Towards a Large Scale Practical Churn Model for Prepaid Mobile Markets

Amit Kumar Meher*, Jobin Wilson, R. Prashanth

R&D Department
Flytxt
Trivandrum-695581, India
{amit.meher, jobin.wilson, prashanth.ravindran}@flytxt.com

Abstract. Communication Service Providers (CSPs) are increasingly focusing on effective churn management strategies due to fierce competition, price wars and high subscriber acquisition cost. Though there are many existing studies utilizing data mining techniques for building churn prediction models, addressing aspects such as large data volumes and high dimensionality of subscriber data is challenging in practice. In this paper, we present a large scale churn management system, utilizing dimensionality reduction and a decision-tree ensemble learning. We consider subscriber behavior trends as well as aggregated key performance indicators (KPIs) as features and use a combination of feature selection strategies for dimensionality reduction. We report favorable results from evaluating our model on a real-world dataset consisting of approximately 5 million active subscribers from a popular Asian CSP. The model output is presented as an actionable lift chart which will help marketers in deciding the target subscriber base for retention campaigns. We also present the practical aspects involved in productionizing our model using the Apache Oozie workflow engine.

Keywords: Prepaid churn management, Large scale prediction, Feature engineering, Dimensionality reduction, Machine learning, Actionability, Operationalization

1 Introduction

Competition in the wireless telecommunication industry has become rampant due to its dynamic nature with new products, services, technologies, and carriers emerging. New rates and incentives are being frequently exercised by Communication Service Providers (CSPs) to attract new subscribers. With the availability of similar services with different price levels, subscribers prefer switching to different CSPs, resulting in significant revenue loss. Other factors such as inadequate connectivity, intermittent call drops and poor customer service also compel subscribers to switch to different CSPs. Wireless carriers across the world report churn rate varying from 1.5% to 5% per month [1, 2]. As subscriber acquisition costs are considerably higher as compared to retention costs [3, 4, 5],

P. Perner (Ed.): ICDM 2017, LNAI 10357, pp. 93–106, 2017.
DOI: 10.1007/978-3-319-62701-4_8

CSPs are increasingly focusing on effective churn management to plan their retention strategies. Data mining and machine learning techniques are increasingly applied for churn prediction. There have been many studies for developing churn prediction models [3, 4],[6, 7, 8]. Existing works generally focus on comparing effectiveness of various machine learning algorithms in predicting churn. However, these studies have the limitation that their test sets were small. In this paper, we present a novel approach towards large scale churn modelling and report results on a real world dataset pertaining to approximately 5 million prepaid subscribers from a popular Asian CSP. We cover practical issues in churn modelling such as high dimensionality, class imbalance, productionization of the churn model and actionability in terms of campaign design based on churn scores.

Lack of contractual obligations between subscribers and CSPs in prepaid mobile markets make predictive churn modelling a difficult problem [9]. Several subscribers sporadically stop consuming the network services and switch to other CSPs, without prior notification. Many CSPs consider a continuous and prolonged period of inactivity in terms of incoming and outgoing service usages to be indicative of churn. In general, inactivity periods of 2 months, 3 months, 6 months or 1 year are used depending upon the CSP, geography and the government regulations, to mark the expiration of a Subscriber Identification Module (SIM). After SIM expiration, all network services to the corresponding subscriber would be terminated. In our opinion, using such long inactivity periods in predictive churn modelling is detrimental, as the CSPs loose actionability in terms of providing counter offers and promotions to proactively retain their customers and to encourage service consumption before they switch to other CSPs. Commonly used definitions of prepaid churn in the literature include two months [10, 11] or three months [8, 10] of continuous inactivity. [12] uses an inactivity period of 6 weeks to model prepaid churn. In this paper, we consider subscribers with no outgoing activity in terms of voice usage, data usage and short messaging service (SMS) usage for a period of 15 consecutive days as churners. Though our definition may not be indicative of network churn, as subscribers may resume service consumption after 15 days of inactivity, we believe that it allows CSPs to engage with subscribers effectively by recommending personalized offers and thereby influencing them to restart their service consumption.

Application of data mining techniques on telecom churn prediction is not new. Most of the studies model churn prediction as a binary classification task. Decision Trees [13], Neural Networks (NN) [14] and Support Vector Machines (SVM) [15] have been extensively used as classifiers for building churn models. Many of the existing studies compare multiple classifiers in terms of their predictive quality [16]. Decision Tree and NN classifiers are compared in terms of their prediction quality on a dataset of 6000 instances and information gain is used for feature selection in [17]. In [4], a binomial logit model was used on a sample of 973 mobile users. [6] compares multiple approaches based on decision trees and Back Propagation Network (BPN) model to predict churn for approximately 160,000 subscribers of a wireless telecom company. The study also provided results for varying sizes of training sets created using different sampling techniques. Data Mining by Evolutionary Learning (DMEL) to mine classification rules in large databases is presented in [18], with specific focus on churn modelling. DMEL searches through the huge rule spaces using an evolutionary approach. Its performance was observed to be better compared to NN and decision trees when applied to a database of 1 million subscribers. Genetic Programming (GP) and AdaBoost based approaches have been attempted in [19] to predict telecom churn. This work explores the evolution process in GP by integrating an AdaBoost style boosting to evolve multiple programs per class. Weighted sum of outputs from all GP programs is used to generate the final prediction. This approach provided better results as compared to KNN and Random Forest in terms of sensitivity, specificity and AUC on Orange Telecom (80,000 subscribers) and Cell2Cell (40,000 subscribers) datasets. However, these studies have limitations since the results are reported on small datasets, in a controlled experimental setting. In this work, we primarily focus on practical aspects involved in large-scale churn modelling such as feature selection, model evaluation, operationalization and providing actionabilty to marketers.

The outline of the paper is as follows. The modelling strategy is discussed in Section 2 and results in Section 3. Actionability and operationalization aspects of the developed system are presented in Sections 4 and 5 respectively. Section 6 concludes the paper.

2 Model Building

The data for the study is from a well known Asian CSP with a subscriber base of approximately 7 million. Uncompressed data, in the form of Call Data Record (CDR) files, with datasize of approximately 80 GB is streamed on a daily basis to our data warehouse. We randomly sampled 0.5 million subscribers for developing the prediction model. Service usage pertaining to voice (incoming and outgoing), data and SMS along with their recharge information for a period of 90 days (3 months), with datasize of approximately 7200 GB, is used for the analysis. Subsequent subsections elaborate our modelling procedure.

2.1 Data Preparation and Preprocessing

Identifying the key KPIs is an important aspect of any modelling task. With the help of domain expertise and preliminary data analysis, we selected 36 candidate KPIs for building our model. Details of these KPIs are provided in Table 1. KPIs were extracted from the raw CDR data using custom PIG scripts [20] on a Hadoop cluster. Dormant users who were inactive for a period of 30 consecutive days were filtered out as some of them may have already churned out of the network. This reduced the sample size from 0.5 million to 0.34 million. Most of the KPIs were derived KPIs, which were obtained by aggregating subscribers' usage behavior over a period of one month or three months, as appropriate. KPIs were further categorized into snapshot or trending KPI. Snapshot KPIs contain a single value whereas trending KPIs comprise of more than one KPI, representing an usage trend. For example, *Days_Since_Last_Usage* is a snapshot KPI whereas *OG_MOU* is a trending KPI that represents total outgoing minutes of usage, for 3 consecutive months separately (represented by $M_1/M_2/M_3$ in Table 1) representing an usage trend. In other words, *OG_MOU*, which is a trending KPI, actually comprises of 3 snapshot KPIs namely *OG_MOU.*M_1, *OG_MOU.*M_2 and *OG_MOU.*M_3, each of them corresponds to the usage for one month respectively. Description of all the KPIs is provided below.

1. **Days_Since_Last_Usage:** Number of days elapsed since the last service usage (data/voice/SMS) has been made by the subscriber
2. **Days_Since_Last_Recharge:** Number of days elapsed since the time the subscriber has made the last recharge
3. **Average_Recharge_Count:** Average number of recharges done per month by a subscriber within a time period
4. **Last_Recharge_MRP:** Monetary value of the last recharge done
5. **Average_Recharge_MRP:** Average monetary value of recharge done per month within a time period
6. **AON:** This is known as Age On Network, which represents the total number of days since the subscriber became active on the network
7. **Voice_OG_Call_Days:** Total number of days within a time period where at least one outgoing call has been made per day by the subscriber
8. **Voice_IC_Call_Days:** Total number of days within a time period where at least one incoming call has been made per day to the subscriber
9. **OG_MOU:** Total outgoing usage in minutes
10. **IN_DEC:** This is known as Intelligent Network Decrement, representing the total amount deducted from balance within a time period. Intelligent Network refers to the prepaid billing system.
11. **On_Net_Share:** Service usage in home network as a proportion of total service usage
12. **STD_Contribution:** STD call usages as a proportion of Total Outgoing call usages
13. **Max_Days_Between_Recharge:** Maximum number days between any two successive recharges done by the subscriber over a time period
14. **Median_Delay_Between_Recharge:** Median number days between any two successive recharges done by the subscriber over a time period
15. **Max_Consistent_Non_Usage_Days:** Maximum number of days at a stretch when no usage has been made by the subscriber over a time period

Churn rate, based on the proposed inactivity-based definition of churn was found to be 19.12%. The extracted dataset was split into a training set and a test set. The training set was balanced by under sampling of the majority class (non churners). In the test set, ratio of churners to non churners was kept same as in the original dataset. Figure 1 depicts the class distribution in the training and test sets.

2.2 Feature Selection

All 36 KPIs identified initially may not be equally important in characterizing the churn phenomena. Hence, feature selection was performed to identify most

Table 1: List of 36 KPIs, Their Types and Duration

KPI	Snapshot (S) or Trending (T)	Raw (R) or Derived (D)	Period To consider (In Months)
Days_Since_Last_Usage	S	D	1
Days_Since_Last_Recharge	S	D	3
Average_Recharge_Count	S	D	3
Last_Recharge_MRP	S	R	3
Average_Recharge_MRP	S	D	3
AON	S	D	NA
Voice_OG_Call_Days.$M_1/M_2/M_3$	T	D	1 x 3
Voice_IC_Call_Days.$M_1/M_2/M_3$	T	D	1 x 3
OG_MOU.$M_1/M_2/M_3$	T	D	1 x 3
IN_DEC.$M_1/M_2/M_3$	T	D	1 x 3
On_Net_Share.$M_1/M_2/M_3$	T	D	1 x 3
STD_Contribution.$M_1/M_2/M_3$	T	D	1 x 3
Max_Days_Between_Recharge.$M_1/M_2/M_3$	T	D	1 x 3
Max_Days_Between_Recharge.3_Months	S	D	3
Median_Delay_Between_Recharge.$M_1/M_2/M_3$	T	D	1 x 3
Median_Delay_Between_Recharge.3_Months	S	D	3
Max_Consistent_Non_Usage_Days.$M_1/M_2/M_3$	T	D	1 x 3
Max_Consistent_Non_Usage_Days.3_Months	S	D	3

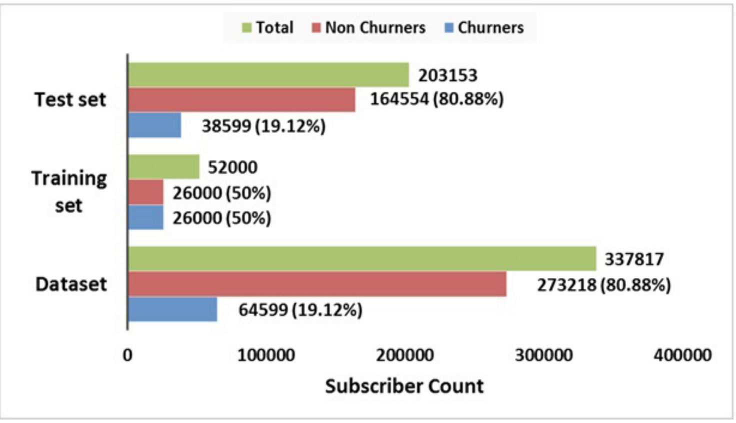

Fig. 1: Data Distribution

prominent KPIs from the initial set. We follow a 2-step strategy in selecting the prominent KPIs.

1. **Step-1 :** Information Gain [21] was used to rank the KPIs based on their information gain scores. Based on empirical experiements, a threshold of 0.12 was chosen and accordingly 19 KPIs were selected. Figure 2 shows the information gain scores of these 19 KPIs.

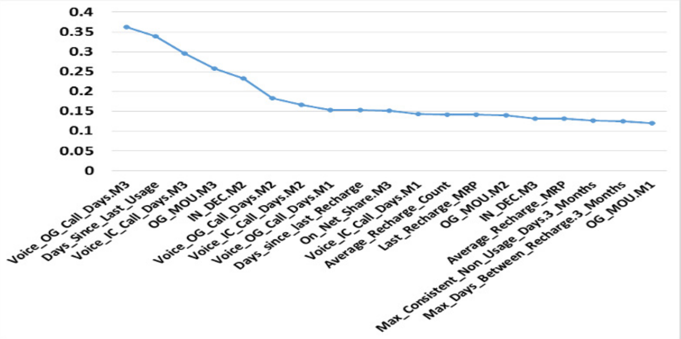

Fig. 2: Information Gain Scores of Top 19 KPIs

2. **Step-2 :** We use a combination of approaches such as wrapper method based on forward selection [22], recency-based filtering and computation-intensive filtering for further reduction of the feature space. Each of these approaches cater to different aspects of the problem as explained below.

   Some KPIs when considered together, convey more meaningful behavioral patterns as compared to the case when they are considered separately. For instance, subscriber S_A, having a smaller value of *Last_Recharge_MRP* and a relatively larger value of *Days_Since_Last_Recharge* is more likely to have stopped its usage as compared to subscriber S_B, who has larger values for both these KPIs, in general. Wrapper based feature selection approach was used for this analysis. In contrary to the information gain (which scores the KPIs separately with respect to the class labels), the wrapper method finds an optimal subset of KPIs which have high discriminating power when considered together. 16 KPIs (listed in Table 2) were obtained after applying a wrapper method based on forward selection on the KPI set obtained from step 1.

   Recency-based filtering is used to consider those KPIs which have the most recent values. This is applicable only to the trending KPIs. For instance, out of the KPIs namely $OG\_MOU.M_1$, $OG\_MOU.M_2$ and $OG\_MOU.M_3$, only $OG\_MOU.M_3$ will be chosen as it represents the most recent outgoing minutes of usage of a subscriber. This is based on the hypothesis that, most recent behavior better explains the churn phenomena as compared to the least recent behavior. In addition to this, we observed significant correlation among the trending KPIs. For instance, $corr(OG\_MOU.M_1,\ OG\_MOU.M_2)$ ≈ 0.78 and $corr(OG\_MOU.M_2,\ OG\_MOU.M_3) \approx 0.65$, where $corr$ represents

Table 2: 16 KPIs using Wrapper method

Days_Since_Last_Recharge	*Voice_IC_Call_Days.*M_2/M_3
Last_Recharge_MRP	*IN_DEC.*M_2/M_3
Average_Recharge_MRP	*Days_Since_Last_Usage*
Average_Recharge_Count	*Max_Consistent_Non_Usage_Days.3_Months*
Voice_OG_Call_Days.$M_1/M_2/M_3$	*Max_Days_Between_Recharge.3_Months*
OG_MOU.M_3	*On_Net_Share.*M_3

the correlation between two KPIs. This motivated us to retain only the recent month's value of the trending KPIs instead of values corresponding to all 3 months.

Apart from this, few KPIs such as *Max_Consistent_Non_Usage_Days.3_Months*, *Max_Days_Between_Recharge.3_Months* pose significant overhead in terms of their computation time. We call these KPIs computationally intensive KPIs and can be ignored if doing so does not the affect the model accuracy significantly. Various combination of above techniques were tried and a Random Forest (RF) classifier (trained on the training set, using an ensemble of 100 Decision Trees) was used to make predictions on the test set, in order to understand the impact of the features on the model quality.

Table 3: Impact of Feature Selection Techniques on Model Quality using RF(Here IG: 'Information Gain')

Feature Selection Combination	Number of KPIs	Precision	Recall	F-measure
All KPIs (No Feature Selection)	36	0.552	0.882	0.679
IG	19	0.551	0.881	0.678
IG + Wrapper	16	0.552	0.88	0.678
IG + Recency Based Filtering	12	0.55	0.879	0.676
IG + Computation Intensive Filtering	17	0.55	0.88	0.677
IG + Wrapper + Computation Intensive Filtering	14	0.551	0.88	0.678
IG + Wrapper + Recency Based Filtering + Computation Intensive Filtering	10	0.547	0.877	0.674

As observed from Table 3, precision, recall and F-measure values are almost similar in all cases. Hence, it can be inferred that the trending KPIs are not contributing much to the discrimination between churners and non churners. So, for a real world deployment, we selected 10 most discriminating KPIs (as listed down in Table 4), without a significant loss in accuracy.

2.3 Machine Learning Classifiers

Several linear and non linear classifiers namely Logistic Regression (LR) [23], Random Forest (RF) [24], Support Vector Machine (SVM) [15] along with ensemble based Random Supspace (RS) [25] were used for this analysis. Many

Table 4: Final 10 KPIs

Days_Since_Last_Recharge	*Days_Since_Last_Usage*
Last_Recharge_MRP	*On_Net_Share.*M_3
Average_Recharge_MRP	*Voice_IC_Call_Days.*M_3
Average_Recharge_Count	*OG_MOU.*M_3
Voice_OG_Call_Days.M_3	*IN_DEC.*M_3

existing studies related to churn prediction have made use of standard classifiers, but prior works have not utilized Random Subspace (RS) classifier in churn prediction to the best of our knowledge. Here, we compare the performances of these alternate classifiers. RS is based on the following principle.

1. Given a training set $X = (X_1, X_2, ..., X_n)$, where $X_i = (x_{i1}, x_{i2}, ..., x_{id})$ is a d-dimensional feature vector, $p < d$ features are randomly selected. Modified bootstrapped training sets $\widetilde{X}^b = (\widetilde{X}_1^b, \widetilde{X}_2^b, ..., \widetilde{X}_n^b)$ are created, each containing p-dimensional training objects $\widetilde{X}_i^b = (x_{i1}^b, x_{i2}^b, ..., x_{ip}^b)(i = 1, ..., n)$.
2. Classifiers $C^b(x)$ are constructed in the random supspaces $\widetilde{X}^b$ and the prediction from each of the classifiers are combined by taking a simple majority voting (the most often predicted label)

$$\beta(x) = argmax \sum_{b} \delta_{sgn(C^b(x)),y} \tag{1}$$

where

$$\delta_{i,j} = \begin{cases} 1, \; if\ i = j \\ 0, \; if\, i \neq j \end{cases} \tag{2}$$

is the Kronecker symbol and $y \in \{-1, 1\}$ is a decision (class label) of the classifier.

2.4 Decile Wise Cumulative Coverage as a Model Evaluation Metric

We define a metric called Decile Wise Cumulative Coverage (DWCC) to assess the overall model quality along with other standard accuracy metrics namely precision, recall and F-measure. Here we describe the procedure to form deciles from the whole subscriber base. We first estimate the probability of churn for each subscriber, using a classfication algorithm as mentioned above. Then subscribers are ranked according to the decreasing order of their churn probabilities. The whole subscriber base is then divided into 10 equal bins, where each bin contains almost 10% of the total observations. Each of these 10 bins is called a decile. Decile wise cumulative coverage metric can be represented as follows:

$$DWCC_N = \sum_{d=1}^{N} \frac{TCC_d \times 100}{TCC}, 1 \leq N \leq 10 \tag{3}$$

where $DWCC_N$ denotes the decile wise cumulative coverage of churners (in percentage) upto N^{th} decile, TCC_d denotes the true churner count in d^{th} decile and TCC denotes the total churner count in the whole subscriber base (all 10 deciles combined). $DWCC_N$ can also be interpreted as the recall of churners in the top N deciles. We choose this evaluation metric as it describes the model efficacy in terms of proportion of actual churners covered if subscribers falling into top K deciles are targeted as part of retention campaigns. More details on subscriber targeting mechanism for retention campaigns is mentioned in Section 4.

Table 5: Comparision among classifiers using DWCC

Classifier	$DWCC_1$	$DWCC_2$	$DWCC_3$	$DWCC_4$	$DWCC_5$	$DWCC_6$	$DWCC_7$	$DWCC_8$	$DWCC_9$	$DWCC_{10}$
LR	38.26	68.13	83.34	91.4	95.63	97.82	99.04	99.61	99.89	100
SVM	32.7	68.13	86.11	93.74	95.86	96.33	96.82	97.66	98.79	100
RF	40.21	69.62	86.59	93.97	97.06	98.34	99.21	99.68	99.91	100
RS	40.31	70.05	86.53	93.86	97.07	98.53	99.33	99.8	100	100

Table 6: Comparison among classifiers using Precision, Recall and F-Measure

Classifier	Precision	Recall	F-Measure
LR	0.521	0.84	0.643
SVM	0.529	0.876	0.66
RF	0.548	0.877	0.674
RS	0.541	0.877	0.669

A grid search technique [26] was used to obtain the best model with respect to each of the classifiers. In this case, the balanced training set was used to train each of the candidate models and predictions were made on the test set. Best results obtained with respect to each classifier using the grid search technique are reported in Table 5 and 6. It is observed that the non-linear classifiers (SVM, RF, RS) performed slightly better than the linear counterpart (LR) in terms of $DWCC$ in top 3 deciles (83.34% for LR, 86.11% for SVM, 86.59% for RF and 86.53% for RS), precision, recall and F-Measure. However, SVM, RF, RS gave comparable performances except that, SVM covered only 32.7% of the actual churners in the 1st decile (which is the lowest among all classifiers used in this analysis). Among the non-linear classifiers, Random Forest (RF) and Random Subspace (RS) performed almost identically. Between RF and RS, RF was chosen, as its distributed implementation is readily available, which would be beneficial in case of a real world production deployment, where model needs to be trained on large-scale datasets, incurring minimal training time.

In our analysis, we have also considered the trending KPIs in terms of their relative values. A relative value indicates the incremental change (positive or negative) in percentage, observed in a KPI value in a month, with respect to the previous month. To this end, KPIs such as $OG\_MOU.M_1$, $OG\_MOU.M_2$, $OG\_MOU.M_3$ were replaced with $OG\_MOU.M_1$, ΔOG_MOU_1 and ΔOG_MOU_2 respectively, where Δ denotes the percentage change in outgoing minutes of usage in a month with respect to the previous month. We repeated similar experiments using the relative representation of the trending KPIs. However, it did not improve the results. Including KPIs such as number of dropped calls and number of calls made to customer care may further improve the model performance as they are known to be good indicators of churn. However, due to data unavailability, we could not use them in our analysis.

3 Large Scale Training and Prediction

It would be inappropriate to assume that the subscribers' behavior remain consistent over time. For example, let us assume that the frequency of recharge of a large number of subscribers decline suddenly because of the increased use of free messaging applications like WhatsApp. In this scenario, the same training set which was used to train the model may no longer be suitable for learning the newly emerging behavioral patterns and hence the generalization capability of the model will degrade. Hence, to capture latest trends in subscribers' usage, we use the most recent window to construct the training set. Once our model is used for prediction, we wait for next 15 days to collect the true labels of the predicted subscribers. A new training set is constructed by utilizing these true labels along with the required KPIs. The model is retrained using this newly created training set to make predictions for the next cycle.

Table 7: Decile wise Coverage of Churners and Non Churners

Decile	Total Subscriber count	Actual Churners Count	Actual Non Churners Count	Cumulative Population in %	% of Churners Covered per Decile	% of Non Churners Covered per Decile	Cumulative % of Churners Covered	Cumulative % of Non Churners Covered
1	456906	348356	108550	10	36.76	3.00	36.76	3.00
2	456906	267441	189465	20	28.22	5.23	64.98	8.23
3	456906	176573	280333	30	18.63	7.74	83.61	15.97
4	456906	85312	371594	40	9.00	10.26	92.61	26.23
5	456906	36382	420524	50	3.84	11.61	96.45	37.84
6	456906	16474	440432	60	1.74	12.16	98.19	50.01
7	456906	8458	448448	70	0.89	12.38	99.08	62.39
8	456906	4691	452215	80	0.50	12.49	99.58	74.88
9	456906	2681	454225	90	0.28	12.54	99.86	87.42
10	456907	1295	455612	100	0.14	12.58	100.00	100.00
TOTAL	4569061	947663	3621398		100.00%	100.00%		

Table 8: Confusion Matrix

	Predicted →	
Actual ↓	C	NC
C	818884	128779
NC	657184	2964214

Table 9: Precision, Recall and F-Measure

Precision	Recall	F-Measure
0.5548	0.8641	0.6757

Table 7,8 and 9 represent the real world prediction results when the model was applied to a full circle subscriber base [27] containing approximately 5 million active subscribers from a popular Asian operator. A balanced training set, consisting of approximately 0.9 million churners and an equal number of non churners was used for model training.

4 Actionability with Lift Chart

Campaign management is a crucial part of retention programs. To have effective campaigns, CSPs need actionability. We use lift chart [28] to provide actionability to marketers. These charts aid marketers in deciding the set of subscribers to be targeted for retention. Lift is a measure of effectiveness of a predictive model, calculated as the ratio between the results obtained with and without the predictive model which can be represented as follows:

$$Lift_{D(PM)} = \frac{DWCC_{D(PM)}}{DWCC_{D(RM)}} \tag{4}$$

where $Lift_{D(PM)}$ denotes lift obtained in top D deciles using our proposed model, $DWCC_{D(PM)}$ and $DWCC_{D(RM)}$ denotes the decile wise cumulative coverage of churners obtained in top D deciles using our proposed model and a random model respectively. In this context, a random model is equivalent to randomly guessing the likely churners. As evident from the decile formation described in the previous section, decile 1 contains subscribers who are most likely to churn and decile 10 contains subscribers who are least likely to churn, making it convenient for marketers to consider top K deciles as the target base for retention campaigns.

Choosing an optimal value of K depends mostly upon a marketer's subjectivity. One ideal way is to choose a value of K, beyond which the model's lift does not fall rapidly. Considering this case, optimal value of K that would have been chosen is 4, as seen from the lift chart in Figure 3. However, marketer can also set a threshold value of model's lift Th_L and select a value of K for which $Lift_{D(PM)} > Th_L$. For instance, if marketer chooses $Th_L = 2.5$, then value of K chosen would be 3. Moreover, marketer can also consider other factors such as cumulative decile wise coverage intended, cost of targeting a subscriber base and historically observed churn rate in choosing a target base for retention campaigns.

Table 10: Proposed Model's Lift Calculation

Decile	$DWCC_{PM}$	$DWCC_{RM}$	$Lift_{PM}$
1	36.74	10	3.674
2	64.98	20	3.249
3	83.61	30	2.787
4	92.61	40	2.315
5	96.45	50	1.929
6	98.19	60	1.636
7	99.08	70	1.415
8	99.58	80	1.244
9	99.86	90	1.109
10	100	100	1

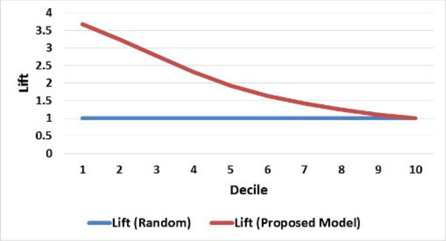

Fig. 3: Lift Chart

5 Operationalization

For the scaled implementation, the preprocessing and data preparation steps were carried out in a MapReduce framework [29] with the help of Apache spark [30] scripts. Apache Spark is a fast and general-purpose cluster computing system. In the production environment, the model was deployed as an Apache Oozie workflow [31] and was scheduled to run in every 15 days. We used MLlib library [32] for the training and prediction tasks. MLlib is Apache Spark's scalable machine learning library which contains several machine learning algorithms and utilities, including classification, regression, clustering, collaborative filtering, dimensionality reduction, as well as underlying optimization primitives. Our model utilities the distributed implementation of Random Forest, that can train multiple decision trees in parallel thereby reducing the training time significantly, especially when the size of the training set is huge. Figure 4 shows the sequence of actions carried out for productionizing the churn model.

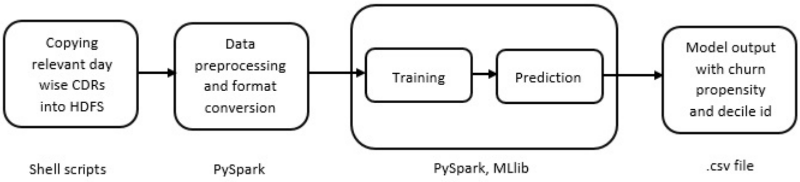
Copying relevant day wise CDRs into HDFS
Data preprocessing and format conversion
Training
Prediction
Model output with churn propensity and decile id
Shell scripts
PySpark
PySpark, MLlib
.csv file

Fig. 4: DAG of Actions in Oozie Workflow

6 Conclusion

In this paper, we have proposed a practical approach for developing a usage based churn management framework for prepaid mobile markets. We have discussed about the KPI selection methodology using a combination of approaches. We also reported results from evaluating our model on a real-world dataset consisting of approximately 5 million active subscribers from a popular asian CSP. We further discussed about the actionability that the model provides in the form of lift charts. Finally, we described the operationalization part of the model in a real world scenario.

As a part of the future work, it would be interesting to apply PCA [33] for further dimensionality reduction. Incremental learning could also be incorporated, where model incrementally learns from the new set of instances, every time the model is trained, which will keep the model up-to-date and robust in case of any change in subscribers' usage patterns.

References

[1] Kang, C., Pei-ji, S.: Customer churn prediction based on svm-rfe. In: Business and Information Management, 2008. ISBIM'08. International Seminar on. Volume 1., IEEE (2008) 306–309

[2] Hadden, J., Tiwari, A., Roy, R., Ruta, D.: Computer assisted customer churn management: State-of-the-art and future trends. Computers & Operations Research **34** (2007) 2902–2917

[3] Wei, C.P., Chiu, I.T.: Turning telecommunications call details to churn prediction: a data mining approach. Expert systems with applications **23** (2002) 103–112

[4] Kim, H.S., Yoon, C.H.: Determinants of subscriber churn and customer loyalty in the korean mobile telephony market. Telecommunications Policy **28** (2004) 751–765

[5] Commission, F.C., et al.: Annual report and analysis of competitive market conditions with respect to mobile wireless, including commercial mobile services. Washington, DC (2010)

[6] Hung, S.Y., Yen, D.C., Wang, H.Y.: Applying data mining to telecom churn management. Expert Systems with Applications **31** (2006) 515–524

[7] Huang, B., Kechadi, M.T., Buckley, B.: Customer churn prediction in telecommunications. Expert Systems with Applications **39** (2012) 1414–1425

[8] Han, Q., Ferreira, P.: Determinants of subscriber churn in wireless networks: Role of peer influence. (2015)

[9] Kraljević, G., Gotovac, S.: Modeling data mining applications for prediction of prepaid churn in telecommunication services. AUTOMATIKA: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije **51** (2010) 275–283

[10] Radosavljevik, D., Van Der Putten, P., Larsen, K.K.: The impact of experimental setup on prepaid churn modeling: Data, population and outcome definition. In: Industrial Conference on Data Mining-Workshops, Citeseer (2010) 14–27

[11] Kusuma, P.D., Radosavljevik, D., Takes, F.W., van der Putten, P.: Combining customer attribute and social network mining for prepaid mobile churn prediction. In: Proceedings of the 23rd Annual Belgian Dutch Conference on Machine Learning (Benelearn). (2013) 50–58

[12] Owczarczuk, M.: Churn models for prepaid customers in the cellular telecommunication industry using large data marts. Expert Systems with Applications **37** (2010) 4710–4712

[13] Quinlan, J.R.: Simplifying decision trees. International journal of man-machine studies **27** (1987) 221–234

[14] Bishop, C.M., et al.: Neural networks for pattern recognition. (1995)

[15] Cortes, C., Vapnik, V.: Support-vector networks. Machine learning **20** (1995) 273–297

[16] Powers, D.M.: Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. (2011)

[17] Umayaparvathi, V., Iyakutti, K.: Applications of data mining techniques in telecom churn prediction. International Journal of Computer Applications **42** (2012) 5–9

[18] Au, W.H., Chan, K.C., Yao, X.: A novel evolutionary data mining algorithm with applications to churn prediction. Evolutionary Computation, IEEE Transactions on **7** (2003) 532–545

[19] Idris, A., Khan, A., Lee, Y.S.: Genetic programming and adaboosting based churn prediction for telecom. In: Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on, IEEE (2012) 1328–1332

[20] Foundation, T.A.S.: Apache pig: https://pig.apache.org/ (2015)

[21] Kent, J.T.: Information gain and a general measure of correlation. Biometrika **70** (1983) 163–173

[22] Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artificial intelligence **97** (1997) 273–324

[23] McCullagh, P., Nelder, J.A.: Generalized linear models. Volume 37. CRC press (1989)

[24] Breiman, L.: Random forests. Machine learning **45** (2001) 5–32

[25] Skurichina, M., Duin, R.P.: Bagging, boosting and the random subspace method for linear classifiers. Pattern Analysis & Applications **5** (2002) 121–135

[26] Wikipedia: Hyperparameter optimization — wikipedia, the free encyclopedia (2017) [Online; accessed 7-April-2017].

[27] Wikipedia: Mobile telephone numbering in india — wikipedia, the free encyclopedia (2017) [Online; accessed 7-April-2017].

[28] Uregina: Cumulative gains and lift charts (2015) [Online; accessed 7-April-2017].

[29] Chu, C., Kim, S.K., Lin, Y.A., Yu, Y., Bradski, G., Ng, A.Y., Olukotun, K.: Map-reduce for machine learning on multicore. Advances in neural information processing systems **19** (2007) 281

[30] Spark, A.: Apache spark-lightning-fast cluster computing (2014)

[31] Islam, M., Huang, A.K., Battisha, M., Chiang, M., Srinivasan, S., Peters, C., Neumann, A., Abdelnur, A.: Oozie: towards a scalable workflow management system for hadoop. In: Proceedings of the 1st ACM SIGMOD Workshop on Scalable Workflow Execution Engines and Technologies, ACM (2012) 4

[32] Meng, X., Bradley, J., Yuvaz, B., Sparks, E., Venkataraman, S., Liu, D., Freeman, J., Tsai, D., Amde, M., Owen, S., et al.: Mllib: Machine learning in apache spark. JMLR **17** (2016) 1–7

[33] Abdi, H., Williams, L.J.: Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics **2** (2010) 433–459