Customer Churn Prediction: A Comparative Analysis of Machine Learning Algorithms

Project Plan

Abdulahi Said

CS4822 - MSci Final Year Project

Supervised By: Dr Anand Subramoney

Department of Computer Science

Royal Holloway, University of London

Contents

1. Abstract

When clients discontinue using a service, customer churn becomes a critical issue across many industries, especially for subscription-based businesses. In such sectors, customer satisfaction and retention are vital to ensuring steady revenue streams. With intense competition, every lost customer can represent a direct gain for competitors [1]. On average, businesses lose between 10% and 25% of their customer base annually due to churn, with some industries, such as wholesale, experiencing rates as high as 56% [2][3].

High churn rates undermine a business's ability to establish and maintain a loyal customer base, essential for long-term profitability. Studies have shown that retaining customers is much more profitable and sustainable [5]. This is because customer retention requires fewer resources compared to acquiring new customers and because loyal customers tend to spend more over time. As a result, businesses that focus on strengthening relationships with existing clients can boost their profitability while ensuring steady growth.

To address this known issue, this project aims to conduct a comparative analysis of various machine learning (**ML**) algorithms designed to predict customer churn effectively. These algorithms include K-Nearest Neighbours (**KNN**), Decision Trees (**DT**), Support Vector Machines (**SVM**), and Random Forest **(RF**). A significant challenge in churn prediction is class imbalance. This occurs as the number of churners is outnumbered by customers that remain. This can lead to biased model predictions, increasing the number of false positives and false negatives. Such inaccuracies have damaging repercussions on businesses as they may waste financial and marketing resources targeting the wrong customers while overlooking actual customers who would leave. To deal with these issues, I will implement robust data preprocessing techniques, such as handling missing values and scaling features. Additionally, I will explore ensemble methods to improve model accuracy and generalisation [6].

Furthermore, my project will apply these ML models to a benchmark dataset. Each algorithm's performance will be measured using various measures, including precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (**AUC-ROC**), thus ensuring comprehensive model assessment. To enhance accessibility for non-technical users, I will develop a GUI that allows users to visualise the performance of selected ML models, making it more user-friendly and intuitive.

2. Specification

- I will use Python, a programming language suited to machine learning projects. It has many supported libraries for this project.
- Jupyter Notebook [7] would be used as I can combine visualisation of results and allow for much easier code documentation.
- Matplot library [8] to visualise data results and carry out analysis.
- Numpy library [9] to handle data and computations.
- Scikit learn [10] for splitting customer data to be trained and tested
- RHUL GitLab [11] repository for version control of this project

3. TimeLine

For term 1, I will focus on understanding ML algorithms and applying the standard algorithms. Term 2 is where I would put much more effort into implementing complex and hybrid algorithms. I will also leave time for extensions such as multi-classifiers of risk of churning and any other ideas that may seem fit to improve my project.

3.1 Term1

Week 1 :

- Carry out my research into churning
- Select my dataset for my project
- Review the fundamentals of ML

Week 2-3 :

- Preprocess the dataset(handle outliers, missing values, convert some features into binary values)
- Investigate class imbalance
- Apply some sampling techniques to handle class imbalances

Week 4 :

- Review KNN algorithm
- Implement the KNN model for churn prediction
- Evaluate the performance of the model using p-value, cross-validation, accuracy
- Investigate to find the optimal value of K

Week 5-6 :

- Implement decision tree model
- Compare performance with the KNN model
- Finetune preprocessing based on performance score

Week 7-8:

- Implement support vector machine model
- Compare performance metrics with other models

Week 8-9:

- Develop prototype GUI
- Carry out further model fine-tuning.
-

Week 10-11:

- Preparation of presentation and interim report

3.2 Term 2

Week 1-2:

- Investigate feature selection techniques
- Research hyperparameter optimisation techniques
- Implement these on existing models
- Evaluate models to see any noticeable improvements

Week 3-4:

- Extend to multi-class churn prediction: low, medium and high risk of churning
- Research more advanced ensemble methods
- Implement ensemble methods such as gradient boosting
- Compare performance to single classifiers such as **KNN** and **SVM**

Week 5:

- Comprehensive performance analysis of all models
- Use various metrics such as **AUC-ROC**, F1 score

Week 6:

- Research more advanced techniques for handling class imbalances, such as Cost Sensitive learning
- Re-assess model performances

Week 7-8:

- GUI Development to allow users to compare performances of selected models
- Any additional ML algorithms can be implemented this week, such as logistic regression/Naïve Bayes

Week 9:

- Codebase refactoring to align with SE principles.
- Any bug fixes and further testing
- Code documentation (make sure all code is fully documented)
- Make sure functions are not overly complex

Week 10-11:

- Prepare for the final report and presentation

4. Risks and mitigation

There are always risks in projects, some with a higher likelihood and some with a low impact on the project. Each risk would be categorised into low, medium, and high for Impact and Likelihood of occurring. I will also discuss ways of mitigating each risk.

4.1 Project Delays due to time mismanagement

Likelihood: Medium

Impact: High

Time allocated to specific tasks, such as developing and testing the SVM model, may take longer than usual. This can create a chain of effects on my project plan. To successfully deal with this issue, I will break weekly tasks into sub-goals, making understanding each task's length easier. I will also use Trello, a project management tool. It would allow me to monitor my progress on weekly tasks, ensuring that I don't fall behind and improve my time efficiency.

4.2 Imbalanced Dataset

Likelihood: High

Impact: Medium-High

Churn datasets usually need to be more balanced. This is because there are often much fewer churners compared to non-churners. This would lead to a skewed model performance as there is a bias towards predicting the majority class(non-churners). As a result, we could face many false negatives where actual churners are predicted as non-churners, defeating the purpose of accurately predicting churners. To combat this issue, I will experiment with different training set resampling techniques, such as SMOTE and adjust my model evaluation metrics [6]. I may also use cross-validation, where I train the model with different subsets of data and test using a subset of data. Repeating the process multiple times with different subsets would help reduce bias. With this, I can find what works best, but it will require much experimentation and be more time-consuming.

4.3 Resource Constraints

Likelihood: Low

Impact: Medium

I could face computing resource limitations while processing and training large datasets. To combat this, I would select a dataset that is not too taxing on computing resources when training and testing different models. Also, I could use external computational resources such as cloud computing to run models on virtual machines.

4.4 Issues with Data Quality

Likelihood: Medium

Impact: High

The dataset might contain missing or inconsistent values, negatively affecting the quality and accuracy of the model's training. Missing data can occur for several reasons, such as incomplete customer records and data entry errors. Inconsistent or missing data values would hinder the model's ability to establish connections between features and labels (churn prediction). To combat this, I will implement preprocessing methods at the start of the project to identify problems before building any models. This would include performing explorative data analysis of data, where findings would be documented clearly in Jupyter notebook.

4.5 Model overfitting

Likelihood: Medium

Impact: High

The models I create could have issues with overfitting data during the training phase, leading to poor results when testing unseen data. I could combat this by using cross-validation, which ensures that models are evaluated on multiple data subsets, reducing overfitting risks [12]. However, I must be careful of data leakage, which can occur in the preprocessing stage of feature scaling and normalisation, which is being done improperly. This would enable the model to access test data during the data training phase. To mitigate this issue, I will ensure that all data processing is done within each fold of cross-validation through pipelining.

4.6 Model Performance and Comparison in GUI

Likelihood: Medium

Impact: Medium

Developing a GUI to visualise and compare model performances could lead to challenges in displaying their results. This is due to the possibility of models utilising different evaluation metrics, making it difficult to assess the overall comparability of models. Furthermore, computing the results of models on the spot may take time, potentially affecting user experience and delaying the evaluation of model performance. A mitigation strategy would be to standardise metrics, allowing users to select specific metrics to compare model performances. Also, implement caching the results of previously computed metrics to speed up the comparisons. The results from the models I build would also be presented visually through graphs to make comparisons much more intuitive and user-friendly

5. Acronyms

AUC-ROC Area Under the Receiver Operating Characteristic Curve

DT Decision Tree.

RF Random Forest

KNN K Nearest Neighbours

SVM Support Vector Machine

ML Machine Learning.

SMOTE Synthetic Minority Oversampling Technique

6. References

[1] Oliver Wyman. (2004). *Customer churn*. Available at:https://www.oliverwyman.de/content/dam/oliver-wyman/global/en/files/archive/2004/CMMJ17_Customer_Churn.pdf

[2] Outsource Accelerator. "Crucial Customer Retention Statistics to Know in 2024." Outsource Accelerator, 2024. https://www.outsourceaccelerator.com/articles/customer-retention-statistics

[3] Tessitore, Sabrina. "What's the Average Churn Rate by Industry?" *CustomerGauge*, 2022. Available at: https://customergauge.com/blog/average-churn-rate-by-industry.

[4] Reichheld, F. F., & Schefter, P. (2000). *E-Loyalty: Your Secret Weapon on the Web*. Harvard Business Review.

[5] Gefen, D. Customer Loyalty in e-Commerce. J. Assoc. Inf. Syst. **2002**

[6] Morrison, J., & Vasilakos, A. (2018). An empirical comparison of techniques for the class imbalance problem in churn prediction.

[7] Jupyter Notebook: **https://jupyter.org**

[8] Matplotlib: https://matplotlib.org

[9] NumPy: https://numpy.org

[10] Scikit-Learn: https://scikit-learn.org

[11] RHUL GitLab: https://gitlab.rhul.ac.uk

[12] Underwood, J. Data preparation for automated machine learning. https://datateam.mx/downloads/datarobot/Data-Preparation-for-Automated-Machine-Learning.pdf