 51 percent accuracy over 10, 25, 50 and 100 folds, which is the same as the overall accuracy from the classification report. Therefore, we can conclude that there is no overfitting.

Results

As per our hypothesis, this model has a lower accuracy level than Is_Good_Professor. The top 10 words which have the highest correlation with each outcome of Professor_Star_Rating (i.e., 1, 2, 3, 4, 5) can be found in the results table (Appendix B, Table 6). From the results, the top 10 words most correlated with a rating of 1 are: 'away', 'unclear', 'doesn't', 'unhelpful', 'terrible', 'awful', rude', 'horrible', 'avoid', 'worst'. The words most correlated with rating 5 are: 'nicest', 'hilarious', 'loved', 'fun,' 'excellent', 'wonderful', 'amazing', 'awesome', 'great', 'best'. These are very similar to the words most and least correlated with Is_Good_Professor. The words most correlated with rating 3 are: 'hw', 'pretty', 'dry', 'attendance', 'lot', 'arent', 'hard', 'alright', 'overall', 'ok'. These results make sense since 3 is a neutral rating on the scale, therefore the overall sentiment is neither good nor bad. However, we can see that some words do not reveal much about the sentiment, such as 'lot' and 'arent'.

<u>Learning Task 3</u>

Model Description

Logistic Regression Model: Best_Worst_Average_Prof (Multinomial Classifier)
Dependent Variables: 'student_star' = 0,1,2

- 0 is mapped to actual star rating of 1
- 1 is mapped to actual star rating of 2,3,4
- 2 is mapped to actual star rating of 5

Independent Variables: TF-IDF vectors

The purpose of this experiment is to build a model, Best_Worst_Average_Prof, to accurately predict a professor's 'student_star' rating is "Worst", "Average", or "Best" based on the TF-IDF vectors extracted from the 'comments' column.

Purpose

The purpose of this model is also to build a multinomial classifier which has more class labels than the binary classifier Is_Good_Professor, but still more generalized than the multinomial classifier Professor_Star_Rating. This model is to provide a model with a higher accuracy than Professor_Star_Rating but provide more information than the Is_Good_Professor.

Hypothesis

Our hypothesis is that the model will have a lower accuracy level than the binary classifier Is_Good_Professor, but a higher accuracy than Professor_Star_Rating. Since the training dataset for this learning model is a modified version of the multinomial dataset which uses less generalization than the binary training dataset but more generalization than the original multinomial dataset (i.e., the outcomes are reduced to three instead of five).

Model Evaluation

As seen from the confusion matrices (Appendix B, Figure 7, Figure 8), the three labels were more evenly predicted than Professor_Star_Rating. The classification report (Appendix B, Table 10) shows that the class label 'worst' has a precision score of 73%, while 'average' and 'best' have precision scores of 61% and 68% respectively. This is since the middle values, ratings of 2,3,4 were combined to provide a larger subset of data points in that range since most ratings were distributed to 1 and 5 for Professor_Star_Rating. The overall accuracy of the Best_Worst_Average_Prof model is 65% which is lower than Is_Good_Professor, but higher than Professor_Star_Rating. The k-fold cross validation report (Appendix B, table 11) shows that the average accuracy over 10, 25, 50, and 100 folds is approximately 64%.

Results

The top 10 words which have the highest correlation with each outcome of Best_Worst_Average_Prof can be found in the results table (Appendix B, Table 9). From the results, for Best_Worst_Average_Prof = Best, 7/10 most correlated words are identical to Is_Good_Professor as well as 7/10 identical to most correlated words in Professor_Star_Rating = 5.

For Best_Worst_Average_Prof = Average, 5/10 of the top correlated words were identical to Professor_Star_Rating = 3. However, the general sentiment among the words were the same. The results for Best_Worst_Average_Prof = Average (i.e., 'approach' 'alot' 'willing' 'ok' 'lot' 'nice' 'overall' 'okay' 'pretty' 'alright'), capture the sentiment more accurately than Professor_Star_Rating = 2,3,4 since it includes the general sentiment of all classes in between and has an approximately even distribution of data points among the various class labels in comparison to Professor_Star_Rating.

For Best_Worst_Average_Prof = Worst, 8/10 words are identical to the words least correlated with Is_Good_Professor, and 9/10 words were the same as words associated with Professor_Star_Rating = 1.

Learning Task 4

Model Description

Naïve Bayes Model: naive_bayes_Is_Good_Prof (Binary Classifier)
Dependent Variables: 'student_star' = 1,0

- 0 is mapped to actual star rating of 1,2
- 1 is mapped to actual star rating of 4,5

Independent Variables: TF-IDF vectors

Purpose

The purpose of this experiment is to build a Naïve Bayes model, naive_bayes_Is_Good_Prof, to accurately predict if a professor has "good" ('student_star' = 1) or "bad" ('student_star' = 0) reviews based on the TF-IDF vectors extracted from the 'comments' column. It is then compared with the binary logistic regression model Is_Good_Professor, to determine the better learning model for this problem.

The Naïve Bayes algorithm is a good model for this problem since it assumes each feature is independent of the other features. Which is suitable for our learning model since we want to treat each word in the TF-IDF vectors array as words independent of each other.

Hypothesis

Our hypothesis is that the word with the highest coefficient for Is_Good_Professor, 'great', will also appear in the top 10 words with the highest coefficients for this model. We also predict that the Naïve Bayes model may have a lower overall accuracy, because unlike logistic regression, Naïve Bayes assumes all features are conditionally independent prior to calculating the probability of belonging to a class label. This may reduce the model's accuracy since words like "pretty" are usually correlated with words like "good" or "bad" since they combine to form one sentiment e.g., "pretty good".

Model Evaluation

As seen from the confusion matrices for the naive_bayes_Is_Good_Prof model (Appendix B, Figure 9, Figure 10), the distribution of actual vs. predicted label was very similar to Is_Good_Professor. However, from the classification report (Appendix B, Table 13), it can be seen that, the overall accuracy for this model was only 79%, which is much lower compared to Is_Good_Professor's overall accuracy of 87%. The k-fold cross validation report (Appendix B,

Table 14) shows that this model has an average accuracy of approximately 80% over 10, 25, 50, and 100 folds.

Results

The top 10 words which have the highest correlation with each outcome of naive_bayes_Is_Good_Prof, can be found in the results table (Appendix B, Table 12). The word with the highest coefficient to good reviews was 'class' instead of 'great' from Is_Good_Professor. For the good reviews 2/10 words were the same as Is_Good_Professor's highest coefficients and 2/10 words for bad reviews were the same as Is_Good_Professor's lowest coefficients. Therefore, the Naïve Bayes binary classifier was drastically different from the logistic binary classifier Is_Good_Professor.

An observation from the classification report (Appendix B, Table 13) shows that the bad reviews (student_star = 1,2) had a higher precision rate of 97% with a support of 1295 in comparison to 76% for good reviews (student_star = 4, 5) with a support of 2761. This result is surprising since the good reviews had more data to learn from but a lower precision rate. This may be due to the fact that the good reviews had more rows of data, and as a result, more TF-IDF vector features. Since the Naïve Bayes algorithm calculates the joint probability and then the posterior probability, it naively assumes independence of words. However, a student who has written a lengthy "good" review for a professor has all words in that review treated as independent in the Naïve Bayes algorithm. Thus, the accuracy of the model is reduced due to this naive assumption.

Another unexpected result for this model is that the words 'professor' and 'class' appear in the top 10 list of words with the highest coefficients as well as in the lowest coefficients list (Appendix B, Table 12). This may be because Naïve Bayes looks at each feature's probability of belonging to a class label separately. Whereas the binary logistic model is a discriminative algorithm that groups a feature as one class label or another, so words don't appear in both class labels.

Learning Task 5

Model Description

Logistic Regression Model: naive_bayes_Prof_Star_Rating (Multinomial Classifier)
Dependent Variables: 'student_star' = 1,2,3,4,5
Independent Variables: TF-IDF vectors

Purpose

The purpose of this experiment is to build a model, naive_bayes_Prof_Star_Rating, to accurately predict a professor's 'student_star' rating based on the TF-IDF vectors extracted from the 'comments' column. It is then compared with the logistic regression model Professor_Star_Rating, to determine the better learning model for this problem.

Hypothesis

Our team hypothesizes that the overall accuracy for this model would be lower than all previous models. This prediction is made due to the nature of the Naïve Bayes learning algorithm as discussed in our hypothesis for learning task 4. It will also have a lower accuracy rate than naive_bayes_Is_Good_Prof, since we have multiple outcomes in this model, instead of just two.

Model Evaluation

As seen from the confusion matrices for the naive_bayes_Prof_Star_Rating model (Appendix B, Figure 11, Figure 12), student_star = 5 was the most correctly predicted label. This model also had lower precision levels in general in comparison to Professor_Star_Rating (Appendix A, Table 7). The classification report for naive_bayes_Prof_Star_Rating (Appendix B, Table 16), shows that overall accuracy for this model was only 43% which is significantly lower than all other models in the previous learning tasks. The classification report also shows that the precision rate for outcomes 2 and 3 were 0%. The k-fold cross validation report shows that the average accuracy over 10, 25, 50 and 100 folds is approximately 51% (Appendix B, Table 17). Under normal circumstances a k-fold accuracy level this low would indicate model overfitting. But, in this case, the overall accuracy was only 43% and the k-fold accuracy is surprisingly higher. Since the original train/test split had a less than <50% accuracy, when the model was put through k-fold cross validation, the wrong predictions from the original testing dataset were correct during k-fold testing.

Results

An unexpected result is that the word ‘class’ appeared in the top 10 correlated words for all of the five outcomes, as shown in the results table (Appendix B, Table 15). The word ‘class’ also appeared in both outcomes of naive_bayes_Is_Good_Prof. As explained in learning task 4, we believe this outcome is a result of the Naïve Bayes algorithm looking at each feature’s probability of belonging to a class label separately.

Lastly, the overall sentiment for naive_bayes_Prof_Star_Rating = 3 for this model is more positive, with the top 10 word being 'time' 'nice' 'lot' 'lecture' 'really' 'easy' 'hard' 'good' 'test' 'class', whereas the words for Professor_Star_Rating = 3 was more neutral, with the words being 'hw' 'pretty' 'dry' 'attendance' 'lot' 'arent' 'hard' 'alright' 'overall' 'ok'. The skewed result for this model is due to the lower accuracy rate.

IV. Unsupervised Learning:

Association rule mining was used to develop rules to determine an interesting relationship between Professor’s characteristics and the student’s perception of whether the Professor is “Good” or “Bad”. For the unsupervised learning tasks, the Apriori Algorithm will be used with a minimum confidence and support threshold to devise correlations and rules between itemsets.

Learning Task 1

Association Rules for Good Professors

The first unsupervised learning task was determining association rules to develop correlations between the Characteristics of Good Professors. Four separate experiments were conducted with different support and confidence thresholds in order to determine the goodness of our association rules.

Experiment 1

Association with minimum support of 10%, minimum confidence of 10%

Purpose

The purpose of this experiment was to understand the impacts that low support and low confidence would have on the association rules for good professors.

Hypothesis

Our hypothesis is that due to such low confidence and support, there would be many rules outputted since fewer itemsets would be pruned.

Findings

There was a total of 875 frequent itemsets that met the minimum criteria of having greater than 10% support. The remaining itemsets were pruned for not meeting the minimum support threshold. Association rules were found for the remaining itemsets who's minimum confidence met a 10% threshold. A total of 11,634 association rules were devised (Appendix C, Table 1). The itemset with the highest support (0.431095) and highest confidence (0.717647) threshold created an association of, "gives good feedback → respected". Since the support and confidence were so low, there were often cases with many items in an itemsets (ex. gives_good_feedback, caring, inspirational as an consequents) .

The findings were aligned with our hypothesis since the low support and confidence does not narrow the search space enough. It is so broad that good association rules are not able to be made.

Experiment 2

Association with minimum support of 25%, minimum confidence of 75%

Purpose

The purpose of this experiment was to understand the impacts that lower support and higher confidence would have on the association rules for good professors.

Hypothesis

Our hypothesis is that due to such low support there would be many frequent itemsets outputted. But, due to the high confidence threshold, the number of association rules would significantly reduce.

Findings

There was a total of 47 frequent itemsets that met the minimum criteria of having greater than 25% support. Itemsets that did not meet the minimum support threshold were pruned. Association rules were found for the remaining itemsets who's minimum confidence met a 75% threshold. A total of 17 association rules were created (Appendix C, Table 2). The association rules also highlighted particular relationships between the Professor's characteristics. For example, "clear grading criteria → gives good feedback", "amazing lectures → respected", and "hilarious → respected".

The findings of the experiment were aligned with our knowledge and hypothesis of itemset pruning. However, we were surprised that the maximum number of items in an itemset for the antecedents was 2. An unexpected result from this experiment was that most consequents were the exact same itemset. In particular, "respected" was a consequent that was shown in 10 out of the 17 association rules. To evaluate the goodness of the model, we can see that the association rules seemed to be accurate with our knowledge of characteristics of Bad Professors. Since the thresholds were not too strict or not too broad, the algorithm was able to output an appropriate number of rules.

Experiment 3

Association with minimum support of 35%, minimum confidence of 10%

Purpose

The purpose of this experiment was to understand the impacts that higher support and lower confidence would have on the association rules for good professors.

Hypothesis

Our hypothesis is that due to the higher support threshold there would be many itemsets pruned, significantly narrowing the search space. But, due to the low confidence threshold, most of the itemsets that were not pruned would form association rules.

Findings

There was a total of 13 frequent itemsets that met the minimum criteria of having greater than 35% support. The remaining itemsets were pruned for not meeting the minimum support threshold. Association rules were found for the remaining itemsets who's minimum confidence met a 10% threshold. A total of 8 association rules were developed (Appendix C, Table 3). The association rules also highlighted particular relationships between the Professor's characteristics. For example, "caring → gives good feedback", "inspirational → respected", and "caring → respected".

The findings of the experiment were aligned with our knowledge and hypothesis of itemset pruning. However, we were surprised that although the minimum confidence was 10%, most association rules had greater than 60% confidence. This means that we can conclude with greater confidence that the strength of the association rules is high. An unexpected result from this experiment was that the same 4 itemsets formed association rules. For example, “caring”, “gives good feedback”, “respected”, and inspirational, were the only itemsets. This was surprising because there were 20 Boolean variables, and this seemed to be the 4 Characteristics with the strongest correlation for Good Professor. To evaluate the goodness of the model, we can see that the association rules seemed to be accurate with our knowledge of characteristics of Good Professors. Since the thresholds were not too strict or not too broad, the algorithm was able to output an appropriate number of rules.

Experiment 4

Association with minimum support of 40%, minimum confidence of 60%

Purpose

The purpose of this experiment was to understand the impacts that higher support and higher confidence would have on the association rules for good professors.

Hypothesis

Our hypothesis is that due to the higher support threshold there would be many itemsets pruned. In addition, the high confidence threshold, should further prune many itemsets, significantly narrowing the amount of association rules.

Findings

There was a total of 9 frequent itemsets that met the minimum criteria of having greater than 40% support. 7 itemsets had only 1 item, and 2 itemsets had 2 items. Itemsets that did not meet the minimum support threshold were pruned. Association rules were found for the remaining itemsets who’s minimum confidence met a 60% threshold. A total of 4 association rules were developed (Appendix C, Table 4). The association rules also highlighted particular relationships between the Professor’s characteristics. For example, “respected → gives good feedback”, “gives good feedback → respected”, “caring→ respected”, and “respected → caring ”.

The findings of the experiment were aligned with our hypothesis of itemset pruning. However, we were surprised that increasing the support threshold more than 45% with this confidence level led to an empty DataFrame, such that no association rules could be made. This was the highest threshold where we were able to output information. However, it is important to note that this support and confidence threshold are not appropriate since it is so strict. This means that it narrows the search space significantly such that very few association rules could be made. An unexpected result from this experiment was that ‘respected’ had shown up in all 4 rules. This was surprising because out of the 20 Boolean characterisitics, “respected” was the one variable that showed the highest correlation to good professor as demonstrated by being on either the left and right side of each association rule.

Learning Task 2:

Association Rules for Bad Professors

The second unsupervised learning task was determining association rules to develop interesting relationships of the Characteristics of Bad Professors. Four separate experiments were conducted with different support and confidence thresholds in order to determine the goodness of our association rules.

Experiment 1

Association with minimum support of 10%, minimum confidence of 10%

Purpose

The purpose of this experiment was to understand the impacts that low support and low confidence would have on the association rules for bad professors.

Hypothesis

Our hypothesis is that due to such low confidence and support, there would be many rules outputted since fewer itemsets would be pruned.

Findings

There was a total of 369 frequent itemsets that met the minimum criteria of having greater than 10% support. The remaining itemsets were pruned for not meeting the minimum support threshold. Association rules were found for the remaining itemsets who's minimum confidence met a 10% threshold. A total of 3,306 association rules were devised (Appendix C, Table 5). It was interesting to note that even rules that show a stronger correlation with good professors were outputted for bad professor. For example, "caring → gives good feedback" was seen consistently as a rule for Good Professors. But, due to such low support and confidence, this rule was also created for Bad Professors as well.

The findings were aligned with our hypothesis since the low support and confidence does not narrow the search space enough. It is so broad that good association rules are not able to be made.

Experiment 2

Association with minimum support of 25%, minimum confidence of 65%

Purpose

The purpose of this experiment was to understand the impacts that lower support and higher confidence would have on the association rules for bad professors.

Hypothesis

Our hypothesis is that due to such low support there would be many frequent itemsets outputted. But, due to the high confidence threshold, the number of association rules would significantly reduce.

Findings

There was a total of 22 frequent itemsets that met the minimum criteria of having greater than 25% support. Itemsets that did not meet the minimum support threshold were pruned. Association rules were found for the remaining itemsets who's minimum confidence met a 65% threshold. A total of 19 association rules were created (Appendix C, Table 6). The association rules also highlighted particular relationships between the Professor's characteristics. For example, "skip class you won't pass → tough grader", "get ready to read → skip class you won't pass", and "lecture heavy → tough grader".

The findings of the experiment were aligned with our knowledge and hypothesis of itemset pruning. However, we were surprised that the maximum number of items in an itemset for the antecedents was 2. To evaluate the goodness of the model, we can see that the association rules seemed to be accurate with our knowledge of characteristics of Bad Professors. Since the thresholds were not too strict or not too broad, the algorithm was able to output an appropriate number of rules.

Experiment 3

Association with minimum support of 35%, minimum confidence of 10%

Purpose

The purpose of this experiment was to understand the impacts that higher support and lower confidence would have on the association rules for good professors.

Hypothesis

Our hypothesis is that due to the higher support threshold there would be many itemsets pruned, significantly narrowing the search space. But, due to the low confidence threshold, most of the itemsets that were not pruned would form association rules.

Findings

There was a total of 9 frequent itemsets that met the minimum criteria of having greater than 35% support. The remaining itemsets were pruned for not meeting the minimum support threshold. Association rules were found for the remaining itemsets who's minimum confidence met a 10% threshold. A total of 8 association rules were developed (Appendix C, Table 7). The association rules also highlighted particular relationships between the Professor's characteristics. For example, "skip class you won't pass → tough grader", "lots of homework → tough grader", and "tough grader→ lecture heavy".

The findings of the experiment aligned with our knowledge and hypothesis of itemset pruning. However, we were surprised that although the minimum confidence was 10%, most association

rules had greater than 50% confidence. This means that we can conclude with greater confidence that the strength of the association rules are high. An unexpected result from this experiment was that the same "tough grader" had shown up in all 7 of the association rules. This means that tough grader should be a highly correlated trait of a Bad Professor. To evaluate the goodness of the model, we can see that the association rules seemed to be accurate with our knowledge of characteristics of Bad Professors. Since the thresholds were not too strict or not too broad, the algorithm was able to output an appropriate number of rules.

Experiment 4

Association with minimum support of 40%, minimum confidence of 65%

Purpose

The purpose of this experiment was to understand the impacts that higher support and higher confidence would have on the association rules for bad professors.

Hypothesis

Our hypothesis is that due to the higher support threshold there would be many itemsets pruned. In addition, the high confidence threshold, should further prune many itemsets, significantly narrowing the search space.

Findings

There was a total of 7 frequent itemsets that met the minimum criteria of having greater than 40% support. 4 itemsets had only 1 item, and 3 itemsets had 2 items. Itemsets that did not meet the minimum support threshold were pruned. Association rules were found for the remaining itemsets who's minimum confidence met a 65% threshold. A total of 4 association rules were developed (Appendix C, Table 8). The association rules also highlighted particular relationships between the Professor's characteristics. For example, "skip class you won't pass → tough grader", "tough grader→ skip class you won't pass", and "get ready to read → tough grader, and "lecture heavy → tough grader".

The findings of the experiment were aligned with our hypothesis of itemset pruning. However, we were surprised that increasing the support threshold more than 45% with this confidence level led to an empty DataFrame, such that no association rules could be made. This was the highest threshold where we were able to output information. However, it is important to note that by making the threshold so strict, it narrows the search space significantly meaning that fewer associations could be made. An unexpected result from this experiment was that the "tough grader" had shown up in all 4 rules. This was surprising because there out of all 20 Boolean Variables, "tough grader" was the only one variable showed up on either the left and right side of each association rule.This means that it has a very strong correlation of being a trait related to Bad Professors.

V. Conclusions

Summary of Main Findings and Discussion for Supervised Learning Results

Throughout the various learning models, it was found that the words that appeared to have the highest or one of the highest correlations with highly rated professors were ‘great’ and ‘best’. The word that had the lowest correlation with the highly rated professors/highest correlation with lowly rated professors was ‘worst’. The models with the highest overall accuracy were the binary logistic regression classifiers Is_Good_Professor and Is_Bad_Professor with accuracy levels of 87% and 88% respectively.

Naïve Bayes Vs. Logistic Regression

An important lesson learned from the experiments is the difference in classification methods between Naïve Bayes and logistic regression. Between the Naïve Bayes and logistic regression algorithms, the logistic regression had significantly higher overall accuracy rates. For Naïve Bayes the word ‘class’ had the highest predictive power since it was the only word that appeared in all outcomes of binary and multinomial classification models for Naïve Bayes algorithm.

Another lesson learned about the Naïve Bayes algorithm is that more rows of data does not mean higher precision rates. As seen from the results of the naive_bayes_Is_Good_Prof (Appendix B, Table 13), the good reviews had a lower precision rate despite having more rows of data. This is due to the nature of the Naïve Bayes algorithm which makes a naive assumption that all features are fully independent of other features before calculating the probability of a feature belonging to a class. The logistic regression however, is a linear algorithm that directly calculates the probability of a feature belonging to a class. Therefore, the model accuracy and precision will depend on the type of algorithm and features and more training data will not automatically increase the model’s accuracy.

Multinomial Vs. Binary Classification

The binary classification models had higher overall accuracy levels than the multinomial classification models. The binary models had accuracy levels of 80% or more, whereas the multinomial models Professor_Star_Rating, Best_Worst_Average_Prof, and naive_bayes_Prof_Star_Rating, had accuracy levels of 51%, 65%, and 43% respectively. However, the multinomial classification models provided more information on the sentiment associated with the various levels of star ratings. Therefore, the best model for our business problem would be the Best_Worst_Average_Prof model as it captures a range of sentiments, good, neutral, and bad, without significantly decreasing the accuracy of the model.

Summary of Main Findings and Discussion for Unsupervised Learning Results

It was determined that very low support and low confidence thresholds was ineffective in developing good association rules (Experiment 1). Due to the low pruning, we could see that for bad professors the rules commonly correlated with good professors were also outputted. An important lesson learned was that broadening the search space significantly led to a lower overall accuracy.

Another lesson learned is that making the search space too narrow was also ineffective in producing good rules (Experiment 4). Although the rules that were outputted seemed to logically

correlate to good/bad professors, too few rules were developed. Therefore, relaxing the thresholds reveals further information about the professor's characteristics.

Experiments 2 and 3 were the most effective in developing good association rules. The rules were consistent in logic with regards to good/bad professors. In addition, there were numerous rules produced, giving a deeper understanding of the correlation between the 20 Boolean characteristics.

Many good rules were developed for Good Professors in Experiment 2 and 3 (Appendix C, Table 2, Table 3). For example, through the various experiments, one specific rule "gives_good_feedback → respected" was outputted for Good Professors. This identifies a correlation such that Professors that actively provide their students valuable feedback are generally sought out to be well respected in the student community.

Likewise, for Bad Professors, an example of a good association rule made was "skip_class → tough_grader" (Appendix C, Table 6, Table 7). This shows a correlation that students who believe that if they skip a class they won't pass, the content tends to be quite difficult. Thus, resulting in a belief that specific professors are extremely difficult in grading.

General Lessons about Machine Learning

From the experiments conducted in this study, several general lessons about machine learning were learned including how to clean data and feature engineering. For our supervised and unsupervised learned tasks it was crucial to appropriately preprocess the data for use in machine learning models. To ensure that the results were reliable, the dataset had to be thoroughly checked for null values and disguised missing values. For instance, for the unsupervised learning datasets, the default value for a characteristic for a professor was 0, even if a student did not fill out any characteristics for that professor. Therefore, rows which had all 0 Boolean values for characteristics had been removed. If the data had not been checked for this, the results would be skewed and incorrect.

For feature engineering it was important to select the appropriate columns for supervised and unsupervised learning. For supervised learning, the features were the words in the comments column, which had to be cleaned by ensuring all words were lowercase, removing punctuation, eliminating commonly occurring words in the English language, as well as lemmatizing the words to their root word. This is key since without proper data preprocessing, the results will not be accurate and thus will not be useful. For unsupervised learning, the different tags for the Professors were chosen as 20 independent Boolean variables. All 20 Boolean variables are independent of one another, making them excellent candidates as part of feature engineering to learn associations.

Lastly, it was learned that in general, some algorithms were better suited for certain types of learning tasks. For instance, as discussed previously, the Naive Bayes algorithm did not perform well as a multinomial classifier. However, in general Naive Bayes learning is good for text mining as it assumes all features to be independent. For our business problem, due to the type of data, logistic regression was the most optimal learning algorithm for classifying different levels of ratings from student comments.

Benefit of the Findings

Association Rules Benefit to Professors

Professors can benefit from the knowledge learned from our machine learning tasks. Specifically, understanding the association rules for Good and Bad professors reveal an underlying correlation in relation to student perception. For example, an association rule developed for Bad Professors was "lecture heavy → tough grader". Professors can utilize this knowledge to improve the learning experience and education quality provided to students. In this specific example, the Professor may consider creating shorter slides and lecture material by developing more class examples to create an interactive lecture experience. In addition, they may also consider developing a fair approach when grading newer topics. Or alternatively, professors may consider developing clear criteria and rubrics that enable students to understand grading expectations.

Overall Sentiment of Students and How Professors Can Benefit from the Findings

From our experiments, we were able to identify specific words which describe certain types of behaviour, moods, actions, etc. associated with professors who received good, bad and neutral reviews. From the results (Appendix B, Table 9), it was found that professors who receive the lowest rating of 1, are generally 'unhelpful', 'useless', and 'rude'. Students think of them to be the 'worst', 'horrible', 'terrible', and believe they should be avoided.

For professors who receive ratings of 2,3 and 4, the overall sentiment was that they were 'okay', 'alright', 'nice' and approachable. Therefore, student's sentiments towards professors who received star ratings in this range were neutral.

The overall sentiment for professors who received the highest rating of 5 were that they were 'great', 'fun', 'wonderful', 'amazing', 'best', 'awesome', etc. Therefore, the general sentiment was that a professor that makes students feel good is highly rated. Another word associated with good professors is 'excellent' which could imply they are good at their job. In addition to the above, 'easy' and 'hilarious' were also recurring words among the different models for 'student_star' = 5. This suggests that professors who have a sense of humour and have relatively easy grading expectations are more preferred by students.

Professors can use the overall sentiments associated with good, bad, and neutral ratings to improve their teaching style. For instance, if a professor is struggling to engage with their class, they can improve their overall review by making the class more fun. For example, giving students interactive exercises to test their knowledge or letting students choose topics they are personally interested in for their projects. One of the common words for bad professors was 'unhelpful'. Therefore, to improve their teaching style, a professor can hold extra help sessions during office hours to help students succeed.

VI. References

He, Jibo (2020), “Big Data Set from RateMyProfessor.com for Professors' Teaching Evaluation”, Mendeley Data, V2, doi: 10.17632/fvtfjyvw7d.2

VII. Appendix A. Data Exploration

Figure 1:

Row 1-5 of Dataset, Columns 1-9

	professor_name	school_name	department_name	local_name	state_name	year_since_first_review	star_rating	take_again	diff_index
0	Leslie Looney	University Of Illinois at Urbana-Champaign	Astronomy department	Champaign\xe2\x80\x93Urbana	IL	11.0	4.7	NaN	2.0
1	Leslie Looney	University Of Illinois at Urbana-Champaign	Astronomy department	Champaign\xe2\x80\x93Urbana	IL	11.0	4.7	NaN	2.0
2	Leslie Looney	University Of Illinois at Urbana-Champaign	Astronomy department	Champaign\xe2\x80\x93Urbana	IL	11.0	4.7	NaN	2.0
3	Leslie Looney	University Of Illinois at Urbana-Champaign	Astronomy department	Champaign\xe2\x80\x93Urbana	IL	11.0	4.7	NaN	2.0
4	Leslie Looney	University Of Illinois at Urbana-Champaign	Astronomy department	Champaign\xe2\x80\x93Urbana	IL	11.0	4.7	NaN	2.0

Figure 2:

Row 1-5 of Dataset, Columns 10-48

tag_professor	...	lots_of_homework	accessible_outside_class	lecture_heavy	extra_credit	graded_by_few_things	group_projects	test_heavy
Hilarious (2) GROUP PROJECTS (2) Gives good ...	...	0	0	0	0	0	1	0
Hilarious (2) GROUP PROJECTS (2) Gives good ...	...	0	0	0	0	0	1	0
Hilarious (2) GROUP PROJECTS (2) Gives good ...	...	0	0	0	0	0	1	0
Hilarious (2) GROUP PROJECTS (2) Gives good ...	...	0	0	0	0	0	1	0
Hilarious (2) GROUP PROJECTS (2) Gives good ...	...	0	0	0	0	0	1	0

Figure 3:
Row 1-5 of Dataset, Columns 49-51

so_many_papers	beware_of_pop_quizzes	IsCourseOnline
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Figure 4:
Histogram of 'student_star' and 'year_since_first_review' (1 = good review, 0 = bad review)

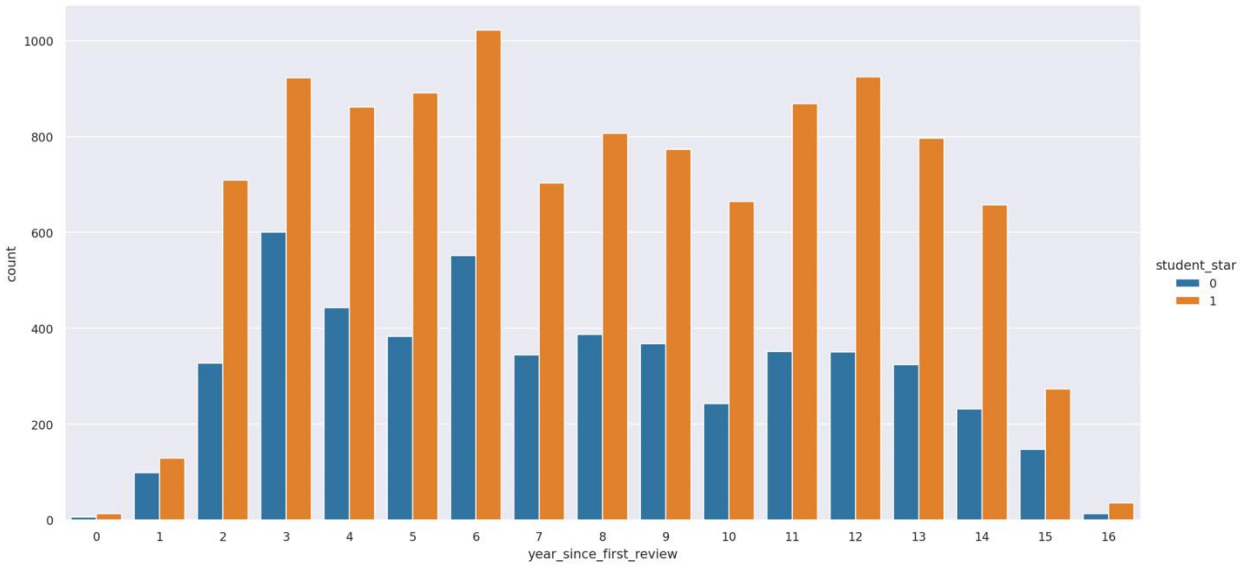


Figure 5:
Histogram of distribution of 'student_star' ratings based on the 'race' of professor

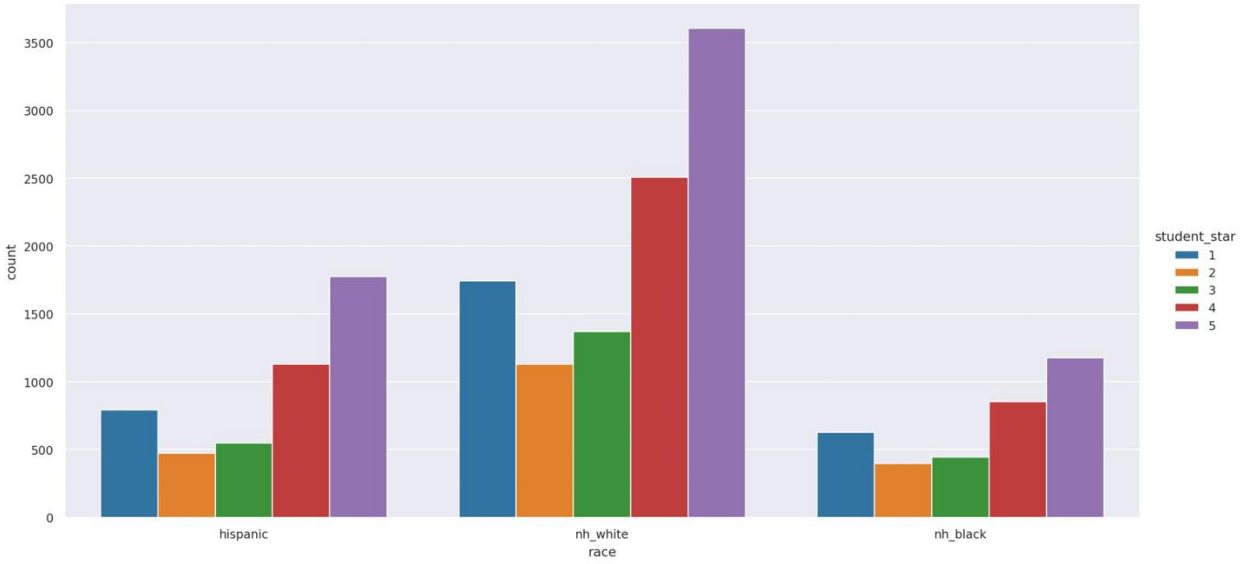


Figure 6:
Histogram of 'student_star' based on difficulty index ('diff_index') of professor (1 = good review, 0 = bad review)

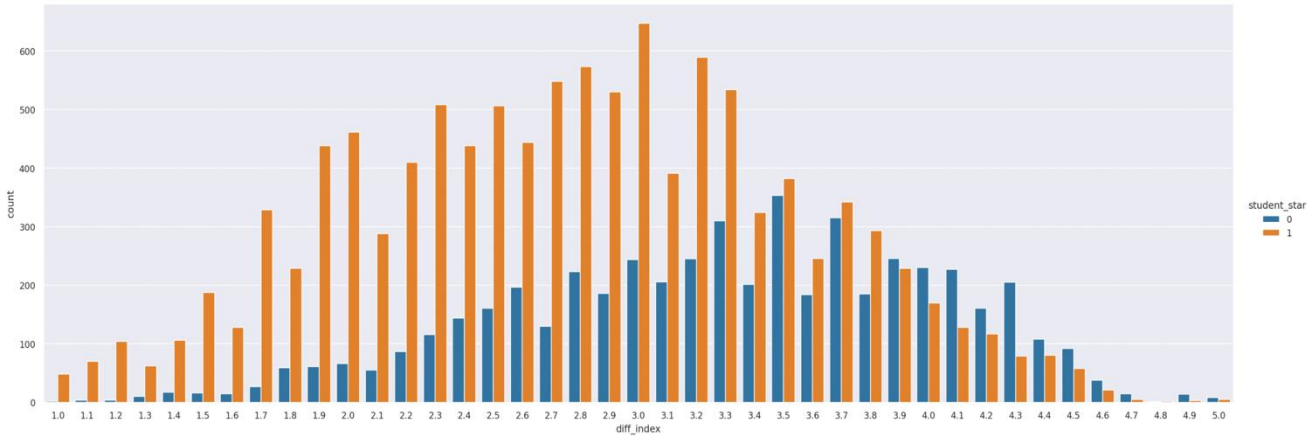
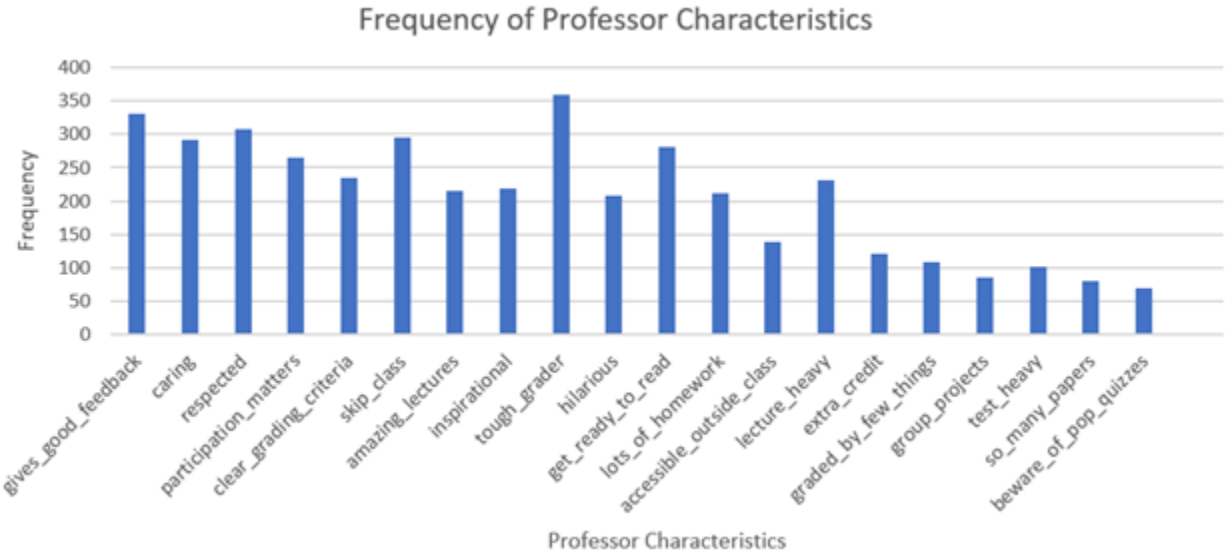


Figure 7:
Histogram of the frequency of professor characteristics



VIII. Appendix B. Supervised Learning

Table 1:
Results for Learning Task 1: Is_Good_Professor and Is_Bad_Professor

Model	Top 10 Words with Highest Coefficients (From Low to High)	Top 10 Words with Lowest Coefficients (From Low to High)
Is_Good_Professor	'fair', 'love', 'fun', 'excellent', 'wonderful', 'easy', 'amazing', 'awesome', 'best', 'great'	'worst', 'horrible', 'avoid', 'rude', 'doesnt', 'terrible', 'unclear', 'confusing', 'awful', 'boring'
Is_Bad_Professor	'boring', 'unclear', 'confusing', 'awful', 'rude', 'doesnt', 'terrible', 'avoid', 'horrible', 'worst'	'great', 'best', 'awesome', 'amazing', 'easy', 'fun', 'wonderful', 'willing', 'dr', 'love'

Figure 1:
Regular Confusion Matrix for Is_Good_Professor

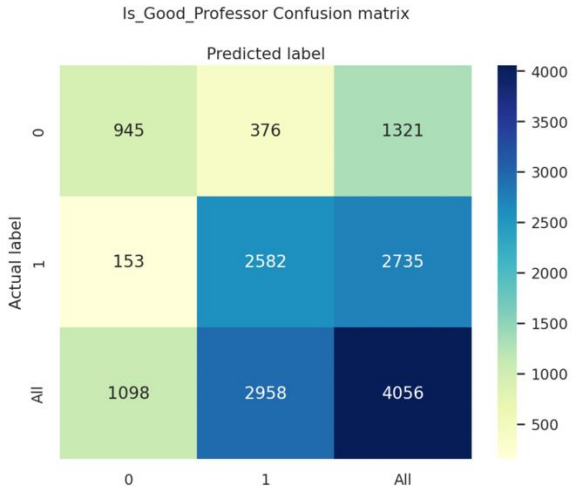


Figure 2:
Normalized Confusion Matrix for Is_Good_Professor

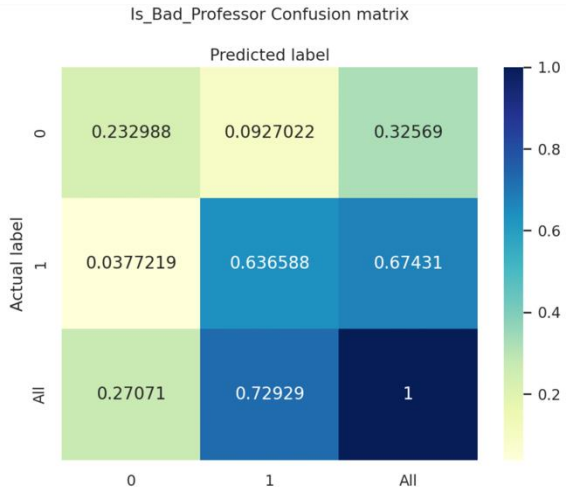


Figure 3:
Regular Confusion Matrix for Is_Bad_Professor

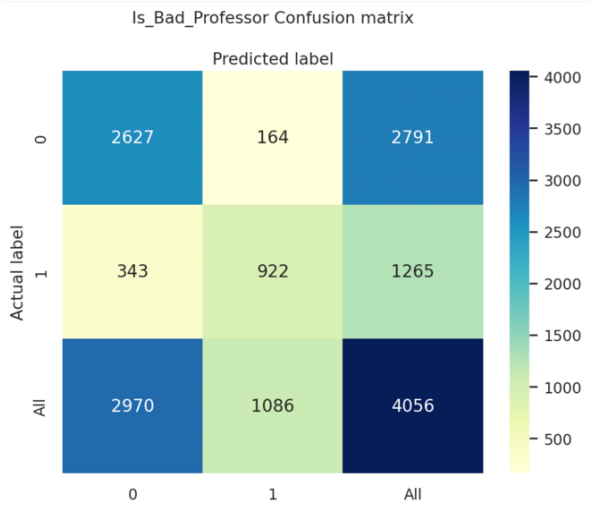


Figure 4:
Normalized Confusion Matrix for Is_Bad_Professor

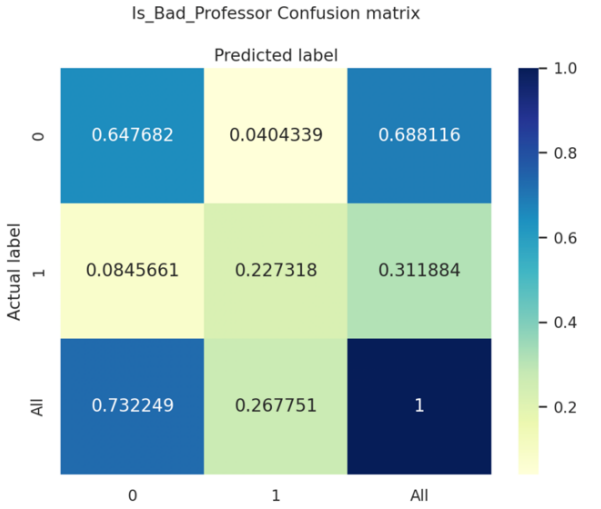


Table 2:
Classification Report for Is_Good_Professor

	Precision	Recall	F1-Score	Support
0 (Bad Review)	0.86	0.72	0.78	1321
1 (Good Review)	0.87	0.94	0.91	2735
Accuracy			0.87	4056
Macro Avg	0.87	0.83	0.84	4056
Weighted Avg	0.87	0.87	0.87	4056

Table 3:
Classification Report for Is_Bad_Professor

	Precision	Recall	F1-Score	Support
0 (Good Review)	0.88	0.94	0.91	2791
1 (Bad Review)	0.85	0.73	0.78	1265
Accuracy			0.88	4056
Macro Avg	0.87	0.84	0.85	4056
Weighted Avg	0.87	0.88	0.87	4056

Table 4:
K-Fold Cross Validation Report for Is_Good_Professor

Number of Folds	Average Accuracy
10	0.8742376655551782
25	0.8758409899370351
50	0.8770797720797718
100	0.8773877149132774

Table 5:
K-Fold Cross Validation Report for Is_Bad_Professor

Number of Folds	Average Accuracy
10	0.8742376655551782
25	0.8758409899370351
50	0.8770797720797718
100	0.8773877149132774

Table 6:
Results for Learning Task 2: Professor_Star_Rating

Outcome	Top 10 Words with Highest Coefficients (From Low to High)	Top 10 Words with Lowest Coefficients (From Low to High)
‘student_star’ = 1	'away' 'unclear' 'doesnt' 'unhelpful' 'terrible' 'awful' 'rude' 'horrible' 'avoid' 'worst'	great' 'best' 'easy' 'lot' 'awesome' 'willing' 'love' 'amazing' 'nice' 'interesting'
‘student_star’ = 2	'explanation' 'horrible' 'avoid' 'unorganized' 'doesnt' 'boring' 'confusing' 'worst' 'unclear' 'ramble'	'great' 'awesome' 'loved' 'fair' 'best' 'professor' 'fun' 'cool' 'dr' 'amazing'
‘student_star’ = 3	'hw' 'pretty' 'dry' 'attendance' 'lot' 'arent' 'hard' 'alright' 'overall' 'ok'	'helpful' 'amazing' 'life' 'best' 'worst' 'awesome' 'ive' 'job' 'excellent' 'prof'

‘student_star’ = 4	'cool' 'participate' 'enjoyed' 'fair' 'easy' 'willing' 'amazing' 'best' 'awesome' 'great'	'avoid' 'worst' 'horrible' 'doesnt' 'confusing' 'rude' 'waste' 'useless' 'unclear' 'luck'
‘student_star’ = 5	'nicest' 'hilarious' 'loved' 'fun' 'excellent' ‘wonderful' 'amazing' 'awesome' 'great' 'best'	'worst' 'unclear' 'confusing' 'terrible' 'horrible' 'rude' 'boring' 'doesnt' 'hard' 'avoid'

Figure 5:
Regular Confusion Matrix for Professor_Star_Rating

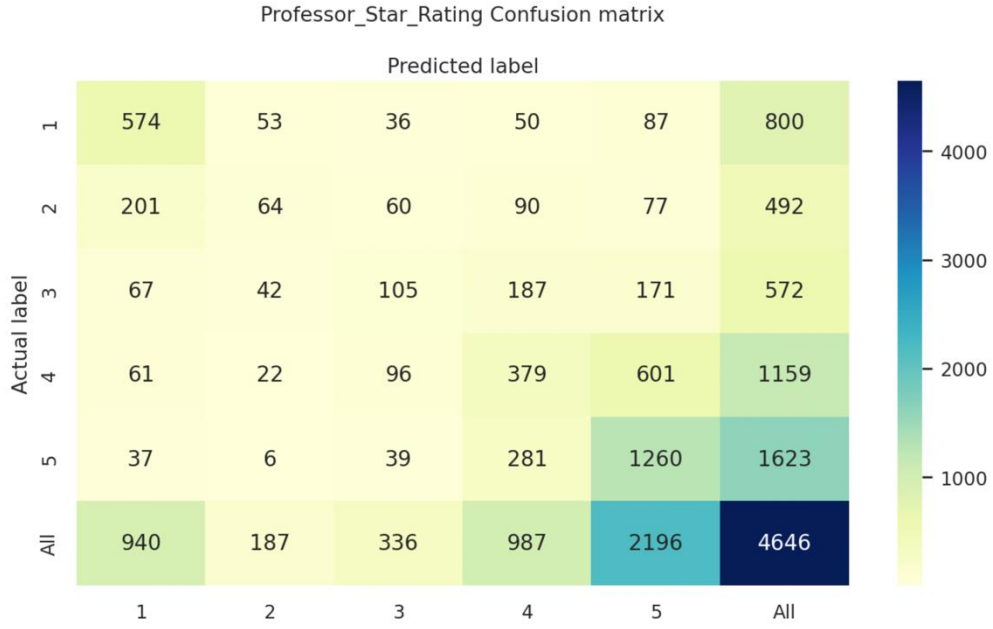


Figure 6:
Normalized Confusion Matrix for Professor_Star_Rating

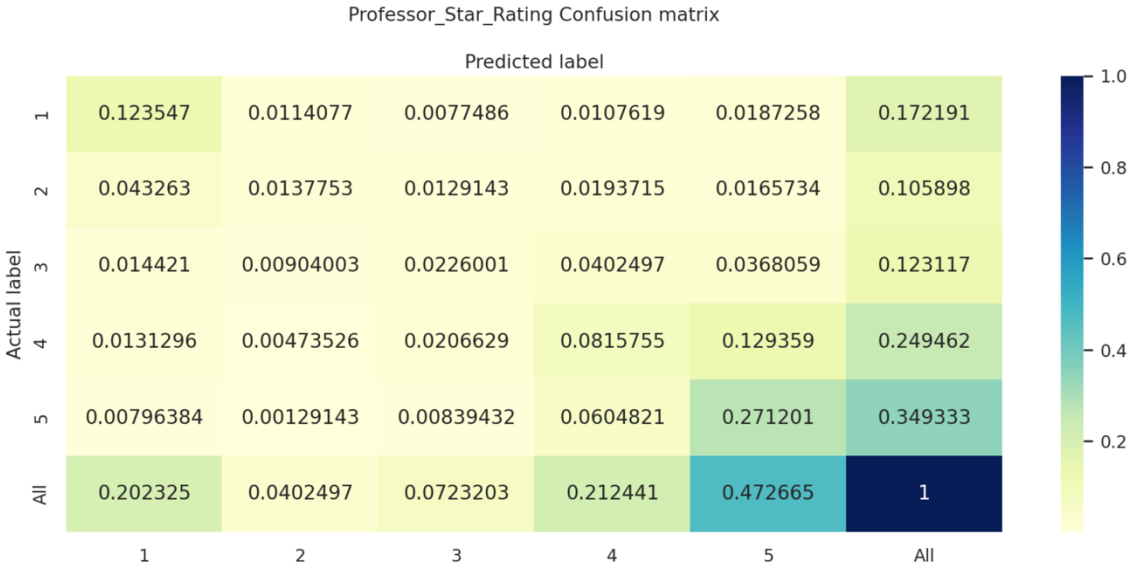


Table 7:
Classification Report for Professor_Star_Rating

	Precision	Recall	F1-Score	Support
1	0.61	0.72	0.66	800
2	0.34	0.13	0.19	492
3	0.32	0.18	0.23	572
4	0.38	0.33	0.35	1159
5	0.57	0.78	0.66	1623
Accuracy			0.51	4646
Macro Avg	0.44	0.43	0.42	4646
Weighted Avg	0.48	0.51	0.48	4646

Table 8:
K-Fold Cross Validation Report for Professor_Star_Rating

Number of Folds	Average Accuracy
10	0.5102796681665606
25	0.5082336936858709
50	0.5098345071443063
100	0.509893344957861

Table 9:
Results for Learning Task 3: Best_Worst_Average_Prof

Outcome	Top 10 Words with Highest Coefficients (From Low to High)	Top 10 Words with Lowest Coefficients (From Low to High)
Worst	'useless' 'unhelpful' 'terrible' 'awful' 'doesnt' 'unclear' 'avoid' 'rude' 'horrible' 'worst'	'great' 'best' 'easy' 'awesome' 'amazing' 'dr' 'lot' 'fun' 'willing' 'funny'
Average	'approach' 'alot' 'willing' 'ok' 'lot' 'nice' 'overall' 'okay' 'pretty' 'alright'	'worst' 'wish' 'retire' 'power' 'helpfull' 'tell' 'paid' 'rock' 'unhelpful' 'prof'
Best	'hilarious' 'fun' 'dr' 'easy' 'wonderful' 'excellent' 'amazing' 'awesome' 'best' 'great'	'worst' 'horrible' 'rude' 'terrible' 'unclear' 'confusing' 'avoid' 'doesnt' 'boring' 'awful'

Figure 7:
Regular Confusion Matrix for Best_Worst_Average_Prof

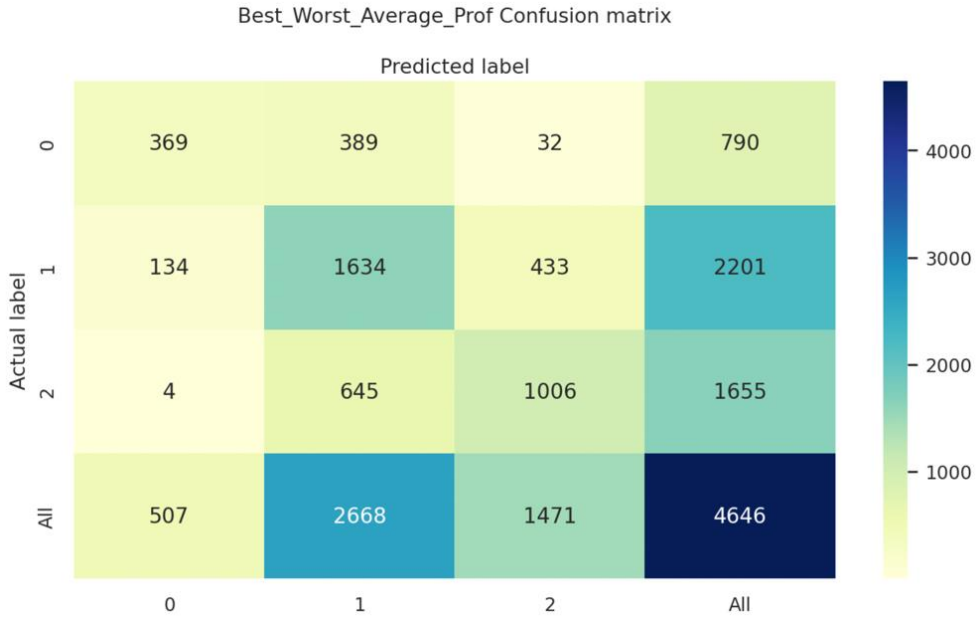


Figure 8:
Normalized Confusion Matrix for Best_Worst_Average_Prof

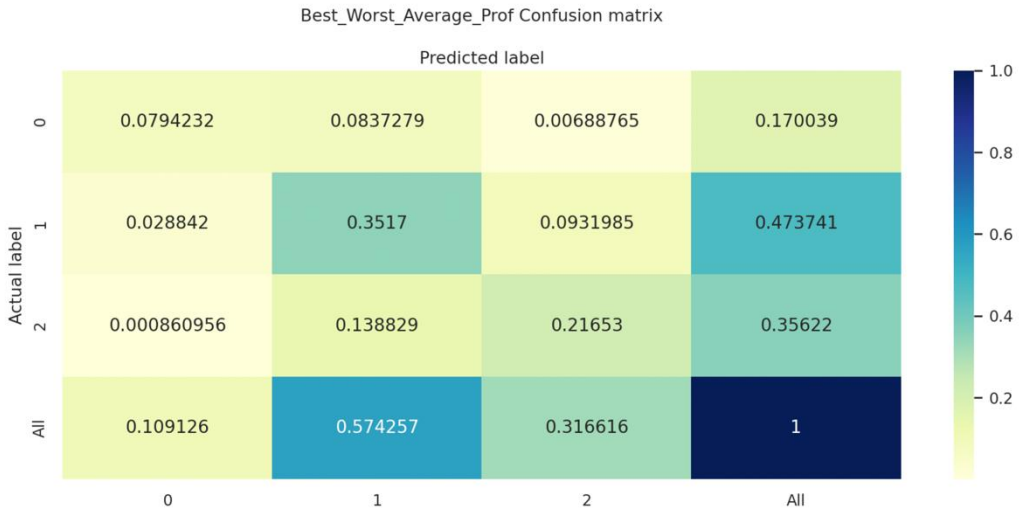


Table 10:
Classification Report for Best_Worst_Average_Prof

	Precision	Recall	F1-Score	Support
Worst	0.73	0.47	0.57	790
Average	0.61	0.74	0.67	2201
Best	0.68	0.61	0.64	1655
Accuracy			0.65	4646
Macro Avg	0.67	0.61	0.63	4646
Weighted Avg	0.66	0.65	0.64	4646

Table 11:
K-Fold Cross Validation Report for Best_Worst_Average_Prof

Number of Folds	Average Accuracy
10	0.6437406594399225
25	0.6435729894788637
50	0.6431887082282701
100	0.6429735541993606

Table 12:
Results for Learning Task 4: naive_bayes_Is_Good_Prof

Top 10 Words with Highest Coefficients (From Low to High)	Top 10 Words with Lowest Coefficients (From Low to High)
'make' 'lot' 'good' 'really' 'best' 'easy' 'professor' 'teacher' 'great' 'class'	'professor' 'student' 'like' 'worst' 'doesnt' 'teacher' 'hard' 'test' 'dont' 'class'

Figure 9:
Regular Confusion Matrix for naive_bayes_Is_Good_Prof

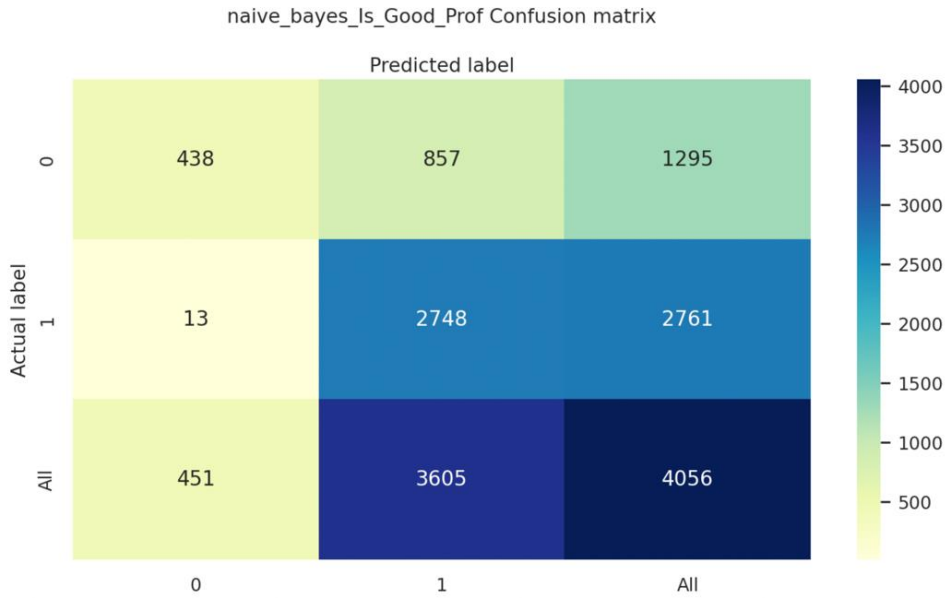


Figure 10:
Normalized Confusion Matrix for naive_bayes_Is_Good_Prof

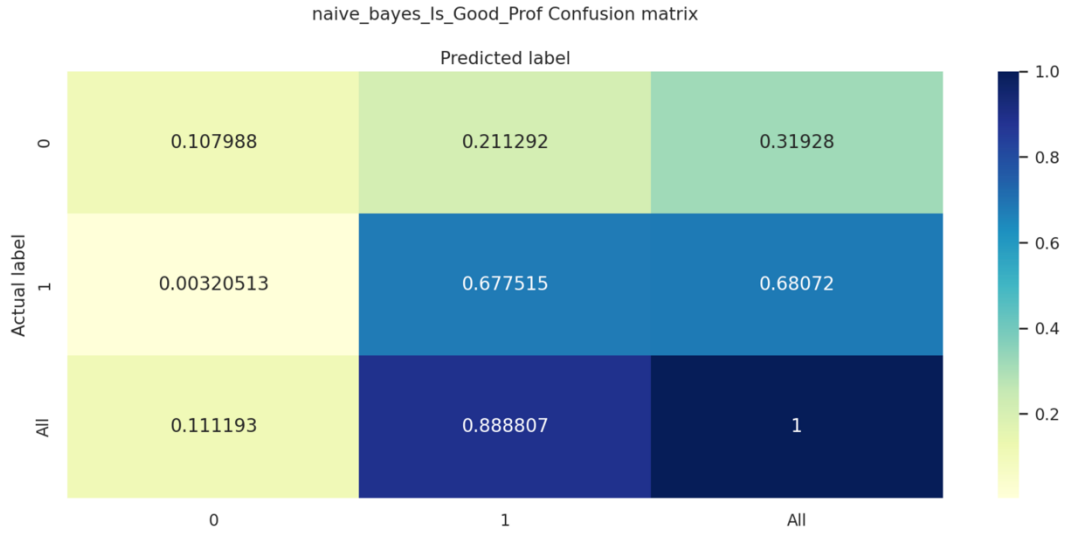


Table 13:
Classification Report for naive_bayes_Is_Good_Prof

	Precision	Recall	F1-Score	Support
0 (Bad Review)	0.97	0.34	0.50	1295
1 (Good Review)	0.76	1.00	0.86	2761
Accuracy			0.79	4056
Macro Avg	0.87	0.67	0.68	4056
Weighted Avg	0.83	0.79	0.75	4056

Table 14:
K-Fold Cross Validation Report for naive_bayes_Is_Good_Prof

Number of Folds	Average Accuracy
10	0.7990894607647618
25	0.8033461736004109
50	0.8050738841405507
100	0.8059615238960843

Table 15:
Results for Learning Task 5: naive_bayes_Prof_Star_Rating

Outcome	Top 10 Words with Highest Coefficients (From Low to High)	Top 10 Words with Lowest Coefficients (From Low to High)
‘student_star’ = 1	'test' 'hard' 'teach' 'professor' 'student' 'doesnt' 'teacher' 'dont' 'worst' 'class'	'00' 'myuniverse' 'mz' 'nacc' 'nad' 'nada' 'nag' 'nagging' 'nailed' 'nait'
‘student_star’ = 2	'good' 'boring' 'time' 'dont' 'grade' 'lecture' 'like' 'hard' 'test' 'class'	'00' 'nowhe' 'nowin' 'nowself' 'noy' 'np' 'npv' 'npvc' 'nsc' 'nscc'
‘student_star’ = 3	'time' 'nice' 'lot' 'lecture' 'really' 'easy' 'hard' 'good' 'test' 'class'	'00' 'nonsci' 'nonsense' 'nonsensical' 'nonsince' 'nontechnical' 'nontextbooky' 'nonthreatening' 'nonviolent' 'nonwestern'
‘student_star’ = 4	'hard' 'lot' 'test' 'professor' 'really' 'teacher' 'good' 'great' 'easy' 'class'	'zzzzzwhere' 'homeworkpaper' 'homeworkpractice' 'sarna' 'homeworkquizzestests' 'sargent' 'homeworkshe' 'homeworksome' 'sarcastici' 'sarcastically'
‘student_star’ = 5	'help' 'student' 'make' 'really' 'easy' 'best' 'professor' 'teacher' 'great' 'class'	'00' 'mineral' 'minibooks' 'minilecture' 'minilectures' 'minimum' 'miniquiz' 'minitest' 'minitests' 'minium'

Figure 11:
Regular Confusion Matrix for naive_bayes_Prof_Star_Rating

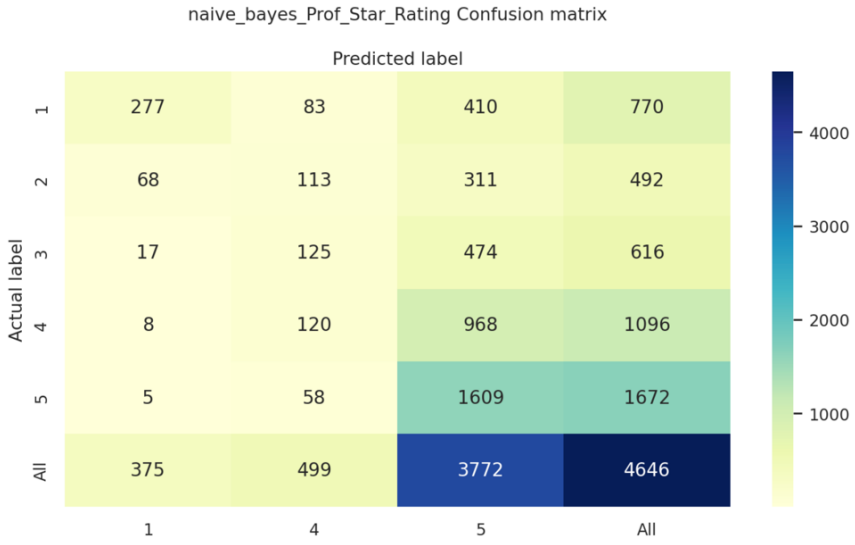


Figure 12:
Normalized Confusion Matrix for naive_bayes_Prof_Star_Rating

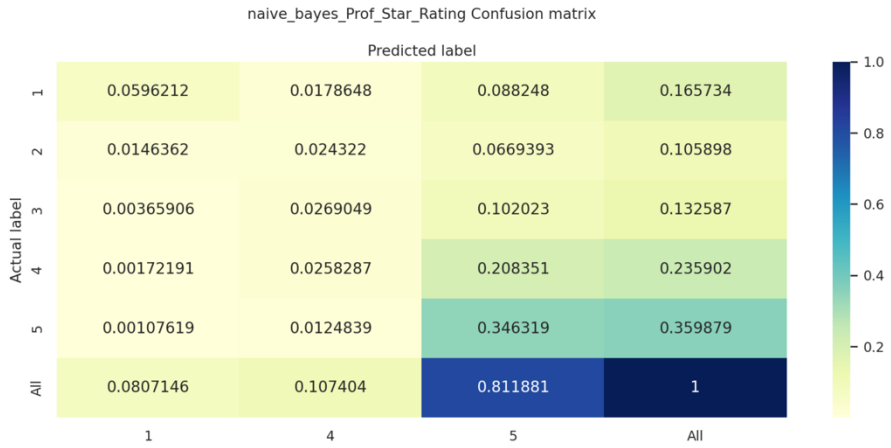


Table 16:
Classification Report for naive_bayes_Prof_Star_Rating

	Precision	Recall	F1-Score	Support
1	0.74	0.36	0.48	770
2	0.00	0.00	0.00	492
3	0.00	0.00	0.00	616
4	0.24	0.11	0.15	1096
5	0.43	0.96	0.59	1672
Accuracy			0.43	4646
Macro Avg	0.28	0.29	0.25	4646
Weighted Avg	0.33	0.43	0.33	4646

Table 17:
K-Fold Cross Validation Report for naive_bayes_Prof_Star_Rating

Number of Folds	Average Accuracy
10	0.5102796681665606
25	0.5082336936858709
50	0.5098345071443063
100	0.509893344957861

IX. Appendix C. Unsupervised Learning

Table 1:
Good Professors - Association with minimum support of 10%, minimum confidence of 10%

	Antecedents	Consequents	Support	Confidence
0	caring	gives_good_feedback	0.385159	0.694268
1	gives_good_feedback	caring	0.385159	0.641176
2	respected	gives_good_feedback	0.431095	0.677778
3	gives_good_feedback	respected	0.431095	0.717647
4	participation_matters	gives_good_feedback	0.275618	0.709091
…				
11629	amazing_lectures	clear_grading_criteria, respected, inspirational…	0.109541	0.252033
11630	inspirational	clear_grading_criteria, gives_good_feedback…	0.109541	0.234848
11631	hilarious	clear_grading_criteria, gives_good_feedback…	0.109541	0.254098
11632	caring	clear_grading_criteria, gives_good_feedback…	0.109541	0.197452
11633	gives_good_feedback	clear_grading_criteria, respected, inspirational…	0.109541	0.182353

[11634 rows x 4 columns]

Table 2:
Good Professors - Association with minimum support of 25%, minimum confidence of 75%

	Antecedents	Consequents	Support	Confidence
0	clear_grading_criteria	gives_good_feedback	0.293286	0.761468
1	clear_grading_criteria	respected	0.300353	0.779817
2	skip_class	respected	0.307420	0.763158
3	amazing_lectures	respected	0.339223	0.780488
4	inspirational	respected	0.378092	0.810606
5	hilarious	respected	0.332155	0.770492
6	caring, respected	gives_good_feedback	0.307420	0.756522
7	caring, gives_good_feedback	respected	0.307420	0.798165
8	respected, amazing_lectures	gives_good_feedback	0.257951	0.760417
9	amazing_lectures, gives_good_feedback	respected	0.257951	0.869048
10	gives_good_feedback, inspirational	respected	0.261484	0.870588
11	caring, inspirational	respected	0.265018	0.882353

12	caring, hilarious	respected	0.254417	0.847059
13	hilarious, respected	caring	0.254417	0.765957
14	hilarious, respected	amazing_lectures	0.257951	0.776596
15	hilarious, amazing_lectures	respected	0.257951	0.879518
16	respected, amazing_lectures	hilarious	0.257951	0.760417

Table 3:
Good Professors - Association with minimum support of 35%, minimum confidence of 10%

	Antecedents	Consequents	Support	Confidence
0	caring	gives_good_feedback	0.385159	0.694268
1	gives_good_feedback	caring	0.385159	0.641176
2	respected	gives_good_feedback	0.431095	0.677778
3	gives_good_feedback	respected	0.431095	0.717647
4	caring	respected	0.406360	0.732484
5	respected	caring	0.406360	0.638889
6	respected	inspirational	0.378092	0.594444
7	inspirational	respected	0.378092	0.810606

Table 4:
Good Professors - Association with minimum support of 40%, minimum confidence of 60%

	Antecedents	Consequents	Support	Confidence
0	respected	gives_good_feedback	0.431095	0.677778
1	gives_good_feedback	respected	0.431095	0.717647
2	caring	respected	0.406360	0.732484
3	respected	caring	0.406360	0.638889

Table 5:
Bad Professors - Association with minimum support of 10%, minimum confidence of 10%

	Antecedents	Consequents	Support	Confidence
0	caring	gives_good_feedback	0.161616	0.500000
1	gives_good_feedback	caring	0.161616	0.457143
2	respected	gives_good_feedback	0.151515	0.652174
3	gives_good_feedback	respected	0.151515	0.428571
4	participation_matters	gives_good_feedback	0.181818	0.529412
…				
3301	lecture_heavy	tough_grader, test_heavy, get_ready_to_read	0.101010	0.204082

3302	test_heavy	tough_grader, lecture_heavy, get_ready_to_read	0.101010	0.416667
3303	get_ready_to_read	tough_grader, lecture_heavy, test_heavy	0.101010	0.196078
3304	skip_class	tough_grader, lecture_heavy, test_heavy	0.101010	0.185185
3305	graded_by_few_things	ttough_grader, lecture_heavy, test_heavy	0.101010	0.434783

[3306 rows x 4 columns]

Table 6:
Bad Professors - Association with minimum support of 25%, minimum confidence of 65%

	Antecedents	Consequents	Support	Confidence
0	gives_good_feedback	tough_grader	0.292929	0.828571
1	skip_class	tough_grader	0.494949	0.907407
2	tough_grader	skip_class	0.494949	0.653333
3	get_ready_to_read	skip_class	0.343434	0.666667
4	lots_of_homework	skip_class	0.292929	0.690476
5	get_ready_to_read	tough_grader	0.444444	0.862745
6	lots_of_homework	tough_grader	0.383838	0.904762
7	lecture_heavy	tough_grader	0.424242	0.857143
8	lecture_heavy	get_ready_to_read	0.323232	0.653061
9	get_ready_to_read, skip_class	tough_grader	0.333333	0.970588
10	get_ready_to_read, tough_grader	skip_class	0.333333	0.750000
11	skip_class, tough_grader	get_ready_to_read	0.333333	0.673469
12	lots_of_homework, skip_class	tough_grader	0.272727	0.931034
13	lots_of_homework, tough_grader	skip_class	0.272727	0.710526
14	lecture_heavy, skip_class	tough_grader	0.292929	0.935484
15	lecture_heavy, tough_grader	skip_class	0.292929	0.690476
16	lecture_heavy, get_ready_to_read	tough_grader	0.303030	0.937500
17	lecture_heavy, tough_grader	get_ready_to_read	0.303030	0.714286
18	get_ready_to_read, tough_grader	lecture_heavy	0.303030	0.681818

Table 7:
Bad Professors - Association with minimum support of 35%, minimum confidence of 10%

	Antecedents	Consequents	Support	Confidence
0	skip_class	tough_grader	0.494949	0.907407
1	tough_grader	skip_class	0.494949	0.653333
2	get_ready_to_read	tough_grader	0.444444	0.862745
3	tough_grader	get_ready_to_read	0.444444	0.586667
4	lots_of_homework	tough_grader	0.383838	0.904762
5	tough_grader	lots_of_homework	0.383838	0.506667
6	lecture_heavy	tough_grader	0.424242	0.857143
7	tough_grader	lecture_heavy	0.424242	0.560000

Table 8:
Bad Professors - Association with minimum support of 40%, minimum confidence of 65%

	Antecedents	Consequents	Support	Confidence
0	skip_class	tough_grader	0.494949	0.907407
1	tough_grader	skip_class	0.494949	0.653333
2	get_ready_to_read	tough_grader	0.444444	0.862745
3	lecture_heavy	tough_grader	0.424242	0.857143