

Sequential User Retention Modelling

HELDER MARTINS

Master in Machine Learning

Date: February 8, 2017

Supervisor: Hedvig Kjellström (KTH) and Sahar Asadi (Spotify)

Examiner: Patric Jensfelt

Swedish title: Detta är den svenska översättningen av titeln

School of Computer Science and Communication

Abstract

English abstract goes here. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Sammanfattning

Träutensilierna i ett tryckeri äro ingalunda en faktor där trevnadens ordningens och ekonomiens upprätthållande, och dock är det icke sällan som sorgliga erfarenheter göras ordningens och ekon och miens därmed upprätthållande. Träutensilierna i ett tryckeri äro ingalunda en oviktig faktor, för trevnadens ordningens och och dock är det icke sällan.

Contents

Contents	iii
1 Introduction	1
1.0.1 Music as a service	1
1.0.2 Spotify	1
1.0.3 Churn prediction	2
1.0.4 User retention	2
1.0.5 Goals	2
1.0.6 Ethical considerations	2
1.0.7 Research Question	2
2 Background	4
2.0.1 Definitions	4
2.0.2 Feature Selection	4
2.0.3 Predictor Models	4
2.0.4 Evaluation Metrics	4
3 Related Work	7
4 Methods	10
Bibliography	12
A Unnecessary Appended Material	13

Chapter 1

Introduction

Service providers, especially those that operates on the mobile market, have experienced a rapid increase in their user base in the last few years. However, the number of companies in each market increased in a similar fashion, and maintaining your current users within your service grew in importance since the process of acquiring new users is costly and difficult to execute properly. One of the important steps towards that goal is to predict accurately what is the chance of a user leaving the service provider at any given time with the intent of acting before it happens, a problem which is called *churn prediction*. Several classical techniques for predicting churn like Decision Trees and Logistic Regression have been used on recent work, however most of them make use of user behaviour data at a single point in time, normally the most recent one when the dataset was created.

Intuitively one can think that latent factors hidden in the temporal axis of the user behaviour data could yield a better prediction accuracy when comparing to a model which leverages a single and static point in time. For instance, a user that is gradually reducing its consumption of the service over time can be easily thought of a prospect churning. While this intuition is trivial to come up with, we are also interested in learning strong correlations between temporal aspects of the data and the churn rate which are not as clear to the eye. Our hypothesis is that making use of the properties hidden on user behaviour data over time, a churn rate predictor could improve its accuracy greatly when compared to methods which are static on time.

Music as a service

TODO: paragraph / plots about the state the music industry is in today

Spotify

TODO: paragraph introducing the company and its goals (activation problem, high rate of churn for new users)

The user data and the required computational resources shall be provided by Spotify AB, a well known music streaming service, which joined this project in a tutoring partnership with the student and the university. Creating a suitable dataset for training our proposed models is also part of the project, and shall be done by first exploring the data at our disposal and identifying the features that highly correlates with churn. An exam-

ple of what this dataset may contain is the user listening habits, usage patterns, user profile, and so on.

Churn prediction

TODO: add churn prediction focused on Spotify

User retention

TODO: add user retention info on Spotify

A churn rate predictor which can identify possible churners accurately is just a trigger on the user retention process that can span several different stages. One of these stages is to identify what actions can be taken by the service provider as to avoid the user abandoning the service. An important problem to solve beforehand is to identify which features of the data have a strong statistical correlation with the churn rate. With that information, the providers could for example set up automated actions as to influence the value on these features. Learning which are the most important features is also of relevance for the task of choosing what data to use for training the predictor models, since commonly service providers have a vast amount of data about the user but only a fraction of that is of importance for predicting churn: training models with a full dataset will commonly introduce error on the system and take a large amount of computing resources to train.

Goals

TODO: review with supervisor

The goal of this project will be to research different models for churn prediction that can make use of the temporal aspect of user behaviour data at its fullest. Our hypothesis is that latent factors hidden on usage patterns may increase the accuracy of state-of-art predictors which considers only the state of the data at a single point in time. We shall experiment on different models that are known to perform well on time series data and compare their accuracy using different evaluation methods.

Detecting possible churners, while important, does not improve user retention without an associated action performed by the service provider. To derive insights on what can be done to improve this metric, a deep data exploration and feature analysis is also a goal of this project. Features shall be correlated to their influence on the churn rate, and feature selection techniques shall be researched and implemented in an attempt to reduce bias on the churn classifiers.

Ethical considerations

TODO: Is user data anonymous? Is it relevant to add this section?

Research Question

TODO: rewrite

Can latent factors hidden on user behaviour data improve the accuracy of a churn predictor model when compared to the ones which does not leverage this property? Can techniques proven to be successful for time series prediction on different domains be

used interchangeably for prediction of churn rate? What feature selection methods can most successfully extract the correlations between the feature value and churn rate? Can we reasonably interpret what was learnt by our predictor models as to allow the service provider to create actions to keep user retention level high? Can we identify different types of users which exerts an influence on the churn rate of the service?

Chapter 2

Background

Definitions

Definition 1 *A session is defined as the time period the user is engaged into the application, be it to stream music or to navigate through the available content. A session starts when a user interaction trigger is captured (eg. a button click), and the session ends when no such trigger is captured and no continuous activity is being performed (eg. listening to a song) for at least 15 minutes.*

The session is the basic measurement of activity of the user on the service. Users are considered *active* if they are currently on a session, and *inactive* otherwise. (TODO Check Sahar)

Definition 2 *A user is defined to have churned if no session of his was registered in the last 30 days.*

Note that this definition is not concerned whether the user is a paying customer or not, but only if he is actually making use of the service. For example, a user may be subscribed to a Premium paid account and still be considered to have churned if no activity was registered in the last 30 days. However, it should be noted that empirical observations by the service provider suggest that there is a significant behavioural difference between the two classes of users that will be taken into consideration when the experiments are performed. (TODO Check Sahar)

Feature Selection

TODO: add feature selection techniques (Information gain, L1-norm, Recursive Feature Elimination, Variance Threshold...)

Predictor Models

TODO: add models (Decision Trees, Logistic Regression, Random Forests, LSTMs, CNNs...)

Evaluation Metrics

Confusion Matrix

A *confusion matrix* (also called contingency table) is a table layout representing the performance of a classifier's output, judging by its predictions against the actual true values.

In a binary classification problem, the confusion matrix is a 2×2 table where commonly the rows represent the true labels while the columns the predicted classes. Each cell contains a count of how many samples were classified on that category, and the values in the diagonal represent the correctly classified samples (if the features are ordered). Table (TODO: Add table) depicts a confusion matrix for a binary classifier.

The confusion matrix is an excellent visualization tool to estimate the performance of a model. The values that should be maximized, the *true positives* (TP) and *true negatives* (TN), represent the samples that were correctly labelled as possessing the feature of interest or not, respectively. On the other hand, the *false positives* (FP) and *false negatives* (FN) are the misclassified samples where the algorithm predicted that the feature was present while in truth it was not and vice-versa. In this work, the positive class will always represent a churning user, and it follows that the negative class represents the non-churners.

Classification Accuracy

Several different metrics can be derived from the confusion matrix table. The *classification accuracy* (CA) of a model is a common metric that corresponds to the fraction of the correctly classified samples on the test set, and can be calculated as follows:

$$CA = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.1)$$

While trivial to understand, this metric may lead to erroneous conclusions when class imbalance is present in the test set, which is a common occurrence on the churn prediction domain. For example, if 9 out of 10 users of a dataset are non-churners, any classifier that simply outputs a negative class for all samples will result in an accuracy of 90%, however its ability of detecting churners is non-existent. For a service provider, detecting churn cases is always more important than detecting the loyal users, and this metric by itself cannot represent this goal. (TODO Add ref? Hassouna 2015)

Precision, Recall, Fall-out and False Alarm

To address the class imbalance problem of the classification accuracy, others metrics are also commonly used. The *positive predictive value* (PPV, also known as precision) is the proportion of the samples labelled as positive which are true positives, and describes the performance of the algorithm. The *true positive rate* (TPR, also called sensitivity and recall) of a model corresponds to the number of correctly predicted positive samples divided by all positive samples. *True negative rate* (TNR, also called specificity and fall-out) is the number of correctly predicted negatives divided by all true negatives. The *false positive rate* (FPR, also called false alarm ratio) is the probability of receiving a false positive as output of an experiment, and is calculated by dividing the number of false positives by the total number of positive samples.

$$PPV = \frac{TP}{TP + FP} \quad (2.2)$$

$$TPR = \frac{TP}{TP + FN} \quad (2.3)$$

$$TNR = \frac{TN}{TN + FP} \quad (2.4)$$

$$FPR = \frac{FP}{TN + FP} = 1 - TNR \quad (2.5)$$

Depending on the distribution of classes of the dataset, it is often difficult (although desirable) to maximize both metrics at the same time. A compromise must be reached that achieves the best trade-off between the two, which is commonly a business decision. For a music streaming service like Spotify, maximizing sensitivity is preferred due to the costs associated with a churning user, however a reasonable specificity is also a metric that should be strived for. (TODO Check with Sahar)

Receiver Operating Characteristic

The *receiver operating characteristic* (ROC) is a visualization tool that plots the relationship between the true positive rate (commonly the y-axis) and the false positive rate (the x-axis) of a binary classifier system. The curve is drawn by selecting different parameters of a model or levels of threshold for the decision boundary between the positive and negative classes. For example, when the output of a classifier is a probability value (like in logistic regression), different thresholds can be chosen to decide whether a user is a churner or not, depending if the goal is to minimize FPR or maximize TPR. An example of a ROC curve can be seen in Figure (TODO Add figure)

A good classifier would score values close to the upper-left corner of the plot, where the point (0,1) represents a perfect classifier with 100% TPR and 0% FPR. On the other hand, an algorithm that outputs a curve alongside the diagonal where TPR and FPR are almost the same at different threshold levels can be considered close to a random guess, like the flip of a coin. A classifier would underperform if its scores are closer to the bottom-right corner of the plot, however this result can always be mirrored by updating the model to simply invert the positive and negative labels of the classified samples.

ROC curves can be used to compare the performance of different models by measuring the *area under the curve* (AUC) of its plotted scores, which ranges from 0.0 to 1.0. The greater this area, the better the algorithm is to find a specific feature. Moreover, models with an area close to 0.5 can be assumed to perform not much better than random guess, since this is the total area under the diagonal line.

TODO: Add eval methods (F-score, Lift chart...)

Chapter 3

Related Work

Pudipeddi, Akoglu and Tong [1] In "User Churn in Focused Question Answering Sites: Characterization and Prediction", Q&A sites like Stack Overflow were the focus of a study on user behaviour characterization and churn rate prediction. An extensive data exploration was made as to correlate features to the chance of a user leaving a service, and with those insights classical modelling techniques were used, where the best performing one was a decision tree. The approach used for extracting and categorizing features (temporal, frequency, gratitude, etc) and the insights that follow the study (like the importance of temporal features for predicting churn) is of high value and can be mapped to concepts on a different domain like a music streaming service with minor modifications.

Wangperawong, Brun, Laudy and Pavasuthipaisit [2] In "Churn analysis using deep convolutional neural networks and autoencoders" users were represented as 2-dimensional images where columns are tracked features and rows are days. The intuition behind this approach is that by using similar techniques successfully applied on the Image Recognition domain, a similar positive result for the prediction of the churn rate could also be achieved. Two different Convolutional Neural Network models were used, and both outperformed a baseline Decision Tree model. An autoencoder was also used as to discover which features influenced churn rate the most, and suggestions were made as to how improve the retention of users on the service.

Tax and Verenich [3] In "Predictive Business Process Monitoring with LSTM Neural Networks" a technique was presented as to predict the next event and its timestamp on a running case of a business process (a help desk process, for example). The problem definition was formally explained, and three different LSTM architectures were experimented on. With these architectures, three problems were tackled: estimating the next activity and its timestamp, all the remaining activities in a use case and the remaining time of a process. This technique could be used for churn prediction by interpreting the user interaction with the service as actions, and "churn" could be an event that may or may not exist in the process chain.

Runge et al. [4] In "Churn prediction for high-value players in casual social games", players responsible for generating the most profit on two casual social games were the focus of a study on churn analysis and prediction. Formal definitions for active players and churning were made, and the problem was defined as a binary classification task where users were labelled as leaving the service or not in the following week of when the models were trained. Four different models were then trained, and a neural network

classifier obtained the best area under the curve (AUC) score, with logistic regression as a close second. An attempt was made as to include temporal data in the neural net by enhancing it with features learned with a hidden Markov model (HMM), however the accuracy was decreased due to parameter overfitting. The formal definitions of "churn" and "activity" can be applied to a music streaming service in a similar way, as the evaluation of performance using a series of receiver operating characteristic (ROC) curves. A detailed A/B test with real users was also performed, but a similar approach falls out of the scope of this project.

Dror et al. [5] In "Churn prediction in new users of Yahoo! answers", an explorative study was made on the Yahoo! answers website as to discover an efficient churn predictor model and also the features that correlates with the user leaving the service, however differently from other works this paper focused on new users with less than a week of activity on the service. Features were grouped into Question, Answer and Gratification categories, and were used for training several different classifiers. For this dataset, random forests performed the best with logistic regression once again a close second. Features were also ordered by the amount of information gain that they provide, and the number of questions and answers are the top features on that regard (inversely correlating with churn), followed by the period of time the user is active and gratification features for answers given and questions made. Similarly to [1], the features exploration section could be mapped to a music streaming service in a similar way (eg. question/answers are sessions, gratitude is the user explicit feedback on recommended content), and the evaluation methods with ROC curves and information gain tables are also of interest.

Hassouna et al. [6] In "Customer Churn in Mobile Markets: A Comparison of Techniques", two popular models for predicting user churn were empirically compared: decision trees and logistic regression. By making use of the dataset provided by a mobile operator, two models were created by independently training and selecting their best performing versions. Evaluation was executed by comparing the AUC of their ROC curves, as also their Lift score and overall accuracy. The conclusion is that decision trees consistently perform better when compared to logistic regression models, and thus should be a preferred choice.

Lu et al. [7] In "A Customer Churn Prediction Model in Telecom Industry Using Boosting", churn classification was performed on users of a telecom provider in a rather unique way. The Gentle AdaBoost boosting algorithm was used alongside a logistic regression base learner as to train a model to do the separation between the churning and non-churning classes. However, one further step was performed where the users were split into two clusters based on the weights assigned by the boosting algorithm. One of the clusters was identified to have a considerably larger churn rate than the other, and thus another logistic model was trained using the clusters as labels, and its performance evaluated using ROC curves. The idea behind this approach is to use a model to mark users who have a high risk of leaving the service (and be the focus of user retention actions), instead of labelling them as churners and non-churners directly.

Khan et al. [8] In "Behavioral Modeling for Churn Prediction: Early Indicators and Accurate Predictors of Custom Defection and Loyalty", billions of call detail records from a mobile phone operator was the focus of a churn prediction study and feature analysis. The initial data set was filtered to the calls made by 100.000 subscribers during a 6 month period. A brute-force approach to feature engineering was then performed to create 12.914 out of the initial 10 features by combining every feature from each of the

manually split 8 different groups. Feature selection is then performed in two distinct ways: first, an individual Student's t-test score is computed for each individual feature to evaluate how well it can differentiate between the churners and non-churners sets. Second, a tree-based method was used as to estimate the the accuracy of a joint classifier by adding features one at a time. Features are then ordered by their statistical correlation with churn, and the top 100 features were selected to train several different classifiers. Evaluation was performed mainly by comparing the AUC of each model, where Ad-aBoost using a decision tree classifier performed the best, followed closely by a logistic regression model.

Burez and Van den Poel [9] In "Handling class imbalance in customer churn prediction", a recurrent problem on data sets used for churn prediction has taken center stage: the uneven distribution between the churning and non-churning classes. It is a common scenario for the number of non-churners to heavily outweigh the number of churning samples on real data sets, and since the main interest of any classifier is the ability to spot this rare class appropriate evaluation methods should be considered. This paper praises the use of AUC and Lift as suitable metrics for model comparison while detailing several techniques used in the field. The performance boost estimated to be received by using a cost-sensitive learning algorithm (where false negatives are assigned a greater cost than false positives) is also evaluated, and it has taken the form of a weighted random forest model which was compared to other classical baseline models like logistic regression and random forests. Under-sampling, where fewer samples from the majority class are incorporated in the training data as to artificially change the distribution between the labels, is proved to significantly improve the performance of the underlying models, however the exact ratio between churners and non-churners is confirmed to be case-dependant, not necessarily being the even distribution the perfect choice. Another contribution of notice is the use of a 5x2 cross-validation technique to train the models, and the use of a formal F-test for judging whether two algorithms are statistically different or not.

Ballings and Van den Poel [10] In "Customer event history for churn prediction: How long is long enough?", this study investigates a common but under-researched property of customer behaviour data sets, which is the time dimension of the data. More specifically, this paper strives to answer how far in the user history models must be trained on as to achieve the best trade-off between accuracy and computational burden. The experiment consisted on training three different models (logistic regression, decision trees with and without bagging) on a newspaper data set consisting of 16 years of user history. The models were evaluated for their performance by training with data ranging on this interval with a 1-year gap between each measurement. It has been concluded that after year 5 no gain in performance (measured by AUC) is statistically relevant enough to warrant the increase in computational power needed for the training. However, the researchers also question whether this result can be extended for other domains or not, leaving that for future work.

Chapter 4

Methods

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam

pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Bibliography

- [1] Jagat Pudipeddi, Leman Akoglu, and Hanghang Tong. User Churn in Focused Question Answering Sites: Characterizations and Prediction. pages 469–474, 2014.
- [2] Artit Wangperawong, Cyrille Brun, Dr. Olav Laudy, and Rujikorn Pavasuthipaisit. Churn analysis using deep convolutional neural networks and autoencoders. pages 1–6, 2016.
- [3] Niek Tax, Ilya Verenich, Marcello La Rosa, and Marlon Dumas. Predictive Business Process Monitoring with LSTM Neural Networks. 2016. URL <http://arxiv.org/abs/1612.02130>.
- [4] Julian Runge, Peng Gao, Florent Garcin, and Boi Faltings. Churn prediction for high-value players in casual social games. *IEEE Conference on Computational Intelligence and Games, CIG*, 2014. ISSN 23254289. doi: 10.1109/CIG.2014.6932875.
- [5] Gideon Dror, Dan Pelleg, Oleg Rokhlenko, and Idan Szpektor. Churn prediction in new users of Yahoo! answers. *Proceedings of the 21st International Conference on World Wide Web*, pages 829–834, 2012.
- [6] Mohammed Hassouna, Ali Tarhini, Tariq Elyas, Mohammad Saeed Aboutrab, United Kingdom, United Kingdom, and Saudi Arabia. Customer Churn in Mobile Markets: A Comparison of Techniques. 8(6):224–237, 2015. doi: 10.5539/ibr.v8n6p224.
- [7] Ning Lu, Hua Lin, Jie Lu, and Guangquan Zhang. A customer churn prediction model in telecom industry using boosting. *IEEE Transactions on Industrial Informatics*, 10(2):1659–1665, 2014. ISSN 15513203. doi: 10.1109/TII.2012.2224355.
- [8] Muhammad Raza Khan, Joshua Manoj, Anikate Singh, and Joshua Blumenstock. Behavioral Modeling for Churn Prediction: Early Indicators and Accurate Predictors of Custom Defection and Loyalty. *Proceedings - 2015 IEEE International Congress on Big Data, BigData Congress 2015*, pages 677–680, 2015. doi: 10.1109/BigData-Congress.2015.107.
- [9] J. Burez and D. Van den Poel. Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, 36(3 PART 1):4626–4636, 2009. ISSN 09574174. doi: 10.1016/j.eswa.2008.05.027. URL <http://dx.doi.org/10.1016/j.eswa.2008.05.027>.
- [10] Michel Ballings and Dirk Van Den Poel. Customer event history for churn prediction: How long is long enough? *Expert Systems with Applications*, 39(18): 13517–13522, 2012. ISSN 09574174. doi: 10.1016/j.eswa.2012.07.006. URL <http://dx.doi.org/10.1016/j.eswa.2012.07.006>.

Appendix A

Unnecessary Appended Material