

Collective Churn Prediction in Social Network

Richard J. Oentaryo, Ee-Peng Lim, David Lo, Feida Zhu, and Philips K. Prasetyo
School of Information Systems, Singapore Management University
Email: {roentaryo, eplim, davidlo, fdzhu, pprasetyo}@smu.edu.sg

Abstract—In service-based industries, churn poses a significant threat to the integrity of the user communities and profitability of the service providers. As such, research on churn prediction methods has been actively pursued, involving either intrinsic, user profile factors or extrinsic, social factors. However, existing approaches often address each type of factors separately, thus lacking a comprehensive view of churn behaviors. In this paper, we propose a new churn prediction approach based on collective classification (CC), which accounts for both the intrinsic and extrinsic factors by utilizing the local features of, and dependencies among, individuals during prediction steps. We evaluate our CC approach using real data provided by an established mobile social networking site, with a primary focus on prediction of churn in chat activities. Our results demonstrate that using CC and social features derived from interaction records and network structure yields substantially improved prediction in comparison to using conventional classification and user profile features only.

I. INTRODUCTION

Churn, which broadly refers to the loss of customers, has been a prominent issue in a variety of service-based industries including telecommunication, banking, online gaming, and social network service [1]–[4]. Threats arising from churn have substantial impact on the profitability of service providers as retaining an existing customer imposes significantly less cost than acquiring a new one. Accordingly, many providers are now shifting their focus from customer acquisition to customer retention. Many research efforts have therefore been directed toward accurately predicting churn in its early stage to target potential churn customers and provide the appropriate incentives to sustain their interest in the service.

Studies have shown that churn can be attributed to *intrinsic* and *extrinsic* factors [5], [6]; the former pertain to the customer profiles and/or inherent features of the service (such as customer’s membership age, pricing, service failure rate, etc.), while the latter portray the service in terms of the value it accrues through its social roles (e.g., community opinion, word-of-mouth effects, etc). Conventional approaches for churn prediction have been largely focused on intrinsic factors, using (pure) feature-based modeling techniques which treat individual users independently [1], [7]. However, these approaches lack account for the role of social ties between individuals in affecting the propensity to churn, which can be examined from their interaction network.

On the other hand, several recent works have been reported which model churn based on the social roles and influence of individuals within a community. Typically, these approaches use information diffusion models, the prime example being the *spreading activation* (SPA) model [2], [3]. The SPA approach suggests that churn propagates from one user to another, i.e.,

a user linked to other users who already churned is likely to churn next, which relates to the notion of social ties and importance. Although SPA can help improve prediction accuracy, it suffers from several limitations. For instance, SPA operates on a set of global parameters (e.g., spreading factor) that may not accurately reflect the different roles of individuals in the community. Also, it makes a rather simplistic assumption that churn emerges only as the result of influence from users who had churned, whereas the importance of other intrinsic user features are not taken into account.

In this paper, we present a data-driven approach that exploits both intrinsic (user profile) and extrinsic (social tie) factors underlying churn behaviors. Deviating from the conventional feature-based and diffusion-based models, we approach the problem of churn prediction using the idea of *collective classification* (CC) [8]–[10]. The CC approach explores the local features of and dependencies among individuals during each classification step, and infers the status (label) of a group of inter-related individuals jointly rather than independently. In this way, both intrinsic and extrinsic features can be accounted for during prediction, leading to a more comprehensive view of the factors underlying churn. We empirically evaluate the efficacy of our CC approach using real data supplied by an established mobile social networking service called myGamma.

To our best knowledge, there has been no work reported on data-driven CC to predict churn. Perhaps the closest is the Markov logic network described in [11], which employs first-order logic and graphical model to simulate churn behaviors. However, it requires extensive human intervention (e.g., hand-coded graph structure) and its scalability is rather limited so far. We summarize our main contributions as follows:

- 1) Through analysis of the social network data, we propose a simple yet robust criterion for identifying churn users based on the last period of inactivity. The application of such a criterion can be generalized to cases whereby an explicit definition of churn is not available.
- 2) We develop a mining framework for churn prediction that incorporates traditional classification and iterative CC techniques. Our framework also provides a method to evaluate and compare various intrinsic, user profile features and extrinsic, social features underlying churn behaviors in a comprehensive manner.
- 3) We demonstrate empirically that using CC together with extrinsic, social features derived from interaction records and network structure can significantly improve the churn prediction accuracy, as opposed to the conventional approaches using intrinsic profile features only.

II. CHURN PREDICTION

A. Dataset

For our churn prediction task, we consider the data from a mobile social networking site called myGamma that offers a range of services for chatting/messaging, friendship linking, application adoption, blogging, and user group formation. The site has 4.8 million registered users, most of which are young adults (aged 20-30) from developing countries. Our main interest is to investigate churn of chat activity, which is one of the most popular types of activity in the site. We have maintained and analyzed a database containing more than 1 year’s worth of various chat activities (e.g., chat session, chat messages, etc.) and user profile (e.g., age, country, language, race, etc) in the social network. For our churn prediction task, though, we focus on the 7-month data taken from 01 June 2011 to 31 December 2011. This would more accurately reflect the recent trends that the service provider is interested in.

In our study, we consider only users who have chatted at least once within a specified time period. A chat user is typically involved in several chat sessions, each of which has a unique session ID as well as a start and an end timestamp. With this, we can measure social tie strength based on how long a user chats with other user(s) in each session, or how many sessions a user has engaged in. On average, there are 4.8K unique users chatting each month, and there are 1.36 million chat sessions in a month. Note also that a user need not declare explicit friendship in order to chat with another user. That is, chatting with strangers is possible, and in fact there are many such cases in the data. Moreover, a chat session can involve more than two users (i.e., a group chat), with slight variation on the timing and chat duration for each user (e.g., a user may join the session later or go offline earlier).

B. Problem Formulation

To define our churn prediction problem, let us first denote $\Delta t_{i,s} = t_{i,s} - t_{i,s-1}$ as the time gap of inactivity between two consecutive chat sessions ($s - 1$) and s that a user i has engaged in, where $s \in \{1, \dots, S_i\}$, S_i is the total number of sessions user i has, and $t_{i,0} = 0$. We next defined, for each user i , the maximum time gap over all chat sessions $\Delta t_i = \max_s \Delta t_{i,s}$. Fig. 1(a) plots the distribution of Δt_i (in days), measured over the recent 1-year period. Interestingly, it can be observed that the Δt_i between any consecutive sessions never exceeds 30 days. This suggests that 30 days is a good threshold to distinguish between churn and active (non-churn) users. Thus, we define the label y_i of a user i using (1):

$$y_i = \begin{cases} \text{churn} & \text{if } (t_m - t_{i,S_i}) > \varphi \\ \text{non-churn} & \text{otherwise} \end{cases} \quad (1)$$

where the threshold φ is set to be 30 days as above, and $(t_m - t_{i,S_i})$ refers to the last inactivity period with respect to time reference point $t_m \geq t_{i,S_i}$ for a particular observation month m . We have also cross-verified with our data provider that $\varphi = 30$ days is a reasonable churn criterion.

It is also worth noting that our proposed churn definition is simpler and more robust than the definition in [5], which

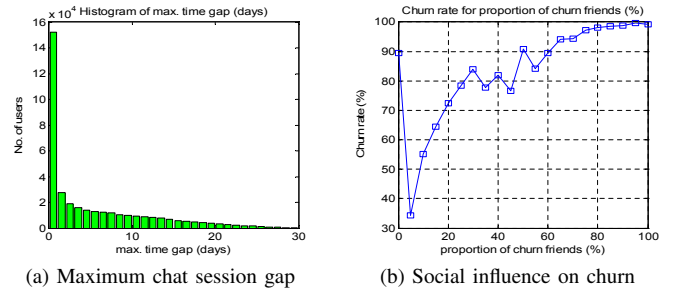


Fig. 1. Key statistics of myGamma dataset

is based on two time windows: a previous activity window and a churn window. That is, a user is labeled as churning if her activity in the churn window drops to some extent relative to the activity in the previous window. Using this definition, however, a user who had churned may still return when her current activity increases beyond some level. Moreover, such definition requires specifying the lengths of the time windows and the results greatly depend on the choice of the lengths.

With this criterion, we formulate the churn prediction problem as follows: *Given profile and/or social features $\vec{x}_i(m)$ of a user i , observed up to time reference point t_m in month m , we predict whether the user will churn by time t_{m+1} in the next month ($m + 1$).* That is, the observation period is one month ahead of the churn period, which can provide insights on the chat activities of a user before he/she decides to churn. Here we choose t_m to be the midnight of the second day of month m (e.g., t_{July} is 2 July 00:00am), so as to cater for users who may chat pass midnight at the last day of the preceding month. It should also be noted that there may be a few users who have churned but are not captured in the dataset, simply because they never chat during the observation period.

C. Social Influence

Another central question we seek to answer is whether the decision of a user to churn depends on her social ties and community effects. To gain insights on the social aspects of churn behavior, we first need to find an evidence of whether social influences on a user’s churn propensity to churn exist. The basic premise here is that an user’s probability to adopt a new behavior (i.e., churn) increases with the proportion of his/her friends who already engaged in the behavior.

To this end, we compute from our social network database the chat churn rate (probability) with respect to the proportion of friends who previously churned. Our findings are summarized in Fig. 1(b), i.e., the likelihood to churn increases as the proportion of churn friends increases. An exception here is when the churn friend proportion is zero. Our investigation reveals that this corresponds to users who have no friend. Hence, the high churn rate is reasonable, as a user who has no friend would likely have no activity and be labeled as churning by definition. We can also see that the churn friend proportion fluctuates between 35% and 55%, suggesting that in this range the proportion imposes little difference on the churn propensity. However, the general trend remains that churn rate increases with higher churn friend proportion.

III. PROPOSED METHODOLOGY

For churn prediction, we consider two classification approaches. The first is *conventional (non-relational) classification*, which assumes that the data instances are independently and identically distributed (i.i.d). That is, the prediction of each instance is made separately, without regard to the inter-dependencies among the instances. Our second, proposed approach is to use *iterative collective classification algorithm* (ICA) [8], [10]. ICA simultaneously classifies a set of inter-linked users (nodes) by considering the correlations between the label of node v and its observed features, between the label of node v and the observed features of its neighbors, and between the label of node v and labels of the unobserved nodes in its neighborhood. We elaborate the non-relational classifier and ICA methods in Sections III-B and III-C, respectively.

A. Chat Graph Construction

As our goal is prediction of chat-activity churn, we consider in our study the construction of *chat graph*, extracted from the chat session table in our database. The nodes in our chat graph represent the chat users, and the directed edges indicate the social ties between any two users. We create an edge between node A and B only when the chat initiated by A is responded by B (and vice versa), i.e., we only regard *reciprocal edges* as a sign of social ties between two individuals.

In turn, the strength of social ties between two nodes are encoded as edge weights. Formally, we calculate the weight $e_{A,B}$ from node A to B according to (2):

$$e_{A,B} = \sum_{s \in S_B} \left[\frac{d_{s,A}}{N_s - 1} \right], N(s) \geq 2 \quad (2)$$

where $d_{s,A}$ is the duration of user A talking in a chat session s , and N_s is the number of users involved in session s , and S_B is the set of chat sessions which involve user B . The denominator term $N_s - 1$ is introduced to prevent excessive updates of the edge weights due to the possibility of group chats having $N_s > 2$. It must also be noted that, though the edges are reciprocal, the edge weights $e_{A,B}$ and $e_{B,A}$ need not be identical (as S_B and S_A are not necessarily the same).

B. Conventional Classification

As mentioned earlier, the traditional non-relational classification approaches require an i.i.d assumption. Let $\vec{x}_i \in \mathbb{R}^d$ be a feature vector in a d -dimensional space and $X = \{\vec{x}_1, \dots, \vec{x}_i, \dots, \vec{x}_N\}$. Also let $y_i \in \{1, -1\}$ be the (binary) class label (1 for churn and -1 for non-churn), and $Y = \{y_1, \dots, y_i, \dots, y_N\}$. The i.i.d-based inference can then be expressed in terms of the conditional probability (3):

$$P(Y|X) \propto \prod_{i=1}^N P(y_i|\vec{x}_i) \quad (3)$$

As our non-relational i.i.d classifier, we employ the popular *support vector machine* (SVM) algorithm. Specifically, we use the linear SVM algorithm developed under the LIBLINEAR

Algorithm 1 ICA Inference Procedure

Input: $G = (V, E, X, Y)$: graph with nodes V , edges E , features X and labels Y , Y^K : labels of observed nodes $V^K \subset V$, LC and RC : local and relational classifiers, J_{max} : maximum iteration (default: $J_{max} = 10$)

Output: Y^U : labels of unobserved nodes in G

- 1: $V^U \leftarrow V - V^K$
- 2: **for all** nodes $v_i \in V^U$ **do**
- 3: $(y_i, conf_i) \leftarrow LC(\vec{x}_i)$ // bootstrapping
- 4: $\vec{g}_i \leftarrow \text{computeLIRelationalFeatures}(v_i, V, E)$
- 5: **end for**
- 6: **for** $j = 1$ **to** J_{max} **do**
- 7: $r \leftarrow |V^U| \times (j/J)$
- 8: $Y' \leftarrow Y^K \cup \{y_i | v_i \in V^U \wedge \text{rank}(conf_i) \leq r\}$
- 9: **for all** nodes $v_i \in V^U$ **do**
- 10: $\vec{f}_i \leftarrow \text{computeLDRelationalFeatures}(v_i, V, E, Y')$
- 11: **end for**
- 12: **for all** nodes $v_i \in V^U$ **do**
- 13: $(y_i, conf_i) \leftarrow RC(\vec{x}_i \cup \vec{f}_i \cup \vec{g}_i)$
- 14: **end for**
- 15: **end for**
- 16: $Y^U \leftarrow \bigcup y_i$

framework [12], which is renowned for its competitive accuracy and efficiency. We also use SVM as a component in our ICA approach, as described shortly in section III-C.

C. Collective Classification

In relational data or information networks, there are complex dependencies among instances for which the i.i.d assumption is not suitable. An effective model for relational data should be able to consider the dependencies among the related instances. The CC approach provides one such model, focused on exploiting the inter-instance label dependencies to improve classification performance [8], [10]. In particular, CC estimates the conditional probability given in (4):

$$P(Y|X) \propto \prod_{i=1}^N P(y_i|\vec{x}_i, Y_{j \in \mathcal{N}_i}) \quad (4)$$

where \mathcal{N}_i is the set of neighbors of \vec{x}_i

In this work, we employ a variant of CC approach called the *iterative CC algorithm* (ICA) [8], [10]. In ICA, a *local classifier* (LC) and a *relational classifier* (RC) are first trained using local and relational features of the observed instances (nodes), respectively. Each classifier can be implemented using the SVM algorithm (cf. section III-B). The local features are static, comprising user profile and/or interaction features. The relational features stem from the structure of the chat graph, and are dynamically computed using aggregation operators such as count, proportion, or mode [8].

During ICA inference phase, the trained LC and RC are applied to unobserved instances whose class labels are unknown. The process is outlined in Algorithm 1. In steps 1-4, the prediction of the labels for unobserved instances are achieved via bootstrapping using LC and local features. In steps 6-7,

label predictions made with highest confidence are deemed as valid and included into data. Using the newly accepted labels, the relational features are recomputed in steps 8-11, based on which RC reclassifies the labels in steps 12-14. Note that, in each inference iteration, a greater percentage of predictions are accepted and new relational features are derived. This constitutes a form of "cautious" CC inference, as it sought to preferentially exploit the more certain relational information. Also, as the prediction is reevaluated each time, the label of a node accepted in one iteration may be discarded in the next if the prediction confidence is no longer sufficiently high.

The ICA method provides an effective means to improve the classification accuracy on relational data. However, because of its dependence on label assignments, its performance may degrade when a large fraction of neighboring nodes are also unlabeled. To compensate for errors arising from such uncertainty, we extend the ICA approach by leveraging *label-independent* (LI) relational features \vec{g}_i [9] in addition to the *label-dependent* (LD) features \vec{f}_i commonly used in conventional CC algorithms. This corresponds to step 4 of Algorithm 1. The LI features provides another source of information that are derived from the network structure, but not dependent on the current nodes' label assignments. Thus, unlike LD features, the LI features can be accurately computed regardless of the availability of label information.

IV. CHURN PREDICTION EXPERIMENTS

A. Features Considered

In Section I, we mentioned several intrinsic and extrinsic features that influence churn behavior in online communities. In this study, we focus on two types of features: *local* and *relational features*. The former refers to the set of features that remain static over the course of prediction and can be used by either conventional or collective classification. We then divide the local features into two groups: user profile and interaction features, portraying static intrinsic and extrinsic aspects of a user, respectively. The interaction features are derived based on multi-modal information, spanning such activities as chat, mobile application, friend action, group, blog, and testimonial. Table I lists the local features considered in this work.

We also take into account several relational features to be exclusively used by our ICA algorithm, as given in Table II. These features are dynamically recomputed during ICA iterations and derived from users' connectivity structure and tie strengths. We can thus view the relational features as different, extrinsic aspects of a user, complementing those portrayed by the user's interaction features. For our ICA-based prediction, we incorporate two types of relational features: label-dependent and label-independent, as already discussed in Section III-C. The complete list of the relational features used in our study is given in Table II. Among them are the Jaccard and cosine similarity indices, defined as $J_{AB} = \frac{|N_A \cap N_B|}{|N_A \cup N_B|}$ and $C_{AB} = \frac{|N_A \cap N_B|}{\sqrt{|N_A| \times |N_B|}}$ respectively for node pair A and B , and N_A is the set of neighbors of node A . Chiefly, they reflect the fraction of common neighbors between any two nodes.

TABLE I
LOCAL FEATURES FOR TRADITIONAL AND COLLECTIVE CLASSIFICATION

Category	Feature
Profile	Birth age of a user (in years)
	Gender of a user (male or female)
	Country at which a user resides
	Race or ethnic group that a user belongs to
	Duration for which a user has joined the social network
Interaction	Number of chat sessions a user has been involved in
	Number of chat messages a user has sent
	Number of applications a user has installed
	Number of friends (positive friendships) a user has
	Number of foes (negative friendships) a user has
	Number of friend adding actions a user has done
	Number of friend removal actions a user has done
	Number of friend blocking actions a user has done
	Number of groups/communities a user has joined in
	Number of group messages a user has sent
	Number of times a user joins a group/community
	Number of times a user leaves a group/community
	Number of blogs a user has posted
	Number of blog comments a user has posted
Number of testimonials a user has posted	

TABLE II
RELATIONAL FEATURES FOR COLLECTIVE CLASSIFICATION

Category	Feature
Label-independent	Number of neighbors of a node (i.e., degree)
	Average in-weight of a node (as computed in (2))
	Average out-weight of a node (as computed in (2))
	Average number of 2-hop neighbors of a node
	Average Jaccard similarity of a node and its neighbors
Label-dependent (\mathcal{L} = churn or non-churn)	Number of \mathcal{L} -neighbors of a node (i.e., \mathcal{L} -degree)
	Average weight of in-edges from \mathcal{L} -neighbors of a node
	Average weight of out-edges to \mathcal{L} -neighbors of a node
	Average degree of the \mathcal{L} -neighbors of a node
	Average Jaccard similarity of a node and its \mathcal{L} -neighbors
	Average cosine similarity of a node and its \mathcal{L} -neighbors
Fraction of \mathcal{L} -neighbors of a node w.r.t. all its neighbors	

B. Training and Testing

Fig. 2 illustrates the data instances and graphs generation for training and testing over a 3 month period, from the beginning of May to end of August 2011. Here our training instances consist of users who chat in the recent 1 month (1 June-2 July 0:00am), and their (ground-truth) labels are derived based on the last inactivity up to the next month's time reference (2 August 0:00am). Meanwhile, the train graph contains nodes corresponding to users who chat in the last 2 months (i.e., 1 May-2 July). We chose 2-month window in order to distinguish between the past churners and the current users under observation. This allows us to investigate on the influence of past churners on the churn propensity of the current users. Specifically, the past churners are users who appear in the train graph but not in the training instances (i.e., they never chat for > 1 month). These users are also treated as *observed* nodes V^K in our ICA inference process. The same instances and graph generation procedure applies to the test case, except that the time window is shifted 1 month ahead.

C. Prediction Results

Using the training/testing setup in Section IV-B, we evaluate the performance of our churn prediction methods. Our

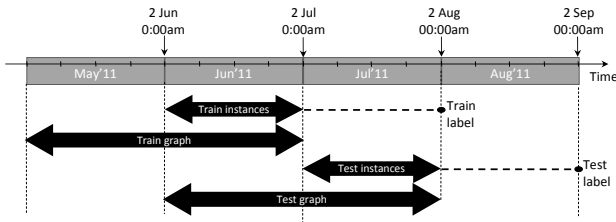


Fig. 2. Data instances and graphs for training and testing

performance evaluation involves four measures popularly used for (binary) classification tasks, including: $Accuracy = \frac{TP+TN}{TP+FN+TN+FP}$, $Precision = \frac{TP}{TP+FP}$, $Recall = \frac{TP}{TP+FN}$, and $F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$, where TP , TN , FP and FN are the true positives, true negatives, false positives, and false negatives respectively. Our particular interest is on the *churn* case, hence we treat churn as the positive label in this study.

Our evaluations also involve comparisons along two dimensions: using conventional (i.i.d) classification vs. collective classification (ICA), and using profile vs. interaction features as the local features (cf. Table I). We cross-validated our results over 6 train and test trials, corresponding to June-November 2011 and July-December 2011 data, respectively. Table III summarizes the testing performances, averaged over 6 trials. As our baseline, we consider a random classifier which predicts the class label equally likely. The expected *Accuracy* of random classifier would be equal to the proportion of instances labeled as churn (53.79% for testing; see the footnote of Table III). In this case, the expected *Precision* would be the same as *Accuracy*, and we can assume that positive and negative misclassifications are made at the same rate (i.e., $FP = FN$), giving $F1 = Precision = Recall$.

From the results in Table III, we can conclude that all methods outperform the baseline, random classifier. Among the four metrics, we are mainly concerned with the *F1* score. When only profile features are used, ICA is superior to the conventional classifiers, i.e., ICA produces higher *F1* and *Accuracy*, with smaller standard deviations implying more robust performance. When interaction features are used, however, the two approaches have marginal differences in performance. This can be attributed to some correlation between (or complementary role of) the interaction and relational features, e.g., friend count in Table I is related to degree in Table II. We also discovered that structural features such as degree and in/out-weight are generally the most predictive, but due to space constraints we do not report the feature ranking results here. All in all, using social features (interaction and relational features) is a crucial facet for improved churn prediction, in comparison to using profile features only.

V. CONCLUSION

This paper put forward a data-driven methodology for churn prediction that facilitates exploration of intrinsic and extrinsic factors underlying churn using collective classification (CC) approach. We evaluated our CC approach on real data from an established social networking site and compared it with the conventional, non-relational classifiers using different feature

TABLE III
CONSOLIDATED TESTING PERFORMANCES (JULY-DECEMBER 2011)

Method	Performance metric	Feature type	
		Profile	Interaction
Conventional classification	Accuracy (%)	56.06 ± 3.19	68.96 ± 1.89
	Precision (%)	58.13 ± 2.93	64.92 ± 2.78
	Recall (%)	67.78 ± 22.51	91.89 ± 1.79
	F1-score (%)	60.38 ± 13.3	76.06 ± 2.17
Iterative CC algorithm (ICA)	Accuracy (%)	58.70 ± 4.32	70.44 ± 1.86
	Precision (%)	58.41 ± 4.84	67.07 ± 2.99
	Recall (%)	83.03 ± 4.88	88.46 ± 2.48
	F1-score (%)	68.38 ± 2.96	76.25 ± 2.24
Baseline accuracy = baseline F1-score = 53.79%			

combinations. The potential of the CC approach in combination with interaction and network structure features has been exemplified in our experimental results. In our future work, we would like to incorporate a richer set of social features derived from graph theory and link analysis in order to capture more complex dependencies underlying churn. Last but not least, we plan to generalize our CC approach to simultaneously exploit multiple graphs as different sources of information.

VI. ACKNOWLEDGMENTS

This work is supported by the National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

REFERENCES

- [1] L. Yan, R. H. Wolniewicz, and R. Dodier, "Predicting customer behavior in telecommunications," *IEEE Intelligent Systems*, vol. 19, no. 2, pp. 50–58, 2004.
- [2] K. Dasgupta, R. Singh, and B. Viswanathan, "Social ties and their relevance to churn in mobile telecom networks," in *Proceedings of the International Conference on Extending Database Technology*, 2008, pp. 668–677.
- [3] J. Kawale, A. Pal, and J. Srivastava, "Churn prediction in MMORPGs: A social influence based approach," in *Proceedings of the IEEE International Conference on Computational Science and Engineering*, vol. 4, 2009, pp. 423–428.
- [4] J. Lang and F. S. Wu, "Social network user lifetime," in *Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 2011.
- [5] M. Karnstedt, T. Hennessy, J. Chan, and C. Hayes, "Churn in social networks: A discussion boards case study," in *Proceedings of the IEEE International Conference on Social Computing*, 2010.
- [6] M. Karnstedt, M. Rowe, J. Chan, H. Alani, and C. Hayes, "The effect of user features on churn in social networks," in *Proceedings of the ACM International Conference on Web Science*, 2011.
- [7] W. Verbeke, K. Dejaeger, D. Martens, J. Hur, and B. Baesens, "New insights into churn prediction in the telecommunication sector: A profit driven data mining approach," *European Journal of Operational Research*, vol. 218, no. 1, pp. 211–229, 2011.
- [8] P. Sen, G. Namata, M. Bilgic, and L. Getoor, "Collective classification in network data," *AI Magazine*, vol. 29, no. 3, pp. 93–106, 2008.
- [9] B. Gallagher and T. Eliassi-Rad, "Leveraging label-independent features for classification in sparsely labeled networks: An empirical study," in *Proceedings of the International Conference on Advances in Social Network Mining and Analysis*, 2008.
- [10] L. K. McDowell, K. M. Gupta, and D. W. Aha, "Cautious collective classification," *Journal of Machine Learning Research*, vol. 10, pp. 2777–2836, 2009.
- [11] T. Dierkes, M. Bichler, and R. Krishnan, "Estimating the effect of word of mouth on churn and cross-buying in the mobile phone market with markov logic networks," *Decision Support Systems*, vol. 51, pp. 361–371, 2011.
- [12] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, and C. J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.