Community-Based Prediction of Activity Change in Skype

Irene Teinemaa
University of Tartu
Estonia, J.Liivi 2
irene.teinemaa@gmail.com

Anna Leontjeva
University of Tartu
Estonia, J.Liivi 2
anna.leontjeva@ut.ee

Marlon Dumas
University of Tartu
Estonia, J.Liivi 2
marlon.dumas@ut.ee

Riivo Kikas
University of Tartu
Estonia, J.Liivi 2
riivokik@ut.ee

***Abstract*—A key problem for facilitators of online communication and social networks is to identify users whose activity is likely to change in the near future. Such predictions may serve as basis for targeted campaigns aimed at sustaining or increasing overall user engagement in the network. A common approach to this problem is to apply machine learning methods to make predictions at the level of individuals. These approaches consider only information about each individual user and, thus, do not exploit the social connections and structure of the network. In this paper, we approach the problem of activity change prediction at the level of communities rather than individuals. We develop predictive models of activity change over communities obtained using state-of-art community detection methods and compare their predictive power with each other and against the single-user baseline and ego networks. The results show that community-level prediction models achieve higher prediction accuracy than the traditional single-user approach, whereas a local community detection algorithm outperforms a global modularity-based method.**

I. Introduction

Being able to predict which users will change their product usage intensity can help businesses to focus their customer care and loyalty initiatives more effectively. On the one hand, knowing in advance which users will become less active in the near future helps businesses allocate resources effectively with the goal of retaining them, thus preventing *customer churn*. On the other hand, early estimation of increasing activity enables businesses to put in place mechanisms to boost loyal customer product usage even further. The question of activity change prediction is particularly relevant in social networks, as active users are essential for a network to be sustainable. Indeed, the value of a social network is considered to be dependent on the number of its connected users and their level of engagement [1], [2].

Traditionally, the problem of predicting changes in activity has been approached at the level of individual users [3]–[8]. The key idea of these approaches is to construct models that, given a vector of features of one individual user, predict whether the user's future level of activity will increase, remain stable or decrease.[1] By focusing on users taken in isolation, these approaches do not fully exploit the structure of the social network. Yet, previous studies have shown that the intensity of a user's activity is dependent on that of their friends [4] and that tight social groups tend to change their level of activity together [9], entailing that: (i) a predictive model that takes into account the structure of the network is likely to achieve higher levels of accuracy; and (ii) a model that identifies groups of users that are likely to change their aggregate level of activity tells us more about what parts of the network to target (e.g. via marketing campaigns) than a model that predicts activity change at the level of individual users. The latter observation holds particularly in very large social networks where targeting a significant percentage of individuals in the network with limited resources is unfeasible.

Previous research has shown that real-world social networks tend to have *community structure* [10], meaning that there are high concentrations of edges within certain groups of vertices and low concentrations between these groups [11]. This observation has led to a broad definition of a *community* as a set of users in a social network who are tightly connected to each other and have relatively few connections with users outside the community. A number of community detection methods have been proposed, which can be broadly classified into two categories: local and global community detection methods [12]. In the former category, the construction of communities starts from the egocentric network of individual users, while in the latter the goal is to partition the network into larger subsets that expose the modular structure of the network seen as a whole.

Given the above observations, this paper addresses the following questions:

RQ1 Are predictive models of activity change for communities of users more (or less) accurate than those constructed for individual users (herein called *single-user approaches*?

RQ2 Are predictive models of activity change of "local" communities more (or less) accurate than those constructed for "global" communities?

We study these questions in the context of a large-scale network, namely the Skype communication network. As a representative of a global community detection method, we consider a well-known community detection algorithm, namely Louvain [13], which is designed to optimize modularity between communities. Meanwhile, as a representative of local-first community detection methods, we consider a derivative of the Demon method [14] – HDemon [15], which constructs communities starting from the egocentric networks and merging adjacent communities based on the percentage of nodes they share, thus leading to denser communities than Louvain.

[1] The extreme version of this problem where the goal is to predict that the level of activity will drop to zero is also known as *churn prediction*.

We compare the accuracy of activity change prediction models constructed for the above two types of communities between them and against three baselines: (i) predictive models constructed for individual users along the lines of those studied in previous work; (ii) predictive models constructed for egocentric networks; and (iii) predictive models constructed for random sets of users.

The rest of the article is structured as follows. Section II introduces the dataset used in this study. In Section III, community detection methods and baselines are explained. Section IV describes the parameters and features in the prediction model. In Section V, the prediction models and the results are evaluated. Finally, Section VI discusses related work while Section VII summarizes the findings.

II. Data Description

In this work, we explore the network of social connections in Skype as of October 2011. The nodes in the network represent users of Skype. The resulting network is undirected, where edges exist between users who belong to each other's contact list. Each edge is accompanied with the time when it was created. Therefore, it is possible to handle the network as dynamic, considering at each timepoint only the edges that have emerged before that time.

The nodes in the network are associated with users' profile data. For each user, we know the date, country and city of account creation. Users have also the possibility to fill in their birth year and gender, but as it is not mandatory, these data are available only for a small subset of users.

In addition to the social network and user profile data, we have at our disposal the following activities: chatting, making an audio or video call. Each of these activities is considered as the users' Skype product usage. The usage is aggregated monthly for each user as the number of days in this month when the user used the given product. The problem of activity change is posed with respect to these products; and the prediction models are built for each of the products separately.

The complete Skype network contains non-active users. These users do not take part in the social engagement, bringing bias to the data set. Therefore, we use a filtered network in the analysis that consists of active users only – those who make an audio call or chat during at least 2 out of 3 months preceding the first observation month. In other words, we limit ourselves to communities where clients are 'recurrent' – i.e. use various products for a relatively long period of time. Thus, we discard 'one-off' users from the prediction task.

The data provided by Skype are anonymous with hashed user ID-s. The product usage data do not contain any information about individual communications, such as the participants, content, length, or time of the interaction. The intensity of the usage is not recorded on finer granularity than a month.

III. Community Detection Methods and Baselines

The common notion of a community states that it should have more edges within itself than between the community and the rest of the network. However, no universal definition of a community exists. A variety of community detection methods have been proposed, resulting in different sets of communities. Thus, a practical approach is to define communities as the products of a given community detection algorithm [11].

A. Description of community detection methods

In this work, we apply the following community detection methods.

- The *Louvain* method [13] is based on modularity maximization [16]. The result is a *partition*, where each vertex belongs to exactly one cluster. The algorithm proceeds in a hierarchical fashion, so that the resulting communities are considered as input vertices for the next level. The final output allows us to explore multiple levels of communities.
- The counterpart of a partition is a division where each node may belong to several *overlapping* communities. In this work, we use *the Hierarchical Demon (HDemon)* [15], which utilizes a local-first approach of discovering communities. The method starts with the extraction of ego networks of each node and discovers local communities in each ego network. Then, two communities are merged if at most $\epsilon\%$ of the nodes in the smaller community are not included in the larger one. Similarly to Louvain, HDemon produces a hierarchical view of the communities through reapplication of the core algorithm.

We compare the community detection algorithms with two baseline methods for extracting groups of users in the network. With the help of such baselines, we are able to evaluate the added value of the community structure for our prediction task.

- As our first baseline we extract *ego networks* – subgraphs that contain a node together with all of its friends and edges between them.
- In order to demonstrate the ability of community detection methods to combine users by their activeness, we introduce the notion of *random groups of nodes*. For a given community size n, we choose n nodes randomly from the total set of nodes in the network. The sizes of random groups are assigned from a power law distribution in order to resemble the distribution of the outcome of the community detection methods.

B. Structural properties

Each of the grouping methods produces different types of communities that can be distinguished by a number of statistical and structural properties. In order to understand and assess the results of the predictive models, we perform descriptive analysis of these properties.

HDemon results in several orders of magnitude more communities than Louvain, while the latter covers about two times the number of nodes that HDemon does (Table I). This is partly due to HDemon's high overlapping ratio – the number of total users divided by unique users in communities. On average, each node belongs to 9 communities. Ego networks have a similar degree of coverage as HDemon with slightly lower overlapping ratio. Moreover, due to the rapid growth of the network, community detection performed on a later snapshot results in a higher number of communities. The number of

TABLE I. COMMUNITY STATISTICS

	Method	# com-s	Coverage	Overlap	Largest size	Median size
January 2009	HDemon	187k	12%	9.0	3.7k	46
	Louvain	10k	22%	1.0	629.7k	45
	Ego	100k	16%	3.4	2.3k	43
	Random	100k	99%	4.3	10k	91
November 2010	HDemon	3 358k	28%	13.8	14.5k	51
	Louvain	93k	52%	1.0	8 401k	50
	Ego	100k	6%	1.3	2.3k	46
	Random	100k	46%	1.3	10k	91

communities produced by the baseline methods is limited to 100 000 groups out of all possible ones.

By far the largest communities are produced by Louvain, even though the median community sizes for Louvain and HDemon are very similar (Table I). This indicates a very skewed community size distribution for Louvain, with a few huge communities (Figure 1a). The maximum size of random groups is limited to 10 000. The minimum size of communities is fixed to 30 members as discussed later in Section IV-A.

It is also relevant to compare the community detection methods in terms of structural characteristics. To this end, we look at the structural features that most discriminate between different community detection methods and baselines, namely internal density, conductance, and relative hub degree. We compare these metrics using probability density plots (Figure 1), where the distribution of the measure for the communities is plotted for each of the methods.

We observe on Figure 1b that the internal density is highest for HDemon communities and lowest for the random groups. In order to capture the linkage between communities, we use conductance – the ratio of internal to outgoing edges. Conductance is highest in Louvain communities as shown on Figure 1c. Looking internally, some communities have a hub-and-spoke structure, while others are more spread. To quantify the existence of a central hub in a community, we use the notion of *relative hub degree* – the maximum internal degree divided by the average internal degree. This measure is highest for ego networks (Figure 1d), which indicates that these communities contain a user who connects with most of the others in the community.

In order to get a better sense of the communities produced by different methods, we present prototypical examples for each method. On Figure 2a we can see a community with high internal density, which serves as a prototypical example of the HDemon communities. Figure 2b illustrates the modular structure of Louvain communities – high ratio of internal to external edges. A typical ego network with a central hub is shown on Figure 2c. Lastly, a random group of nodes is illustrated on Figure 2d.

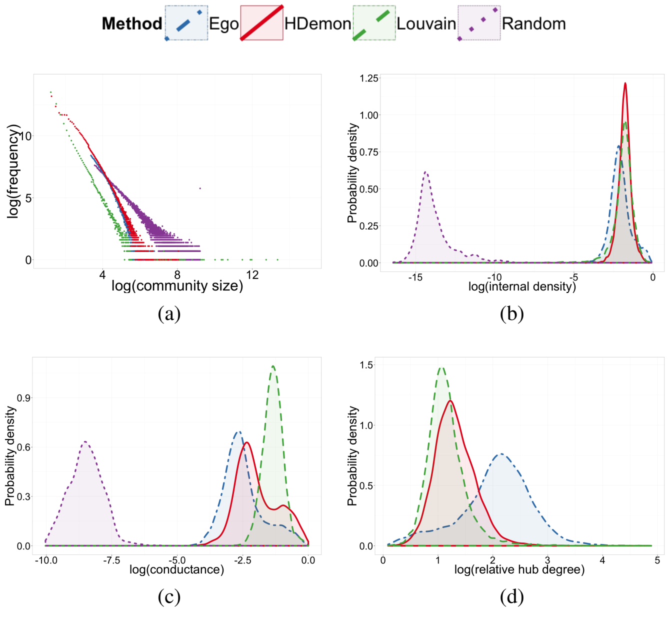
C. Entropy of activeness in communities

The approach of this paper relies on the idea that users who are connected behave similarly in terms of Skype product usage. In order to confirm this hypothesis we adopt the notion of Shannon entropy for the communities and compare it with the entropy for random groups of nodes. To this end, we apply the same filtering as introduced in Section II at the end of the data period and identify users who are not active anymore. This way, each user is labeled as being active or non-active at the end of the data period. For each community C we compute the proportion of active members (p_1) and calculate the entropy as

$$H(C) = -(p_1 \log_2 p_1 + (1-p_1)\log_2(1-p_1)) \qquad (1)$$

Entropy shows the level of uncertainty in the community, where 1 indicates that active and non-active users are mixed together, while entropy 0 implies that the community consists of either all active or all non-active users.

The results on Figure 3 show that random groups of nodes indeed have entropy close to 1. All of the community methods perform better at grouping similar users, while the entropy of ego networks and HDemon are very similar and both are better than Louvain. This suggests that the internal structure of the Skype network to some extent affects user product usage.



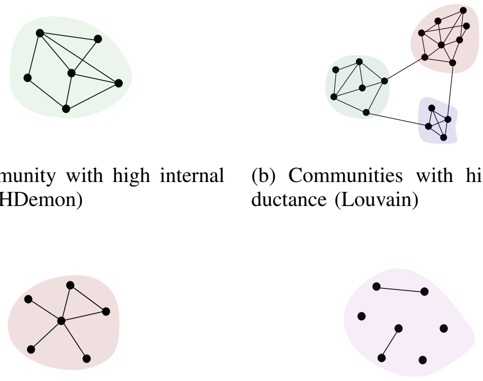
(a)

(b)

(c)

(d)

Fig. 1. Structural characteristics



(a) Community with high internal density (HDemon)

(b) Communities with high conductance (Louvain)

(c) Community with high relative hub degree (Ego network)

(d) Community with mostly disconnected nodes and few edges (Random group)

Fig. 2. Prototypical examples of communities

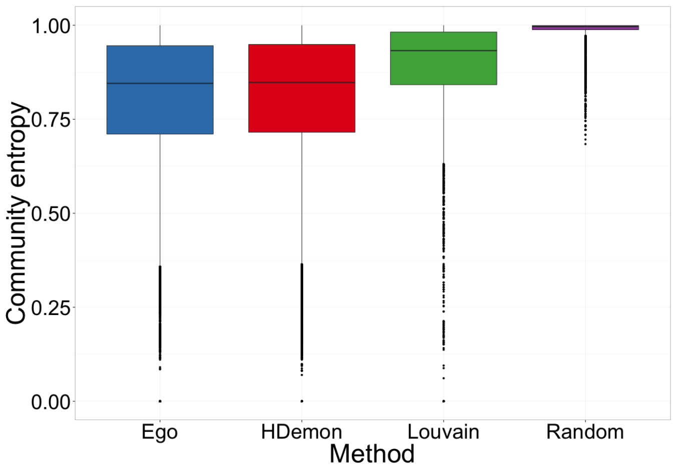


Fig. 3. Community entropy of individual users' activeness

IV. Model Construction

In this section, we describe the design of the experiments, including the temporal split of the data for training and testing purposes, and provide the specification of the model with its parameters.

A. Temporal Split and Parameter Setting

As we aim to predict the activity change taking into consideration the temporal evolution of the communities, the splitting scheme has to be carefully specified. Namely, we want to avoid making predictions about past, using data from the future. In our case, we choose a solution which follows a similar splitting strategy for time series as described by Kuhn and Kjell [17].

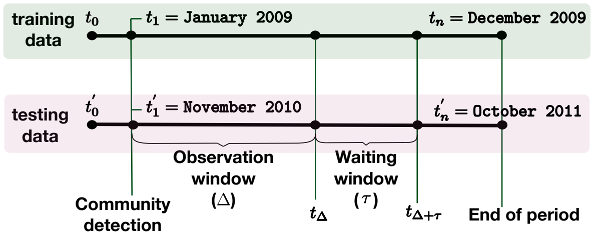


Fig. 4. Division of time into train and test periods

More specifically, we divide the timeline into train and test sets as shown on Figure 4. The training and testing periods are non-overlapping and both contain data from 12 months. By performing community detection twice, independently for training and testing periods, we make sure that patterns persist over time. Moreover, there is a gap between the periods, which ensures that the network has significantly changed and, thus, the predictions are not overly optimistic.

There are multiple parameters that are used for the set-up of the experiments (Figure 4). The parameters are the following:

- t_1 - the first month of the considered period. The observation window always starts from this month. Community detection is performed at the beginning of this month.
- t_n - the last month of the considered period. Data can not be acquired beyond this month.
- t_Δ - the last month of the observation window. For training and predicting, data from t_1 to t_Δ are used, so the length of the observation period is Δ.
- $t_{\Delta+\tau}$ - the month for which we want to predict the state (for training set, the month of the label). τ is the number of months between t_Δ and the month in the future that we want to predict, $1 \leq \tau \leq n - \Delta$.
- α - activity change threshold. A community is considered as meaningfully changing in product usage if its usage between t_Δ and $t_{\Delta+\tau}$ has changed by at least the fraction α.
- β - interestingness threshold. In case of activity decrease prediction, a community is only included in the analysis if its product usage at t_Δ is higher than the interestingness threshold. In case of activity increase, this threshold is irrelevant, as we are interested also in cases where the community is initially inactive and increases to a positive value.

Additionally, we fix a few variables related to community detection. Firstly, we choose $\epsilon = 25\%$ for HDemon, meaning that two communities are merged if at most 25% of nodes in the smaller one are not included in the larger one.

Moreover, we focus the analysis on medium-sized and large communities by excluding communities smaller than 30 nodes, mainly for two reasons. Firstly, medium-sized communities are of higher interest from the business perspective as it potentially may lead to a higher number of users reached by marketing. Secondly, the mean product usage estimates for small-sized communities are highly volatile, and, thus, unreliable as predictive values. The experiments show that the margin of error (E) of mean product usage stabilizes for communities larger than 30 (e.g. $E = 5\%$ under $C.I. = 95\%$ for chat days).

B. Features

For each community we extracted a set of features, some of which are common in the community detection literature and others derive from the peculiarities of the provided data. The features that can be divided into three main groups: *product usage*, *structural* and *profile* features (Table II). The first set of features encompasses the mean product usage of community members in the preceding months. Structural features describe the internal structure of the community and relations with other communities. Some examples of structural features are clustering coefficient, internal density and conductance [18]. Profile features give insight into the geographical dispersion of the community members, as well as the average account age and the extent of voluntary profile information provided.

The features have different variability with respect to time. Namely, they can be dynamic (changing each month) or static (same value in each month or growing linearly). The static features occur once in the predictive feature set. Each dynamic feature is included in the model Δ times, once for each month during the observation period.

TABLE II. FEATURES

	Feature	Description
Usage	Chat days	number of days the user chatted in a month
	Audio days	number of days the user made a call in a month
	Video days	number of days the user made a video call in a month
	Connected days	number of days the user connected (login) in a month
	Active members	number of users with connected days ≥ 1 in given month
Structural	Edge count	number of edges inside the community
	Size	number of community members at time t_1
	Local nodes	number of nodes having neighbors only inside the community
	Outgoing edges	number of edges leaving the community
	Outside nodes	number of neighboring nodes from other communities
	Internal density	ratio of existing edges to all possible edges between community members
	Global CCF	number of closed triplets over all triplets in the community
	Local CCF	ratio of connected neighbors (average over all members)
	Assortativity	preference of nodes in a community to attach to others of a similar degree
	Conductance	ratio of edges inside the community to edges leaving the community
	Avg total degree	avg. total degree of community members (counting both internal and external edges)
	Max total degree	maximum total degree
	Avg internal degree	avg. internal degree of community members (counting only internal edges)
	Max internal degree	maximum internal degree
	Hub degree ratio	maximum total degree divided by the number of links inside the community
Profile	Entropy countries	Shannon entropy of the country distribution of its community members
	Entropy cities	Shannon entropy of the city distribution of its community members
	Num countries	number of different countries represented in the community
	Num cities	number of different cities represented in the community
	Geo max distance	maximum distance between members using city-level location data
	Geo avg distance	average distance between members using city-level location data
	Gender unknown	percentage of users who have not provided information about their gender
	Age unknown	percentage of users who have not provided information about their age
	Account age	years from account creation date, average over members
	Oldest account age	age of the oldest account in the community
	Diff of account ages	difference of first and last account creation date
	Mean ies	mean inter-arrival time, with respect to the account creation date
	Std ies	standard deviation of the inter-arrival time
	Males percentage	average over users who have provided gender information in their profile
	Average age	average over users who have provided age information in their profile

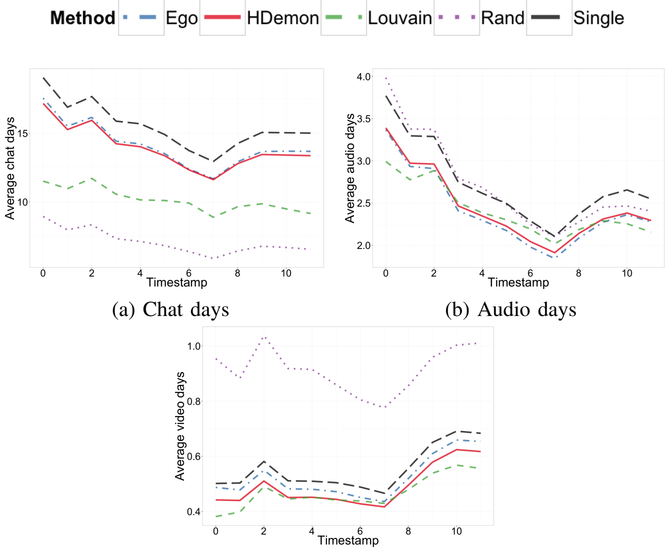
In case of communities and group-based baselines, the features are calculated as the average over the group members. In the single user approach, the features correspond directly to the user's profile data or product usage; features that are only defined for groups of users are calculated based on the user's ego network.

Our goal is to predict activity change in terms of three Skype products: chat, audio, and video days. The average product usage follows the same pattern over time across all community detection methods and baselines (see Figure 5), where chat days is the most frequently used product, followed by audio, and video days. The absolute values of usage differ across methods with chat days higher for single users, ego networks and HDemon. The audio and video usage is highest for random communities. The differences in product usage over time across the data sets affects and complicates the analysis, resulting in different optimal parameters for the prediction models (see Section V-A).

V. MODEL EVALUATION

In this work, we use the supervised machine learning approach to achieve our goal of predicting activity change. As we aim to identify areas of the network with meaningful changes in activity, we decided to use classification instead of regression. We build the models using random forest [19].

We approach the problem of activity change prediction as two subtasks, where the first task is predicting increasing vs. not increasing communities and the second – decreasing vs. not decreasing communities.



(a) Chat days

(b) Audio days

(c) Video days

Fig. 5. Average product usage in communities over time

Before evaluation, we determine the optimal parameters for each product – the settings, where the model achieves the highest accuracy. After that, we measure the prediction accuracy in terms of AUC achieved by different methods.

A. Parameter Tuning

The optimal parameters are determined using *grid search* – exhaustive searching on a manually specified subset of the hyperparameter space. For parameter tuning, we sampled randomly half of the train and half of the test data.

After conducting experiments with different levels of hierarchical community detection methods, we discovered that the optimal level for Louvain is the 1st, while for HDemon – the 4th. On Figure 6 we can see a heatmap of AUC scores over combinations of activity increase threshold and length of waiting period, where length of observation period is fixed to 1. The results show that the higher the activity increase threshold, the easier it is to predict, but it comes at the cost of higher imbalance with smaller number of communities-increasers. The length of the waiting period, on the other hand, does not have a clear effect on the prediction accuracy. The same patterns persist in the case of decrease prediction.

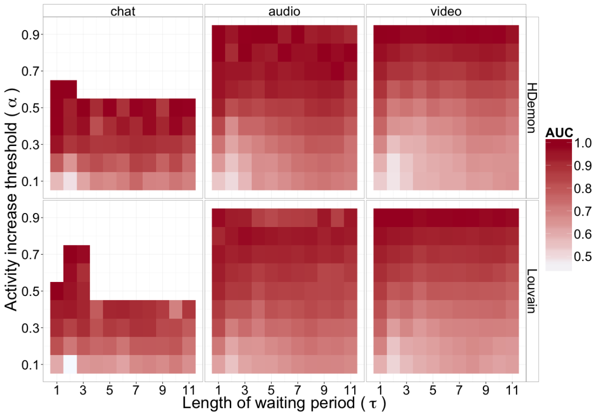


Fig. 6. Grid search over activity increase thresholds and waiting periods with fixed observation period length ($\Delta = 1$)

When determining the final set-up parameters, we introduced the constraint that the length of the waiting window must be at least 3 months in order to make the prediction task more applicable in practice. The resulting optimal parameters are presented in Table III.

TABLE III. EVALUATION PARAMETERS

	Product	Δ	τ	α	β
Increase	Chat	8	3	0.3	-
	Audio	8	3	0.5	-
	Video	5	6	0.7	-
Decrease	Chat	4	4	0.5	0
	Audio	9	3	0.7	0
	Video	9	3	0.9	0

For example, in case of video days the highest accuracy in terms of AUC is gained if communities are observed for 5 months (Δ) and the predictions about the increase of activity are made after 6 months (τ). In this case the communities are labeled as increasing if their activity has increased by at least 70% (α) and all of them are included in the data set, regardless of their initial activity (β).

An interesting observation is that the higher the overall usage of a product, the lower the optimal activity change threshold. It may suggest that it is a more difficult task to predict products with lower usage among customers.

B. Prediction Accuracy

TABLE IV. PREDICTION RESULTS

AUC $+/-$ 95% C.I.

		Chat	Audio	Video
Increase	HDemon	**0.934** +/- 0.002	**0.928** +/- 0.002	0.947 +/- 0.002
	Ego	0.919 +/- 0.007	0.890 +/- 0.009	**0.950** +/- 0.005
	Louvain	0.885 +/- 0.007	0.885 +/- 0.008	0.923 +/- 0.006
	Single	0.825 +/- 0.003	0.770 +/- 0.004	0.717 +/- 0.004
Decrease	HDemon	0.876 +/- 0.002	0.923 +/- 0.006	0.942 +/- 0.005
	Ego	**0.879** +/- 0.007	**0.955** +/- 0.007	**0.963** +/- 0.007
	Louvain	0.752 +/- 0.007	0.886 +/- 0.011	0.918 +/- 0.008
	Single	0.662 +/- 0.004	0.725 +/- 0.005	0.771 +/- 0.005

In this section, we compare the accuracy of predictions, which is estimated using the AUC metric [20]. The AUC expresses the probability that a randomly chosen positive sample is ranked higher than a random negative one. It has been shown to be a suitable measure for evaluating accuracy of models in the context of imbalanced data [21], which is the case here.

The results in Table IV show all the AUC values with corresponding 95% confidence intervals (C.I.). These results suggest that groups of users enable more accurate predictions than single users for all the products.

In both cases (increase and decrease) HDemon and ego network models achieve similarly high accuracy, which indicates that communities produced by local-first approach contain more information about product usage. The lower accuracy of Louvain communities suggests that modularity-based approach is less suitable for the activity prediction, which is attributable to the resolution problem described in [22].

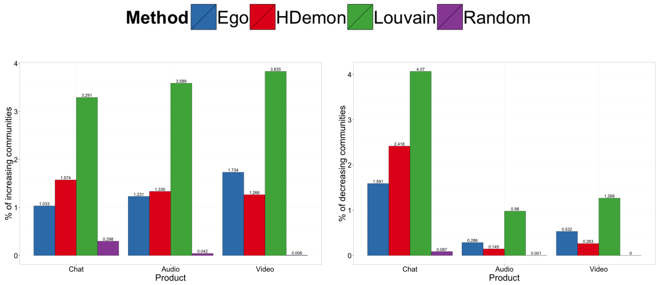
The overall best result for both decrease and increase is achieved for video with the model for ego networks (AUC increase $= 0.95$ and AUC decrease $= 0.963$). However, taking into account the internal structure of produced communities, HDemon may be a better choice in terms of targeting strategy as the internal density of nodes for HDemon is higher. The actual targeting strategies are out of the scope of this paper and are left for future work.

Another aspect of comparison of community detection methods is the extent of imbalance in the data set as compared to random groups of users. As can be seen on Figures 7a and 7b, the ratio of communities that are considered as changing in their activity is very low across all the methods. Still, the imbalance is much more extreme for the random groups. For example, only 6 out of 100 000 communities increase in video days; furthermore, there are no random groups of users that decrease in video days. Therefore, a comparison with random groups in terms of accuracy is not reasonable. As we observe from these figures, in all cases Louvain has the highest number of changing communities.

VI. RELATED WORK

a) Community detection: The problem of community detection in static networks has been extensively investigated in the literature [11]. Several different approaches have been proposed and algorithms have been designed that enable detecting communities in graphs of up to billions of nodes and edges.

The notion of modularity has been widely used in practice to discover communities, such as the method of Clauset et



(a) Ratio of increasing coms

(b) Ratio of decreasing coms

Fig. 7. Ratio of changing communities

al. [23] and Blondel et al. [13](Louvain). The complexity of the Louvain method is almost linear in the number of nodes in the network, making it usable in very large networks. However, Fortunato and Barthlemy discuss the effects of *resolution limit*, which may cause modularity-based methods to miss the real structure of the smaller-sized modules [22].

Mostly, real-world social networks tend to have an overlapping community structure by nature. One of the first methods for finding an overlapping community structure is the method by Palla et al. [24]. Their method finds maximal cliques in the network and merges them to form larger communities.

Community detection methods that approach the large-scale network as a whole tend to produce large formations of users which are not easily interpretable as communities in real life [25]. An algorithm designed to avoid this problem is introduced by Coscia et al. [14]. Their method discovers communities using a local-first approach, starting from the ego networks of each user. The result is a set of relatively dense overlapping communities.

Lescovec et al. show that communities have an impact on viral marketing, as users in densely connected communities make more purchases [26]. The importance of communities from the marketing perspective has also been studied by Oestreicher-Singer and Zalmanson, who show that engagement in communities in an online social media network has a positive effect on users' willingness to pay for services [27].

b) Churn prediction: A question of practical interest with respect to activity of customers is *churn*. A common concept in telecom companies, traditionally churners are users who leave the service for the benefit of a competitor. According to a wider definition used in many sources, churn is the significant decrease in user's activeness. Karnstedt et al. propose a definition for churn in social networks [28]. They introduce the concepts of *previous activity window* (from t_1 to t_{n-1}), *churn window* (from t_n to t_{n+m}), and *threshold factor* ($0 \leq T < 1$). A user is recognized as a churner if his mean activity during the churn window divided by the mean activity during the previous activity window constitutes a fraction smaller than the threshold factor. Additionally, several alternations are discussed, such as taking the median activity of the windows instead of the mean, requiring the activity to be below the threshold for a number of timesteps, or using an absolute threshold factor.

In this study, we use a similar approach to define the class of communities that change in terms of product usage. Instead of comparing the mean activity scores from the previous activity window and churn window, we compare only the last month of both windows. The reason behind is that the activity scores of communities are more stable than for individual users and do not need to be smoothed over time.

The most common approach for churn prediction is to build the prediction model based on users' features [3], [5]. In addition, social network analysis has been exploited for churn prediction [6]–[8]. Dasgupta et al. used social network data to predict churners in a telecom network [4]. They show that the number of friends who have churned increases the probability of the user to churn. Furthermore, they build a diffusion model to describe the propagation of churn between users. The results show that social ties significantly influence churn, and reasonably good predictions can be achieved by using the social network data alone. These findings provide a basis for our hypothesis that groups of tightly connected users exhibit similar behaviour in terms of activity and, thus, justify the approach of considering communities instead of single users in the context of product usage.

Richter et al. followed the hypothesis that groups of users tend to churn together [9]. They extracted dense groups of users from the network and predicted whether at least 1/3 of the group will churn. Their results show that smaller groups are significantly more likely to churn. Also, they show that a group is more likely to churn when there is a clear leader in the group. They achieve lift between 3 and 8, depending on the population size covered by the model.

Our approach differs from the work of Richter et al. in several aspects. Firstly, the definition of change in communities is defined differently. Richter et al. consider the problem of churn prediction, where a group is a churner when 1/3 of the group members leave the service. Our definition is more flexible, enabling to predict both decrease and increase in activity, rather than expecting the complete churn of users. Secondly, they have at their disposal a data set consisting of calls made between users. This enables them to quantify social relations between any two users and form groups based on the heaviest edges. In our case, the edges are unweighted, so the approach of Richter et al. is not suitable. Furthermore, we use an extended feature set in our analysis, adding several structural features of communities and the members' profile data.

VII. Conclusions

We have studied the problem of predicting which users in a communication network will either increase or decrease their activity after a given period of time and beyond a given change threshold. In contrast to traditional approaches to this problem, which focus on making predictions for individuals, we have approached the problem at the level of communities. Specifically, we applied two representative community detection algorithms: a global modularity-based community detection method (Louvain) and a local-first method that produces denser communities (HDemon). Furthermore, we used single users, ego networks and random groups as the baselines for comparison.

The evaluation conducted on a large communication network (Skype) covering three different products confirms that communities tend to group together users with similar activity change patterns. The activity change prediction models built for community-based approaches achieve higher accuracy (AUC) than those built at the level of individuals (cf. RQ1 in Section I). Additionally, models built for HDemon communities and ego networks are more accurate than those built for Louvain communities (cf. RQ2 in Section I).

To sum up, the study shows that prediction of activity change at the level of communities (particularly denser ones) has advantages over prediction for single users. Thus, community-level targeting has some potential as an efficient marketing strategy.

This study paves the way for several research directions. Firstly, the approach of targeting communities could be combined with studies on influence and diffusion in networks. For instance, one open question is how many users in a community should be targeted in order to achieve sufficient coverage and hence prevent for example a given community from churning collectively. The number of targeted community members could be optimized for example by determining the most influential users in each community.

Secondly, instead of detecting communities at a particular snapshot in time, an evolutionary community detection algorithm could be used, which updates the communities as new nodes and edges appear in the network. This way, users who join the network during the observation period could be included in the analysis. Also, the events that a community goes through, such as growing, merging or splitting, might give additional insight into their future life-cycle.

ACKNOWLEDGMENT

This work is supported by Microsoft (Skype) and ERDF via the Software Technology and Applications Competence Centre (STACC). We acknowledge the valuable input of Ando Saabas and Adriana Dumitras.

REFERENCES

[1] B. Metcalfe, "Metcalfe's law: A network becomes more valuable as it reaches more users," *Infoworld*, vol. 17, no. 40, pp. 53–54, 1995.

[2] B. Briscoe, A. Odlyzko, and B. Tilly, "Metcalfe's law is wrong — communications networks increase in value as they add members — but by how much?" *Spectrum, IEEE*, vol. 43, no. 7, pp. 34–39, 2006.

[3] C.-P. Wei and I.-T. Chiu, "Turning telecommunications call details to churn prediction: a data mining approach," *Expert Systems with Applications*, vol. 23, no. 2, pp. 103–112, 2002.

[4] 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 *Proceedings of the 11th international conference on Extending database technology: Advances in database technology*. ACM, 2008, pp. 668–677.

[5] Y. Xie, X. Li, E. Ngai, and W. Ying, "Customer churn prediction using improved balanced random forests," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5445–5449, 2009.

[6] R. J. Oentaryo, E.-P. Lim, D. Lo, F. Zhu, and P. K. Prasetyo, "Collective churn prediction in social network," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*. IEEE Computer Society, 2012, pp. 210–214.

[7] Y. Zhu, E. Zhong, S. J. Pan, X. Wang, M. Zhou, and Q. Yang, "Predicting user activity level in social networks," in *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. ACM, 2013, pp. 159–168.

[8] W. Verbeke, D. Martens, and B. Baesens, "Social network analysis for customer churn prediction," *Applied Soft Computing*, vol. 14, pp. 431–446, 2014.

[9] Y. Richter, E. Yom-Tov, and N. Slonim, "Predicting customer churn in mobile networks through analysis of social groups." in *SDM*. SIAM, 2010, pp. 732–741.

[10] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[11] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010.

[12] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: inferring user profiles in online social networks," in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 251–260.

[13] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[14] M. Coscia, G. Rossetti, F. Giannotti, and D. Pedreschi, "Demon: a local-first discovery method for overlapping communities," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 615–623.

[15] ——, "Uncovering hierarchical and overlapping communities with a local-first approach," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 9, no. 1, p. 6, 2014.

[16] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.

[17] M. Kuhn and J. Kjell, *Applied Predictive Modeling*. Springer, 2013.

[18] Y. Yang, Y. Sun, S. Pandit, N. V. Chawla, and J. Han, "Perspective on measurement metrics for community detection algorithms," in *Mining Social Networks and Security Informatics*. Springer, 2013, pp. 227–242.

[19] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[20] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (roc) curve." *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.

[21] J. Burez and D. Van den Poel, "Handling class imbalance in customer churn prediction," *Expert Systems with Applications*, vol. 36, no. 3, pp. 4626–4636, 2009.

[22] S. Fortunato and M. Barthelemy, "Resolution limit in community detection," *Proceedings of the National Academy of Sciences*, vol. 104, no. 1, pp. 36–41, 2007.

[23] A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," *Physical review E*, vol. 70, no. 6, p. 066111, 2004.

[24] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[25] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, "Statistical properties of community structure in large social and information networks," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 695–704.

[26] J. Leskovec, L. A. Adamic, and B. A. Huberman, "The dynamics of viral marketing," *ACM Transactions on the Web (TWEB)*, vol. 1, no. 1, p. 5, 2007.

[27] G. Oestreicher-Singer and L. Zalmanson, "Content or community? a digital business strategy for content providers in the social age," *MIS Quarterly*, vol. 37, no. 2, 2013.

[28] M. Karnstedt, T. Hennessy, J. Chan, and C. Hayes, "Churn in social networks: A discussion boards case study," in *IEEE Second International Conference on Social Computing (SocialCom)*. IEEE, 2010, pp. 233–240.