5.2 Assignment: Predictive Analytics Case Study

Help Desk Case Study

Harlan Wittlieff

Data Science, Bellevue University

DSC 630: Predictive Analytics

Dr. Brett Werner

April 16, 2022

Introduction

A large U.S. corporation sought to improve their help desk operations by gaining insight into their service desk tickets. Some of the service requests required a physical part to resolve the reported issue. The ability to predict whether a new ticket would require the replacement and installation of a part could remove inefficiencies in the company's process, thus impacting their bottom line.

To begin the process, the team gathered all of their historical data from past service tickets. The dataset was sizeable, containing over one million tickets. Numerous variables were available within the dataset including the following:

- Call timestamp

- Ticket originator

- Country

- Error codes

- Ticket reason

- Warranty details

- Problem description

- Outcome

- Required parts

The target of the study was to predict not only if a ticket needed a part, but also which specific part is required.

Method and Results

Several issues arose with the initial data. For example, there were instances where another issue was found during service that required a part. Since the part was not needed for the original problem captured in the initial service ticket, a new ticket had to be opened to capture the subsequent issue(s) discovered. Another issue that was found in the data was that a part would be used to resolve a ticket temporarily, and then after a short period of time the issue would reappear, and a new ticket would be opened.

The next step for data preparation was to create features from the problem description text field. The original analysis team was fortunate to have a member of their team who was extremely skilled in SQL. This individual was able to translate various synonyms and misspellings into key stem words.

Finally, a dummy variable was created for the target variable of needing a part. Zero was classified as no part needed whereas one was used on tickets requiring a part.

After the data was cleaned, the next step was to prepare a model. The team settled on attempting to build a decision tree model due to the fact that the ticket data was mostly categorical and contained a large number of candidate inputs. Decision trees also provide the benefit of being easy to understand. The conciseness of decision trees would allow for support engineers to quickly discover the reasoning behind a ticket's predicted part need.

After building hundreds of decision trees, the most successful terminal node predicted a 50.5% likelihood of needing a part. Unfortunately, this result failed to meet the business objective. This caused the analysts to redouble their efforts and attempt to find a better solution.

The team then turned to Random Forests due to their increased flexibility over decision trees. The result was higher accuracy. However, the lack of transparency was a downside of this model in addition to not meeting the business objective.

Due to the lack of success in the previous models, the team went back and created a new feature. Instead of having a one zero dummy variable as the target, the team pivoted to using a percentage likelihood of needing a part for a specific ticket based on historical data prior to each individual ticket's opening. This enriched the predicted field. The team then built a variety of random forest models. Instead of using the entirety of one model, they cherry-picked interesting terminal nodes that identified the subsets that contained the highest and the lowest likelihood of requiring a part.

Conclusion

After identifying the most interesting rules, the team converted them to SQL for implementation. Calls with a high likelihood of requiring a part were diverted to a technician, while calls not likely to require a part were handled by the call center. The implementation of these rules allowed for increased efficiency and significant cost savings for the company.

Overall, the team learned that with this type of data it is best to identify key relationships instead of trying to build an all-encompassing model. Going forward, starting with this mindset could save analysts time by avoiding the development of unsuccessful models that fail for these same reasons. Additionally, implementing best practices while gathering data for tickets to reduce the risk of misclassifying tickets that contain another issue or tickets that contain reoccurring issues could help identify more key relationships in the future.

References

Abbott, D. (2014). *Applied predictive analytics: Principles and techniques for the professional data analyst*. Wiley.