

# Predicting customer churn in mobile networks through analysis of social groups

Yossi Richter \*

Elad Yom-Tov †

Noam Slonim ‡

## Abstract

Churn prediction aims to identify subscribers who are about to transfer their business to a competitor. Since the cost associated with customer acquisition is much greater than the cost of customer retention, churn prediction has emerged as a crucial Business Intelligence (BI) application for modern telecommunication operators.

The dominant approach to churn prediction is to model individual customers and derive their likelihood of churn using a predictive model. Recent work has shown that analyzing customers' interactions by assessing the social vicinity of recent churners can improve the accuracy of churn prediction.

We propose a novel framework, termed Group-First Churn Prediction, which eliminates the a priori requirement of knowing who recently churned. Specifically, our approach exploits the structure of customer interactions to predict which groups of subscribers are most prone to churn, before even a single member in the group has churned. Our method works by identifying closely-knit groups of subscribers using second order social metrics derived from information theoretic principles. The interactions within each group are then analyzed to identify social leaders. Based on Key Performance Indicators that are derived from these groups, a novel statistical model is used to predict the churn of the groups and their members.

Our experimental results, which are based on data from a telecommunication operator with approximately 16 million subscribers, demonstrate the unique advantages of the proposed method. We further provide empirical evidence that our method captures social phenomena in a highly significant manner.

## 1 Introduction

**1.1 Overview** Over the last two decades, we have seen mobile telecommunication become the dominant communication medium. In many countries, especially developed ones, the market has reached a degree of saturation where each new customer must be won over from the competitors. At the same time, public regulations and the standardization of mobile communication now allow customers to easily move from one carrier to another, resulting in a very fluid market. Since the cost of winning a new customer is far greater than the cost of preserving an existing one [11], mobile carriers have been shifting considerable attention from customer acquisition to customer retention. As a result, *churn prediction* has emerged as a crucial mobile Business Intelligence (BI) application that aims at identifying customers who are about to transfer their business to a competitor (i.e., to *churn*) [8].

A good churn prediction system should not only pinpoint potential churners successfully, but further provide a sufficiently long horizon forecast in its predictions. Once a potential churner is identified, the retention department usually makes contact and, if the customer is established to be a churn risk, takes appropriate measures to preserve her business. Thus, a long forecast horizon is an obvious advantage since the further away the customer is from actually making the churn decision, the easier it is to prevent that decision at a significantly lower cost.

Naturally, retention efforts are allocated limited resources and thus only a tiny fraction of the subscriber pool can be contacted at any given time. With this constraint in mind, churn prediction systems are usually measured by their ability to identify actual churners among the top 0.1% to 5% of the customers predicted to have the highest risk of churning.

**1.2 Existing solutions** The mainstream approach to churn prediction (for example, [1, 4, 9, 14, 16]) considers each customer individually. The goal is to predict each customer's likelihood of churning in the near future, where usually a forecast horizon of a month to three months is considered. To this end, dozens

\*IBM Haifa Research Lab, 165 Aba Hushi st., Haifa 31905, Israel. richter@il.ibm.com

†IBM Haifa Research Lab, 165 Aba Hushi st., Haifa 31905, Israel. yomtov@il.ibm.com

‡IBM Haifa Research Lab, 165 Aba Hushi st., Haifa 31905, Israel. noams@il.ibm.com

to hundreds of complex Key Performance Indicators (KPIs) are generated per customer; these KPIs span the customer's personal characteristics as well as trends in their call activities over a long period. The information then serves as input to a statistical regression model (usually a logistic regression variant) that outputs a churn score. In other words, this approach focuses on identifying patterns that are uncommon to a given customer, and are correlated with churn.

Other systems [10] employ a different, ad-hoc, approach to the problem. By monitoring customers' calls to the mobile carrier's call center, such systems apply speech and emotion analysis to the calls, and together with additional information (number and length of calls by the customer, number of transfers, hold period, etc.) try to quantify the customer's dissatisfaction level and hence the associated churn risk. The system can then react by prioritizing pending 'churners', even suggesting retention packages. This approach has a major shortcoming: although it may accurately pinpoint the potential churners, the forecast horizon it provides is very short as the system identifies customers that have already expressed dissatisfaction with the service. At this stage, retention prospects are lower while cost is significantly higher.

Even when combining the long term and ad-hoc churn prediction systems, one drawback is fairly obvious: we clearly rely on the assumption that a churning customer either changes calling patterns or contacts the carrier's call center to express dissatisfaction prior to switching carriers. While this may be true in some cases, there are certainly many scenarios in which these assumptions are violated. For example, this may occur when customers come to believe that they have found a better deal with a competitor and churn immediately.

Another, less obvious, drawback of traditional solutions is that they focus exclusively on the individual customer without taking into account any social influence. Clearly, there are many social aspects to churn, as witnessed in other consumer areas [6, 18], where a dominant example is when a churning customer influences other customers to churn as well. In fact, one can imagine scenarios where a churn decision is attributed solely to social influence. Thus, developing churn prediction systems that take social aspects into account poses an emerging theoretical challenge with potentially great practical implications.

While commercial solutions have started exploring this direction <sup>1</sup>, to the best of our knowledge the only published work directly in this context is the work of Nanavati et al. [12, 3]. In that work, the underlying

assumption is that recent churners are known and they are likely to affect the churning decisions of their social neighborhood. The network of subscribers is then modeled as a weighted directed graph where nodes represent the customers, and weights on the edges correspond to the strength of the social connections between them. Next, a diffusion process is used to model the flow of information from recent churners to their social environment. Specifically, each node in the network that corresponds to a recent churner is assigned an initial weight. A decaying diffusion process propagates this weight across the global network until convergence. At that point, each subscriber in the network has some associated weight corresponding to the amount of "churning" information that has reached him. The individual churn scores are then derived directly from these weights [3].

### 1.3 The group-first social networks approach

We take a somewhat more refined approach to the social aspects of churn. Our initial working hypothesis, which we later established experimentally, was that social influence on churn is highly dominant in relatively tight social groups. We had several reasons to believe this hypothesis. First, information flows rapidly in tight social groups and is considered very reliable. Therefore, positive information about an appealing deal with another carrier, or alternatively negative feedback concerning dissatisfaction with the service provided by the current carrier, are likely to circulate quickly and heavily influence churn decisions. Second, members of dense social groups tend to make many calls within their group, thus they have a strong incentive to remain together with the same carrier to save inter-network fees. Finally, small social groups often have dominant social leaders that considerably affect the overall group's consumer preferences, and specifically influence the group's decision to churn. Therefore, a challenging goal is to pinpoint social groups that are at a high churn risk, in spite of the fact that none of their members have already churned. This is precisely the goal we tackle in this study.

Our social networks approach to churn prediction can be sketched as follows. First, we employ a novel, information-theoretic based measure to quantify the connectivity between each pair of subscribers in the network. Unlike previous approaches, our measure takes into account second-order social factors. We keep only the strongest connections, and partition the network into a collection of small disjoint clusters, each representing a dense social group. We further analyze the social interactions within each cluster and establish the relative social status of each of its members. We

<sup>1</sup><http://www.datanetis.com>

then query a statistical model to establish a churn risk score for each group. This model was developed by designing novel *group* KPIs and using machine learning techniques to fit a model that correlates these group KPIs with group churn. Finally, we assign an individual churn score to each subscriber based on the churn score of her social group as well as her personal characteristics.

Although stemming from similar assumptions, our approach is distinguished from the mainstream in two important aspects. The first is group-based analysis *vs.* global analysis of the social structure. Although a global approach has the advantage of capturing inter-group influences, a group-first approach facilitates deeper analysis of each group, and allows advanced machine learning techniques to be applied at the group-level, as we demonstrate in our results.

The second aspect in which we deviate from the established social networks approach is that we aim to predict the beginning of a churn process rather than predicting behavior in the next period based on the last one. Although the latter approach can reliably identify potential churners with higher risk (which is good), one can fairly assume that these churners will often be the hardest to preserve, as members of their social reference group have already churned. In our approach we tackle a more difficult problem (no recent churning in the local vicinity) but with a larger expected payback, since the early identification of potential churners makes them easier to retain.

**1.4 Our results** To test the effectiveness of our approach we conducted two successful pilots over real world data. In the first pilot, we integrated our solution with the churn scores that were provided by the carrier's existing system. Although we are unauthorized at this point to describe the details of this experiment, it is worth mentioning that the results obtained were far better than the original results of the carrier's existing system, suggesting that our social group-first viewpoint is exploiting data aspects that are not utilized by conventional methods, and hence can be easily integrated into an existing system with immediate benefits. We further conducted a second pilot over a challenging dataset of approximately 16 million users at another mobile carrier. In Section 4 we provide a comprehensive review of the results obtained. Our data included only the calls data over a short period of time, and we were not given access to any additional information frequently used in making churn predictions, such as demographic details, individual mobile tariffs, commitment periods, etc. Nevertheless, our results were highly encouraging compared to state-of-the-art results reported

in the literature. Specifically, our results demonstrate that groups of users that are at a high churn risk can be efficiently detected using our approach, before even a single group member has churned.

## 2 Churn prediction - problem definition

From a machine learning perspective, churn prediction is a supervised (i.e., labeled) problem defined as follows: Given a predefined forecast horizon, the goal is to predict the future churners over that horizon, given the data associated with each subscriber in the network. The input includes data on past calls for each mobile subscriber, together with all personal and business information that is maintained by the carrier. In addition, for the training phase, labels are provided in the form of a list of churners together with their corresponding churn dates.

Specifically, in both our experiments we were provided with limited training data that included only the calls data for a period of about a month. Our goal in the pilot described in Section 4 was to construct a model that predicted the churners for the next two weeks when given three days of calls data.

The nature of the churn prediction problem dictates a specific non-standard performance measure. Recall that once the prediction system produces its churn scores, the retention department makes contact with the subscribers that are most likely to churn, in an attempt to preserve each customer that is established to be a churn risk. Naturally, only a small fraction of the subscriber pool can be contacted at any given time, and the subscribers with the highest churn scores are assigned top priority. Therefore, a churn prediction system should be measured by its ability to identify churners within its top predictions. Formally, performance is measured using the *lift* metric [5]. For any given fraction  $0 < T < 1$ , lift is defined as the ratio between the number of churners among the fraction of  $T$  subscribers that are ranked highest by the proposed system, and the expected number of churners in a random sample from the general subscribers pool of equal size. For example, a lift of 3 at a fraction  $T = 0.01$  means that if we contact the 1% of subscribers ranked highest by the proposed system, we expect to see three times more people who planned to churn in this population than in a 0.01-fraction random sample of the population.

The performance of a churn prediction system is completely characterized by its derived *lift curve*, which maps each fraction  $0 < T < 1$  (horizontal axis) to the lift (vertical axis) that is obtained by the system. In general, the lift curve is monotonically decreasing, since it is usually harder to provide a substantial lift for larger fractions. (Note that by definition, for  $T = 1$  the lift is

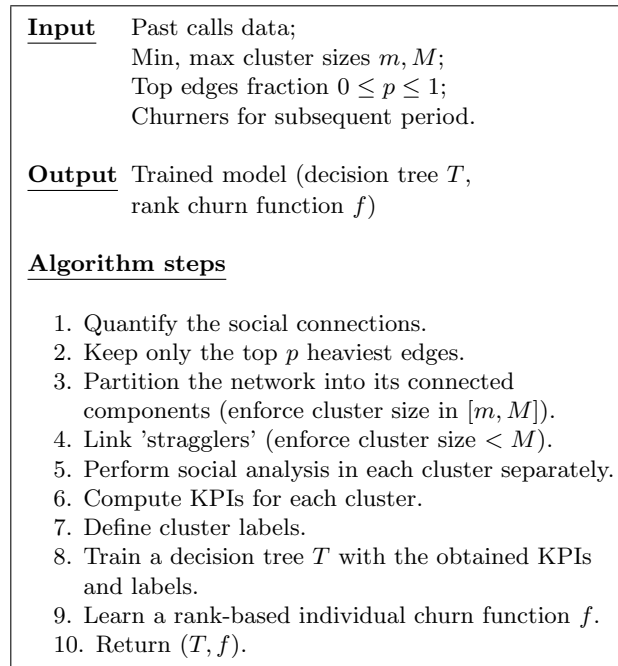


Figure 1: Group-First model training procedure

equal to 1.)

Lift is highly sensitive to various network parameters such as the size of the subscriber pool, the specific segments that are being targeted, local trends in the population, and so forth. As a result, there is no “absolute optimum” to strive for, and it is usually meaningless to compare the performance of different systems that were used in different networks. However, as a general rule of thumb, a lift larger than 4 for the top percentile (i.e., when  $T = 0.01$ ) is considered very significant.

### 3 The Group-First approach to churn prediction

In this section, we formally define our Group-First approach and our derived model for churn prediction. We begin with a description of our model training algorithm and then show how individual subscriber churn predictions can be obtained.

**3.1 Model training** Figure 1 shows the steps of the Group-First model training procedure, the goal of which is to identify social groups and use them for building a churn prediction model. In the following we explain each of these steps.

**3.1.1 Quantifying social connections** The network of subscribers can be easily obtained from the call data. We represent the network as a weighted directed graph, where nodes represent the subscribers, and an

edge  $(i, j)$  is drawn if subscriber  $i$  called subscriber  $j$ . Throughout this section we refer to subscribers and nodes in the graph interchangeably. The nodes in the graph correspond to subscribers of the given operator (termed *local subscribers* henceforth), as well as subscribers that belong to other operators (termed *remote subscribers*). However, all edges are adjacent to local subscribers (at least on one end), since we only see calls that passed through the network of a given carrier. The first question we tackle is the assignment of weights to edges such that the social closeness is represented, i.e., we would like a larger weight to be assigned when the corresponding subscribers are socially closer.

In order to quantify the social relatedness of two subscribers, we rely on the intuitive idea that if two subscribers call a relatively similar set of persons they should be considered highly socially related. To formally capture this intuition we propose to use a similarity measure, which is based on the concept of Mutual Information (MI) [2]. Specifically, we define the *recent calls* of the  $i$ -th subscriber as all the calls she made over the last three days. If she performed more than 100 calls during these three days, only her most recent 100 calls are considered. Next, we associate the  $i$ -th subscriber with a vector  $V_i$ , where  $V_i[t] = 1$  iff the  $i$ -th subscriber called the  $t$ -th subscriber among her recent calls, otherwise  $V_i[t] = 0$ . Thus,  $V_i$  is a vector with  $N \approx 16M$  entries (corresponding to the number of subscribers in the network), of which at most 100 entries are 1, and all the remaining are 0. In order to quantify the similarity between the  $i$ -th subscriber and the  $j$ -th subscriber, we construct a  $2 \times 2$  counts matrix out of  $V_i$  and  $V_j$ , denoted  $C_{ij}$ . Specifically,  $C_{ij}(0, 0)$  counts the number of entries where both  $V_i$  and  $V_j$  equal zero;  $C_{ij}(0, 1)$  counts the number of entries where  $V_i$  equals zero and  $V_j$  equals one;  $C_{ij}(1, 0)$  counts the number of entries where  $V_i$  equals one and  $V_j$  equals zero; and  $C_{ij}(1, 1)$  counts the number of entries where both  $V_i$  and  $V_j$  equal one. Normalizing  $C_{ij}$  by  $N$  – the total number of entries – we obtain a joint distribution, denoted  $P_{ij}$ . For example,  $P_{ij}(1, 1)$  is the probability of the event where both  $V_i$  and  $V_j$  equal one, namely the probability that both the  $i$ -th subscriber and the  $j$ -th subscriber have called the same person during their recent calls. Finally, given this joint distribution, we quantify the similarity of subscribers  $i$  and  $j$  via the Mutual Information contained in  $P_{ij}$  (see [2])

$$S(i, j) \stackrel{def}{=} \begin{cases} \sum_{0 \leq k, k' \leq 1} P_{ij}(k, k') \log \frac{P_{ij}(k, k')}{P_i(k)P_j(k')}, & V_i[j] = 1 \\ 0, & \text{otherwise} \end{cases}$$

where  $P_i$  and  $P_j$  represent the marginal distributions, extracted from  $P_{ij}$ . Specifically,  $P_i(0)$  is the num-

ber of entries in  $V_i$  that equal zero, divided by  $N$ ,  $P_i(1)$  is the number of entries in  $V_i$  that equal one, divided by  $N$ , and  $P_j$  is defined similarly. Due to the properties of the mutual information,  $S(i, j)$  is symmetric and non-negative, and is bounded by the maximum of the entropies of  $P_i$  and  $P_j$ . If both subscribers called precisely the same set of persons in their 100 most recent calls,  $S(i, j)$  will be maximal. As the overlap between the set of persons called by the  $i$ -th subscriber and those called by the  $j$ -th subscriber decreases,  $S(i, j)$  will decrease correspondingly, and if these two sets have no overlap  $S(i, j)$  will tend to 0. In other words, given that we know that the  $i$ -th subscriber called a particular person,  $S(i, j)$  quantifies the number of bits we gained regarding the question of whether the  $j$ -th subscriber called the same person, when averaging across all  $N$  persons in the data. Exploiting the mutual information as a similarity measure has been proposed in the past (e.g., [15]), although to the best of our knowledge it was not proposed in the context of the data we consider in this work, nor in the context of churn prediction.

**3.1.2 Keeping heaviest edges** Recall that our goal is to identify dense social clusters. One can fairly assume that in terms of the weight function we defined, edges that are contained in such clusters will be heavier compared to edges connecting the clusters. We therefore use a parameter  $0 \leq p \leq 1$  that controls the fraction of the heaviest edges we consider, and allows us to focus on the heavier ones. For example, setting  $p = 0.1$  means that we only keep the edges whose weights are among the top 10%, while discarding the rest. The next stages in our algorithm will then consider the induced graph over the surviving edges. We note that the coverage we obtain is monotonic in  $p$ , i.e., the larger  $p$  is, the more nodes we cover in the induced graph. We shall return to this point in Section 4.

**3.1.3 Network partition** At this stage, if we chose the fraction  $p$  appropriately, the resulting network should be composed primarily of relatively small and dense clusters. Since our goal is to identify those disjoint connected clusters in the graph, we consider the graph to be undirected and unweighted for the moment, and partition it into its connected components (with some additional modifications). Two parameters govern this process:  $m$  and  $M$  – the minimum and maximum (respectively) cluster size allowed. Here we assume that either very small or very large connected components are likely to be less informative. Therefore, clusters with sizes smaller than  $m$  are discarded, while we aim to further divide large clusters (larger than  $M$ ) into smaller “islands” that are relatively tight. We employ several

heuristics to accomplish exactly that. We note that since this stage of the algorithm involves the entire network (which may include tens of millions of subscribers), only fast, linear algorithms are viable. Although there is a plethora of research on these fundamental graph partitioning problems, it is mostly irrelevant to such massive datasets, where fast and simple heuristics work best in practice.

**3.1.4 Linking stragglers** As a result of discarding the light edges in the original network (Section 3.1.2), we may have disregarded numerous subscribers that were not covered by the surviving edges. In this stage, after constructing the core clusters in the network, we would like to examine them again and consider adding them to one of the existing clusters. To this end, we compute the connectivity of each subscriber to each of the clusters in terms of the number and duration of calls with cluster members. If a subscriber is found to have a significant connectivity to any cluster, she is added as a new member of that cluster, but marked as having joined at a later stage. This procedure can be executed iteratively to add a new circle of subscribers at each round. The parameter  $M$ , denoting the maximum cluster size allowed, is enforced and clusters are not grown beyond that size.

**3.1.5 Social influence analysis** One of our original hypotheses in this work, which was later illustrated experimentally, was that social leaders have significant influence over the churn decisions of their group members (see Sections 1 and 4). After identifying the tight groups in the network, our next step is to analyze each cluster separately in order to establish the relative social influence of each of its members. We represent each cluster by the directed graph induced by its members and initiate a standard random walk with restarts over this graph. Specifically, we consider a Markov chain whose states are the nodes in the cluster with the following transition matrix. At each node  $v$  with probability  $p$  (tunable parameter, we used  $p = 0.15$ ) we move to a random node (restart); The remaining probability mass  $1 - p$  is evenly divided between the neighbors of  $v$  in the directed graph. Note that since the resulting Markov chain is irreducible and aperiodic, it has a unique stationary distribution. We continue the random walk until convergence to the stationary distribution (or sufficiently close to) and then assign each member its corresponding probability in the distribution obtained. The value assigned to each subscriber should be highly correlated with their local social influence.

1. Number of group members
2. Number of group members who are subscribers of the analyzed provider
3. Ratio of feature (1) to feature (2)
4. Maximal social strength in the group
5. Minimal social strength in the group
6. Ratio of feature (4) to feature (5)
7. Number of calls made by the leader
8. Number of calls received by the leader
9. Average number of calls made by group members
10. Average number of calls received by group members
11. Features (7)-(10) normalized by group size

Table 1: Features (KPIs) used to predict churn of groups

**3.1.6 Group KPIs** Following our hypothesis that social groups tend to churn together, our main goal is to predict group churn. In order to treat this problem as a standard supervised learning problem, we need to extract KPIs for each cluster and provide corresponding labels. This is done in this stage and the next. The set of KPIs that we used is specified in Table 1.

**3.1.7 Cluster labels** We formalize our notion of group churn by considering a group as churned if more than a third of its members have churned. Given the list of individual churners, we can easily label each cluster as churning or not.

**3.1.8 Decision tree training** We experimented with several predictive models and concluded that decision trees [7] provided the best results. The last step in our model training was therefore to train a decision tree with the group KPIs that were extracted according to Sections 3.1.6 and the labels defined in Section 3.1.7. We denote this tree by  $T$ .

**3.1.9 Rank based individual churn function** While examining the experimental data, we observed (see Section 4 for details), that the leader is the most likely to churn, and that churn likelihood decays exponentially for members with lower ranks. In order to exploit this phenomenon, we fit an exponentially decaying function to the probability of churn as a function of subscriber rank, to be used later in making individual subscriber churn predictions. We denote this exponential function as  $f$ .

**3.2 Model test** Figure 2 shows the steps of the Group-First model testing procedure, that is, applying the churn model to new data. In the following, we

**Input** Decision tree  $T$ ; Rank based churn function  $f$ ;  
Parameters  $p, m, M$  as in model train.  
Call data for test period;

**Output** Subscriber churn scores

**Algorithm steps**

- 1–6. Identical to model training (Result: group KPIs).
7. Obtain group churn scores by applying  $T$  to group KPIs.
8. Combine group churn scores with subscriber relative cluster score to obtain individual scores.
9. Return subscriber churn scores.

Figure 2: Group-First model testing procedure

explain these steps.

**3.2.1 Group churn scores** Following our model hypotheses, our first prediction task is to forecast group churn. For this end, we first follow the same steps as in model training in order to construct the network, identify the social groups and derive their KPIs. We then score each group based on its KPIs by using the trained decision tree  $T$ . The assigned score is a real-valued number representing the average tree node purity multiplied by the class label (i.e.,  $-1, +1$ ).

**3.2.2 Subscriber churn scores** Our final prediction task is to forecast individual subscriber churn. For each subscriber we first compute their relative churn score by applying  $f$  to their social rank within their cluster. We then multiply that value by the group churn score of their associated cluster.

**4 Experiments**

**4.1 Description of the data** Our experiments were performed on data from a large mobile operator. This operator logged an average of 117 million call data records (CDRs) every day. We received 28 days of this data.

We found approximately 28 million subscribers during the data period. The subscribers included those of the operator and people who had called subscribers of the operator. Of these, approximately 16 million were subscribers of the operator we analyzed.

In the month after the data was collected, approximately 800,000 subscribers churned. We were provided with the last day on which each of the churning subscribers made a call; this was used as the indicator for

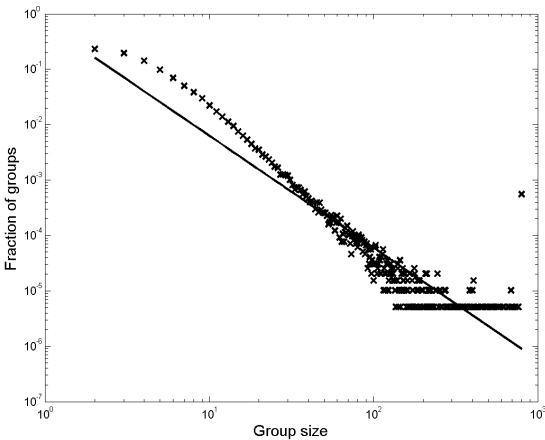


Figure 3: Group size distribution when minimal group size is 2, maximal group size is 800, and the strongest 2.5% of edges are used for building the groups. The black line denotes a power-law regression curve with  $R^2 = 0.87$ .

the day of churn.

For each subscriber we considered only the last 100 calls made during the previous three days.

**4.2 Effect of group size parameters and percentage of edges on groups** Three parameters determine the groups found by the proposed algorithm: minimum and maximum group size, and the fraction of the strongest edges (denoted respectively as  $M$ ,  $m$ , and  $p$  in Section 3).

Figure 3 shows an example of the distribution of group sizes, computed for a minimal group size of 2, a maximum of 800, and using the 2.5% of strongest edges in the data. The distribution closely follows a power law distribution, with a scaling factor of -2.02 ( $R^2 = 0.87$ ).

The effect of the three parameters on the number of subscribers that fall inside one of the groups was estimated using linear regression. Minimal group size and the percentage of strongest edges both had a statistically significant effect on the number of subscribers ( $p < 0.001$ ). However, the maximal group size did not have a statistically significant influence, which may be due to the fact that these groups are simply broken into additional groups once they grow beyond the set threshold.

Table 2 gives a summary of model coverage (i.e., the fraction of subscribers who are members of a group) as a function of the percentage of strongest edges. The table also shows the fraction of outgoing calls associated with each population. This fraction is a useful approximation to the revenue provided by this population. As the

Percentage of strongest edges	Median percentage of subscribers in groups	Percentage of outgoing calls	Ratio of calls to subscribers
15.0	28.1	52.6	1.9
10.0	25.0	44.1	1.9
5.0	16.7	31.6	2.3
2.5	8.3	17.2	2.5

Table 2: Percentage of subscribers and their outgoing calls as a function of the percentage of strongest edges retained

table shows, lowering the percentage of strongest edges focuses the system on those subscribers who are most valuable to the telecom operator. As results below show, the prediction is also most accurate for these valuable subscribers.

**4.3 Indicators for group churn** In this section we report indicators of churn related to group parameters and especially to group leaders, as defined in Section 3.1. We report results obtained at a threshold that retained 2.5% of the strongest edges between subscribers.

The probability of churn for subscribers who are members of small groups (defined as groups with 20 members or fewer) is 2.7 times greater than that of subscribers who are members of larger groups (statistically significant at  $p < 10^{-10}$ , chi square test). Clearly, being part of a larger group makes churn less likely, possibly because of the difficulty in connecting to multiple colleagues across networks.

In the following results, we define a group as having a leader when its highest ranked member has a social strength that is larger by at least 0.2 compared with the weakest member in the group. At this threshold, 70% of the groups have a leader.

In 99% of the groups, the leader is a member of the telecom provider analyzed. This is not surprising as data is strongly biased towards the operator we analyze. In the remaining 1% of groups, however, the probability of churn is 19.4 times greater than in groups where the leaders are from the provider analyzed ( $p < 10^{-10}$ , chi square). Members of these groups are 1.6 times more likely to churn compared to their peers in groups where the leader is from the company analyzed ( $p < 0.005$ , chi square).

Interestingly, leader strength (when the leader is a member of the competitors' network) has an important effect on the probability of group churn. Figure 4 shows the increase in churn probability for a group as a function of leader strength. As the figure shows, there is a high correlation (Spearman,  $R^2 = 0.99$ ,  $p < 10^{-4}$ ) between the strength of a leader and the increase

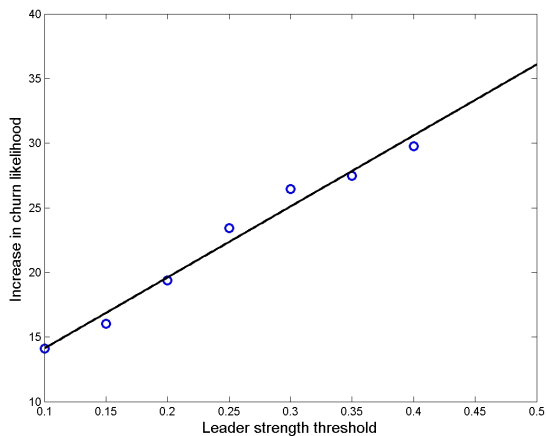


Figure 4: Increase in churn likelihood as a function of leader strength, when the leader is not a member of the analyzed provider. The solid line denotes a linear fit with  $R^2 = 0.99$ .

in churn probability, indicating that the stronger the leader, the more likely there is to be churn from her group.

The probability of churn is highly dependent on subscriber strength: The median strength of a churning subscriber was 0.30 compared to 0.12 of non-churning subscribers (statistically significant, ranksum test,  $p < 10^{-10}$ ). Therefore, the most likely person to churn in a group is the leader, who is 3 times more likely to churn compared to the churn probability of other group members. In groups where two subscribers churned, the probability that one of them is a leader is 12 times that expected by chance. In groups where more than two subscribers churned, the probability that one of them is the leader is 11.8 times that expected by chance. Therefore, the leader seems to be an indicator of churn likelihood in the group.

The current data cannot be used to imply causation, that is, that churn of a leader causes churn of additional subscribers. Further experiments are needed to validate if this is just a matter of correlation or also of causation.

**4.4 Churn prediction results** Our task was to predict which users churned in the 14 days after data was collected. Prediction was performed as described in Section 3.2.

In our experiments we used a single three-day dataset for training, and another seven similar (and non-overlapping) periods for testing. We used lift as our performance measure (see Section 2 for details).

As noted in Section 3, we fit an exponentially decaying function to the probability of churn as a

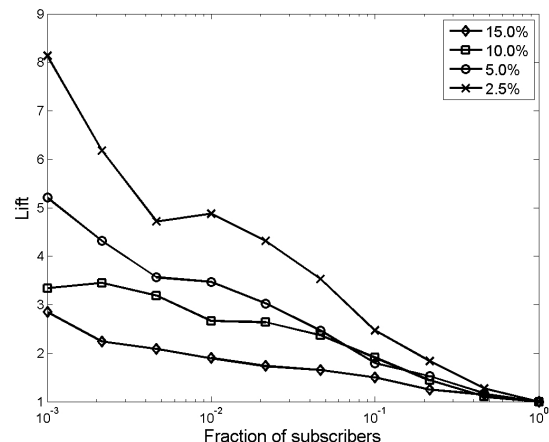


Figure 5: Lift obtained by the churn prediction model for four coverage levels

function of subscriber rank (in the training data), and used it to score subscribers regarding their likelihood of churn. Our experiments showed that the best scoring was obtained using the function  $score = 0.85^{(rank-1)}$ .

The combined churn score for subscribers was obtained by the product of the group churn score with the subscriber churn score.

Figure 5 shows the lift obtained by the proposed algorithm. As this figure shows, when a large population is covered by the model, significant lift of approximately 3 can be obtained using the proposed method. If one focuses on the highest-spending (and thus most valuable) segment of the population, a very high lift of up to 8 can be obtained by our model. This "free lunch", whereby the best performance is reached for the most important part of the subscriber population is likely due to the similarity measure we employed.

Analysis of the decision tree learned for predicting group churn reveals those groups in the most danger of churning and those least likely to churn. For example, groups with more than four members are less likely to churn. Groups at a higher risk of churn are those where the leader made and received few calls, groups with few non-voice interactions (i.e., SMS messages), and groups where the leader is not a subscriber of the analyzed carrier (as described above).

**4.5 Execution times** The execution of the proposed Group-First approach is computationally demanding, but several of its stages lend themselves considerably to parallel processing [13], specifically, the measurement of the strength of connections between subscribers and the analysis of each group. We used the IBM Parallel

Number of processors	Processing time (minutes)	Speedup (compared to 2 CPUs)
2	118	1.0
4	71	1.7
8	61	1.9

Table 3: Run-time and speedup of the proposed algorithm

Machine Learning toolbox [17] (available from <http://www.alpha-works.ibm.com/tech/pml>), as the basis for implementing the proposed method.

The algorithm begins by measuring the strength of connections between every pair of subscribers. This is performed by multiple processing nodes (CPUs), each responsible for a fraction of the subscribers. The strongest pairs are sent to a master node, which executes the connected components stage. Each node is then assigned a fraction of the groups for linking stragglers and for the analysis of each cluster. Finally, extraction of KPIs and prediction is done in the master node.

We used a P575 machine with 8 CPUs (Power5, 1.6 GHz) and 16 GB memory. The processing time for the above-mentioned data is approximately 61 minutes, most of which is required for measuring the strength of connections between subscribers. Therefore, even though we process very large data, processing times are very reasonable.

Table 3 shows the run-time using two, four, and eight processors, as well as the speedup [17]. (The latter is compared to two CPUs because single CPUs cannot handle the memory required.) As this table shows, scaling is efficient in the range we tested. Therefore, even larger customer bases can be analyzed by our method, if necessary by adding processing nodes.

## 5 Discussion and future directions

Churn prediction is one of the most important tasks for any modern telecommunications company, because of the financial penalty associated with churn and the high cost of winning new customers. In this work we proposed a new method that complements current churn models. Our approach is based on an analysis of group behavior, in contrast with both individual-based models and whole-network models currently used by telecommunications companies.

Our approach has important benefits in modeling the interactions between customers, which are known to have an important influence on customer behavior. Additionally, by predicting whole-group churn, we are able to overcome the limitation of social-network based approaches that require knowing which customers have

churned recently. Moreover, because only call data records are used, we do not require the use of financial indicators and demographic information, which are the cornerstone of most individual-based models. This means that our method is applicable to both pre- and post-paid customers without the need to retrain the model, as is often done for individual-based models. Finally, the fact that our prediction is based on the last calls made by each customer in a relatively short time period means that the proposed method can be used on data before it passes through the operator's data warehouse, saving substantial processing time and effort.

The results presented in this paper indicate that using the Group-First approach we can accurately predict churn in a large population.

We have begun investigating the merging of predictions made by individual-based methods with those made by our approach. By training a meta-model that accepts the churn predictions of both models as input, we are able to provide churn prediction accuracy that greatly exceeds the accuracy of either system. This hints that indeed there are several modes of churn, which should be captured by different models, each focusing on different aspects of the problem. We plan to provide comprehensive results of this work in a future paper.

Additionally, we plan to investigate whether the correlation between leader behavior and churn can be used for churn prevention. If this is indeed the case, the proposed algorithm will have significant additional benefits in that churn prevention costs would be significantly reduced by approaching leaders rather than members, without taking their social standing into consideration.

Our work provides the basis for future work in several directions. First, individual-based models have been researched extensively over the years and the most relevant features for them are well-known. While we have provided a list of group features and of individual features for group members, a systematic study that will aim to characterize the general relevance of group-related features to churn is necessary. It will also be of interest to ascertain whether these features are operator and country invariant, or related to cultural effects and the business environments of specific geographies.

Another interesting alley for research is the use of the proposed method in additional applications. For example, the method can be used for campaign management, by modeling the best groups to be approached with a specific marketing campaign and by pinpointing individuals who are the most influential over their peers. Our method can also be used in other domains where

the links between people can be measured or inferred, such as social networking sites on the Internet, in order to predict customer behavior.

## References

- [1] K. Coussement and D. Van den Poel. Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. *Expert Syst. Appl.*, 34(1):313–327, 2008.
- [2] T. M. Cover and J. A. Thomas. *Elements of Information Theory, 2nd Edition*. John Wiley and Sons, Inc, New-York, USA, 2006.
- [3] K. Dasgupta, R. Singh, B. Viswanathan, D. Chakraborty, S. Mukherjea, A. A. Nanavati, and A. Joshi. Social ties and their relevance to churn in mobile telecom networks. In *EDBT '08: Proceedings of the 11th international conference on Extending database technology*, pages 668–677, New York, NY, USA, 2008. ACM.
- [4] P. Datta, B. Masand, D. R. Mani, and B. Li. Automated cellular modeling and prediction on a large scale. *Artif. Intell. Rev.*, 14(6):485–502, 2000.
- [5] E. de Oliveira Lima. *Domain knowledge integration in data mining for churn and customer lifetime value modelling: new approaches and applications*. PhD thesis, University of Southampton, 2009.
- [6] S. Doyle and S. Barn. The role of social networks in marketing. *The Journal of Database Marketing & Customer Strategy Management*, 15(1):60–64, October 2007.
- [7] R. Duda, P. Hart, and D. Stork. *Pattern classification*. John Wiley and Sons, Inc, New-York, USA, 2001.
- [8] R. Fildes. Telecommunications demand - a review. *International Journal of Forecasting*, 18:489–522, 2002.
- [9] R. K. Gopal and S. K. Meher. *Customer Churn Time Prediction in Mobile Telecommunication Industry Using Ordinal Regression*, volume 5012 of *Lecture Notes in Computer Science*, pages 884–889. Springer, 2008.
- [10] J. Hadden, A. Tiwari, R. Roy, and D. Ruta. Churn prediction using complaints data. *Proceedings of world academy of science, engineering, and technology*, 13:158–163, 2006.
- [11] J. Hadden, A. Tiwari, R. Roy, and D. Ruta. Computer assisted customer churn management: State-of-the-art and future trends. *Computers and Operations Research*, 34(10):2902–2917, 2007.
- [12] A. A. Nanavati, S. Gurumurthy, G. Das, D. Chakraborty, K. Dasgupta, S. Mukherjea, and A. Joshi. On the structural properties of massive telecom call graphs: findings and implications. In *CIKM '06: Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 435–444, New York, NY, USA, 2006. ACM.
- [13] B. Parhami. *Introduction to parallel processing: algorithms and architectures*. Springer, 1999.
- [14] P. C. Pendharkar. Genetic algorithm based neural network approaches for predicting churn in cellular wireless network services. *Expert Syst. Appl.*, 36(3):6714–6720, 2009.
- [15] N. Slonim, G. S. Atwal, G. Tkacik, and W. Bialek. Information-based clustering. *Proceedings of the National Academy of Science*, 102(51):18297–18302, 2005.
- [16] G. Song, D. Yang, L. Wu, T. Wang, and S. Tang. A mixed process neural network and its application to churn prediction in mobile communications. In *Proceedings of the Sixth IEEE International Conference on Data Mining Workshops (ICDM Workshops)*, pages 798–802, 2006.
- [17] H. Toledano, E. Yom-Tov, D. Pelleg, E. Pednault, and R. Natarajan. Support vector machine solvers: Large-scale, accurate, and fast (pick any two). Technical Report H-0260, IBM Research, 2008.
- [18] J. Yang, X. He, and H. Lee. Social reference group influence on mobile phone purchasing behaviour: a cross-nation comparative study. *International Journal of Mobile Communications*, 5(3):319–338, 2007.