UNIVERSITY OF
WATERLOO

Faculty of Engineering
Department of Management Sciences

MSCI 446

Engineering Quiz Re-design

Laura Dobson
ID: 20536100

Rudra Bhatt
ID: 20516603

Johnson Kan
ID: 20270951

Sarah Watts
ID: 20515933

Professor: Lukasz Golab
TA: Shivangi Chopra

December 5th, 2016

Table of Contents

Business Insight **3**

Abstract 3

Motivation 3

Related work 3

Hypotheses 4

Beneficiaries 4

Data **4**

Description of data 4

Exploratory data analysis 7

General Exploration *7*

Exploration on Gender *7*

Exploration on Happy=No *9*

Data cleaning 10

Technical Analysis **10**

Association 10

Association Rule mining *10*

Apriori *11*

Clustering 18

Classification **18**

One R 18

Decision Tree and Random Forest 20

Naive Bayes with one class variable 24

Feature Selection *24*

Naive Bayes with two class variables 30

Extensions *33*

Naive Bayes Iterations *34*

Comparison of models 35

Recommendation **37**

Future Steps **38**

Conclusion **38**

Business Insight

Abstract

This report will explore prediction and association models utilized to determine which program engineering students should take based on their answers to categorical features. These models were developed by collecting 944 responses from University of Waterloo Engineers. The data was cleaned and features that were not required to predict the class variable were also removed. The results of the models based on accuracy and k-fold validation were compared for the Naïve Bayes theorem, Random Forest, Decision Trees, the One R algorithm and through Exploratory Data Analysis.

Motivation

This project was undertaken because the current engineering quiz does not include Biomedical Engineering which is a new Engineering program introduced in the University of Waterloo. Another reason this project was undertaken is because all group members are genuinely interested in improving the Waterloo Engineering quiz. All members are all interested in helping potential first years decide which Engineering discipline best suits them. Improvements will be made to the current methodology with the current programs, while incorporating the new Biomedical Engineering program. The current engineering quiz is based on 598 responses, 50% of which could not be used due to missing attributes. The goal of the new engineering quiz is to improve the response rate and quality of data. This is being done so more accurate data is collected, and more accurate data can be used so potential first years can pick the best possible discipline for themselves. The current engineering quiz had a weighted average performance of 52%, where the goal of this re-design is to improve the accuracy of the engineering quiz.

Related work

To develop this report, the methodology in the current engineering report, which was done a couple of years ago without including the Biomedical Engineering program was analyzed. The current engineering quiz utilizes J48 decision trees, Naive Bayes, Averaged One-dependence estimators with subsumption resolution (AODEsr), binary classification and One R. The features utilized in current model that provided a high accuracy were carried over to help create the re-designed features for Naïve Bayes theorem, Random Forest method, Exploratory Data Analysis, and Decision Trees. Related articles involving information on Naïve Bayes theorem and its benefits has been studied to form a stronger understanding of the topics discussed.

Hypotheses

Results from exploratory interviews with current University of Waterloo Engineering students indicate that students select their program because of the domain they want to expand their knowledge in. For example, computer engineers enjoyed their program because they wanted to learn about hardware and software. Based on over qualitatively 40 interviews, key features that focused on the area of study of each program were incorporated with a hypothesis that they would improve the results of the quiz.

Beneficiaries

This report has been developed to provide a new method for the University of Waterloo to decide which engineering program high school students should choose.

Data

Description of data

To re-design the engineering quiz, 944 responses to a preliminary quiz have been collected for 21 features and a class variable (Table 1).

Table 1: Features and class variable

	Feature	Reference Name*	Values of the feature
1	What program are you in currently?	Program (class variable)	1. Biomedical Engineering 2. Chemical Engineering 3. Civil Engineering 4. Computer Engineering 5. Electrical Engineering 6. Environmental Engineering 7. Geological Engineering 8. Management Engineering 9. Mechanical Engineering 10. Mechatronics Engineering 11. Nanotechnology Engineering 12. Software Engineering 13. Systems Design Engineering
2	What year are you in?	Year	1. Year 1 2. Year 2 3. Year 3 4. Year 4 5. Alumni

3	Are you happy with your program?	Happy	1. Yes 2. No
4	What field interests you the most?	Field_interest	1. Finance 2. Manufacturing 3. Software Development 4. Research 5. Construction
5	I would like to have been part of the team that was responsible for:	Team_responsible	1. The Automobile 2. The Internet 3. The CN Tower 4. Solar Panels 5. The Robot 6. Penicillin
6	Who would be your dream professor?	Dream_professor	1. Mark Zuckerberg, Founder of Facebook 2. Elon Musk, Founder of Tesla 3. Madam Marie Curie, Noble Price Winner for work on radioactivity 4. Warren Buffett, Billionaire American Business Men 5. David Suzuki, Environmental Activist
7	I would enjoy working...	Work_setting	1. In a laboratory 2. In a manufacturing facility 3. In an office 4. In the outdoors
8	An ideal outfit to wear to work would be:	Work_outfit	1. Formal 2. Business Casual 3. Extremely Casual 4. Protective Clothing
9	When you start work, ideally you would want to...	Start_work	1. Work for a large corporation 2. Work for a small company 3. Work in a start up
10	I would rather design something that...	Design_something	1. I can see it function and prove it works, but can't necessary touch 2. I can see function and tangibly touch the parts
11	I would rather analyze...	Analyze_something	1. Blue prints and design drawings 2. Graphs and charts 3. Test results
12	I would rather gather information for solving a problem...	Solve_problem	1. By analyzing the design and implementation of the system 2. By talking to the people who use the system 3. By analyzing the inputs and outputs of the system 4. By analyzing the internal components of the system
13	I prefer to...	Prefer_to	1. Listen 2. Talk 3. Write 4. Read

14	I would rather work as part of a...	Work_partof	1. Focused team 2. Multidisciplinary team
15	When working on a project would you rather	On_project	1. Know all the requirements for the project at the beginning 2. Continuously change and update requirements as you go
16	What cause do you care about the most?	Care_most	1. Health 2. Environment 3. Human Rights 4. Bringing technology to developing countries
17	When setting goals for yourself you find yourself thinking about...	Goals	1. Five years from now 2. One year from now 3. A week from now 4. Today
18	You are most likely to eavesdrop in a conversation regarding	Eavesdrop	1. A groundbreaking metal that will create stronger cars, planes and spaceships for a cheaper cost 2. A new way to deal with movement in the Earth's surface to protect buildings from earthquakes 3. A microchip that is smaller than usual but twice as powerful as before 4. A factory that is able to 100% rely on robotics while producing no defected products
19	What subject would you like to develop your skills in?	Develop_skills	1. Math 2. Science 3. Physics 4. Technology
20	In an ideal setting, you are working with:	Working_with	1. People 2. Computers 3. Business Processes 4. Machinery
21	Would you prefer to be the leader on a team?	Leader	1. Yes 2. No
22	What is your gender? Your answer will only be used for data collection purposes.	Gender	1. Male 2. Female 3. I choose not to specify

reference name to be used throughout the remainder of the report

Only 19 features will be used to predict the class variable, which engineering program the student is currently in, and the 3 remaining features, Year, Happy and Gender of the participant have been used for data cleaning purposes. The records spanned all engineering disciplines, demographics and personality traits, where all features

and class variables were required fields for respondents to complete to ensure quality of the data. A threshold of 10% representation of responses by program was set to ensure there was a sufficient representation of all values of the class variable.

Exploratory data analysis

General Exploration

Each feature was graphed based on their responses by frequency of each response. The features with the highest skew towards a particular response were the questions “What area would you want to develop your skills in” and “Who would be your dream Professor” (Fig 2). These features showed a large bias towards a particular response and were removed during the data cleaning stage because these bias responses will not have strong predictive powers.

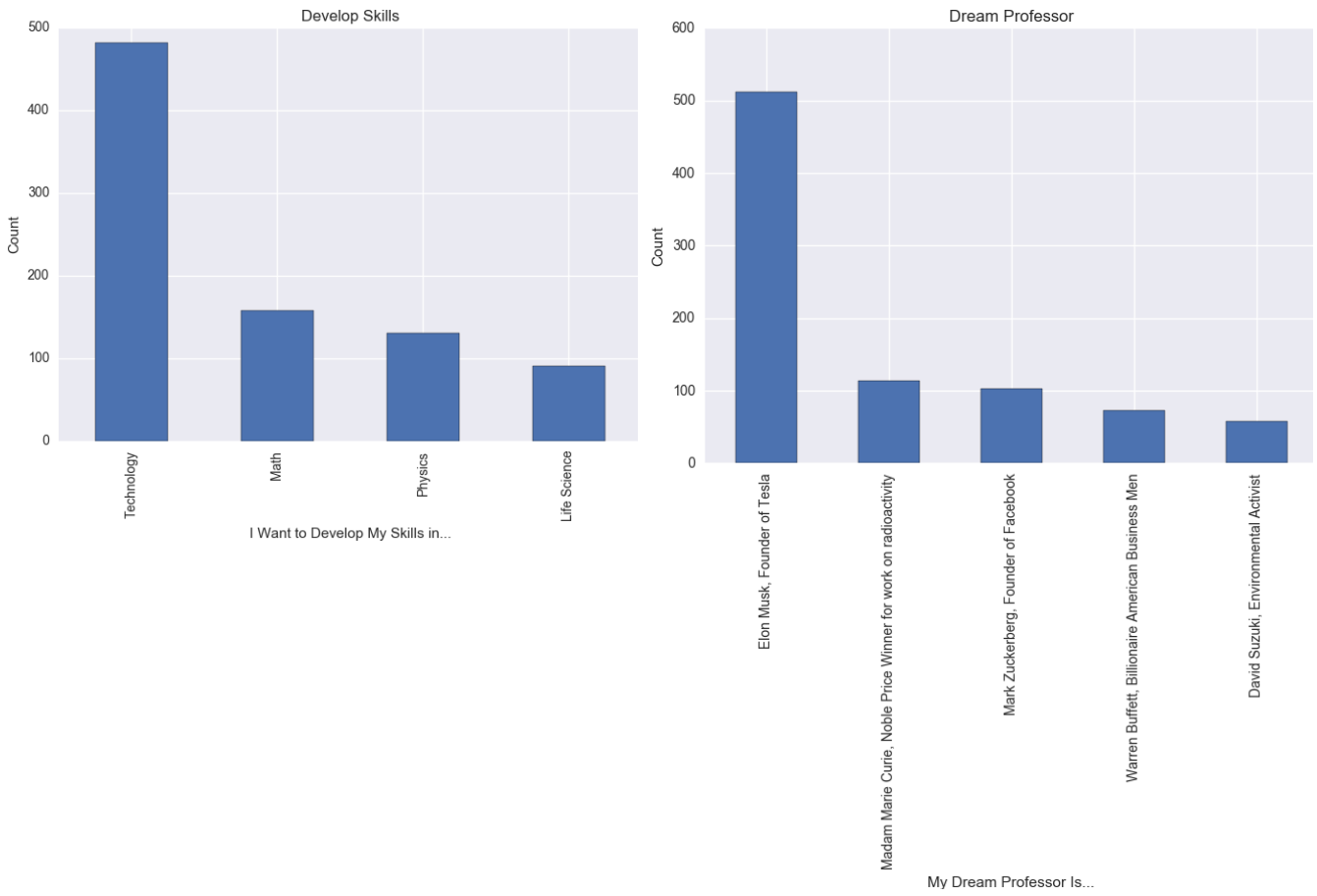


Figure 1: Features removed due to bias

Exploration on Gender

Overall the quiz was filled out by 34% by females and 66% by males. The breakdown of number of people per gender by program who completed the survey is shown in Figure 2.

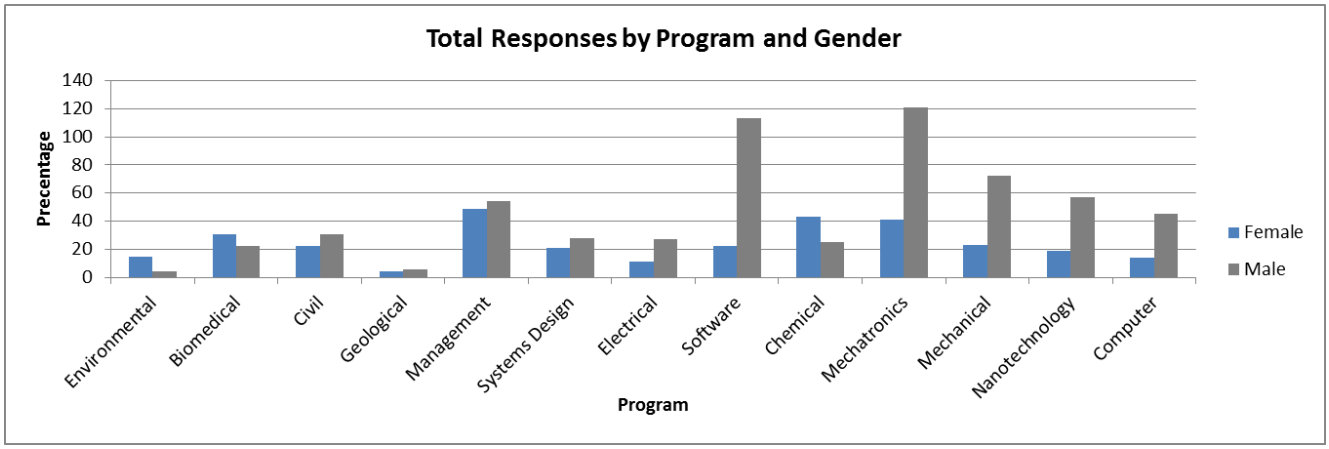
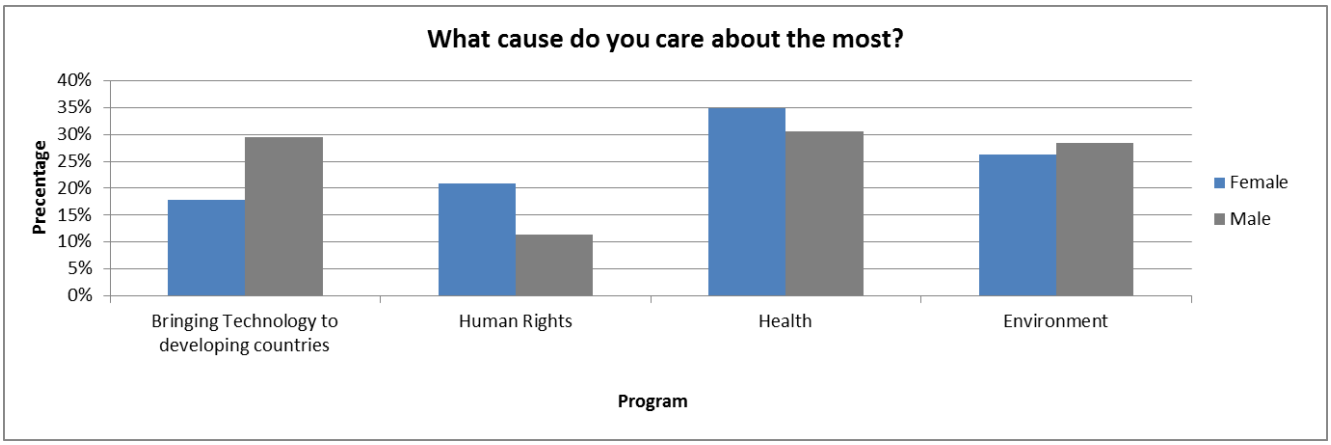


Figure 2: Total number of responses by Gender

All features have been analyzed to determine if there is an underlying gender bias. The responses to the questions were graphed as a percentage of the total number of female or male responses to normalize the male and female response quantity.

Two questions showed a gender bias by over a 10% deviation for the expected values of the responses. The questions are shown graphically in Figure 3.



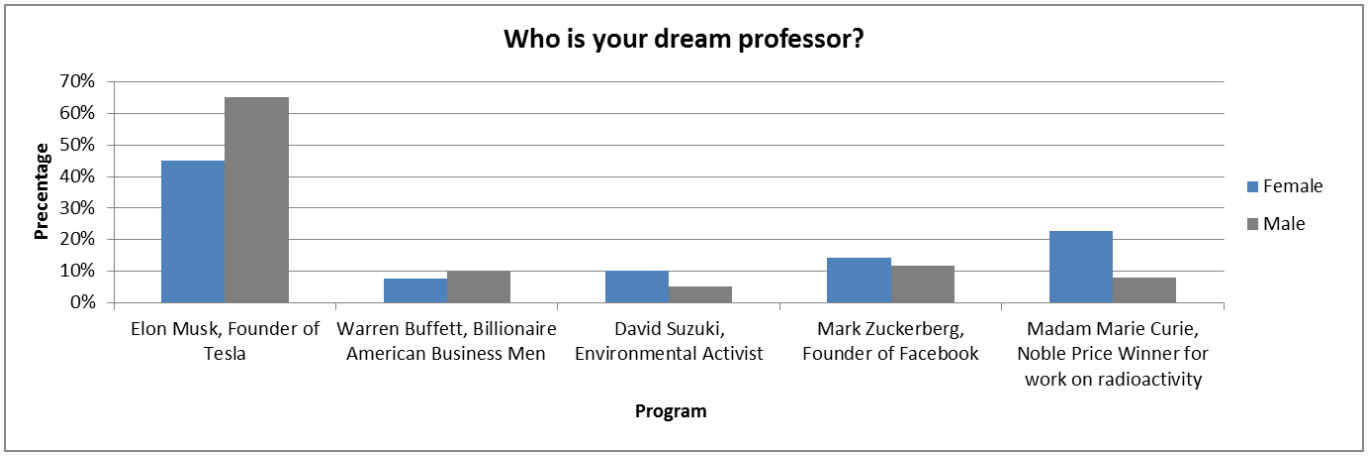


Figure 3: Questions with an underlying gender bias

Based on the responses to these questions, males have a bias towards bringing technology to developing worlds as a "cause they would like to help" and females have a bias towards human rights. The question "who is your dream professor" found that females have a bias towards Madam Marie Curie and males have a bias towards Elon Musk. The dream professor question has been removed from the feature set due to an overall bias towards Elon Musk and this will help ensure this gender bias does not interfere with the quiz results.

Exploration on Happy=No

The number of students who were unhappy in their program has been analyzed. The number of unhappy students as a percentage of their total program is presented in Figure 4. It was found that only 13% of students were unhappy in their programs overall.

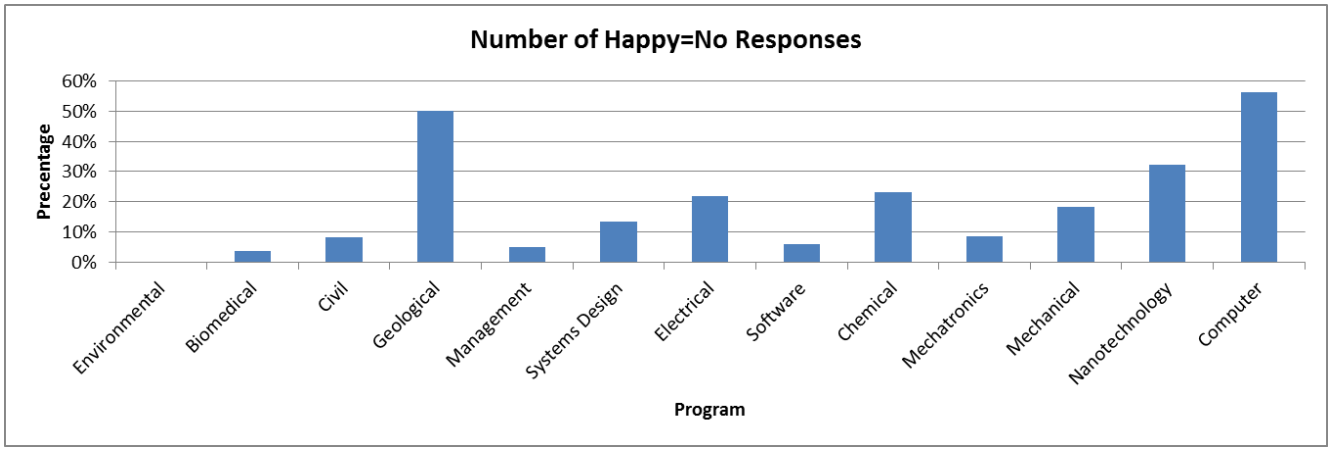


Figure 4: Number of unhappy students by program

The three programs with the highest level of unhappiness were Computer, Geological and Nanotechnology. It is important to note that Geological Engineering only had 8 responses present in our records, 4 of which were unhappy. Due to the small quantity of responses Geological Engineering has not been further analyzed. Computer and Nanotechnology Engineering were further analyzed and are represented in Figure 5.

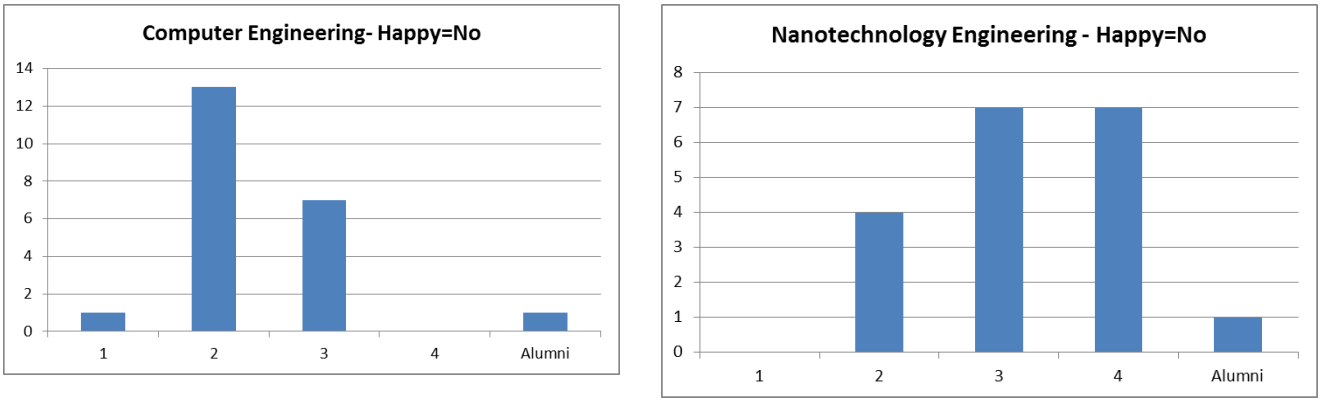


Figure 5: Unhappy students in Computer and Nanotechnology Engineering by year

Computer Engineering students were predominately unhappy in their second year of study. Through further interviews with Computer Engineering students a hypothesis was created that due to the large workload present in the second year of study for Computer Engineers they are more likely to be unhappy with their program. Additional investigation was also conducted with a sample of Nanotechnology Engineers in third and fourth year, the time when they are more likely to be unhappy. Based on the interviews it is hypothesized that due to the nature of their program it is difficult to secure fulltime employment outside of research fields and this has led to some dissatisfaction of students in their program.

Data cleaning

The data was cleaned to remove the "happiness" field, where only records where happiness = "yes" are retained. This is because people who select "no" will not have answers that support why they are happy with their degree. Gender and year of study were also removed as features to predict which discipline students should be predicted in, because the goal was to remove year and gender bias from the results. All features were required fields to be filled out by the participants so no data cleaning was required for missing fields.

Technical Analysis

Association

Association Rule mining

Association rule mining aids in the process of determining if any relationships exist amongst the attributes in a given dataset. If a rule is discovered, an if-then statement is produced concerning the selected values for the attribute. In detail, a list of *n* attributes denoted as $I = \{I_1, I_2, \dots, I_n\}$ represents attributes called items. Let $D = \{T_1, T_2, \dots, T_n\}$ represent a list of transactions defined as the database. This can be thought of as a record within a database. Each transaction in the database consists of a subset of the items in I. A rule is then formed by the

notation $I_1 => I_2$, where the item sets on the left hand side are antecedents and the item sets on the right hand side are consequents.

Association rule mining determines all possible rules that satisfy a minimum support and confidence that are pre-defined by the user. The goal of the algorithm is to determine which rules exceed these threshold limits, indicating that those given item sets are large/frequent and appear in a high percentage of the data. In detail, the support of an item set is equal to the number of transactions that contain a specific value for the corresponding item. This number is represented as a percentage of the number of transactions containing I_1 out of all transactions in D. The confidence of a particular rule is defined as the number of transactions that contain $I_1 => I_2$ out of the total number of transactions in D. For example, if the confidence for a given rule is 95%, this provides insight to the user that 95% of the total transactions in D that contain I_1 also contain I_2.

Apriori

To run association rule mining on the engineering quiz dataset, the Apriori algorithm was selected. This is because it is an easy algorithm to implement that scales efficiently to find an association with a desired support threshold, s, and confidence threshold, c. All features were considered, including the class feature, which program each transaction belongs to. However, gender was removed to ensure the integrity of the quiz by respecting all participant's wishes of not wanting to classify any results based upon their sex. Additionally, following the data cleaning process, all transactions where the answer to the Item "Are you happy in your program?" is "No" were removed.

Picking the appropriate threshold values for confidence and support is difficult as this is an unsupervised process. For high values of support and confidence, Apriori may generate very few or no rules. This implies some rules may be missed that provide good insight into the data. Contrary, if low threshold values are used, the algorithm may generate a large number of rule, providing too much information that may not be significant to the data.

Therefore, the Apriori algorithm will be run in an "iterative" fashion. With each iteration, the hyper-parameters will be altered to try and determine if the algorithm can output significant and insightful rules. Please note that more than three iterations were performed. However, the following sections show the thought process between each iteration and how the results of the algorithm were analyzed. The following three iterations show significant difference between the rules, thus why they were chosen to explain.

Iteration 1

Hyper-parameters

For the first iteration, the confidence and support thresholds were as follows:

minSupport = 0.15

minConfidence = 0.75

This produced an output of 99 rules. Some of the generated rules provided insight and matched accurately to attributes of certain disciplines. For example, the following rule relates to the engineering programs that require building physical objects.

Rule: ('Blue prints and design drawings', 'Know all the requirements for the project at the beginning') ==> ('I can see function and tangibly touch the parts',), 0.777

When engineers design and build a product, they begin by determining the requirements and developing a technical drawing to reflect these. They then will build the object based upon the drawing. It is suspected that the disciplines relating to this rule are the programs that are not software/technology related.

However, many of the generate rules provided very little to no insight on the data, such as:

Rule: ('People', 'Technology') ==> ('Multidisciplinary team',), 0.772

This indicates that the minimum support and confidence thresholds may be increased to produce more meaningful rules. Therefore, for this iteration, the support hyper-parameter was increased to

minSupport = 0.2

minConfidence = 0.75

Results

Using these hyper-parameters, 26 rules were generated summarized in table 2.

Table 2: Rules Generated from Iteration 1

Rule 1: ('Business Casual',) ==> ('Yes',) , 0.751
Rule 2: ("I can see it function and prove it works but can't necessary touch", 'In an office') ==> ('Software Development',) , 0.751
Rule 3: ('Multidisciplinary team',) ==> ('Yes',) , 0.753
Rule 4: ('Yes', "I can see it function and prove it works but can't necessary touch") ==> ('Multidisciplinary team',) , 0.754
Rule 5: ("I can see it function and prove it works but can't necessary touch", 'Multidisciplinary team') ==> ('Yes',) , 0.760
Rule 6: ('Software Development', 'In an office') ==> ("I can see it function and prove it works but can't necessary touch",) , 0.761
Rule 7: ('Elon Musk Founder of Tesla', 'Business Casual') ==> ('Yes',) , 0.761
Rule 8: ('Elon Musk Founder of Tesla', 'Multidisciplinary team') ==> ('Yes',) , 0.761
Rule 9: ('People',) ==> ('Yes',) , 0.764
Rule 10: ('Software Development', "I can see it function and prove it works but can't necessary touch") ==> ('In an office',) , 0.764
Rule 11: ('Yes', 'Business Casual') ==> ('Multidisciplinary team',) , 0.766

Rule 12: ('The Internet',) ==> ("I can see it function and prove it works but can't necessary touch",) , 0.768
Rule 13: ('I can see function and tangibly touch the parts', 'People') ==> ('Yes',) , 0.774
Rule 14: ('People',) ==> ('Multidisciplinary team',) , 0.776
Rule 15: ('The Internet',) ==> ('Software Development',) , 0.784
Rule 16: ('People', 'Business Casual') ==> ('Multidisciplinary team',) , 0.784
Rule 17: ('Continuously change and update requirements as you go', 'Multidisciplinary team') ==> ('Yes',) , 0.786
Rule 18: ('People', 'Business Casual') ==> ('Yes',) , 0.794
Rule 19: ('Business Casual', 'Multidisciplinary team') ==> ('Yes',) , 0.796
Rule 20: ('People', 'Multidisciplinary team') ==> ('Yes',) , 0.803
Rule 21: ('Yes', 'Continuously change and update requirements as you go') ==> ('Multidisciplinary team',) , 0.806
Rule 22: ('Elon Musk Founder of Tesla', 'People') ==> ('Yes',) , 0.813
Rule 23: ('Yes', 'People') ==> ('Multidisciplinary team',) , 0.816
Rule 24: ('Five years from now',) ==> ('Yes',) , 0.817
Rule 25: ('Five years from now', 'Multidisciplinary team') ==> ('Yes',) , 0.844
Rule 26: ('Talk',) ==> ('Yes',) , 0.899

Insights

By increasing the support by 5% this reduced the number generated rules by 76. This means that the accuracy of the rule also increased, by ensuring all items in the rule are in at least 20% of the total transactions in the data.

Similarly, to the insights previously discussed, these threshold values generate some responses that are extremely insightful while others do not provide any interesting similarities or regularities to the transactions within the dataset.

The following rule by itself does not provide enough insight into why people who selected they would prefer to work in a multidisciplinary team would also prefer to be the leader:

Rule: ('Multidisciplinary team',) ==> ('Yes',), 0.753

As it is a multidisciplinary team, someone can take on many roles such as (but not limited to) developer, tester, manufacturer or project manager. However, a more interesting rule containing these two items is as follows:

Rule: ('Yes', 'People') ==> ('Multidisciplinary team',) , 0.816

With 81.6% confidence, the participants who selected they would prefer to be a leader and work with people also selected they would like to work on a multidisciplinary team. This may mean that the participants relating to this transaction may like to take a “project manager” role in the work force post-graduation.

Unfortunately, a majority of these rules had a low confidence by falling below 80%. Therefore, for the next iteration, the confidence will be increased by 10% while the support will be lowered by 5%.

Iteration 2

Hyper-parameters

The confidence and support intervals were altered until reaching values that provided more significant results. For this iteration, the following support and confidence levels were used:

minSupport = 0.15

minConfidence = 0.85

Results

Using the threshold limits listed in the previous section, 7 rules were generated that provided more general insight into the engineering data. These rules are summarized in table 3.

Table 3: Rules Generated from Iteration 2

Rule 1: ('In an office', 'The Internet') ==> ('Software Development',) , 0.851
Rule 2: ('Five years from now', 'Business Casual') ==> ('Yes',) , 0.851
Rule 3: ("I can see it function and prove it works but can't necessary touch", 'Computers') ==> ('Software Development',) , 0.857
Rule 4: ('In an office', "I can see it function and prove it works but can't necessary touch", 'The Internet') ==> ('Software Development',) , 0.862
Rule 5: ('Talk',) ==> ('Yes',) , 0.899
Rule 6: ('Multidisciplinary team', 'Talk') ==> ('Yes',) , 0.912
Rule 7: ('Software Engineering',) ==> ('Software Development',) , 0.970

Insights

Interesting insight may be drawn from each of the rules listed in Table 3. Rule 1 describes students who may be in a technology/software related discipline such as Software, Computer, Systems or Management engineering. The common jobs for each of these faculties tend to take place in office settings and have aspects of software development. The intent of including the option “The internet” or the attribute “I would want to be on the team responsible for creating...” was to capture these students from the disciplines previously listed.

With the same reasoning, Rules 3 and 4 capture these subset of students as well. By including the option of “developing a product that you can see however not necessarily touch “for the question “I want to design something…”, this proves the intent of trying to capture people who are interested in software development. In software development, you can always see function (e.g. when you successfully build code and it works) as it correctly outputs what it is intended to, however, you may not necessarily be able to touch it (e.g. a website). Thus, Rules 1, 3 and 4 agreed with the hypothesis of these options being selected in conjunction with each other.

It is important to note that with 91.2% confidence, Rule 6 “adds on” to the rule discussed in iteration 1 regarding working multidisciplinary teams and being a leader. In this iteration, Rule 6 better depicts why people who work in multidisciplinary teams also like being a leader. This is because they also prefer to talk rather than read, write or listen. By preferring to talk, this indicates that these participants like to stay connected with all team members, possibly indicating they like to organize and have a valued opinion on the activity occurring within the team, hence, why they may like to be the leader. For this same reasoning, it is suspected that Rule 5 was generated with 89.9% this way.

Although these results are extremely insightful, the next iteration includes increasing the confidence by 5% and removing the attributes of “Who is your dream professor?” and “What would you want to further develop your skills in?” These attributes were removed to reflect the data we used in the prediction models. These attributes were removed because they displayed bias in the Exploratory Data Analysis portion of the project. However, based on the nature of the rules already generated, it was hypothesized that this would not alter the rules generated in the third iteration.

Iteration 3

Hyper-parameters

The following minimum support and confidence intervals were used in the third iteration. Also, the Dream Professor and Develop Skills attributes were removed.

minSupport = 0.15

minConfidence = 0.9

Results

Two rules were generated utilizing the parameters listed in the previous section. These rules are summarized in Table 4.

Table 4: Rules Generated from Iteration 3

Rule 1: ('Talk', 'Multidisciplinary team') ==> ('Yes',) , 0.912
Rule 2: ('Software Engineering',) ==> ('Software Development',) , 0.970

Insights

Table 4 lists two rules with extremely high confidence. Rule 1 was discussed in iteration 2, however, it is important to note that by increasing the support to 20%, this rule still holds true. That means that the support of these three items in conjunction with each other represents 20% of the transactions for these three attributes. It is also

interesting that 91% of the participants who selected they like to talk and work in an interdisciplinary team also selected they like to be the leader.

Rule 2 generates the highest confidence out of iterations performed of the Apriori algorithm. With 97% confidence, if the program of the participant is software engineering then their field of interest is software development. This rule is extremely intuitive and it resulted in 129 records of the database.

A total of 132 software engineers took the quiz, thus this is a great finding to explain that almost all software engineers are interested in software development. This encouraged more exploratory data analysis to see what the remaining 4 participants chose. It was interesting to find that all four of these participants chose "Research" for this attribute instead.

General Insights

There are three major insights drawn from the rules generated through all iterations of the Apriori algorithm. The first major insight is that many of the rules showed bias towards the attributed "Would you want to be a leader on a team?" and "I want to work on…" Both of these teams only included two possible answers. Thus, the likelihood of these two answers being selected in conjunction with other is extremely high. This is proven in Figure 6. Also, it is apparent that selecting "Yes" and "Multidisciplinary Teams" for the respective attributes were the most common answers, both with approximately 600 entries in total. This gives a good understanding as to why these two items frequently appeared in the rules generated.

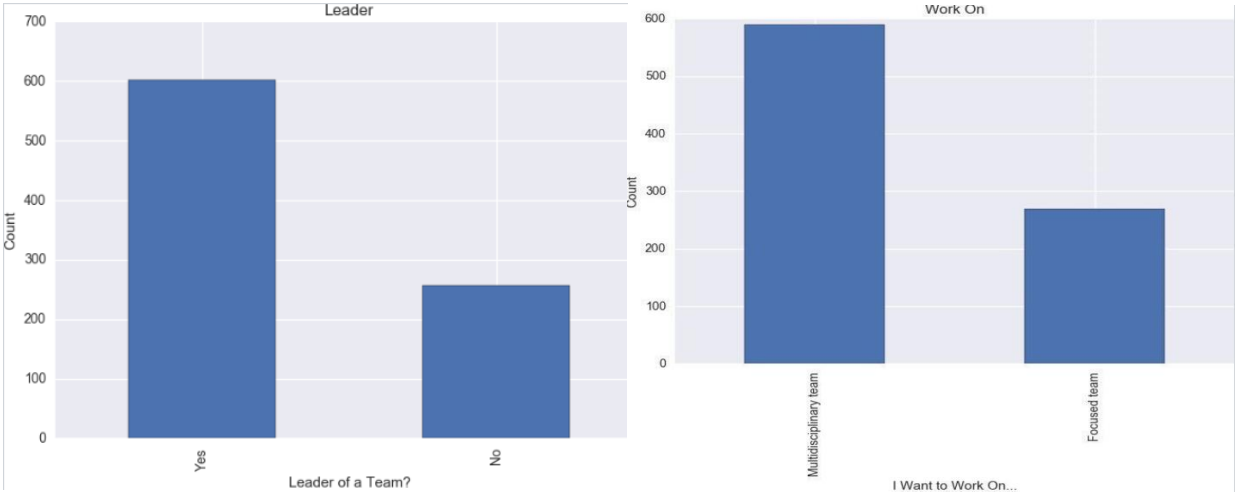


Figure 6: Features with bias towards particular values

The second general insight is that there is a huge bias towards software engineering and software development. In every iteration, a majority of the rules could be applicable to these two item sets. This probed more exploratory data analysis regarding these two features. One hundred and twenty-nine Software Engineers took the survey.

That represents 15% of our total data. Secondly, 326 participants selected “Software Development” as their field of interest. This represents 35% of the total data. This provides reasoning into why many of the rules were applicable to these two features.

The third general insight is a hypothesis as to why the response to field of interest being software development is so high. It is suspected that because the disciplines with the highest count of software development are all programs that have a heavy focus on technology, specifically with software. These results are summarized in Table 5.

Table 5: Count of programs that selected Software Development as their Field of Interest

Program	Count of program
Software Engineering	129
Mechatronics Engineering	70
Management Engineering	35
Computer Engineering	33
Systems Design Engineering	30
Electrical Engineering	11
Biomedical Engineering	9
Mechanical Engineering	5
Nanotechnology Engineering	2
Civil Engineering	2
Grand Total	**326**

This is because the number of responses per each discipline fell below the expected threshold of 10%. This target was set to ensure we have an accurate population of responses for each discipline. As depicted in Figure 7, the programs who had a low count of “Software Development” fell below the threshold. These programs include Civil, Chemical, Environmental, Geological and Mechanical. This may be a reflection of the number of people in each discipline.

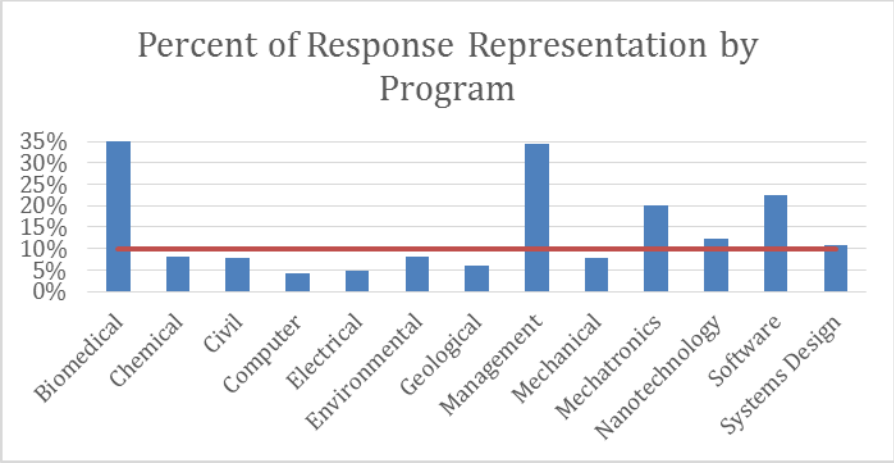


Figure 7: Percentage of Responses per Program versus 10% Threshold

Clustering

Clustering was not used for this particular dataset because clustering analysis is best for interval-scaled variables, binary variables and nominal/ordinal/ratio variables. The collected data consists of purely categorical data. A type of clustering is the k-Means algorithm that operates by updating the distance between centroids of clusters. This means that the Euclidean or Manhattan distance between two points that have categorical dimensions does not make sense. These distance metrics fail to capture the similarity of data elements when the attributes are categorical. For example, one cannot measure the distance for the attribute “Prefer to” with the answers “Talk”, “Read”, “Write” and “Listen”.

Classification

One R

Motivation

The One R method was used for simplicity and as a baseline to determine which feature after data exploration and cleaning had the highest predictive power. The results from One R will be used as a preliminary check for accuracy for multi-level decision trees, where the root node should be the same as the One R feature selected.

Modification to data

No modifications were made to the data outside of the initial data cleaning.

Sample of labelled training dataset

```
Ran 1R on Full data set using the following 19 attributes:

field_interest, team_responsible, work_setting, conversation, cause,
working_with, gather_information, design, prefer, gender, analyze, working_on,
leader, goals, develop, part_of, start_work, professor, work_outfit
```

Model

The One R model will select the feature to predict the class variable that has the highest coverage and accuracy. The model follows the following logic:

```
For each attribute:
        For each value of the attribute, make a rule as follows:
                count how often each class appears
                find the most frequent class
                make the rule assign that class to this attribute-value
        Calculate the error rate of the rules
Choose the rules with the smallest error rate
```

Prediction

The model uses the "What team would you want to be a part of" feature to predict the value of the class variable. The model produced the following rules:

```
team_responsible:
      The Robot   -> Mechatronics Engineering
      Penicillin  -> Biomedical Engineering
      Solar Panels      -> Nanotechnology Engineering
      The Internet      -> Software Engineering
      The Automobile    -> Mechanical Engineering
      The CN Tower      -> Civil Engineering
```

Evaluation

This feature has only 6 possible values to reach the class variable. This means that One R does not provide the ability to reach all programs. This is an issue because the model must be able to direct students to any of the 14 engineering programs offered at the University of Waterloo. Since the model is unable to classify many records within the dataset, the overall accuracy is 41.2%.

Analysis of results

The results from the model are interesting because they show that of the 6 programs that are classified, they are correctly classified 63% of the time. This indicates that the team_responsible feature is a very strong indicator of which program students should choose to be in.

Decision Tree and Random Forest

Iteration 1 – No change to hyper parameters

Motivation

An entropy based decision tree was the next model selected to build because it allowed for the opportunity to reach more values of the class variable through more levels of the decision tree. Decision trees also have the ability to be easily consumed by humans, which is helpful to see which paths will take records to particular outcomes.

Random forest will be compared to the results from the decision trees because they are used to correct entropy based decision tree's overfitting. This is because the random forest is an ensemble learning method that operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes of the trees.

Modification to data

All Happy=No records have been removed from the data. The Year feature has also been removed because the year a person is in will not help to predict which program they should be in when high school students are taking the quiz.

Iteration 1 – No change to hyper parameters

Table 6: Decision Tree and Random Forest with no hyper parameters and features

Model	Dataset	Hyper parameters	Accuracy	K-Fold	Insight
Decision Tree	field_interest, team_responsible, work_setting, conversation, cause, working_with, gather_information, design, prefer, gender, analyze, working_on, leader, goals, develop, part_of, start_work, professor, work_outfit	The branching criteria is on entropy	1	0.37	Overfitting indicated by large disparity between k-fold and accuracy Tree finds features "field_interest" and "team_ responsible" to be most important.

Random Forest	field_interest, team_responsible, work_setting, conversation, cause, working_with, gather_information, design, prefer, gender, analyze, working_on, leader, goals, develop, part_of, start_work, professor, work_outfit	The branching criteria is on entropy The number of trees generated has been limited to 100	0.98	0.42	Overfitting indicated by large disparity between k-fold and accuracy Tree finds features "field_interest" and "team_ responsible" to be most important.

Analysis of results

Both trees are severely overfitting; iteration is required to limit the hyper parameters to negate the effect. Specifically, the maximum depth the tree is allowed to reach, the maximum number of features utilized by the tree and the minimum number of records that have to be classified to generate a root node will be limited.

Iteration 2 – Select hyper parameters and features

Table 7: Decision Tree and Random Forest with select hyper parameters and features

Model	Dataset	Hyper parameters	Accuracy	K-Fold	Insight
Decision Tree	field_interest, team_responsible, work_setting, conversation, cause, working_with, gather_information, design, prefer, gender, analyze, working_on, leader, goals, develop, part_of, start_work, professor, work_outfit	The branching criteria is on entropy The maximum depth of the tree has been limited to 4 The maximum number of features utilized has been set to 17 A minimum of 2 records have to be classified by a root node for the tree to generate it	0.47	0.37	The k-fold score has not changed but the accuracy has by limiting tree depth, minimum number of records and maximum features, there is less

					overfitting in the model now
Random Forest	field_interest, team_responsible, work_setting, conversation, cause, working_with, gather_information, design, prefer, gender, analyze, working_on, leader, goals, develop, part_of, start_work, professor, work_outfit	The branching criteria is on entropy The maximum depth of the tree has been limited to 4 The number of trees generated has been limited to 100 The maximum number of features utilized has been set to 17 A minimum of 2 records have to be classified by a root node for the tree to generate it	0.53	0.49	The random forest has higher accuracy and is no longer overfitting, the result of the random forest is better than the decision tree as expected because the random forest tasks the average of the nodes

Prediction

Decision Tree

The decision tree has an overall 47% accuracy, where the results fall as per the confusion matrix in table 8.

Table 8: Confusion matrix for the decision tree

	Biomedical	Chemical	Civil	Computer	Electrical	Environmental	Geological	Management	Mechanical	Mechatronics	Nanotechnology	Software	Systems Design
Biomedical	27	3	0	0	0	1	0	0	4	16	0	2	0
Chemical	6	33	1	0	0	2	0	0	8	0	6	0	0
Civil	0	0	35	0	0	2	0	4	5	0	1	0	2
Computer	0	0	0	0	0	0	0	0	3	9	0	25	2
Electrical	1	4	1	0	0	1	0	3	4	9	2	5	2
Environmental	0	1	1	0	0	7	0	1	3	0	7	0	0
Geological	0	0	4	0	0	0	0	2	2	0	0	0	0
Management	1	15	2	0	0	3	0	18	23	5	1	14	17
Mechanical	4	8	4	0	0	4	0	2	43	15	1	1	0
Mechatronics	1	4	2	0	0	0	0	1	17	103	3	12	8
Nanotechnology	24	7	0	0	0	0	0	1	7	4	15	1	0
Software	0	0	0	0	0	0	0	0	4	23	0	89	17
Systems Design	1	1	0	0	0	4	0	1	4	6	0	9	19

It is important to note that the decision tree did not classify 3 values of the class variable: Computer, Electrical and Geological engineering.

The accuracy of prediction by program is shown in table 9. Only 6/14 values of the class variables are classified with accuracy higher than 50%.

Table 9: Confusion matrix for the decision tree

Discipline	Predicted Correctly
Biomedical	50.94%
Chemical	58.93%
Civil	71.43%
Computer	0.00%
Electrical	0.00%
Environmental	35.00%
Geological	0.00%
Management	18.18%
Mechanical	52.44%
Mechatronics	68.21%
Nanotechnology	25.42%
Software	66.92%
Systems Design	42.22%

Random Forest

Random Forest has an overall 52% accuracy, where the results fall as per the confusion matrix in table 10.

Table 10: Confusion matrix for random forest

	Biomedical	Chemical	Civil	Computer	Electrical	Environmental	Geological	Management	Mechanical	Mechatronics	Nanotechnology	Software	Systems Design
Biomedical	8	4	0	0	0	0	0	0	4	10	19	8	0
Chemical	1	34	1	0	0	0	0	2	7	0	11	0	0
Civil	0	0	38	0	0	2	0	4	2	0	1	2	0
Computer	0	0	0	0	0	0	0	3	1	8	0	27	0
Electrical	0	1	1	0	0	0	0	6	5	8	3	8	0
Environmental	0	2	2	0	0	6	0	1	2	0	7	0	0
Geological	0	0	4	0	0	0	0	2	2	0	0	0	0
Management	1	12	3	0	0	0	0	41	10	1	1	30	0
Mechanical	0	9	5	0	0	2	0	1	44	14	5	2	0
Mechatronics	0	3	1	0	0	0	0	1	18	94	4	30	0
Nanotechnology	0	7	0	0	0	0	0	1	7	3	39	2	0
Software	0	0	0	0	0	0	0	1	3	4	0	125	0
Systems Design	0	1	0	0	0	2	0	6	4	8	1	18	5

It is important to note that random forest also did not classify 3 values of the class variable: Computer, Electrical and Geological engineering.

The accuracy of prediction by program is shown in table 11. Only 50% of the values of the class variables are classified with accuracy higher than 60%.

Table 11: Confusion matrix for random forest

Discipline	Predicted Correctly
Biomedical	15.09%
Chemical	60.71%
Civil	77.55%
Computer	0.00%
Electrical	0.00%
Environmental	30.00%
Geological	0.00%
Management	41.41%
Mechanical	53.66%
Mechatronics	62.25%
Nanotechnology	66.10%
Software	93.98%
Systems Design	11.11%

Analysis of results

The accuracy from random forest was higher than the accuracy of the decision tree and was less susceptible to overfitting. However, this method did not successfully reach root nodes for Computer, Electrical and Geological engineering and is a drawback of this model. Further models that can reach all values of the class variable and produce a higher accuracy than 53% should be considered.

Naive Bayes with one class variable

Feature Selection

Feature selection is defined as the process by which the best subset of attributes is selected within a given dataset. The best attributes imply a prediction score for a given model. Three benefits of performing feature selection are reducing overfitting, improves accuracy and reduces the training time of the model. By reducing the overfitting of the model, there is a less likely chance the model will make a decision based upon noise within the data. This means there is less misleading data, thus increasing the accuracy. As a result, less data implies the algorithm for each model may run faster.

Weka provides a feature selection tool called “Select Attributes” that breaks the process of selection into two parts: attribute evaluator and search method. The attribute evaluator is the method that assesses subsets of the attributes. The Search Method is defined by the space of possible subsets to be searched.

For Naïve Bayes, the selected attribute evaluator to be used is "WrapperSubsetEval." This evaluator was then specified to the Naïve Bayes classifier. This evaluator assesses the accuracy of a subset of the data using Naïve Bayes on a k-fold validation.

The Search Method is the process in which the search space of all potential feature subsets is navigated based upon the attribute evaluator. For this model, the "BestFirst" method was selected, as it is the most common and uses a best-first search strategy to determine attribute subsets.

As a result of running this feature selection process, Table 12 displays the top attributes given as output as the most effective when running Naïve Bayes. The effectiveness is measured in terms of accuracy as the model. It is proven that if any of these nine attributes are removed, the accuracy of the model decreases. With these nine features, the accuracy of the model is 77%. If the top two attributes, field_interest and team_responsible, are removed, the accuracy of the model decreases by 16 percent to 61%.

Table 12: Features Selected for Naïve Bayes

Features Selected
field_interest
team_responsible
start_work
design_something
work_partof
on_project
care_most
Eavesdrop
working_with

Motivation

The Bayesian approach is explored as a method because it is an intuitive model involving statistics. By involving Bayes' Theorem from statistics, we determine the probability of an event A occurring when event B is observed. It is a simple rule in statistics, when used as a classifier can provide many benefits. The Naïve Bayes classifier can be used to find the probability of value of a class variable C, given the values of specific feature variables (X_1, X_2,

…, X_K). In the case of the Engineering quiz, the Naïve Bayes classified is able to find the probability of value of a specific discipline given multiple different features based on personality and preferences.

Another major motivation for using Naïve Bayes is the fact that it is one of the top 10 algorithms in data mining. The algorithm can be used when involving classification, which is what the different Engineering disciplines represent. The algorithm is simple to construct, and does not require any complicated, iterative parameters. The algorithm can be applied to large data sets, and is simple to interpret. The Engineering quiz aims to predict the best possible Engineering discipline for potential first year students, and the Naïve Bayes' classifier is robust and simple enough to predict for this model.

Modification to data

Two sets of data are being studied in order to show progression of the models when different numbers of features are used. At first, all 19 features are used to predict a class variable. This is done to see how accurate the model is when all quiz questions that were included in the initial survey are used to predict the best discipline.

For the second part of the class variable predicting model, the data has been modified to only include specific features that would work optimally with the Naïve Bayes' classifier. The features that are chosen to predict with Naïve Bayes are decided based upon the results received from the explanatory data analysis. Nine of the 19 available features are used to predict the class variable.

Sample of labelled training dataset

All features are used in the development of parts of the Naïve Bayes model. In some instances, only 9 features are used.

Model

The Naïve Bayes' model is based off Bayes theorem in statistics. Bayes' theorem attempts to find the probability of a particular event A when event B is observed. This equation is converted to predict the probability of a class variable C. The aim is to use class memberships from the training set to construct a score to define relations between the features and the class variable.

The Naïve Bayes algorithm assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. This allows each of the features in our quiz to have an impact on the final result of expected discipline. All questions will independently contribute to the probability of what the most matched discipline is.

The Naïve Bayes algorithm has been programmed in Python code, and the quiz results are inputted into this code as two models - first with all 19 questions, second with only 9 of the 19 total questions as features.

Prediction

Table 13: Confusion matrix for Naïve Bayes algorithm with 19 features

	Biomedical	Chemical	Civil	Computer	Electrical	Environmental	Geological	Management	Mechanical	Mechatronics	Nanotechnology	Software	Systems&Design
Biomedical	24	4	0	0	1	0	0	3	2	3	10	5	1
Chemical	4	39	1	0	0	1	0	1	4	0	6	0	0
Civil	0	0	40	1	0	1	0	1	2	0	2	0	2
Computer	0	0	0	3	0	0	0	2	0	9	0	23	2
Electrical	0	0	1	2	2	2	0	5	3	8	2	5	2
Environmental	1	0	1	0	0	15	0	0	0	0	2	1	0
Geological	0	0	4	0	0	0	3	1	0	0	0	0	0
Management	2	6	1	0	0	0	0	68	3	1	0	11	7
Mechanical	1	2	3	0	1	1	0	0	57	15	1	1	0
Mechatronics	4	0	0	1	1	2	0	5	17	100	2	17	2
Nanotechnology	8	4	0	0	0	1	0	3	5	3	33	2	0
Software	1	0	0	4	2	0	0	4	0	10	1	108	3
Systems&Design	2	1	1	1	0	3	0	6	1	4	0	12	14

Table 14: Confusion matrix for Naïve Bayes algorithm with 9 of 19 features

	Biomedical	Chemical	Civil	Computer	Electrical	Environmental	Geological	Management	Mechanical	Mechatronics	Nanotechnology	Software	Systems&Design
Biomedical	28	3	1	0	0	0	0	3	2	6	4	5	1
Chemical	4	32	2	0	1	2	0	2	8	0	5	0	0
Civil	0	0	41	1	0	1	0	1	3	0	1	0	1
Computer	0	0	0	0	0	0	0	4	0	9	0	24	2
Electrical	0	0	1	2	2	2	0	5	3	9	1	6	1
Environmental	1	0	4	1	0	13	0	0	0	0	0	1	0
Geological	0	0	8	0	0	0	0	0	0	0	0	0	0
Management	3	9	1	1	0	0	0	53	5	2	2	17	6
Mechanical	2	4	3	0	0	3	0	0	54	13	2	1	0
Mechatronics	2	0	0	1	3	1	0	4	16	99	2	23	0
Nanotechnology	8	5	1	1	1	0	0	1	4	2	34	2	0
Software	1	0	0	0	0	0	0	5	0	9	1	113	4
Systems&Design	2	1	2	0	0	2	0	5	1	6	0	13	13

Valuable data can be collected from the confusion matrices above. It is evident that the majority of the diagonal portion of the matrix contains the highest numbers for each discipline. This proves that most students are placed into the discipline they belong in to versus other disciplines. Both matrices show valuable results, and are further evaluated.

Evaluation

All 19 Features:

The Naïve Bayes classifier that includes all 19 features, has created a model with an accuracy of 61%. When a k-fold evaluation was conducted, with k = 5, an average accuracy of 54% was received.

Pros

Despite the fact that most of the false positives are software engineering; geological and chemical engineering disciplines are not miss-classified as software engineering. This is a good sign because it shows that even though many of the features cause the class variable to be software, there are features that are strong enough to not sway the results too much. According to course concepts, geological and chemical engineering are not highly associated with software, and so the fact that the model does not predict software for those engineers, it shows the model is still strong in predicting.

Cons

The model gives the most number of false positives for software engineering, 87 in total. In reality, it does not make sense that most students are classified as software engineering. It is possible that this is occurring because many of the 19 features are highly associated with the software engineering class variable, as well as the fact that a large amount of survey data comes from software engineering students.

Further Analysis

Table 15 below shows how accurately this model predicted each of the disciplines. Disciplines such as electrical, and computer have a very low accuracy compared to disciplines such as civil and software. This is due the fact that not many electrical and computer engineering students have completed the survey coupled with the fact that over 25% are being classified incorrectly as mechatronics or software. This is because more students from those disciplines have completed the survey and these disciplines are similar to one another.

Table 15: Naïve Bayes/1 Class Variable/ 19 Features Prediction Percentages

Discipline	Predicted Correctly
Biomedical	45.28%
Chemical	69.64%
Civil	81.63%
Computer	7.69%
Electrical	6.25%
Environmental	75.00%
Geological	37.50%
Management	68.69%
Mechanical	69.51%
Mechatronics	66.23%
Nanotechnology	55.93%
Software	81.20%
Systems Design	31.11%

9 of 19 Features:

The Naïve Bayes classifier has predicted an accuracy of 58% overall when using 9 features. When a k-fold evaluation was conducted, with k = 5, an average accuracy of 54% was received. These numbers include data from any student that is happy in their program. As the simplicity of Naïve Bayes was expected to give similar results as the other models, its capacity to handle the large engineering quiz dataset has proved otherwise. So far, the Naïve Bayes algorithm that uses all 19 features has given the highest accuracy prediction model to predict 1 class variable. However, the accuracy of using only 9 variables is not much lower. If the model is able to give an accuracy lower than less than 5% by asking 10 less questions, using 9 features is definitely a better option.

Pros

Removing 10 features has continued to yield a similar prediction accuracy and k-fold validation score. By choosing specific features instead of all, it further simplifies the models, makes the model run faster, and makes them easier to interpret. Reducing the amount of features reduces overfitting as well, due to enhanced generalization in the data. The gap between the k-fold validation score and the overall accuracy of the model go from 7% with 19 features to 4% with 9 features. The small amount of difference means the model is not overfitting. Features that had low predictive power, were removed.

Cons

This model has a great average accuracy; however, this is not good enough for a quiz that will affect several potential first-year engineering students. In order to minimize students choosing the incorrect program, a higher accuracy model needs to be created. A higher accuracy can be obtained by outputting 2 class variables.

Further Analysis

Table 16: Naïve Bayes/1 Class Variable 9 Features Prediction Percentages

Discipline	Predicted Correctly
Biomedical	52.83%
Chemical	57.14%
Civil	83.67%
Computer	0.00%
Electrical	6.25%
Environmental	65.00%
Geological	0.00%
Management	53.54%
Mechanical	65.85%
Mechatronics	65.56%
Nanotechnology	57.63%
Software	84.96%
Systems Design	28.89%

Table 16 above shows interesting data about the different disciplines. Most of the disciplines have prediction accuracy that is not highly affected. However, geological engineering does stand out, as it has 0% students predicted accurately. According to the confusion matrix, all 8 geological engineering students have been classified as civil engineering. Even though they are not accurately placed in their own discipline, they are placed in the second best discipline for them. Geological engineering students spend their first year of engineering classes in combined classes with civil engineering students. This further proves predicting 2 class variables can be highly effective for this model.

Analysis of results

Interesting conclusions can be derived from the confusion matrices the Naïve Bayes algorithm has formed. It can be seen that the false positives coincide with the false negatives for the model with 19 features as well as with 9 features. For example, according to both the confusion matrices, the second most predicted discipline for actual Mechatronics Engineering students is Software Engineering. Similarly, the second most predicted discipline for Software Engineering students, is Mechatronics Engineering. It is also seen that Systems Design Engineering students are most likely to be placed into Software Engineering if not Systems Design, which according to course concepts taught in both disciplines makes sense. The two disciplines have similar learning objectives, and so many systems design engineers are predicted as software engineers. Due to these results, it was decided that exploring Naïve Bayes that predicts 2 class variables will provide higher accuracy.

Naive Bayes with two class variables

Motivation

After running multiple classification algorithms, it was found that the Naïve Bayes algorithm had provided the most accurate results. Due to this, it was decided that further improving the algorithm would provide better results. It is understood that when a student fills out the AIF form and submits it to the First-Year Engineering office for entrance into Waterloo Engineering, the student must choose their top choice engineering discipline, and their 2nd favorite option. Thus, it is desirable that the Engineering quiz produces two possible disciplines that the student fits into. By coding the Naïve Bayes algorithm to output two possible disciplines for the student, it causes the model to have a higher accuracy, as well as fulfils the needs of the Engineering quiz.

Modification to data

The results will now include two predicted class variables from the features that are used. The data that is used to predict 2 class variables in Naïve Bayes is the same data that is used to predict 1 class variable in Naïve Bayes. The reason for maintaining the same features as prediction for 1 class variable prediction, is to truly be able to capture if adding a class improves the accuracy of the model.

A model using all 19 features, as well as one with 9 features will be used to predict 2 class variables.

Sample of labelled training dataset

All features are used in the development of parts of the Naïve Bayes model. In some instances, only 9 features or 6 features are used.

Model

The model is revised to output 2 class variables predicted from the given features. The reason for creating a model like this is because the 1 class variable predicting Naïve Bayes algorithm provided high accuracy predictions. Thus, by predicting 2 class variables, the probability of a student being matched with the program they belong in increases.

The model is developed by using the two disciplines that have the highest Naïve Bayes probability score. For example, for a student who is truly in Systems Design Engineering, Naïve Bayes will calculate the probability of how well the student fits into each of the disciplines given their survey responses. See Table 17 below.

Table 17: Naïve Bayes/2 Class Variables Output Expectations

Discipline	Naïve Bayes Score	Rank
Biomedical	3%	-
Chemical	0%	-
Civil	0%	-
Computer	10%	-
Electrical	1%	-
Environmental	0%	-
Geological	0%	-
Management	15%	-
Mechanical	0%	-
Mechatronics	2%	-
Nanotechnology	2%	-
Software	49%	1
Systems Design	18%	2

As it is evident from the table 17 above, Naïve Bayes predicts that the student is best fit for Software Engineering, and second best fit for Systems Design Engineering. If the algorithm had only calculated one discipline, the student would have only been matched with Software Engineering. Therefore, the 2 class variable predicting model is expected to predict the students' discipline at a higher accuracy than simple Naïve Bayes that only predicts 1 class

variable. Second, by predicting 2 class variables, the model will coincide with the original quiz requirements of outputting 2 of the best programs according to the students' personality and preferences.

Prediction

The following table (18) represents the data received from expanding the model 2 class variables:

Table 18: Naïve Bayes/2 Class Variables/Data

Model	Accuracy	K-Fold Validation
Naïve Bayes with 2 classes - All 19 features	0.81	0.74
Naïve Bayes with 2 classes - Only 9 of 19 features	0.77	0.71

Evaluation

As expected, the model predicted the 2 class variables with a higher accuracy than predicting only 1-class variable.

Table 19: Naïve Bayes/2 Class Variables/ 9 Features Prediction Percentages

Discipline	Predicted Correctly
Biomedical	79.25%
Chemical	75.00%
Civil	87.76%
Computer	64.10%
Electrical	25.00%
Environmental	80.00%
Geological	87.50%
Management	66.67%
Mechanical	82.93%
Mechatronics	80.79%
Nanotechnology	76.27%
Software	93.98%
Systems Design	53.33%

Table 19 above shows increases in prediction accuracy compared to Table 15 where the Naïve Bayes algorithm used 9 features to calculate 1 class variable. The prediction percentage for computer engineering has increased from 0% to over 64% accurately predicted students. Similarly, the prediction accuracy for geological engineering has increased from 0% to 87.5%, instantly improving the results. This shows how much better the model is when predicting 2 class variables.

Similar to predicting 1 class variable, the 2 class variable predicting model works best when using only 9 out of 10 features instead of all 19. This is because some of the features are highly associated with some disciplines and skew the results. Also, having 9 features predict only a little less accurate than 19 features provides a much more efficient model, and useful quiz for students to complete.

Overall, predicting 2 class variables has improved the Naïve Bayes model significantly. Predicting 2 class variables instead of 1 has increased the model's accuracy by 20% with all 19 features, and by 19% with 9 features. This is a significant improvement for both models. The k-fold validation method has increased accuracy by over 15% for both models of 19 and 9 features.

It is important to notice that the average k-fold validation score is similar to the overall model accuracies, which proves that the algorithm does not over-fit any data.

It is interesting to see that removing features does not improve the accuracy score, however it does improve the overall model. Using 9 features takes away any bias the features have towards certain disciplines or outliers, as well as creates a clear and concise quiz.

Analysis of results

Analyzing the results has proved that the Naïve Bayes algorithm that predicts 2 class variables by using 9 features is the model that outputs the best accuracy compared to the amount of features used. The comparison of models will show this is true with greater detail. The reason predicting 2 class variables is better is because now there is a higher chance a student will be matched with the program they actually belong in. The second class variable is chosen according to its percentage ranking compared to all other disciplines, and so the top 2 best fit disciplines are returned.

Extensions

3-Class Variable Prediction

In order to ensure that stopping at 2 class variable predictions is the best option, the Naïve Bayes algorithm was run once again with all 19 features, and 9 of 19 features were used to predict 3-class variables. This study found the accuracy of the model did not increase significantly. This proves that adding extra variables is unnecessary since the improvements are minor, and the aim is to output the 2 best fit disciplines for the students.

Stratified

The K-fold validation scores are calculated using the stratify method. Stratified is a type of data sampling that divides the data into separate groups in which a probability sample is drawn from each of these data groups.

Stratify is an advantageous sampling method because it selects the groups to have a proportionate amount of variables in each of the data set.

In terms of the engineering quiz, it is evident that not many environmental or geological engineering students completed the survey, the stratified method ensures that each of the folds include a proportional amount of geological/environmental student results. This ensures that the 5 folds do not have highly varying average prediction accuracies. The stratified sampling method allows data to be selected proportionally when performing k-fold evaluations. This method has been used throughout the analysis of different algorithms to create models that are highly accurate.

Naive Bayes Iterations

In order to ensure that the best models are analyzed, multiple models were studied. Table 20 below explains the iterations that were done to show progression with the Naïve Bayes model, as well as how the best results were predicted accurately.

Table 20: Naïve Bayes Iterations

Naïve Bayes Model	Accuracy	K-Fold Validation	Thoughts on Iteration
Class Variables: 1 Features: 19	61%	54%	This iteration has been successful overall, and has provided good insight on the data. This was the first iteration and has been a foundation to all other iterations.
Class Variables: 2 Features: 19	81%	74%	This iteration improved by 20% from the model above while maintaining a strong k-fold score.
Class Variables: 1 Features Variables: 9	58%	54%	Removing 10 features actually dropped the accuracy score, as expected. This happened because the more features a model has, the better it can predict. The fact that the prediction accuracy has not decreased by much is a good sign. This proves that the 10 features removed had low predictive power, the 9 that are kept are strong and accurate.
Class Variables: 2 Features: 9	77%	71%	This iteration had a high and feasibility of all Naïve Bayes models that were explored. It uses 9 features. This proves that this

			model has highly accurate predicting features that create a strong model. This is the best choice for the Engineering quiz.
Class Variables: 1 Features: 7	43%	39%	The top 2 features are removed from the 9 best features. This has driven accuracy down by over 30%. This tells us that the top 2 predicting features are truly predicting the data accurately.
Class Variables: 2 Features: 7	61%	57%	The top 2 features are removed from the 9 best features. This has driven accuracy down by over 30%. This tells us that the top 2 predicting features are truly predicting the data accurately.
Class Variables: 3 Features: 9	80%	79%	The accuracy has increased by adding a 3rd class variable. However, this is not a significant change, proving that stopping at the prediction of 2 class variables is good. It also does not follow the format of the AIF form Waterloo requires students to completes.
Class Variables: 4 Features: 9	88%	84%	The accuracy has increased as expected by adding a 4th class variable into the predictions. This occurs because the more disciplines the model predicts, the higher chance of the model being accurate about the students' discipline. However, outputting 4 results to a potential first year engineering student would cause confusion and would not help the student decide which discipline is truly best for them. The increase in accuracy is not significant enough to outweigh the risks of confusing the student and wasting their time with the quiz.

Comparison of models

The following table depicts the four different prediction models with corresponding features used, hyper-parameters, accuracy and k-fold validations.

Table 21: Comparison of models

Model	Subset of features	Hyper-parameter	Accuracy	K-fold Validation
One R	19 features, excluding year of study and gender	Based on entropy	40%	N/A
Decision Tree Classifier (best result)	The maximum number of features utilized has been set to 17	The branching criteria is on entropy The maximum depth of the tree has been limited to 4 A minimum of 2 records have to be classified by a root node for the tree to generate it	47 %	37%
Random Forest (best result)	The maximum number of features utilized has been set to 17	The n_estimators was set to 100 The branching criteria is on entropy The maximum depth of the tree has been limited to 4 A minimum of 2 records have to be classified by a root node for the tree to generate it	53%	49%
Naive Bayes with two class variables (best result)	9 features, excluding year of study and gender	Refer to list of 9 features	77%	71%

Recommendation

Based upon the current dataset, it is recommended that the model used to predict which engineering discipline is right for future engineers is Naïve Bayes with 2 class variables, using 9 features. This is because this model produced the highest accuracy of 77%. The model will output two potential disciplines to the quiz-takers. This number agrees with the fact that the Waterloo Engineering Admissions Office is changing the number programs that a student is able submit an application for to 2. The following nine questions are to be included in the quiz are summarized in Table 22.

Table 22: Features and class variable to be Used in Engineering Quiz

Feature	Reference Name*	Values of the feature
What field interests you the most?	Field_interest	6. Finance 7. Manufacturing 8. Software Development 9. Research 10. Construction
I would like to have been part of the team that was responsible for:	Team_responsible	7. The Automobile 8. The Internet 9. The CN Tower 10. Solar Panels 11. The Robot 12. Penicillin
When you start work, ideally you would want to...	Start_work	4. Work for a large corporation 5. Work for a small company 6. Work in a start up
I would rather design something that...	Design_something	3. I can see it function and prove it works, but can't necessary touch 4. I can see function and tangibly touch the parts
I would rather work as part of a...	Work_partof	3. Focused team 4. Multidisciplinary team
When working on a project would you rather	On_project	3. Know all the requirements for the project at the beginning 4. Continuously change and update requirements as you go
What cause do you care about the most?	Care_most	5. Health 6. Environment 7. Human Rights 8. Bringing technology to developing countries

You are most likely to eavesdrop in a conversation regarding	Eavesdrop	5. A groundbreaking metal that will create stronger cars, planes and spaceships for a cheaper cost 6. A new way to deal with movement in the Earth's surface to protect buildings from earthquakes 7. A microchip that is smaller than usual but twice as powerful as before 8. A factory that is able to 100% rely on robotics while producing no defected products
In an ideal setting, you are working with:	Working_with	5. People 6. Computers 7. Business Processes 8. Machinery

Future Steps

The programs with the lowest responses are Electrical, Systems, Computer and Management Engineering all with accuracy levels under 67%. To improve the accuracy and prediction of the Naïve Bayes model, the first next step is to gather more survey responses from these disciplines. After more data is gathered, feature selection will be performed again to determine if the same nine selected attributes are the best for this model. The last step will be to use the features selected and re-run the Naïve Bayes algorithm to determine the accuracy and predictions for the data.

It is important to note that it is logical that these four programs have the lowest accuracy. This is because of two reasons, the first being that not many students from these disciplines took the quiz (with the exception of Management). The second reason is that a lot of the answers selected for the participants in these four programs show great similarity in the answers selected for Software and Mechatronics Engineering, both of which have the highest number of responses of the survey. Therefore, these four programs were commonly mistaken for Software and Mechatronics Engineering.

Conclusion

Four prediction models were run on the engineering quiz dataset. The dataset consists of 19 questions that were used to predict which program is best suited for a future student based upon their personality, interests and attributes. The data collected consisted of purely categorical data, with the exception of what year each participant is currently in.

The best prediction model was Naïve Bayes with 2 class variables, followed by Naïve Bayes with 1 class variable. The model with the third highest accuracy is Random Forest, followed by the Decision Tree algorithm. Lastly, the model with the lowest accuracy for prediction was 1R.

Prior to data collection, it was hypothesized that if more questions relating to future employment opportunities and industries to work in were included in the survey, the models used would result in a higher accuracy in comparison to the first time the project was done. This hypothesis was validated within the recommendation section, noting that six out of the nine questions are relating to jobs. The previous project reported an accuracy of 55% utilizing an AODER model. Thus, our full hypothesis has been confirmed.