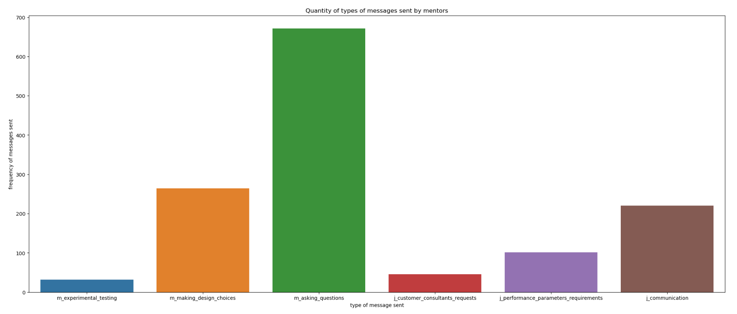
ADS2001 Report components

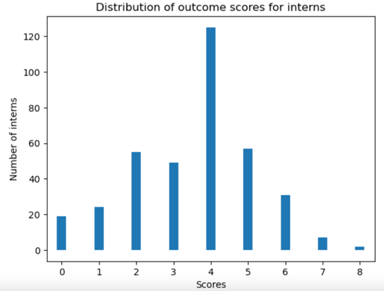
Exploratory data analysis

In this project, all the data was used albeit in different sections. For example, the subsequent figures \_,\_,\_... and \_ were generated using a quantitative data set, created by the processes explored during data wrangling. For example, a bar graph was generated from the cumulative totals of the types of messages sent by interns and mentors.

Chart, bar chart

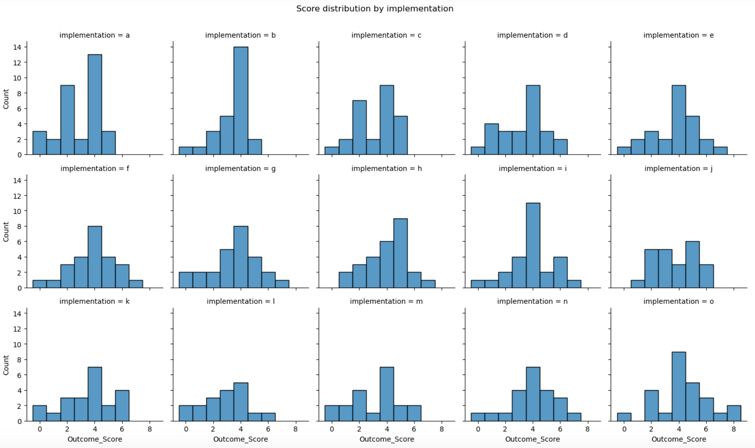
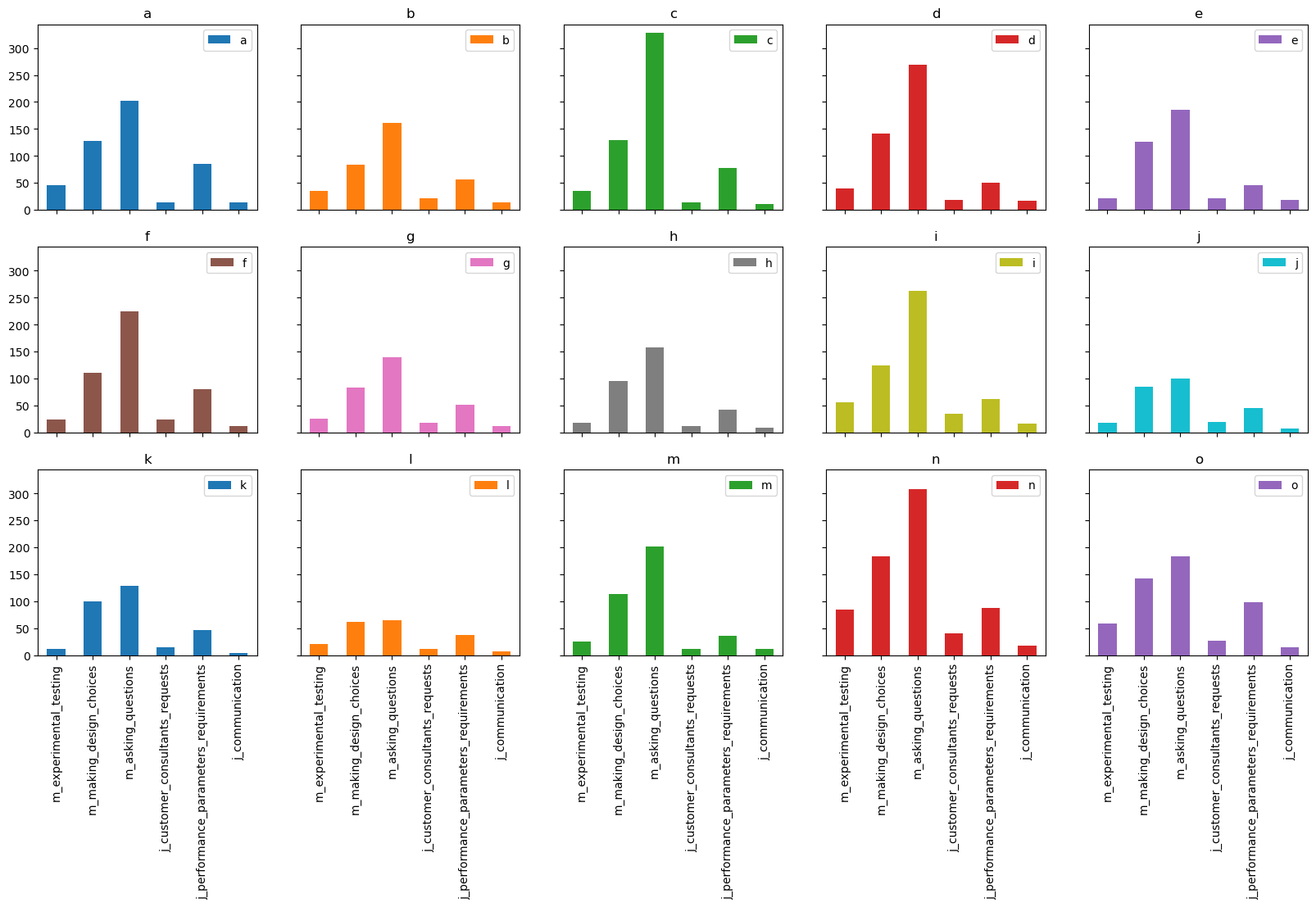
Description automatically generated

As observed in figure \_ on the distribution of the types of messages sent, interns most frequently asked questions. This is likely because interns could only raise concerns with their mentors through WorkPro, in the form of questions to seek clarification on different aspects of their project that they were uncertain of. Interns also expressed their design choices very frequently through the WorkPro applications, perhaps because they could subsequently receive responsive feedback on their choices from mentors. Interns least commonly sent out messages intended to communicate with one another as they may have used other tools to communicate with one another or the communication category could have been used as a placeholder category that ambiguous messages could be classified as. They also seemed to rarely ask for advice from customer consultants, possibly because of the limited access that they had to consultants or their lack of willingness to engage with consultants. Both reasons deduced from the analysis of the relationship between consultants and interns may help explain the overall poor performance of interns, indicated in figure \_ . For example, the lack of correspondence between consultants and interns may have left interns underinformed about what constitutes an effective product for kidney dialysis, causing performance issues with the prototypes they may have designed. Regarding the distribution of messages sent by mentors, the most common type of content contained in messages was questions which may imply that majority of the contact between interns and mentors was in the form of providing hints through posing potential questions that may have assisted interns. For example, a “question” type message sent by a mentor was “How did you choose the best surfactant?” which was intended to encourage interns to think about their thought process more critically when deciding on the best surfactant, rather than revealing the surfactants which they should consider. The least common type of messages sent mentors was the consultant type, likely because mentors rarely had a presence in interns’ discussions with consultants apart from organising the logistics of the sessions between consultants and interns. Moreover, mentors sent significantly less messages than interns. This may be because one mentor had to manage multiple groups and had limited ability to sustain a heavy amount of communication with any single group, thus reducing the number of messages they were able to send.



In figure \_, the mode and average score is clearly 4. As previously mentioned during the data wrangling process, each mentor’s assigned score was 4 but these mentors’ messages were excluded from the data frame before the analysis was conducted, avoiding any errors with regards to this insight. In addition, the figure shows that scores are mildly skewed to the left therefore suggesting that the internship program at Nephrotex has an above average degree of difficulty because majority of students either performed below average or at an average level.

Furthermore, because each implementation represents a ‘round’ of internships, the distribution of each internship can be observed for both the scores and frequency of messages sent to determine the presence of a potential relationship between interns’ performance and the types of messages they sent at an increased depth.

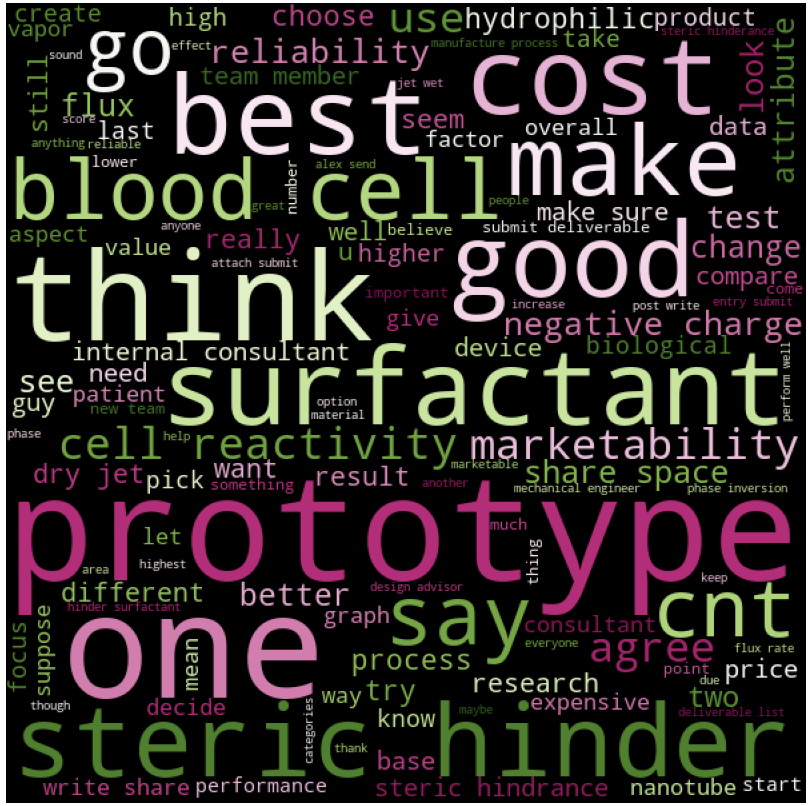
In both figure \_ and figure \_, there seems to be a lack of a concrete relationship between the distribution of the outcome scores achieved by interns and the distribution of the types of messages they send. However, more concrete findings are shown in the form of a correlation matrix at figure \_.

Chart, bar chart

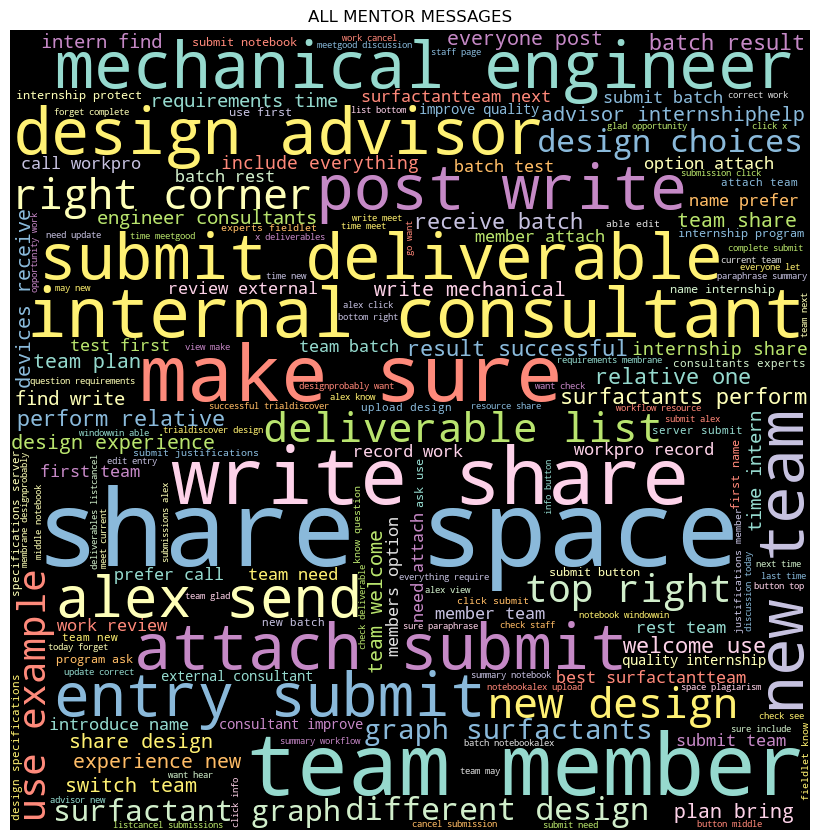
Description automatically generated

The observations to pay attention to in figure \_ are the correlation scores between outcome score and the rest of the variables. All these scores are very low which indicates the lack of a definite relationship between any other variable and the outcome score of interns. Since the correlation score between some variables are reasonably high, for example, the correlation score between m\_asking\_questions and m\_making\_design\_choices, there will be some issues with multicollinearity later, meaning that when modelling is performed, the importance of both features in terms of their impact on outcome score may become ambiguous which could hinder the process of answering the research question.

Moreover, another important component in the analysis was the use of the techniques provided by natural language processing to determine the frequency of words used by both interns and mentors to determine whether high-performing students used more of certain key words and conversely whether low-performing students used certain key words frequently or did not use them. A concise visual is provided in figure \_ to show which words were commonly associated with the content that interns talked about using WorkPro.



From figure \_, scientific words and phrases like “blood cell” and “surfactant” seem to have a mild presence perhaps suggesting that they may not have influenced outcome score as much as commanding words like “think” and “hinder” or words associated with the designing process, like “prototype”. A reason why “command” words may have had an impact on score is that commands tend to affect team dynamics and therefore may influence individual interns’ performance as a result. Designing words could affect the outcome of an individual’s performance because the use of such words may alter how the thought processes of interns are expressed. For example, if interns frequently refer to their prototypes during the process of assessing their products, this may imply that they are relying heavily on the insights produced by their prototypes, therefore significantly influencing outcome score. The results from using similar natural language processing techniques may suggest that a series of features may not be influencing outcome score, but rather, the content of the messages being sent could be the defining trait in predicting outcome score.



In figure \_, further analysis done on mentors’ responses to the messages sent by interns, using the same mechanism as before, shows that action and command words such as “write”, “share” and “submit” were the most frequent type of words contained within the messages sent by mentors, perhaps implying that mentors approached interns in a mainly instructive capacity, providing them with required tasks to fulfil rather than advising them of potential ways to improve. There was a significantly lesser presence of messages of a technical nature as words like “surfactant” and “graph” were used a lot less than other phrases, perhaps suggesting that mentors were either restricted in providing technical knowledge to interns or that interns did not probe mentors for such information. This may lead to interns producing results that may be lacking with respect to the technical aspect of their project. In addition, a significant proportion of words and phrases, such as “make sure” and “deliverable list” were intended to clarify requirements for the project. This may imply that interns’ main source of assistance were their mentors but because mentors were either restricted in the information they were allowed to give or the very nature of their role was to provide instructions, the feedback they gave to interns when asked for further clarification may not have been as helpful. This could potentially explain the poor performance of majority of interns.

Decision trees

The next algorithm chosen was the decision trees classifier algorithm. This technique was chosen because of the ability for decision trees to express feature importance and operate on a relative scale, which was important as not all features had the same measurement system. The parameters were optimised using the typical GridSearch cross validation method. The parameters chosen to be optimised were the “maximum number of features considered when looking for the best split” (Ceballos, 2019) and the maximum depth. The maximum number of features was chosen as a tuning parameter because the goal of this project was to find the factors that affected outcome score the most and setting this parameter could assist in the selection process. The maximum depth was also altered using the optimisation mechanism because accuracy is affected by the depth at which a decision tree terminates and another important consideration when deducing the importance of features was that the observations associated with the results had to be as accurate as possible. The decision tree generated is shown in figure \_ and displays the features that had a prominent effect on score. The cross-validation accuracy achieved by the optimisation mechanism was approximately 34% and the accuracy of the algorithm when applied to the data set had an accuracy of approximately 18%. Since this algorithm is a supervised learning type, it requires a separated training and testing set which was randomly split in half. The choice to split the data in half was made because decision trees are sensitive to overfitting meaning that slight alterations with regards to the amount of data used to train the algorithm would drastically alter results. By splitting the data in half randomly, an equal amount would be inputted for both training and testing sets, tackling the issue of overfitting to some extent.

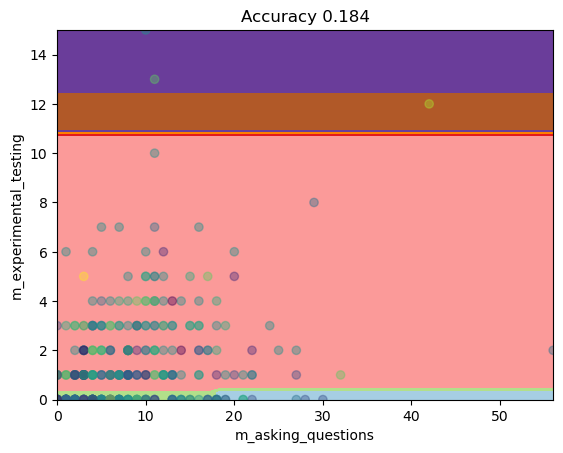
Diagram

Description automatically generated

In figure \_, the important features were determined based on how much they reduced the Gini scores at each node. Therefore, by this reasoning, some important features could be m\_experimental\_testing, m\_asking\_questions and m\_making design\_choices The reason why word count was discounted was because, like previously mentioned, the inclusion of word count as an important feature does not produce useful insights and word count includes noisy data from messages which could be filled with meaningless words. Upon closer inspection, the Gini scores shown are still extremely high which indicates strong signs of misclassification, deducing the reason for such low accuracy scores being produced. Using a more precise function, the importance of features in terms of their influence on outcome scores is shown in figure\_ on the table.

|  |  |
| --- | --- |
| Feature | Importance |
| wordCount | 0.296 |
| m\_experimental\_testing | 0.242 |
| m\_asking\_questions | 0.174 |
| j\_performance\_parameter\_requirements | 0.133 |
| m\_making\_design\_choices | 0.117 |
| j\_communication | 0.04 |
| j\_customer\_consultants\_requests | 0.0 |

The quantity of ‘importance’ of a feature was measured by what proportion of variability each feature could ‘explain’ in the data. As observed, the most important features besides wordCount were m\_experimental\_testing and m\_asking\_questions. However, all features have very low feature importance quantities, signifying a lack of a definite relationship between these variables and the outcome scores of interns. A decision boundary can be used to visualise the performance of the algorithm.



In figure \_, the model is very inaccurate because there are clear signs of underfitting. (?) Evidence of this can be seen through the extremely large pink (?) region which classifies majority of students into the ‘4’ category, likely because there is a high degree of tendency for students to achieve that score. Furthermore, because of this occurrence, whereby majority of students achieved a score of ‘4’, there seems to be a lack of a relationship between the number of questions asked for each of these categories despite them being two of the most relatively important features. This suggests that perhaps, no relationship between the frequency of the types of questions asked exists at all.

A potential way to improve this model would be to allow GridSearchCV to cross validate over a wider range for each parameter, or to increase the number of cross validations. However, this would not necessarily be effective because of the extremely sensitive nature of decision trees to noise in the data set, leading to increasing randomness. The next sensible choice would be to utilise random forests to combat the primary issue of sensitivity to noisy data.

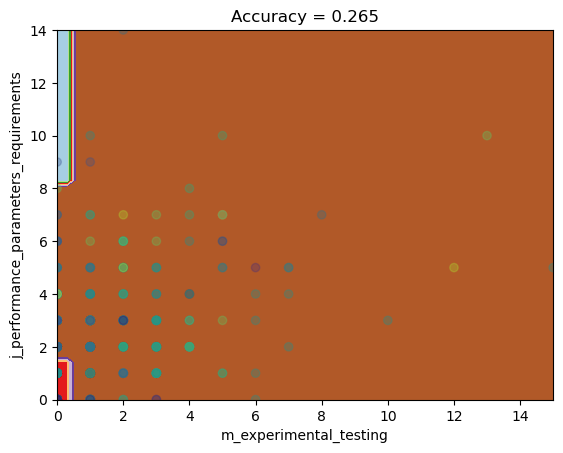
Each paragraph must answer these 5 questions: Why did we choose the model?  How did we get the hyperparameters? Why did we modify the hyperparameters we did? Why did we use a certain optimisation method in a certain way? How can we improve the model?

Random forests

The random forests classifier model was chosen to overcome the issues encountered by decision trees, such as sensitivity to noisy data and having a high variance. The random forest classifier algorithm operates analogously to a series of repeated decision tree classifiers under different randomised conditions and retains the aggregate result of the repeated operations as the output. This also combats the issue of bias because of the repeating mechanism running the decision tree classifier on a greater area of the data set. The method chosen to tune the hyperparameters was grid search cross validation, which was used to maintained consistency. The hyperparameters chosen for tuning were the number of estimators, maximum depth and minimum number of samples required before reaching a leaf. These hyperparameters were chosen to retain some consistency when comparing the performance of the random forests classifier algorithm with the decision trees algorithm. Both algorithms achieved very similar results, which was to be expected to some degree since random forests produced an aggregated result of a series of decision trees. For example, similar accuracies were achieved as demonstrated in figure \_. However, another important feature was discovered by using the random forests algorithm. The order of feature importance is shown in the table at figure\_.

|  |  |
| --- | --- |
| Feature | Importance |
| m\_experimental\_testing | 0.299 |
| j\_performance\_parameters\_requirements | 0.211 |
| m\_making\_design\_choices | 0.195 |
| m\_asking\_questions | 0.170 |
| j\_customer\_consultants\_requests | 0.071 |
| j\_communications | 0.054 |

An issue with using random forests was that majority of interns’ scores were now classified into a single target label which likely occurred because of the repeated nature of random forests, as opposed to the non-iterative nature of decision trees, which naturally decreased variance in the potential targets that interns could be assigned.



The model can be greatly improved by exploring more hyperparameters to tune during cross validation rather than performing a comparison between the results of a single decision tree classifier and the aggregated results of multiple decision trees.

<https://towardsdatascience.com/scikit-learn-decision-trees-explained-803f3812290d>

<https://www.kdnuggets.com/2022/10/hyperparameter-tuning-grid-search-random-search-python.html>