User Community Detection via Embedding of Social Network Structure and Temporal Content

Hossein Fani[a,*], Eric Jiang[b], Ebrahim Bagheri[b], Feras Al-Obeidat[c], Weichang Du[a], Mehdi Kargar[b]

[a] *University of New Brunswick, Fredericton, NB., Canada*
[b] *Ryerson University, Toronto, ON., Canada*
[c] *Zayed University, United Arab Emirates*

Abstract

Identifying and extracting user communities is an important step towards understanding social network dynamics from a macro perspective. For this reason, the work in this paper explores various aspects related to the identification of user communities. To date, user community detection methods employ either explicit links between users (link analysis), or users' topics of interest in posted content (content analysis), or in tandem. Little work has considered temporal evolution when identifying user communities in a way to group together those users who share not only similar topical interests but also similar temporal behavior towards their topics of interest. In this paper, we identify user communities through *multimodal* feature learning (embeddings). Our core contributions can be enumerated as (a) we propose a new method for learning neural embeddings for users based on their temporal content similarity; (b) we learn user embeddings based on their social network connections (links) through neural graph embeddings; (c) we systematically interpolate temporal content-based embeddings and social link-based embeddings to capture both social network connections and temporal content evolution for representing users, and (d) we systematically evaluate the quality of each embedding type in isolation and also

*Corresponding author:
Email address: `hfani@unb.ca` (Hossein Fani)
URL: `https://sites.google.com/site/hosseinfani/` (Hossein Fani)
The implementation is available at `http://tiny.cc/i9fj7y`

when interpolated together and demonstrate their performance on a Twitter dataset under two different application scenarios, namely *news recommendation* and *user prediction*. We find that (1) content-based methods produce higher quality communities compared to link-based methods; (2) methods that consider temporal evolution of content, our proposed method in particular, show better performance compared to their non-temporal counter-parts; (3) communities that are produced when time is explicitly incorporated in user vector representations have higher quality than the ones produced when time is incorporated into a generative process, and finally (4) while link-based methods are weaker than content-based methods, their interpolation with content-based methods leads to improved quality of the identified communities.

Keywords: User Community Detection, Social Network Analysis, Twitter

1. Introduction

Online social networks such as Twitter have become an effective medium for users to not only express their interests in real time through short textual snippets, but also to bond with other users. The analysis of users' dynamic preferences at the macro level through community detection methods can exhibit the overall properties of the network, its evolution, and future functions. Community-level analysis of a social network has shown to be more effective than their user-level counterparts in some tasks, e.g., in social recommender systems [1, 2], information diffusion modeling [3], financial and political analytics [4, 5], and churn prediction [6], just to name a few.

There is already an abundant number of user community detection methods, especially for social network platforms, in the literature that approach the problem from various perspectives such as the use of min-cuts [7], modularity maximization [8], or clique identification [9, 10], to name a few. However, one of the essential aspects of social networks, which is often overlooked in community detection methods, is their dynamic evolving nature. For instance, given content on the social network are often reflective of issues in the real world, the topics discussed on the network constantly change and hence users' interests towards these topics also change as the community evolves and new topics and connections are made [11]. Let us consider an example scenario of three users from the dataset that we used in our experiments in this paper. In Figure 1, we depict how these three users' interests evolve over a time period of November and December 2010 with regards to the specific topic of 'War in Afghanistan'. All of the three users expressed interest in this topic at some point in time, as such, it is safe to assume that these users are generally cognizant of and interested in the topic. However, there is a noticeable difference between the evolution pattern of the users' interests in this topic over time. Two of the users, namely `@teerasay` and `@WingsofCrystal`, are specifically interested in this topic starting from mid November to early December. However, the other user, `@ClaraListenspre` only becomes engaged with this topic in the last third of December. Therefore, although these users are essentially interested in the same topic but their interests are temporally distributed differently over time. This difference is quite important when deciding to cluster users into similar communities. Most existing community detection methods overlook the slight yet important aspect of temporal evolution when deciding on user communities. Assume the identified user communities for the users in Figure 1 were to be used for performing news recommendation. If all three users were placed in the same cluster, then `@teerasay` and `@WingsofCrystal` would receive news recommendation for the 'War in Afghanistan' topic in late December, a time at which these two users are no longer interested in the topic. On the other hand, `@ClaraListenspre` would also receive news recommendation on this topic around mid November, a time which again would not suit this user.

For this reason, temporal content-based user community detection aims at finding latent communities whose members share higher similarity with respect to topics of interest over *time*. As such, those users who share not only similar topical interests but also share similar temporal behavior are considered to be like-minded and hence members of the same community. In contrast, those users who are simply dissimilar in topics of interest or share similar topical interests

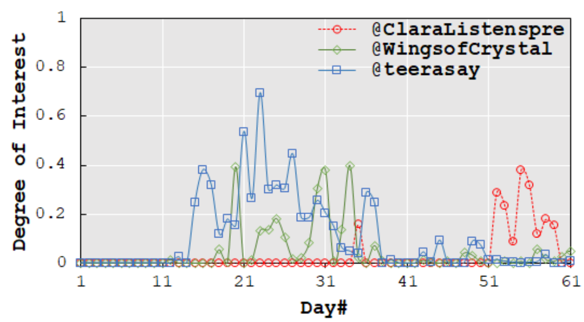

Figure 1: Different temporal behaviour of three Twitter users with respect to the *'War in Afghanistan'* topic.

but in different time intervals are not considered like-minded and need to end up in different communities.

More recently, a few temporal content-based user community detection methods have been introduced [12, 13]. Fani et al. [13] have proposed a multivariate timeseries representation of users in topic and time spaces. Hu et al. [12] have devised a unified probabilistic generative model of both topics and users. Our work in this paper moves beyond the work of Fani et al. and Hu et al. for identifying user communities by employing neural embeddings from user's temporal interests as well as their social network connections. More specifically, we extend our previous work [14] by interpolating information from two sources when identifying user communities: i) users' interests over time and ii) users' social network connections.

Earlier *non*-temporal user community detection methods have already shown improvement when incorporating social network structure (links) with topics of interest (content) compared to those in which links and content are used separately [15, 16]. However, to the best of our knowledge, all existing *temporal* user community detection methods are only content-based and none has studied the effect of social network structure and temporal evolution of user content simultaneously. Our experiments show that while social network structure is not a discriminative enough feature on its own for identifying high quality user communities, it does improve the quality of the identified user communities when effectively interpolated with content-based methods.

In order to simultaneously consider users' temporal content and their social network structure when identifying user communities, we embed both users' temporal interests and their social network structure into a dense vector representation using neural embedding mechanisms. The user embeddings, which are derived from two different information sources (modalities), i.e., *i*) temporal content-based embeddings based on users' topics of interest over time, and *ii*) network embeddings based on social network neighborhoods, are linearly interpolated to build a single final *multimodal* user embedding. The linear interpolation of two user embeddings at the embeddings level allows us to investigate how and to what extent users' dynamic topics of interest and/or users' social network structure contribute to the quality of the inferred user communities. We perform experiments on Twitter data and evaluate our work in two application scenarios: news recommendation and user prediction, to explore the impact of the different user embeddings and their interpolation.

In summary, the main contributions of this paper are as follows:

1. We propose a community detection method that considers users' topical interests and their temporal evolution in tandem by learning neural user representations, which embeds users in an embedding space where those users who have similar inclination towards similar topics in similar time intervals will be embedded close to each other.
2. We employ neural graph embedding techniques to embed information from users' social network structure into user representations.
3. We build a single set of multimodal embeddings from embeddings of temporal social content and social network structure through their linear interpolation in order to elucidate the contribution of users' temporal content on the one hand, and social network structure, on the other hand, for finding user communities.
4. We identify temporal content-based user communities which are topically, temporally and structurally cohesive, based on our multimodal user em-

beddings.

5. We demonstrate the performance of the various variations of our work in the context of news recommendation and user prediction, and compare them to the state of the art on a Twitter dataset.

It is worth noting that in our work we consider that the evolution of user-generated content is dynamic over time (temporal). In other words, users' interests can evolve over different time intervals. In contrast, we assume that the social network structure is static and remains stable over time. The main reason for this assumption is that the social network structure has a significantly lower pace of change compared to how fast content is generated over time and distributed across the social network [17]. The work in the current paper extends our previous work [14] by (1) additionally incorporating social network structure, (2) embedding it through a neural architecture and (3) systematically interpolating it with content-based representations. We integrate the neural embedding model of the social network structure into temporal content-based embeddings of users through a linear interpolation strategy. The exclusive experiments reported in the current paper highlight the fact that incorporating social network structure in user representations can achieve improved results over the state-of-the-art for identifying user communities.

The rest of the paper is organized as follows: we first present the related work in Section 2, then we continue with the problem definition and the details of our proposed approach in Sections 3 and 4, respectively. The experimental setup and evaluation is described in Section 5, followed by a study on performance of the proposed method under different settings in Section 6. Finally, Section 7 concludes the paper.

2. Related Work

While the literature on community detection is broad, the related works to this paper are largely centered around two areas: 1) user community detection and 2) neural representation learning (neural embeddings).

2.1. User Community Detection

A social network is essentially a platform for sharing content and making connections, as such, it can formally be viewed as a collection of user relationships and their content engagements. Based on these two main aspects in the social network, community detection methods have primarily focused on either detecting communities by considering the structure of the social network from the user relationships, the similarity of the content shared by the users or both of such information types in tandem. These methods are generally classified as (1) link-based (topological), (2) content-based (topical), and (3) hybrid (topological and topical) methods. In our review of the literature, we will cover these three classes of community detection methods. We will also explore a fourth group of methods that have recently examined *temporality* in community detection as a part of the literature review.

2.1.1. Link-based Methods

Link-based user community detection methods are primarily based on the *homophily* principle [18] where links between users are considered important clues for interest similarity and, as a result, densely connected groups of users imply a user community. In this line of work, the social network is modeled as a graph with nodes representing users and edges representing relationships or interactions. The primary principle considered in this line of work is connectedness which means that connections within each communities are dense and connections among different communities are relatively sparse. Methods that identify connectedness within a graph representation look for and identify subgraph structures such as cliques and components and consider those to represent user communities [9, 19, 10]. There are also other similar techniques which focus on a different optimization function whose objective is to minimize the number of links connecting users across communities and maximize the number of links between users within the same community. Approaches such as iterative bisection that iteratively divide the user set into smaller sub-communities and Girvan-Newman, which gradually removes edges from the network, are well known implementations of these techniques [20]. Other graph partitioning approaches include modularity optimization [8], spectral methods [21], max-flow min-cut theory [7], and conductance cut minimization [22]. Not all link-based methods perform well on large real-world networks that have many complex structural features such as sparsity, heavy tailed degree distributions and small diameters, among others. For a recent empirical comparison of these algorithms in practice, see [23, 24].

Nonetheless, link-based methods inherently fall short when the communities of interest need to take users' content similarity into account. This is mainly due to two reasons: *i*) there are many users on a social network that have similar interests but are not explicitly connected to each other; and, *ii*) an explicit social connection does not necessarily indicate user interest similarity but could be owing to sociological processes such as conformity, aspiration, and sociability or other factors such as friendship and kinship that do not necessarily point to inter-user interest similarity [25, 26]. There are also some special cases where link-based methods are not applicable like when the network is not available [27] or misleading, e.g., when links are fraudulent because of link-farmers (social capitalist) [28].

2.1.2. Content-based Methods

Given the abundance of user-generated content on online social networks, several researchers have utilized the similarity of social content to detect user communities. Most of these content-based methods have been inspired by latent Dirichlet allocation (LDA) [29] in one way or another and focused on probabilistic generative models based on textual content [30, 15]. For example, Zhou et al. [30] have modeled communities based on topics of interest through a community-user-topic generative process to identify user communities. In their work, communities follow a multinomial distribution over topics with Dirichlet prior where each user is posting about her topics of interest based on the conditional probability of a topic given each community. Another class of work attempts to transform the content-based community detection problem into a graph clustering problem [31, 32] where a user distance matrix is computed according to the similarity of their topical interests. The distance matrix is then used to identify clusters of users. The work by Peng et al. [31] is an instance of such techniques that focuses on identifying user communities on SINA Weibo by hierarchically clustering of users based on their relations to the predefined categories available on this social networking platform. Huang et al., [33] have built a pairwise similarity matrix for users based on the shortest path on the users' retweet graph. A spectral clustering algorithm has been used to find user communities in order to identify influential users and topical changes in the face of natural disasters. Barbieri et al. [27], however, proposed a *network-oblivious* probabilistic framework based on stochastic diffusion processes to identify like-minded users. They argue that users adopt topics of interest based on underlying diffusion processes over the unobserved social graph where the diffusion process itself is based on community-level influence.

2.1.3. Hybrid Content and Link-based Methods

Some researchers have argued that neither the consideration of link nor content alone is sufficient for identifying user communities. Sparse and noisy link information and irrelevant content could mislead the process of user community detection. Several approaches have been proposed that combine link and content information for community detection to achieve better performance. Most of these approaches adopt LDA as the generative model behind link, content, and community membership [34, 15, 35, 36, 37].

For instance, the work in [34] introduces a generative model which recursively defines a community through an integrated relationship between users' social relation and social topics. Simply put, in this model, communities are formed around multiple correlated topics where each topic can be reused in several different communities. Similarly, Sachan et al [15] also propose generative models for community detection but different from the work by Yin et al., they consider three types of information namely, topics, social connections and interaction types such as retweeting and replying. In both approaches, a user can be a member of different communities but with varying degrees of membership.

Differently from these two techniques, Yang et al. [16], however, have proposed a non-generative probabilistic model to find user communities in citation networks. They estimate the conditional probability that a user is cited given her popularity and her membership to a community according to her weighted content vector (topics of interest) so that modeling the absent links, as in generative models, is avoided. In addition to probabilistic models, some other approaches that combine link and content information include matrix factorization [38], kernel fusion [39] and graph union [40] for spectral clustering.

2.1.4. Temporal Analysis

While the methods introduced in the previous sections cover various data types such as social connections and users' content, they do not explicitly consider the temporal evolution of social networks when determining user communities. As a matter of fact, many of the content based and link based methods assume that the structure of the network and the topics discussed by the users remain stable over time, which can be a limiting assumption in practice.

More recently, there have been a few works that have considered time as an explicit dimension when identifying communities in social networks [12, 13, 14]. The work by Hu et al. [12] is among the pioneers to consider temporality through a generative process, which models how users and topics are related to each other and co-evolve over time. Their model learns a specific time-aware probability distribution known as the community-topic-time distribution addressing how communities and topics are associated with each other over time. While the work by Hu et al. is based on a generative process and extends topic modeling techniques, in our earlier work [13], we have addressed the same problem but from a time series analysis perspective. We model users based on a multidimensional time series representation where each of the time series depict to what extent the user has contributed to social topics in consecutive time intervals. This time series representation allows one to compute a user similarity matrix for the users based on the cross-correlation similarity of users' time series, which can then be effectively used to extract clusters of users. Both of these methods, including our proposed work and Hu et al's work, have shown to have superior performance compared to other non-temporal community detection methods on applications such as news recommendation and user prediction.

In our previous work [14], we propose a neural embedding approach to model the users' temporal contribution towards topics of interest by introducing the notion of similarity regions between users. These regions cover users who share not only similar topical interests but also similar temporal behavior. By considering the identified set of regions as a context, we train a neural network such that the probability of a user in a region be maximized given other users in the same region.

2.2. Neural Representation Learning

The notion of *distributional semantics* states that words that occur in similar contexts are semantically similar. Recent neural representation learning models [41, 42] approximate the semantics of a word with a dense low-dimensional vector (embeddings) so that the semantic similarity of words can be measured in terms of geometric distance between the respective vectors. The success of these methods has extended beyond computational linguistics to graph representation learning. Inspired by these works, methods such as node2vec [43] and deepwalk [44] employ a second order random walk to sample network neighborhoods in a graph and output vector representations (embeddings) that maximize the likelihood of preserving topological structure of each node neighborhood in the graph. While previous work used hand-engineered statistics like node degrees to extract network's structural information, graph representation learning employs a data-driven approach to automatically encode graph elements, nodes, edges, or even the entire graph, to a dense low-dimensional vector space. This not only saves time and effort in the feature engineering process, but also is agnostic to the downstream task. The embeddings can be easily fed into tasks such as user classification or link prediction. In user community detection, it offers an unsupervised way to encode homophily (Section 2.1.1) into a vector of real values so that its fusion with other information types such as content become effectively straightforward. More sophisticated methods based on deep autoencoders such as deep neural graph representations (DNGR) [45], structural deep network embeddings (SDNE) [46] have been also proposed to generate user embeddings.

So far most social graph embedding methods have been *amodal* which only rely on the social network graph elements and fail to leverage other heterogeneous information about the user during vector representation learning process. In contrast, author2vec [47], for instance, is bimodal which augments the social network with textual content to learn user embeddings. Author2vec includes content-info and link-info neural models. In the content-info model, given a user and a text, it predicts whether the given user has authored the text. In the link-info model, given two users, it predicts whether they are connected. There have also been work in the literature [48] the employ canonical correlation analysis for integrating different user representations. Recently, convolutional encoders such as the graph convolutional network (GCN) [49] and GraphSAGE [50] have been introduced, which are able to leverage user information (e.g., user profiles) and their social relations, simultaneously.

None of the proposed neural embeddings take the *time* dimension into consideration. Although Benton et al. [48] offer the opportunity to integrate different information types, it is not clear how to integrate temporality, which can be considered to be an aspect, rather than a new information type. In this paper, we propose to build multimodal user embeddings in order to incorporate users' temporal social content, and their social network neighbourhood into a single representation vector. We first propose a novel approach for building user embeddings where users' temporal and topical content are both taken into account. We then employ graph representation learning to encode information from users' social network neighbourhood into a feature vector in order to plug in homophily as well. Finally, based on a linearly weighted interpolation strategy, we integrate user embeddings from these two different modalities, i.e., *i*) temporal content-based embeddings based on topics of interest over time, and *ii*) network embeddings based on social network neighborhoods.

Beyond the works that have been reviewed in this section, there are abundant number of methods that propose application specific user embeddings for tasks such as sarcasm and irony identification [51], gender classification [52], or recommendation [53]. These works also fall short in effectively considering the notion of temporality when building the user representations.

3. Task Description

Within the context of an online social network, at least two different questions may be raised about user communities: 1) how to identify all user communities, and 2) given a user in the social network, what is the best community for the given user if a set of communities already exists. This paper is focused on addressing the former problem (1), known as user community detection; also referred to as community discovery or mining. In other words, our goal is to group users in separate non-overlapping groups. The identified user communities need to consist of user members that exhibit similar temporal behaviour towards similar content and be densely connected. Here, we provide a formal statement of the problem as follows:

Problem Definition. Given a set of users U, we aim to partition U into non-overlapping subsets in which each $u \in \mathsf{U}$ is only a member of one subset. More formally, $\mathsf{P} = \{\mathsf{C} : \mathsf{C} \subseteq \mathsf{U}, |\mathsf{C}| > 1\}$ such that $\forall \mathsf{C}_i, \mathsf{C}_{j \neq i} \in \mathsf{P} : \mathsf{C}_i \cap \mathsf{C}_j = \emptyset$. The objective of our work is to identify a configuration for P such that members of each C_i in P show highly similar temporal disposition with regards to active topics on the social network and high dissimilarity with members of any other $\mathsf{C}_{j \neq i} \in \mathsf{P}$.

4. The Proposed Approach

Having formally laid out the problem, we seek to find P through three pipelined phases: 1) temporal content-based and topological user representation learning (Sections 4.2 and 4.3 respectively), 2) interpolation of user embeddings

Algorithm 1 Overview of the proposed approach to find user communities

Inputs:

U, the set of users;

$D = (\mathsf{U}, \mathsf{M}, \mathrm{T})$, temporal social content;

$G = (\mathsf{U}, \mathsf{A})$, the social network;

Output:

$\mathsf{P} = \{\mathsf{C} : \mathsf{C} \subseteq \mathsf{U}, |\mathsf{C}| > 1\}$ such that $\forall \mathsf{C}_i, \mathsf{C}_{j \neq i} \in \mathsf{P} : \mathsf{C}_i \cap \mathsf{C}_j = \emptyset$

---

parallel_exec: //User representation learning - parallel execution

$\mathbf{W}_{\mathcal{D}} = f(D)$; //Temporal content-based user embeddings, § 4.2.

$\mathbf{W}_{\mathcal{G}} = g(G)$; //Link-based user embeddings, § 4.3.

$\mathbf{W} = h(\mathbf{W}_{\mathcal{D}}, \mathbf{W}_{\mathcal{G}})\{\textbf{return } \alpha \mathbf{W}_{\mathcal{D}} + (1 - \alpha)\mathbf{W}_{\mathcal{G}};\}$; //Interpolation, § 4.4.

$\mathsf{P} = \mathrm{Cluster}(\mathsf{U}, \mathbf{W})$ //User community detection, § 4.5.

(Section 4.4), and 3) user community detection (Section 4.5). Foremost, we provide an overview of this process after which the details of each step will be presented.

4.1. Overview

The overview of the approach discussed in this paper to find user communities is outlined in Algorithm 1. We define temporal social content as $D = (\mathsf{U}, \mathsf{M}, \mathrm{T})$ where U is the user set, M is the textual user-generated content corpus (e.g., tweets), and T is the time period broken down into time intervals. We define the social network structure as a directed graph $G = (\mathsf{U}, \mathsf{A})$ whose vertices are users in U and edges are ordered pairs of user elements such as $(u, v) \in \mathsf{A}$ indicating a social tie from u to v (e.g., u is following v). Herein, we use the terms 'graph' and 'network' interchangeably as well as the terms 'vertex', 'node', and 'user'.

Our proposed approach consists of creating user representations from two different information sources (modalities), i.e. 1) temporal content-based embeddings from temporal social content $D = (\mathsf{U}, \mathsf{M}, \mathrm{T})$, and 2) link-based embeddings from the social network structure $\mathsf{G} = (\mathsf{U}, \mathsf{A})$. On Line 2 of Algorithm 1, we learn user vector representations $\mathbf{W}_{\mathcal{D}}$ from users' content with the expectation that temporally like-minded users end up closer to each other in the vector space. To build this type of user embeddings, we first formally formulate what we mean by a like-minded pair of users with respect to social content only. Then, we propose a representation learning method, which preserves pairwise proximity of the users through maximizing the likelihood that two like-minded users stay close to each other in vector space. Likewise, on Line 3, we learn user vector representations $\mathbf{W}_{\mathcal{G}}$ but from users' social network neighbourhood with the assumption that similar users are those that are densely connected to each other due to homophily. We use unsupervised random-walk based graph representation learning to learn user representations such that geometric relationships in the learned vector space reflect the structure of the original social network. Learning vector representations from temporal social content and social network structure are independent and could be run in parallel (Line 1). These monomodal user representations are then linearly interpolated into a single consolidated multimodal representation on Line 4 tailored for the task of user community detection on Line 5.

4.2. Temporal Content-based User Embeddings

In order to learn temporal content-based neural embeddings ($\mathbf{W}_{\mathcal{D}}$) for social network users, we consider social content to be in the form of a triple $D = (\mathsf{U}, \mathsf{M}, \mathrm{T})$ where U is the set of users, M is the collection of content generated by U and T is the number of consecutive time intervals. We identify a set of topics Z from M over the T time intervals using a topic detection method (e.g., LDA [29]). Based on Z, we represent the temporal topic preferences of each user $u \in \mathsf{U}$ towards each topic $z \in \mathsf{Z}$ over time intervals $1 \leq t \leq \mathrm{T}$ as a timeseries $\mathbf{X}_{uz} = [x_{uz,1} .. x_{uz,\mathrm{T}}]$, which we refer to as the user's *topic preference timeseries*, where $x_{uz,t} \in \mathsf{R}^{[0,1]}$ indicates the preference of user u towards topic z at time interval t. The stacking of all users' topic preference timeseries will generate a cuboid $\mathbf{X} = \{x_{uz,t} : u \in \mathsf{U}, z \in \mathsf{Z}, 1 \leq t \leq \mathrm{T}\}$.

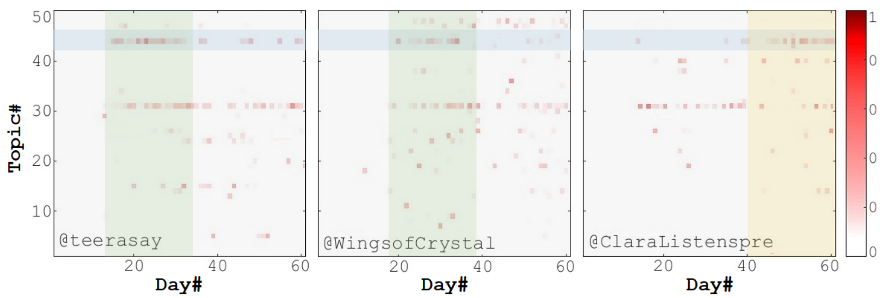

Figure 2: Topic preference timeseries for three sample Twitter users.

It is possible to visualize the topic preference time series of each user by projecting it onto a heatmap, which has been done in Figure 2 for the three sample users introduced earlier in Figure 1. In Figure 2, Topic 44 (highlighted horizontally with blue) represents the 'War in Afghanistan' topic while Topic 30 refers to the 'New Year' topic. As seen in the projection shown in Figure 2, all three users have shown consistent interest in Topic 30 and have started talking about the 'New Year' topic starting from late November. However, their temporal interest pattern with regards to Topic 44 is not as consistent and, as discussed earlier, while `@teerasay` and `@WingsofCrystal` are heavily engaged with this topic in November (as highlighted with the vertical green column), `@ClaraListenspre` only becomes involved with the topic in late December (specified with an orange column on the right most figure of Figure 2). The power of our multivariate time series representation is in its ability to accurately capture the temporal evolution of user interests towards the active topics of the social network.

To instantiate the topic preference timeseries, we need to find *i*) a set of topics Z that have been observed up until time interval T, and *ii*) each user's degree of preference at time interval t towards each topic $z \in \mathsf{Z}$, i.e., $x_{uz,t}$. We use Latent Dirichlet Allocation (LDA) [29] to extract both the topics available in the collection of users' content and the users' degrees of preference as suggested in [54, 55]. We concatenate all of the tweets posted by a given user at each time interval into a single document, the collection of which over all users and in all time intervals produces our document corpus. By applying the LDA topic modeling technique over this corpus, a set of topics Z is learnt such that each $z \in \mathsf{Z}$ is a multinomial distribution of terms denoting how much each term contributes to that topic. We infer each user's inclination towards the topics in Z at each time interval by inferring the distribution of topics over the document curated for that user in the given time interval through the concatenation of the users' tweets in that interval.

4.2.1. Temporal Context Model

In order to be able to learn neural embedding representations for the users, each user needs to be defined in the context of other users. Such context information for each user is not explicitly available. As such, the purpose of this section is to define context for each user that could then be used for neurally embedding the users. More specifically, in order to build user embeddings, we first formally formulate what we mean by a like-minded pair of users with respect to social content only within time. Then, we propose an embedding method which preserves pairwise like-minded proximity of the users through maximizing the likelihood that two like-minded users stay close to each other in vector space.

The premise of our approach is that the more two users share common interests in similar time intervals, the more similar these users would be and hence the likelihood of these users being in the same community should increase. As an example, let us consider the same three users that were introduced in Figures 1 and 2 earlier. Figure 3 shows a subset of the topic preference time series of these three users for a 10 day time period for a limited set of topics. An interesting observation is that while the visualization of the users' topic preference time series based on a heatmap in Figure 2 showed us that users `@teerasay` and `@WingsofCrystal` share similar temporal interests, which is different from `@ClaraListenspre`, it becomes clear that the actual degree of interest is not within the same range. For instance, even for the two users who are considered to be quite similar, their degree of interest for Topic 44 is 0.35 and 0.14, respectively, which are quite different. This shows that it would be quite difficult to identify users that not only have similar temporal trends but also similar degrees of interest. For this reason, we relax the similarity condition to allow for cells with similarity values within a range to be considered to be similar. The softened condition of similarity is referred to as condition of homogeneity. For the sake of clarifying the concept of condition of homogeneity, let us assume that any degree of interest below 0.1 is insignificant and can be ignored (shown in grey in Figure 3). Assuming the condition of homogeneity considers values above 0.1 to be similar, users `@teerasay` and `@WingsofCrystal` will now share four regions of similarity in Figure 3. This would not be possible without this relaxed condition. On the other hand, `@ClaraListenspre` still maintains its difference with the other two users with only one and zero regions of interests with the other two users. Based on the condition of homogeneity, we now consider `@teerasay` and `@WingsofCrystal` to be similar as they share the many similar regions and `@ClaraListenspre` to be distant from them.

The condition of homogeneity and the number of shared regions between users allows us to formally define an objective function for learning user embeddings. Our objective function will endeavor to place those users who share many regions of similarity close to each other and far away from those users who do not share any regions of similarity with them. Expressed more formally, the shared regions between two users act as a context for the users when they are embedded into a neural embedding space. For instance, the four shared regions for `@teerasay` and `@WingsofCrystal` act as context for each of the users and allows our embedding model to learn similar representations for these two users. In the following, we will propose a deterministic method for finding shared regions between any two users, which will be later used as context for learning user embedding representations. We first define the shared regions as follows:

Definition 1. ***Region of Like-mindedness.*** *Let us recall that the stacking of all users' topic preference timeseries is referred to as* $\boldsymbol{X}$*. A subspace of* $\boldsymbol{X}$*, such as* R*, is defined to be a region of like-mindedness iff (1) all the values in this subspace are equal with respect to a certain condition of homogeneity* c*; notationally,* $\forall x, x' \in R; c(x) = c(x')$ *and (2) it is maximal such that there exists*

		20	21	22	23	24	25	26	27	28	29	30
u_1	z_{40}			0.15					0.14			
	z_{41}											
	z_{42}											
	z_{43}	0.16	0.15	0.29								
	z_{44}	0.15	0.54	0.26	0.7	0.35	0.32	0.35	0.45	0.19	0.19	0.26
	z_{45}							0.18				
u_2	z_{40}	0.38		0.14	0.62				0.17	0.18	0.15	0.3
	z_{41}			0.14					0.17			
	z_{42}								0.17			
	z_{43}	0.25		0.14		0.92	0.52		0.17		0.35	
	z_{44}	0.39		0.14	0.13	0.14	0.18	0.17	0.17	0.29	0.83	0.34
	z_{45}				0.51	0.18	0.52	0.22	0.17	0.39	0.17	
u_3	z_{40}					0.36	0.21	0.75				
	z_{41}											
	z_{42}											
	z_{43}					0.13	0.29	0.75				
	z_{44}											
	z_{45}											

time interval t

Figure 3: Topic preference timeseries for three sample Twitter users in Figure 2 with $\{u_1 =$ `@teerasay`, $u_2 =$ `@WingsofCrystal`, $u_3 =$ `@ClaraListenspre`$\}$ $\{z_{40} .. z_{45}\}$ $t \in \{20 .. 30\}$. The values are *unnormalized* probabilities for every topic in each document, most of which are equal to the smoothing parameter alpha ($\alpha = \frac{5.0}{|Z|}$) in the LDA topic modeling method. Also, the values are rounded upward to two digit precision.

no other regions of like-mindedness such as R' such that R is subsumed by R'. The set of all regions of like-mindedness is called R.

We adopt a similar strategy to [56] to find the set of all regions of like-mindedness R in $\mathbf{X}$. First, we find R in user and topic dimensions at each time interval t. The output is two-dimensional (2-d) regions indexed by time interval $1 \leq t \leq$ T, i.e., R_t. Then, we merge each of the R_t from different time intervals to build the required R. We refer the interested reader for more technical details to Appendix A.

4.2.2. Temporal Content-based User Vector Representation

We approach the problem of learning user representations as a maximum likelihood (ML) problem through which similar users to a given user are identified based on the user's context. We define the context for each user to consist of all those users who have been observed with this user in similar region's of like-mindedness (R). As such, the more two users are seen in each others' contexts,

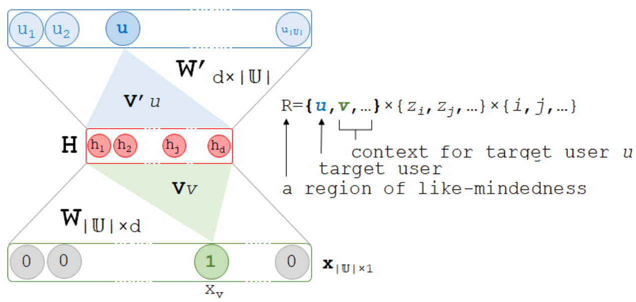

Figure 4: The neural network architecture to learn temporal content-based user vector representation.

the more likely it would be for them to be similar to each other. We adopt the continuous bag-of-word (CBOW) model from [41] to learn user representations.

Definition 2. ***Temporal Content-based User Embedding Objective.*** *Given the set of all regions of like-mindedness $\mathcal{R}$, the embedding function $f : \mathbb{U} \to \mathbb{R}^d$ maps each user $u \in \mathbb{U}$ onto a d-dimensional real space $[0,1]^d; d \ll |\mathbb{U}|$, such that the following objective is optimized:*

$$\arg\max_f \sum_{\mathrm{R}\in\mathcal{R}, u\in\mathrm{R}} \log \Pr(u|\mathrm{R} \setminus u) \tag{1}$$

In order to make the optimization tractable, we assume conditional independence for observing users in a region of like-mindedness. So,

$$\Pr(u|\mathrm{R} \setminus u) = \prod_{v\in\mathrm{R}\setminus u} \Pr(u|v) \tag{2}$$

We adopt the architecture shown in Figure 4 to learn user representations. It should be noted that the size of the hidden layer (d) will be the size of the user representation vectors. Furthermore, given the model learns to predict a user given its context, the size of the input and output layers is equivalent to the number of users. We use a one-hot encoding representation to refer to users in the input ($\mathbf{I}$) and output layers. The structure of the hidden layer neurons is linear $\mathbf{H} = \mathbf{W}_{\mathcal{D}}^{\top}\mathbf{I}$ where $\mathbf{W}_{\mathcal{D}}$ has a size of $|\mathbb{U}| \times d$ and is the input to the hidden layer. Similarly, the weights between the nodes in the hidden and output layers are denoted by $\mathbf{W}'_{\mathcal{D}}$ of size $d \times |\mathsf{U}|$. Also, we refer to a user v's corresponding row in $\mathbf{W}_{\mathcal{D}}$ as $\mathbf{V}_v$ The network performs user prediction given its context through a softmax function by approximating the likelihood of observing the target user v given some other user u observed together in at least one region of like-mindedness. This conditional probability is defined as follows:

$$\Pr(u|v) = \frac{\exp(\mathbf{V}'^{\top}_{u}\mathbf{H})}{\sum\limits_{w \in \mathsf{U}} \exp(\mathbf{V}'^{\top}_{w}\mathbf{H})} = \frac{\exp(\mathbf{V}'^{\top}_{u}\mathbf{V}_v)}{\sum\limits_{w \in \mathsf{U}} \exp(\mathbf{V}'^{\top}_{w}\mathbf{V}_v)} \tag{3}$$

Given the conditional independence assumption in Equation (2) and the above conditional probability in Equation (3), we can simplify Equation (1) as:

$$\arg\max_{f} \sum_{\mathrm{R} \in \mathcal{R}, u \in \mathrm{R}} \Big[\sum_{v \in \mathrm{R} \setminus u} \big[(\mathbf{V}'^{\top}_{u}\mathbf{V}_v) - \log \sum_{w \in \mathsf{U}} \exp(\mathbf{V}'^{\top}_{w}\mathbf{V}_v)\big]\Big] \tag{4}$$

However, this formulation is computationally intractable as its time complexity is proportional to the size of U. Morin and Bengio [57] have proposed hierarchical softmax to approximate the full softmax efficiently in practice. Accordingly, instead of a matrix, the hidden layer to output layer connection is a binary Huffman tree whose leaves are users. For each user u, there is a path $u_1, u_2, ..., u_{h(u)}$ of height $h(u)$ from the root, u_1, to her respective leaf, $u_{h(u)}$. This choice leads to speedup from $O(|\mathsf{U}|)$ to $O(log|\mathsf{U}|)$. Hierarchical softmax defines $\Pr(u|v)$ as follows:

$$\Pr(u|v) = \prod_{i=1}^{h(u)-1} s((-1)^{(u_{i+1} != child(u_i))} \cdot {\mathbf{V}'_{u_i}}^{\top}\mathbf{V}_v) \tag{5}$$

where $s(x)$ is the sigmoid function. ${\mathbf{V}'_{u_i}}^{\top}\mathbf{V}_v$ shows the similarity between the vector representation of user v and the internal user u_i. At each internal user u_i, if we choose the left (right) child as the correct u_{i+1} in the path from the root to the user's leaf, we have the probability $s((-1)^0 \cdot x) = s(x)$, else the right (left) child would result in $s((-1)^1 \cdot x) = s(-x)$ such that $s(x) + s(-x) = 1$. The intuition is that the more an output user u is similar with the ancestors of input user v, the higher the probability would be that they are the same.

Our neural network is trained using stochastic gradient descent and updates $\mathbf{W}_{\mathcal{D}}$ and $\mathbf{W}'_{\mathcal{D}}$ gradually via backpropagation. After the training converges, a pair of like-minded users $u, v \in \mathsf{U}$ will have highly similar vector representations, denoted by $\mathbf{V}_u$ and $\mathbf{V}_v$ in $\mathbf{W}_{\mathcal{D}}$ with respect to the temporal social content $\mathsf{D} = (\mathsf{U}, \mathsf{M}, \mathrm{T})$.

The next step of our work is to learn vector representations of users with respect to social network structure $\mathsf{G} = (\mathsf{U}, \mathsf{A})$, denoted by $\mathbf{W}_{\mathcal{G}}$. More specifically, we are interested in providing a concrete implementation for $g(\mathsf{G})$ on Line 3 of Algorithm 1.

4.3. Link-based User Embeddings

Given a social network structure in the form of a double $\mathsf{G} = (\mathsf{U}, \mathsf{A})$ where U is the set of users and A is the connections between the users, our objective in this section is to learn neural user representations based on the global position of a user in G and the structure of her local neighborhood. We employ an unsupervised representation learning method to encode this information into a low-dimensional dense feature vector in latent space such that the geometric relations in this latent space correspond to social connections (e.g., link or path) in G. Specifically, user embeddings are inferred by maximizing the probability of observing subsequent users in random walks of the graph conditioned on the source user. We formulate user embeddings learnt from the social network structure in a unified framework as follows.

4.3.1. Neighborhood Context Model

Based on the *homophily* principle, similar users tend to form ties in a social network [18]. As such, groups of densely connected users could be a sign of a user community. In the context of the social network structure, users would be considered to belong to similar communities, if they share similar neighborhoods and as such, are to be placed close to each other in the embedding space. The shared neighborhood, hence, presents a context with respect to the social network structure as opposed to the regions of like-mindedness in the temporal context model (Section 4.2.1) or co-occurrence context in word embeddings. There are different strategies for building a neighborhood for a user.

For instance, depth-first-search (DFS) and breadth-first-search (BFS) are two immediate, yet extremely biased ways to generate different samples of neighborhoods for a user. BFS favours *structural equivalence*, that is, those users who share similar structural roles such as hubs and are not necessarily connected and could be anywhere in the network, should be embedded closely together. Being more community aware, DFS in contrast, respects homophily and leads to similar (close) embeddings for densely connected users. In practice, online social networks exhibit mixture behaviors through which some parts show homophily while the other parts reflect structural equivalence. For this reason, stochastic sampling methods, such as random walk, have been introduced to randomly sample different neighborhoods of the same source user. Random walks are also computationally efficient in terms of both space and time [43]. As a result, we form network neighborhood of a user based on random walks in this work, which is formally defined as follows:

Definition 3. ***Network Neighborhood.*** *Network neighborhood of a given user $u \in \mathsf{U}$, denoted by N_u, is a set of random walks of length l rooted at u on a possibly infinite social network structure $\mathsf{G} = (\mathsf{U}, \mathsf{A})$ generated by a stochastic process with random variables $[x_{1:l}]$ such that $x_1 = u$ and x_l is a user chosen from the neighbors of x_{l-1} according to a probability distribution $Pr(x_l = w | x_{l-1} = v)$ if $(v, w) \in \mathsf{A}$ and 0 otherwise. The set of network neighborhoods for all users is denoted by N.*

While graph embedding methods such as deepwalk [44] use a pure (unbiased) random walk based on the uniform distribution, other methods [43] introduce parametric biased random walk to trade-off between breadth-first or depth-first searches to preserve community structure as well as structural equivalence between users. For instance, the work in [43] proposes second order random walk with two parameters p (return parameter) and q (in-out parameter) in

$\Pr(x_l = w | x_{l-1} = v)$ to bias the walk as follows:

$$\Pr(x_l = w | x_{l-1} = v) = \begin{cases} 1/p & \text{if } d(x_{l-2}, v) = 0 \\ 1 & \text{if } d(x_{l-2}, v) = 1 \\ 1/q & \text{if } d(x_{l-2}, v) = 2 \end{cases} \tag{6}$$

where $d(.,.)$ denotes distance of the shortest path between users in an unweighted graph. While higher p values favour exploration and avoid revisiting already seen users, higher q allows the search to obtain a local view and approximate BFS behavior. Unbiased random walks can be seen as a special case when $p = q = 1$.

4.3.2. Link-based User Vector Representation

Once network neighborhoods for all users have been obtained, we learn a user vector representation for each user by optimizing the conditional probability of observing users in the same walk as her. The process is similar to Section 4.2.2 as network neighborhoods can be seen as similar to regions of like-mindedness. To infer the user embeddings, we optimize the following embedding function:

Definition 4. ***Link-based User Embedding Objective.*** *Given the set of network neighborhoods* $\mathsf{N} = \bigcup_{u \in \mathsf{U}} \mathsf{N}_u$*, the embedding function* $g : \mathsf{U} \to \mathsf{R}^d$ *maps each user* $v \in \mathsf{U}$ *onto a* d*-dimensional real space* $[0,1]^d; d \ll |\mathsf{U}|$*, such that the following objective function is optimized, assuming conditional independence:*

$$\begin{aligned} \arg\max_g \sum_{\mathsf{N}_v \in \mathcal{N}} \log \Pr(\mathsf{N}_v \setminus v | v) &= \arg\max_g \sum_{\mathsf{N}_v \in \mathcal{N}} \log\Big(\prod_{u \in \mathsf{N}_v \setminus v} \Pr(u|v)\Big) \\ &= \arg\max_g \sum_{\mathsf{N}_v \in \mathcal{N}} \sum_{u \in \mathsf{N}_v \setminus v} \Pr(u|v) \end{aligned} \tag{7}$$

We use the same neural architecture as shown in Figure 4 but here, given user v, we predict observing users such as u from v's neighborhood, adopting skip-gram model from [41]. The hidden layer $\mathbf{H}$ is of size d, the input to hidden layer connections is represented by matrix $\mathbf{W}_{\mathcal{G}}$ of size $|\mathsf{U}| \times d$ with each row representing a vector for each user. The input layer $\mathbf{I}$ is a one-hot encoded vector and the hidden layer's neurons are all linear such that $\mathbf{H} = \mathbf{W}^\top \mathbf{I}$. Given a user v in the input layer, $\mathbf{H}$ is the transpose of v's corresponding row in $\mathbf{W}_{\mathcal{G}}$, denoted as $\mathbf{V}_v$. In the same way, the connections from the hidden layer to the output layer can be described by matrix $\mathbf{W}'_{\mathcal{G}}$ of size $d \times |\mathsf{U}|$. The softmax function approximates the probability of observing user u taken from N_v from the same random walk, i.e.,

$$\Pr(u|v) = \frac{\exp(\mathbf{V}'^{\top}_u \mathbf{H})}{\sum\limits_{w \in \mathsf{U}} \exp(\mathbf{V}'^{\top}_w \mathbf{H})} = \frac{\exp(\mathbf{V}'^{\top}_u \mathbf{V}_v)}{\sum\limits_{w \in \mathsf{U}} \exp(\mathbf{V}'^{\top}_w \mathbf{V}_v)} \tag{8}$$

where $\mathbf{V}'_u$ is u's corresponding column of matrix $\mathbf{W}'_{\mathcal{G}}$. However, calculating the normalization factor in the denominator is not feasible. Hierarchical softmax and negative sampling are two promising alternatives to accelerate the computation. Stochastic gradient descent is used to train the neural network and the derivatives are estimated using backpropagation. Users' vector representations with respect to social network structure $\mathsf{G} = (\mathsf{U}, \mathsf{A})$ are vectors of $\mathbf{W}_{\mathcal{G}}$.

4.4. Embeddings Interpolation

Having learnt two different user vector representations of users from the temporal social content $\mathsf{D} = (\mathsf{U}, \mathsf{M}, \mathrm{T})$ and the social network structure $\mathsf{G} = (\mathsf{U}, \mathsf{A})$, denoted by $\mathbf{W}_{\mathcal{D}}$ and $\mathbf{W}_{\mathcal{G}}$, respectively, the next step is to integrate them into a single vector representation, denoted as $\mathbf{W}$, by an interpolation function $h(\mathbf{W}_{\mathcal{D}}, \mathbf{W}_{\mathcal{G}})$ defined on Line 4 of Algorithm 1. We adopt a linear weighting mechanism to interpolate the embeddings mined from the social network structure and temporal social content. Formally,

$$h(\mathbf{W}_{\mathcal{D}}, \mathbf{W}_{\mathcal{G}}) = \alpha \mathbf{W}_{\mathcal{D}} + (1 - \alpha) \mathbf{W}_{\mathcal{G}} \tag{9}$$

where α denotes a weighting coefficient to interpolate between temporal content and social network structure in the final user vector representation. For instance, if $\alpha = 0$, the interpolated embeddings lead to the conventional link-based user community detection on the one extreme. On the other extreme, it will solely rely on temporal content if $\alpha = 1$ and becomes a pure temporal content-based method. The effect of embedding interpolation to the overall performance of user community detection is evaluated by choosing $\alpha \in \mathsf{R}^{[0,1]}$. Although simple, linear weighting is uninformed, easy to implement, interpretable, and could achieve competitive performance across a wide span of different data types and domains [58, 59, 60]

4.5. Community Detection

Given the interpolated user vector representation $\mathbf{W} = h(\mathbf{W}_{\mathcal{D}}, \mathbf{W}_{\mathcal{G}})$, we identify communities of users through graph-based partitioning heuristics. We represent users and their pairwise distances through a weighted undirected graph. Precisely, let $\mathrm{G} = (\mathsf{U}, \mathsf{E}, w)$ be a weighted user graph such that $\mathsf{E} = \{e_{u,v} : \forall u, v \in \mathsf{U}\}$ and the weight function $w : \mathsf{E} \to \mathsf{R}^{[0,1]}$ defined as $w(e_{u,v})$ be the dot-product, or angle, between u and v's embeddings in $\mathbf{W}$. It is possible to employ a graph partitioning heuristic to extract clusters of users that form latent communities. We leverage the Louvain Method (LM) [8] as it *i*) can be applied to weighted graphs, *ii*) does not require *a priori* knowledge of the number of partitions, and *iii*) has an efficient linear time complexity for the problem of graph partitioning. As a result of the application of LM, a set of subgraphs such as $\mathrm{G}[\mathsf{C}]$ are induced where the edges in each subgraph have both ends in the same subgraph. The collection of these subgraphs form the set of user communities P desired in the problem definition presented in Section 3.

5. Performance Evaluation

Our work in this paper has been based on three components: 1) learning neural embeddings based on users' temporal social content; 2) learning neural embeddings based on users' social network structure; and 3) interpolating these two distinct neural embeddings to form a multimodal neural embedding-based user representation. As such in this section, we seek to answer four research questions that would provide insight into the role of each embedding type as well as the impact of their interpolation on the quality of the identified communities. The four research questions (RQ) are formulated as follows:

RQ1. Does the consideration of temporal evolution of content lead to higher quality communities compared to when time is overlooked (temporal vs. non-temporal content-based methods)?

RQ2. Does the incorporation of time into users' neural representations lead to higher quality communities compared to when time is incorporated as a component into a generative process (neural vs. probabilistic temporal content-based methods)?

RQ3. Do temporal content-based methods lead to higher quality communities compared to link-based methods?

RQ4. Do link-based and temporal content-based methods have synergistic impact on each other and reinforce the quality of the identified communities when applied in tandem?

5.1. Dataset and Experimental Setup

In our experiments, we use a publicly available Twitter dataset collected and published by Abel et al. [61]. It consists of 2,948,742 tweets posted by 135,731 unique users between November 1 and December 31, 2010. In addition to its text, each tweet includes a user id and a timestamp. Figure 5 depicts the distributions of different types of tweets. The whole two months time period is sampled on a daily basis, i.e., T = 61 days. Additionally, we collected the followership networks of the users using Twitter api. We provide the implementation details and the setup of each step in our approach in the following.

Finding topics (Section 4.2). Extracting topics from tweets suffers from the sparsity problem when topic modeling methods such as LDA are used [62]. Some authors have addressed this issue by extending the context of tweets with knowledge graph entities as suggested in [63, 64]. We annotate each tweet with knowledge graph entities derived from Wikipedia to obtain better topics. As an example, a semantic annotator tool would be able to identify and extract entity links for a tweet such as *'The war in Afghanistan is 18, older than the new wave of Marine recruits fighting it'* and connect

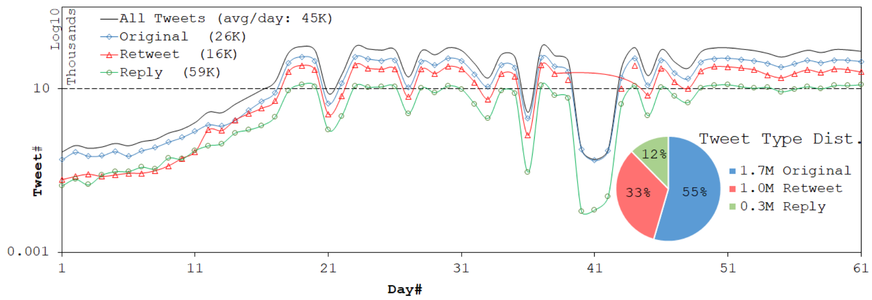

Figure 5: Temporal distribution of different types of tweets in Abel et al. [61]'s dataset from November till end of December 2010.

them with Wikipedia entries including 'War_in_Afghanistan_(2001present)', 'United_States_Marine_Corps', 'Military_recruitment' and 'Combat'. The reason we use these entities instead of the explicitly observed terms in the tweet is primarily because tweets are quite noisy and can abundantly consist of abbreviated or slang terms, which would not necessarily perform well given the sparsity of their mentions across different users. Petkos et al. [65] have argued that adopting the entity representation of the tweet would provide a more meaningful representation compared to n-gram-based term representations. We have annotated each tweet in our corpus using the TAGME RESTful API[2], which produced 350,731 unique entities.

In order to find topics of interest (Z) in our dataset, we have applied MALLET[3] for LDA to discover topics. LDA-based approaches to topic detection need *a priori* knowledge for the number of topics. The number of appropriate topics for this dataset has been already investigated in previous work and determined to be 50 [13]. We populate the topic preference timeseries for all users on a daily basis, i.e., T = 61 days, and screen out values less than 0.1. This threshold is equal to Dirichlet prior ($\alpha = \frac{5.0}{|\mathsf{Z}|}$) for topic distribution over a document in LDA. Thereafter, the condition for homogeneity c is set such that

[2]services.d4science.org/web/tagme/documentation
[3]mallet.cs.umass.edu/topics.php

the difference of values falls in the range [0, 0.1).

Topic modelling method and condition of homogeneity are two main parameters of our proposed method whose impacts are studied in a separate section (Section 6).

Learning temporal user vector representation (Section 4.2). We adopt the implementation of triCluster[4] [56] to find the regions of like-mindedness in users' topic preference timeseries. We proceed to extend CBOW architecture in Gensim[5] to learn user vector representations. The training phase uses a learning rate of 0.025 and in each epoch we decrease it by 0.002 for 200 epochs. The window size for the representation learning process is set to 2. We perform the experiments on different vector sizes of $d = \{100, 200, ..., 500\}$.

Learning link-based user vector representation (Section 4.3). In order to infer the user embeddings from the social network structure $\mathsf{G} = (\mathsf{U}, \mathsf{A})$ whose vertices are users U and edges are ordered pairs of user elements such as $(u, v) \in \mathsf{A}$, we use the formulation presented in Section 4.3 owing to its scalability ($O(|\mathsf{U}|)$) and unsupervised representation learning as opposed to more sophisticated neural-based graph embedding techniques such as deep neural graph representations (DNGR) [45] and structural deep network embeddings (SDNE) [46] with higher time complexity ($O(|\mathsf{U}|^2)$ and $O(|\mathsf{U}||\mathsf{A}|)$, respectively). Graph convolutional networks (GCN) [49] with running complexity of $O(|\mathsf{A}|)$ and its variations [66, 67] are the state of the art in inductive tasks, i.e., they are able to generalize previously unseen users which is crucial in evolving social networks. In our work, however, we assume that the social network structure G remains stationary and, hence, employing GCN-based methods does not add much value to our experiments.

We created 10 random walks of length $l \in \{40, 80\}$ for each user and the window size for the training process is set to $\{5, 10\}$ while the learning rate and the number of epoches are set to 0.002 and 200, respectively. The return (p)

[4] www.cs.rpi.edu/ zaki/software/TriCluster.tar.gz

[5] radimrehurek.com/gensim/models/word2vec.html

and in-out (q) parameters are set to a default value 1.

Interpolating embeddings (Section 4.4). We performed linear interpolation of temporal content-based and link-based user vector representation according to Equation 9 with an increasing values of $\alpha \in \mathbb{R}^{[0,1]}$.

Detecting user communities (Section 4.5). We apply the Louvain Method (LM) with resolution parameter 0.1 using Pajek[6] to identify subgraphs. The output subgraphs are considered to be the user communities $\mathbb{P}$ which we sought to find in Section 3.

5.2. Baselines

In our work and in order to answer the four research questions, we systematically compare the following baseline methods with the neural embedding methods proposed in this paper:

Non-temporal Content-based Community Detection (LDA-CD). We build non-temporal content-based communities over the set of users. We project daily topic preference timeseries of each user to the topic space by aggregating the values over the whole time period. Then, we calculate the topic-based similarity of users based on the cosine similarity of their corresponding topic vectors. Finally, we create a weighted graph over the users and their pairwise similarity and apply LM to find communities.

Fani et al. [13]. This baseline is based on the representation of users as a multivariate time series where each data point is a representative of the intensity of the user's interest in a given topic at a specific time interval. We train the LDA topic modeling technique to identify $|\mathbb{Z}|$=50 topics over a 61 day time intervals. For computing user similarities, we use two dimensional cross-correlation as proposed by the authors implemented in MATLAB and use the implementation of the Louvain method from Pajek for graph clustering.

Hu et al. [12]. This baseline is a probabilistic generative process that considers the temporal evolution of users' when identifying user communities.

[6] vlado.fmf.uni-lj.si/pub/networks/pajek/

Unlike the work by Fani et al., this method probabilistically assigns each user to more than one community, as such, we consider the community with the highest probability to be the community for each user. Similar to the previous baseline, we set the number of topics to 50 and evaluate the approach with differing number of communities ranging in 5, 10, ..., 30. We set the number of epochs for this approach to 1,000.

Temporal Content-based Community Detection (LDA-TCD). This is a temporal content-based method based on temporal user embeddings, proposed in Section 4.2, that does not consider the network structure.

Link-based (N2V-CD). This is a link-based method based on link-based user vector representations, proposed in Section 4.3, which does not consider user content.

Multimodal Community Detection (TCD(α)). This baseline interpolates temporal content-based embeddings with the link-based ones based on Equation 9, proposed in Section 4.4, where $\alpha \in \mathbb{R}^{[0,1]}$. N2V-CD could be considered as a variation of this method where $\alpha = 0$ to filter out temporal content-based embeddings. Also, LDA-TCD could be considered as a variation of this method when $\alpha = 1$ to filter out link-based embeddings.

5.3. Evaluation Methodology and Gold Standard

Contrary to small real-world social networks or synthetic ones, true gold standard user communities are not available in most cases for real world applications [68]. As such, well-defined quality measures such as Rand index, Jaccard index, or normalized mutual information (NMI) that require comparison to the gold standard cannot be used for evaluation. On the other hand and in the absence of a golden standard, quality functions such as modularity [20] are not helpful either since they are based on the explicit links between the users (structural). In our approach and the baselines, the links between the users are inferred through a learning process and are not always explicit. For instance, a near perfect method may result in a low modularity because graph edges are sparse and do not form densely connected user sets. Conversely, a weak method may connect topically dissimilar users together forming communities of users that do not share similar interests but result in a high modularity. So, the communities that achieve high structural quality in an inferred similarity graph are not necessarily optimal [69].

Fortunately, the performance of community detection methods can be measured through observations made at the application level, as suggested in [68, 69]. In these evaluation strategies, a user community detection method is considered to have *better quality* iff its output communities improve an underlying application. We deploy two applications, namely *news recommendation* and *user prediction*. By using these applications, we explore whether and which community detection method is able to provide stronger performance compared to the other state of the art community detection techniques and hence systematically answer the four research questions.

As the first step, we curate a gold standard dataset, which consists of the set of news articles that have been mentioned in the users' tweets. The reason we collect such a gold standard is because it can be safely assumed that users would only post links to news articles if they are interested in the topic of that news. Given our work is based on entity mentions in tweets, we also semantically annotate the news articles that have been collected in the gold standard dataset. Each entry in the gold standard can be viewed as a triple (u, a, t), which refers to user u posting article a at time interval t. Formally, gold standard is defined as $\mathsf{G} = \{(u, a, t) : u \in \mathsf{U}, a \in \mathsf{D}, 1 \leq t \leq \mathrm{T} = 61\}$ where U and A are the set of users and news articles, respectively. In our experiments, the gold standard consisted of 25,756 triples derived from 3,468 articles shared by 1,922 users. To avoid leakage, tweets which include a URL in the golden standard have been removed from training set. It is worth noting that almost half of tweets in our dataset include at least one URL, precisely 1,437,713 out of 2,948,742 tweets with 787,680 unique URL, among which we could only crawl 3,468 news articles to build the gold standard. This leads to removing 13,742 tweets and left 2,935,000 tweets for training purpose.

5.3.1. News Recommendation

Our first set of experiments rely on the assumption that an accurate clustering of users into communities would place those users who have similar topical interest evolution over time next to each other in the same community. As such, recommending news articles to the users of the same community should be possible and effective due to the similarity between user interests. Based on the gold standard, an effective recommendation for a user would be one that has been observed as one of the triples $(u, a, t) \in \mathsf{G}$. In order to make recommendations based on the identified communities, we perform the following two steps:

1. We consider each identified community separately in every time interval $t(1 \leq t \leq \mathrm{T} = 61)$ and compute the selected community's overall topic of interest at that time. The overall topics of interest for a community is calculated as the sum of topic preference time series of all users in that community. More formally, it is computed as $\sum_{u \in \mathsf{C}} x_{uz,t}$. All news articles in the gold standard are ranked descendingly based on their similarity to the overall topic of interest for the community in each time interval.
2. Each user member of a community is recommended the ranked list of news articles that are assigned to the community.

The news recommendation application will perform best when the users that are placed within the same community exhibit the same temporal topical interests and hence are interested in similar news articles at each time interval; therefore, it is a suitable extrinsic evaluation method to measure how well the community detection method has been able to effectively partition users into different communities based on their temporal interests.

Metrics. We evaluate the ranked list of news articles for recommendation by standard information retrieval metrics: Precision at rank k (P_k), Mean Reciprocal Rank (MRR), and Success at rank k (S_k). P_k is the proportion of relevant news articles in the top-k recommended items:

$$\mathrm{P}_k = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{tp_u}{k} \tag{10}$$

where tp_u (true positive) is the number of relevant news articles for user u in her top-k rank list of recommendation. MRR is the inverse of the first position that a correct item occurs within the ranked list,

$$\mathrm{MRR} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{rank_u} \tag{11}$$

where $rank_u$ refers to the rank position of the first relevant news article for the user u. S_k shows the probability that at least one correct item occurs within the top-k items of the ranked list:

$$\mathrm{S}_k = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} (rank_u \leq k) \tag{12}$$

In case $k = 1$, S_1 would be equal to P_1.

Results. We begin by considering research question **RQ1**, i.e., whether the consideration of time plays a role in the quality of the identified communities or not. Figure 6 summarizes the results for news recommendation in terms of different information retrieval metrics. From the figure, it is evident that all the methods that consider time outperform the state of the art non-temporal baseline (LDA-CD). This shows that considering users' temporal behaviour is an influential contributor to the identification of high quality user communities in the context of news recommendation. Furthermore and with regards to **RQ2**, i.e., whether the explicit embedding of time within users' vector representation lead to higher quality communities compared to when time is incorporated into a generative process, we compare the temporal content-based baselines, namely Fani et al. [13], and Hu et al. [12], with LDA-TCD in which only temporal user vector representations has been utilized in Figure 6. As shown, LDA-TCD achieves better performance compared with the temporal approaches proposed by Hu et al. and Fani et al. for different dimension sizes. Specifically, the result shows that LCD-TCD with $d = 300$ is the best and Hu et al. is the runner

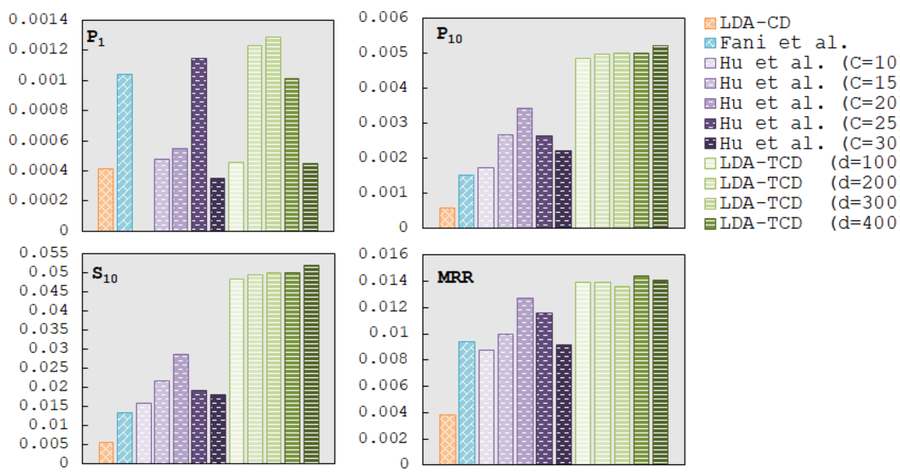

Figure 6: The performance of temporal and non-temporal content-based community detection methods in terms of information retrieval metrics in the context of news recommendation.

up. We attribute the better performance of LDA-TCD to the fact that the embedding function preserves both topical and temporal proximity of users more effectively and, consequently, the extracted user communities capture temporal content-based similarity of users more coherently than the other two baselines. This demonstrates the effectiveness of explicitly embedding time into user vector representations. Based on the results in Figure 6 and in response to **RQ2**, we conclude that the explicit embedding of time in user vector representations leads to higher quality user communities compared to when time is incorporated as a component in a generative process.

In order to answer research question **RQ3**, i.e., whether temporal content-based user community detection methods show better performance compared to link-based methods, we compare the quality of the output communities in Figure 7. As seen, linked-based methods (N2V-CD) show their best performance with $d = 300$ and a random walk length $l = 80$ but still perform worse than the poorest version of LDA-TCD with $d = 100$. As an example, all the variations of N2V-CD produce *zero* in terms of P_1. This points to the fact that link-based methods produce lower quality communities compared to temporal content-

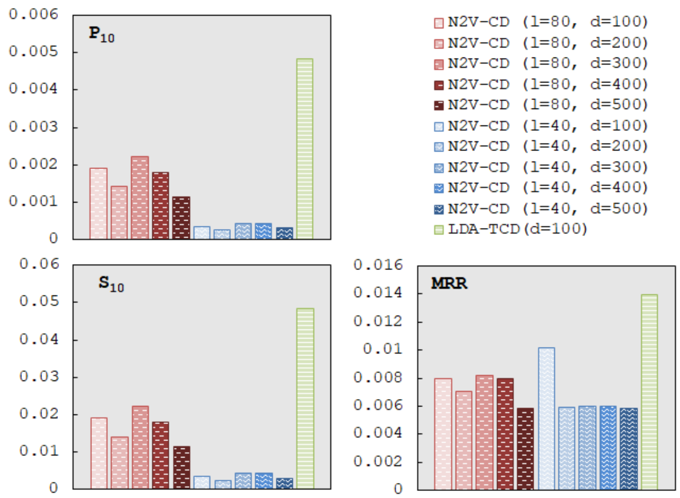

Figure 7: The performance of link-based community detection baseline (N2V-CD) vs. worst case of LDA-TCD ($d = 100$) in terms of information retrieval metrics in the context of news recommendation. All N2V-CD variations had a performance of *zero* in terms of P_1.

based counterparts.

In order to answer research question **RQ4**, i.e., whether link-based and temporal content-based community detection methods have synergistic effect on each other, we use TCD(α) in which the user vector representation from temporal social content is interpolated with link-based ones. As both types of user representation yield best results for user communities at $d = 300$, i.e., LDA-TCD ($d = 300$) and N2V-CD ($l = 80, d = 300$), we investigate the effect of social structure in temporal user community detection only for user vector representation of size $d = 300$ in TCD(α). Figure 8 shows the results for decreasing values of α in order to show the impact of link-based methods on improving the quality of content-based methods. As shown, we start with $\alpha = 1$ where there is no link-based user vector representation involved and the representation is essentially equivalent to LDA-TCD. As we gradually put more weight on the link-based user vector representation, the results improve up to an extremum, which happens at $\alpha = 0.6$. This demonstrates the fact that the link-based user representation is helping with user community detection and identifying user

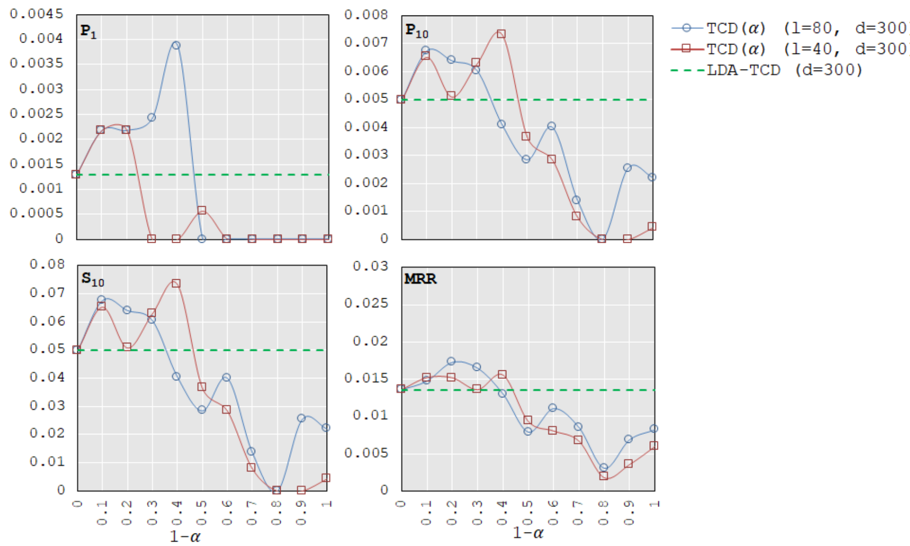

Figure 8: The performance of user communities through linear interpolation of temporal content-based and link-based user vector representations of size $d = 300$ in the context of news recommendation.

relationships that cannot be otherwise derived based solely on user content. However, the impact of link-based user embeddings need to be controlled as the increase in the weight of the link-based user representation beyond $\alpha = 0.6$ leads to declining community quality.

5.3.2. User Prediction

We perform a second set of experiments based on the user prediction application. Given the gold standard G and the user communities P, this time the goal is to predict which users posted a news article a at time interval t. To do so, we find the closest community to the news article in terms of topics of interest at time interval t. This is done based on the cosine similarity of the community's overall topics of interest at time t and news article a. Then, the members of the community would constitute the predicted users. The logic behind why this approach helps us qualify the output communities of the different approaches are the same as the news recommender application. However, while the performance of the news application is evaluated based on information retrieval metrics, the user prediction application is evaluated based on *classification* metrics.

Metrics. We adopt three standard classification metrics, i.e., Precision, Recall, and F-measure, to report user prediction performance. Precision is the probability that a predicted poster of a news article is the actual poster of the article:

$$\text{Precision} = \frac{tp}{(tp + fp)} \tag{13}$$

where tp is the true positive count, i.e., the number of users correctly assigned to the news article and fp is the false positive count, i.e., the number of users assigned incorrectly. Recall, or hit rate, is the probability that a true poster of a news article has been correctly assigned to the posted news article:

$$\text{Recall} = \frac{tp}{(tp + fn)} \tag{14}$$

where fn is the false negative count, i.e., the number of actual posters that have not been assigned to their posted news articles. F-measure is the harmonic mean of Recall and Precision and is defined as:

$$\text{F-measure} = 2 \quad \frac{\text{Precision} \quad \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

Results. Similar to the news recommendation task, we seek to answer the four research questions but in the context of the user prediction application. To answer research question **RQ1**, we summarize the performance of user prediction for temporal baselines in terms of classification metrics in Figure 9. As shown, user communities identified by the non-temporal content-based baseline (LDA-CD) have a lower quality compared to the temporal baselines. While non-temporal communities do consider the topics of news articles, they fail to take *time* into account and fall short in identifying changes in user interests. While a user may have had an interest in a certain topic in the previous time intervals, she may have lost interest in that topic over time and therefore naturally be much less likely to post about that topic as time passes. As it turns out in our experiments, non-temporal community detection methods were not able to identify this transition and hence predict the same user as the poster of a news article throughout different time intervals. This will result in many false

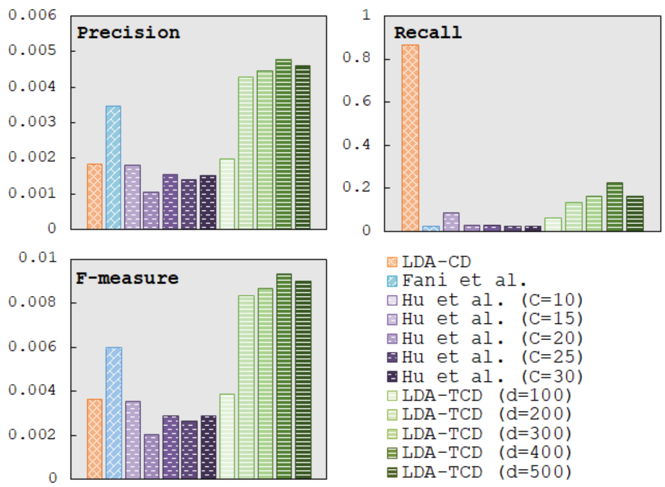

Figure 9: The performance of temporal and non-temporal content-based community detection methods in terms of classification metrics in the context of user prediction.

positives, leading to a poor Precision. In terms of Recall, however, LDA-CD competes with all temporal baselines. The reason for such high Recall can be attributed to the lower number of communities in this method. The lower the number of communities is, the higher the Recall of the method would be. In other words, if we only have one community that includes all users, Recall would be *one*. Overall, F-measure shows a higher quality for communities identified based on temporal approaches compared to non-temporal baselines.

Based on Figure 9, we are also able to answer **RQ2**. As shown, LDA-TCD outperforms other baselines in all metrics (except for $d = 100$). This reinforces the fact that when time is explicitly embedded in the user representations that it will lead to higher quality communities compared to representations that incorporate time within a generative process.

With respect to **RQ3**, Figure 10 shows that the temporal content-based user community detection methods outperform link-based methods. Specifically, the best link-based baseline (N2V-CD with $d = 300$ and random walk length $l = 80$) performs worse than the poorest version of LDA-TCD with $d = 100$. This reinforces our findings in the news prediction application that link-based methods

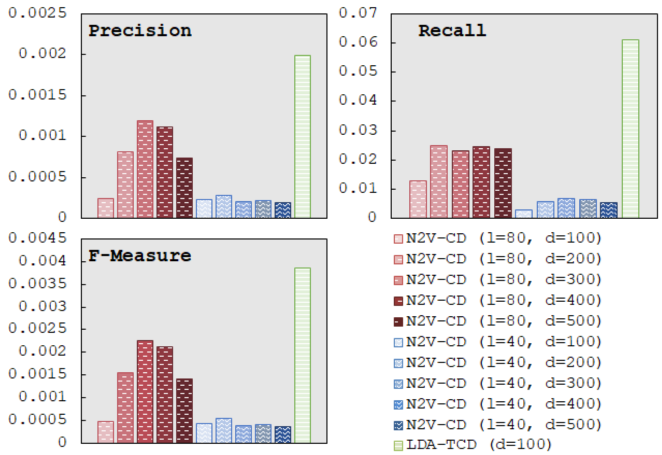

Figure 10: The performance of link-based community detection baseline vs. worst case of LDA-TCD ($d = 100$) in terms of classification metrics in the context of user prediction.

produce lower quality communities compared to content-based baselines.

In order to answer research question **RQ4** with regards to the synergistic impact of content-based and link-based user embeddings, similar to the new prediction application, we employ TCD(α) baseline with embedding dimension size of $d = 300$. Figure 11 shows the results for decreasing values of α. The left corner of each diagram in Figure 11 represents the performance of LDA-TCD due to $\alpha = 1$ and as such no link-based user vector representation is involved. As seen, the gradual increase in the weight of the link-based user representation leads to improved performance up to $\alpha = 0.5$ and 0.6 for $l = 80$ and $l = 40$, respectively. However, we observe declining performance as α decreases till the end when TCD(α) becomes pure link-based N2V-CD method at $\alpha = 0$. This demonstrates the fact that while link-based user representations alone do not produce high quality user communities, they can help improve the performance of content-based methods if interpolated effectively.

5.3.3. Findings

Based on our experiments on the news recommendation and user prediction tasks, we can summarize our findings with regards to the four research questions

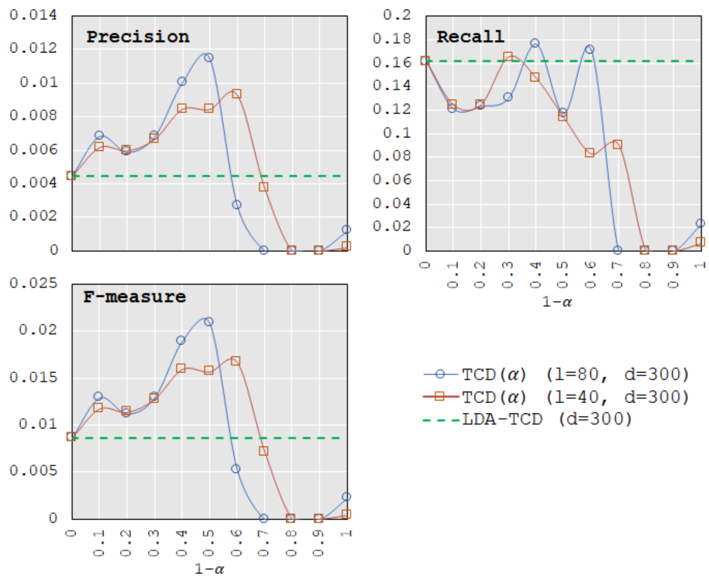

Figure 11: The quality of the identified user communities as a results of the linear interpolation of link-based and temporal content-based user vector representation in TCD(α) in the context of user prediction.

as follows:

1. We find that the consideration of temporal evolution of user-generated content is key in finding effective user communities. Our observations show that the incorporation of time in the user representations leads to higher quality user communities compared to when time is not considered.
2. Further, we find that the neural embedding of time into the user representation leads to higher quality communities compared to when time is included as a part of a generative process.
3. We observed that the communities identified through link-based methods are poorer compared to when temporal content-based methods are employed.
4. Finally, we find that while link-based methods show poorer performance compared to temporal content-based methods, they can still have synergistic impact on the performance of temporal content-based methods. In

other words, the interpolation of link-based and temporal content-based methods lead to higher quality user communities.

In summary, we conclude that when embeddings learnt based on temporal content-based methods are interpolated with the embeddings learnt from link-based community detection methods, they result in the highest quality communities as shown within the context of news recommendation and user prediction tasks. The findings have been evaluated from both the perspective of information retrieval and classification metrics.

6. Performance of Model Variations

In this section, we aim to study the impact of the choice of the topic modeling method and a variation of the condition of homogeneity (c) on the performance of our proposed approach.

Topic modeling method. While the experiments reported in the previous section were based on the standard LDA method, there have been other topic modeling methods in the literature that are designed specifically for short textual content such as tweets. It would be appropriate to understand whether the choice of the topic modeling technique has any impact on the performance of our proposed method. As such, we repeat our experiments with two most recent alternatives for topic modeling over short textual content. These two methods include the word network topic model (WNTM) [70] and biterm topic model (BTM) [71] in addition to LDA. The number of topics $|Z|$, alpha and beta priors are set to 50, $\frac{5.0}{|Z|}$, and 0.01, respectively and models were trained for 1,000 iterations.

Condition of Homogeneity (c). As we show in Figure 3, it is very unlikely that two users have the exact same probability value for a given topic in the same time interval. As such, we have introduced the condition of homogeneity to relax the condition for matching users with each other in a given time interval over some topic. One option (c_1) for defining the condition of homogeneity is to allow for slight variations between topic contributions by different users.

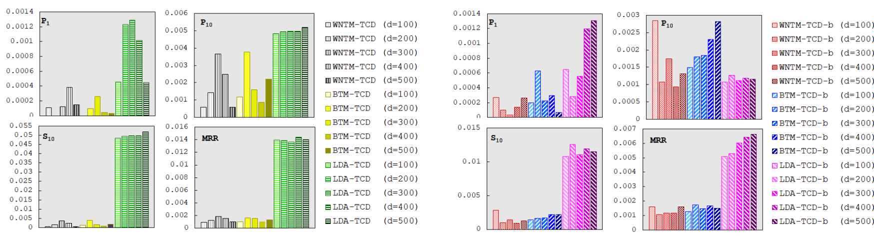

Figure 12: The effect of topic modeling and condition of homogeneity on our proposed method in the context of the news recommendation application.

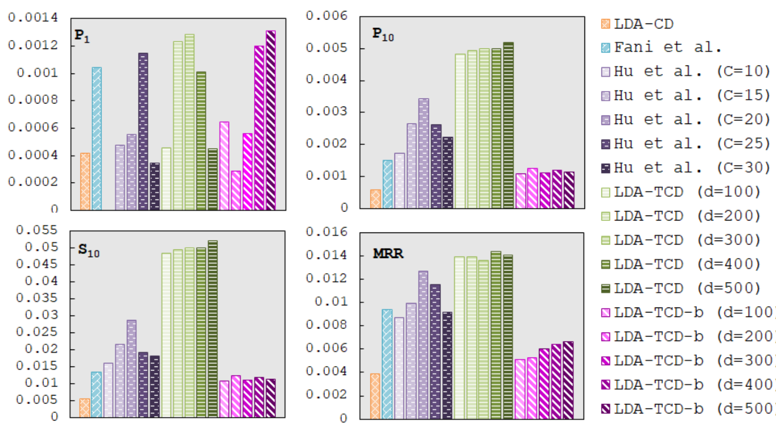

Figure 13: The effect of condition of homogeneity on our proposed method using LDA in the context of news recommendation application.

For instance, we could allow the difference to be between a certain range, e.g., [0, 0.1), which is the strategy that we have adopted in the previous set of experiments reported earlier. It is alternatively possible to define the condition of homogeneity (c_2) in a way that two values would be considered similar if they both have a value higher than a given threshold, e.g., greater than 0.1, as shown in Figure A.16(a). This way, we are treating values as binary value; hence, a pair of users with a degree of interest towards a given topic at a same time interval are considered like-minded only if both users have a value higher than the threshold. We have additionally performed experiments with this alternative condition of homogeneity for all of the LDA, WNTM, and BTM topic modeling methods, denoted by the '-b' suffix in the figures.

Results. In Figure 12, we report the performance of our proposed method when adopting WNTM, BTM, and LDA topic modeling methods and two alternative conditions of homogeneity (c_1 and c_2) in the news recommendation application. As seen, our approach with LDA (LDA-TCD and LDA-TCD-b) consistently excels over both of the other topic modeling techniques, i.e., WNTM and BTM, for both c_1 and c_2 conditions for varying user embedding dimensions in terms of ranking metrics. The only exception is precision at 10 (P_{10}) where BTM-TCD-b shows the best performance. We attribute LDA's reasonable performance to its well-defined inference procedures for previously unseen news articles. As explained in Section 5.3.1, in order to evaluate our method in news recommendation application, we recommend news articles based on the similarity of the topic distributions in a news article and each community's overall topics of interest at a specific time. In light of the fact that news articles are part of the golden standard, we make sure that the news articles are unseen documents during the topic detection step of our method (training phase) and as such, we use the inference mechanism provided by topic modeling technique to identify the distribution of topics in the unobserved news articles for the same set of topics identified in the training phase. Both WNTM and BTM fall short in introducing effective inferencing methods and as a result their performance is not comparable to LDA.

Comparing the two alternative conditions of homogeneity, we observe that c_1 outperforms c_2 for all topic detection methods. For instance, we review the performance of baselines and our proposed method using c_1 (LDA-TCD) and c_2 (LDA-TCD-b) in the news recommendation application in Figure 13. As seen, LDA-TCD-b is not even able to outperform Hu et al.'s baseline. The reason might be the fact that considering two users who have a value more than a threshold (0.1, LDA's alpha prior) to be like-minded regardless of their degrees of interest has a confounding effect on the user embeddings. That is, dissimilar users end up with close embeddings and finally become members of the same communities.

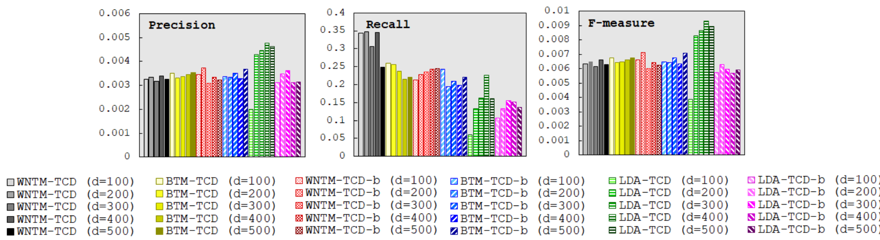

Figure 14: The effect of topic modeling and condition of homogeneity on our proposed in the context of user prediction application.

We study the impact of the topic modelling methods and conditions of homogeneity in user prediction application as well and report the results in Figure 14. As seen, our method using WNTM and BTM topic modellings for both c_1 and c_2 conditions is performing weaker than our approach with LDA for c_1 (LDA-TCD) for varying sizes of user embedding dimensions in terms of precision and f-measure. Also, our method using LDA for c_2 (LDA-TCD-b) shows poorer performance, as does the variations with WNTM and BTM, compared to LDA-TCD.

Contrary to the news recommendation application where LDA-TCD-b did not outperform Hu et al.'s baseline, in user prediction application it performs better than all baselines as seen in Figure 15. Since the performance of WNTM-TCD, WNTM-TCD-b, BTM-TCD, BTM-TCD-b are all close to LDA-TCD-b (Figure 14), they are the best as well compared to the baselines. In summary, in user prediction, our method is able to outperform the baselines regardless of the topic detection methods and the condition of homogeneity.

In summary and based on our experiments on different topic detection methods and conditions of homogeneity in the news recommendation and user prediction tasks, we can conclude that:

1. While the overall performance of our proposed method is sensitive to the choice of topic detection method, it offers better performance compared to the baselines regardless of the topic modeling method in most of the cases for news recommendation and in all cases for user prediction. The

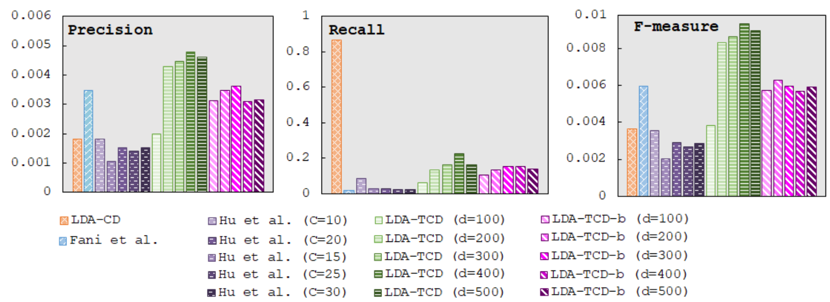

Figure 15: The effect of condition of homogeneity on our proposed method using LDA in the context of user prediction application.

best performance of our proposed method has been obtained when the LDA topic modeling method was adopted.

2. The choice of the condition of homogeneity also impacts the performance of our proposed method. We find that condition c_1, which considers the difference between the degree of users' topical interests, is the more effective from among the two alternatives.

7. Concluding Remarks

In this paper, we have proposed an approach to detect communities through multimodal feature learning (embeddings) of users from their *i*) *temporal* content *ii*) social network neighborhood. With respect to the temporal content, we model the users' temporal contribution towards topics of interest by introducing the notion of regions of like-mindedness between users. These regions cover users who share not only similar topical interests but also similar temporal behavior. Given the regions of like-mindedness as context, we train a neural network such that the probability of a user in a region is maximized given other users in the same region (Section 4.2). With regard to the social network neighborhood, we learn user embeddings based on their social network connections (links) through neural graph embeddings (Section 4.3). We then interpolate temporal content-based embeddings with social link-based embeddings to capture both sources of information for representing users (Section 4.4). Our evaluation on a Twitter dataset under two different application scenarios, namely news recommendation and user prediction showed that (1) content-based methods produce higher quality communities compared to link-based methods; (2) methods that consider temporal evolution of content, especially our proposed method, show better performance compared to their non-temporal baselines; (3) Communities that are produced when time is explicitly incorporated in user vector representations have higher quality than the ones produced when time is incorporated into a generative process, and finally (4) while link-based methods are weaker than content-based methods, their interpolation leads to improved quality of the identified communities.

Possible future directions of our work would be as follows:

1. In our approach, we linearly interpolated temporal content and social network structure at user vector representation level for the task of temporal user community detection. This inherently limits the vectors for both types of representation to have a same embedding size. One possible future direction would be to explore temporal content-based and link-based user vectors at score level, i.e., the final similarity scores of temporal content-based user vector representations be interpolated with the similarity scores of link-based user vectors. This way, the embedding size of information sources become irrelevant. Another direction for our future research is to learn the embedding interpolation function through joint representation learning instead of weighted linear function.
2. One of the parameters that may impact performance of the community detection method is the length of the adopted time interval. In our future experiments, we will systematically explore the impact of time interval size on the quality of the derived communities and will additionally explore ways in which the optimal time interval length can be learnt through hyper-parameter search techniques.

Acknowledgements

The authors graciously acknowledge funding from Natural Sciences and Engineering Research Council of Canada (NSERC) and SOSCIP[7]. They are also thankful to Fattane Zarrinkalam at Ryerson University, whose comments helped improve and clarify this manuscript.

References

[1] D. Lalwani, D. V. L. N. Somayajulu, P. R. Krishna, A community driven social recommendation system, in: 2015 IEEE International Conference on Big Data, Big Data 2015, Santa Clara, CA, USA, October 29 - November 1, 2015, 2015, pp. 821–826. `doi:10.1109/BigData.2015.7363828`. URL `https://doi.org/10.1109/BigData.2015.7363828`

[2] H. Feng, J. Tian, H. J. Wang, M. Li, Personalized recommendations based on time-weighted overlapping community detection, Information & Management 52 (7) (2015) 789–800. `doi:10.1016/j.im.2015.02.004`. URL `https://doi.org/10.1016/j.im.2015.02.004`

[3] Z. Hu, J. Yao, B. Cui, E. P. Xing, Community level diffusion extraction, in: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015, 2015, pp. 1555–1569. `doi:10.1145/2723372.2723737`. URL `http://doi.acm.org/10.1145/2723372.2723737`

[4] S. Y. Yang, S. Y. K. Mo, X. Zhu, An empirical study of the financial community network on twitter, in: IEEE Conference on Computational Intelligence for Financial Engineering & Economics, CIFEr 2014, London, UK, March 27-28, 2014, 2014, pp. 55–62. `doi:10.1109/CIFEr.2014.6924054`. URL `https://doi.org/10.1109/CIFEr.2014.6924054`

[5] L. A. Adamic, N. Glance, The political blogosphere and the 2004 u.s. election: Divided they blog, in: Proceedings of the 3rd International Workshop

[7] www.soscip.org

on Link Discovery, LinkKDD '05, ACM, New York, NY, USA, 2005, pp. 36–43. doi:10.1145/1134271.1134277.
URL http://doi.acm.org/10.1145/1134271.1134277

[6] Y. Richter, E. Yom-Tov, N. Slonim, Predicting customer churn in mobile networks through analysis of social groups, in: Proceedings of the SIAM International Conference on Data Mining, SDM 2010, April 29 - May 1, 2010, Columbus, Ohio, USA, 2010, pp. 732–741. doi:10.1137/1.9781611972801.64.
URL https://doi.org/10.1137/1.9781611972801.64

[7] F. T. Leighton, S. Rao, Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms, J. ACM 46 (6) (1999) 787–832. doi:10.1145/331524.331526.
URL http://doi.acm.org/10.1145/331524.331526

[8] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, Journal of Statistical Mechanics: Theory and Experiment 2008 (10) (2008) P10008.
URL http://stacks.iop.org/1742-5468/2008/i=10/a=P10008

[9] A. Conte, T. D. Matteis, D. D. Sensi, R. Grossi, A. Marino, L. Versari, D2K: scalable community detection in massive networks via small-diameter k-plexes, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018, 2018, pp. 1272–1281. doi:10.1145/3219819.3220093.
URL http://doi.acm.org/10.1145/3219819.3220093

[10] A. Lancichinetti, S. Fortunato, Community detection algorithms: A comparative analysis, Phys. Rev. E 80 (2009) 056117. doi:10.1103/PhysRevE.80.056117.
URL https://link.aps.org/doi/10.1103/PhysRevE.80.056117

[11] D. Rafailidis, P. Kefalas, Y. Manolopoulos, Preference dynamics with multimodal user-item interactions in social media recommendation, Expert

Syst. Appl. 74 (2017) 11–18. doi:10.1016/j.eswa.2017.01.005.
URL https://doi.org/10.1016/j.eswa.2017.01.005

[12] Z. Hu, J. Yao, B. Cui, User group oriented temporal dynamics exploration, in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada., 2014, pp. 66–72.
URL http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8398

[13] H. Fani, E. Bagheri, F. Zarrinkalam, X. Zhao, W. Du, Finding diachronic like-minded users, Computational Intelligence 34 (1) (2018) 124–144. doi:10.1111/coin.12117.
URL https://doi.org/10.1111/coin.12117

[14] H. Fani, E. Bagheri, W. Du, Temporally like-minded user community identification through neural embeddings, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017, 2017, pp. 577–586. doi:10.1145/3132847.3132955.
URL http://doi.acm.org/10.1145/3132847.3132955

[15] M. Sachan, D. Contractor, T. A. Faruquie, L. V. Subramaniam, Using content and interactions for discovering communities in social networks, in: Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012, 2012, pp. 331–340. doi:10.1145/2187836.2187882.
URL http://doi.acm.org/10.1145/2187836.2187882

[16] T. Yang, R. Jin, Y. Chi, S. Zhu, Combining link and content for community detection: a discriminative approach, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009, 2009, pp. 927–936. doi:10.1145/1557019.1557120.
URL http://doi.acm.org/10.1145/1557019.1557120

[17] S. A. Myers, J. Leskovec, The bursty dynamics of the twitter information network, in: Proceedings of the 23rd International Conference on World Wide Web, WWW '14, ACM, New York, NY, USA, 2014, pp. 913–924. doi:10.1145/2566486.2568043.
URL http://doi.acm.org/10.1145/2566486.2568043

[18] M. McPherson, L. Smith-Lovin, J. M. Cook, Birds of a feather: Homophily in social networks, Annual Review of Sociology 27 (1) (2001) 415–444. arXiv:https://doi.org/10.1146/annurev.soc.27.1.415, doi:10.1146/annurev.soc.27.1.415.
URL https://doi.org/10.1146/annurev.soc.27.1.415

[19] S. Fortunato, Community detection in graphs, Physics Reports 486 (3) (2010) 75 – 174. doi:https://doi.org/10.1016/j.physrep.2009.11.002.
URL http://www.sciencedirect.com/science/article/pii/S0370157309002841

[20] M. Girvan, M. E. J. Newman, Community structure in social and biological networks, Proceedings of the National Academy of Sciences 99 (12) (2002) 7821–7826. arXiv:http://www.pnas.org/content/99/12/7821.full.pdf, doi:10.1073/pnas.122653799.
URL http://www.pnas.org/content/99/12/7821

[21] R. Andersen, F. R. K. Chung, K. J. Lang, Local graph partitioning using pagerank vectors, in: 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), 21-24 October 2006, Berkeley, California, USA, Proceedings, 2006, pp. 475–486. doi:10.1109/FOCS.2006.44.
URL https://doi.org/10.1109/FOCS.2006.44

[22] I. S. Dhillon, Y. Guan, B. Kulis, Weighted graph cuts without eigenvectors A multilevel approach, IEEE Trans. Pattern Anal. Mach. Intell. 29 (11) (2007) 1944–1957. doi:10.1109/TPAMI.2007.1115.
URL https://doi.org/10.1109/TPAMI.2007.1115

[23] J. Leskovec, K. J. Lang, M. W. Mahoney, Empirical comparison of algorithms for network community detection, in: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010, 2010, pp. 631–640. doi:10.1145/1772690.1772755.
URL http://doi.acm.org/10.1145/1772690.1772755

[24] M. Wang, C. Wang, J. X. Yu, J. Zhang, Community detection in social networks: An in-depth benchmarking study with a procedure-oriented framework, PVLDB 8 (10) (2015) 998–1009. doi:10.14778/2794367.2794370.
URL http://www.vldb.org/pvldb/vol8/p998-wang.pdf

[25] C. P. Diehl, G. Namata, L. Getoor, Relationship identification for social network discovery, in: Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada, 2007, pp. 546–552.
URL http://www.aaai.org/Library/AAAI/2007/aaai07-086.php

[26] T. A. Snijders, A. Lomi, Beyond homophily: Incorporating actor variables in actor-oriented network models, arXiv preprint arXiv:1803.07172.

[27] N. Barbieri, F. Bonchi, G. Manco, Efficient methods for influence-based network-oblivious community detection, ACM TIST 8 (2) (2017) 32:1–32:31. doi:10.1145/2979682.
URL http://doi.acm.org/10.1145/2979682

[28] V. Labatut, N. Dugué, A. Perez, Identifying the community roles of social capitalists in the twitter network, in: 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2014, Beijing, China, August 17-20, 2014, 2014, pp. 371–374. doi:10.1109/ASONAM.2014.6921612.
URL https://doi.org/10.1109/ASONAM.2014.6921612

[29] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent dirichlet allocation, Journal of

Machine Learning Research 3 (2003) 993–1022.
URL http://www.jmlr.org/papers/v3/blei03a.html

[30] D. Zhou, E. Manavoglu, J. Li, C. L. Giles, H. Zha, Probabilistic models for discovering e-communities, in: Proceedings of the 15th international conference on World Wide Web, WWW 2006, Edinburgh, Scotland, UK, May 23-26, 2006, 2006, pp. 173–182. doi:10.1145/1135777.1135807.
URL http://doi.acm.org/10.1145/1135777.1135807

[31] D. Peng, X. Lei, T. Huang, DICH: A framework for discovering implicit communities hidden in tweets, World Wide Web 18 (4) (2015) 795–818. doi:10.1007/s11280-014-0279-z.
URL https://doi.org/10.1007/s11280-014-0279-z

[32] H. Liu, H. Chen, M. Lin, Y. Wu, Community detection based on topic distance in social tagging networks, 2014.

[33] Y. Huang, H. Dong, Y. Yesha, S. Zhou, A scalable system for community discovery in twitter during hurricane sandy, in: 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, CCGrid 2014, Chicago, IL, USA, May 26-29, 2014, 2014, pp. 893–899. doi:10.1109/CCGrid.2014.122.
URL https://doi.org/10.1109/CCGrid.2014.122

[34] Z. Yin, L. Cao, Q. Gu, J. Han, Latent community topic analysis: Integration of community discovery with topic modeling, ACM TIST 3 (4) (2012) 63:1–63:21. doi:10.1145/2337542.2337548.
URL http://doi.acm.org/10.1145/2337542.2337548

[35] R. Nallapati, A. Ahmed, E. P. Xing, W. W. Cohen, Joint latent topic models for text and citations, in: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, 2008, pp. 542–550. doi:10.1145/1401890.1401957.
URL http://doi.acm.org/10.1145/1401890.1401957

[36] L. Dietz, S. Bickel, T. Scheffer, Unsupervised prediction of citation influences, in: Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007, 2007, pp. 233–240. doi:10.1145/1273496.1273526.
URL http://doi.acm.org/10.1145/1273496.1273526

[37] A. Gruber, M. Rosen-Zvi, Y. Weiss, Latent topic models for hypertext, in: UAI 2008, Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, Helsinki, Finland, July 9-12, 2008, 2008, pp. 230–239.
URL https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=1319&proceeding_id=24

[38] S. Zhu, K. Yu, Y. Chi, Y. Gong, Combining content and link for classification using matrix factorization, in: SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007, 2007, pp. 487–494. doi:10.1145/1277741.1277825.
URL http://doi.acm.org/10.1145/1277741.1277825

[39] A. Mojahed, J. H. Bettencourt-Silva, W. Wang, B. de la Iglesia, Applying clustering analysis to heterogeneous data using similarity matrix fusion (SMF), in: Machine Learning and Data Mining in Pattern Recognition - 11th International Conference, MLDM 2015, Hamburg, Germany, July 20-21, 2015, Proceedings, 2015, pp. 251–265. doi:10.1007/978-3-319-21024-7_17.
URL https://doi.org/10.1007/978-3-319-21024-7_17

[40] Y. Ruan, D. Fuhry, S. Parthasarathy, Efficient community detection in large networks using content and links, in: 22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013, 2013, pp. 1089–1098. doi:10.1145/2488388.2488483.
URL http://doi.acm.org/10.1145/2488388.2488483

[41] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word

representations in vector space, CoRR abs/1301.3781. arXiv:1301.3781. URL http://arxiv.org/abs/1301.3781

[42] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States., 2013, pp. 3111–3119.

[43] A. Grover, J. Leskovec, node2vec: Scalable feature learning for networks, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, 2016, pp. 855–864. doi:10.1145/2939672.2939754. URL http://doi.acm.org/10.1145/2939672.2939754

[44] B. Perozzi, R. Al-Rfou, S. Skiena, Deepwalk: online learning of social representations, in: The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014, 2014, pp. 701–710. doi:10.1145/2623330.2623732. URL http://doi.acm.org/10.1145/2623330.2623732

[45] S. Cao, W. Lu, Q. Xu, Deep neural networks for learning graph representations, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA., 2016, pp. 1145–1152.
URL http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12423

[46] D. Wang, P. Cui, W. Zhu, Structural deep network embedding, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, 2016, pp. 1225–1234. doi:10.1145/2939672.2939753. URL http://doi.acm.org/10.1145/2939672.2939753

[47] G. J, S. Ganguly, M. Gupta, V. Varma, V. Pudi, Author2vec: Learning author representations by combining content and link information, in: Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11-15, 2016, Companion Volume, 2016, pp. 49–50. doi:10.1145/2872518.2889382.
URL http://doi.acm.org/10.1145/2872518.2889382

[48] A. Benton, R. Arora, M. Dredze, Learning multiview embeddings of twitter users, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers, 2016.
URL http://aclweb.org/anthology/P/P16/P16-2003.pdf

[49] T. N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: International Conference on Learning Representations ICLR, 2017.

[50] W. L. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, in: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, 2017, pp. 1025–1035.
URL http://papers.nips.cc/paper/6703-inductive-representation-learning-on-large-graphs

[51] S. Amir, B. C. Wallace, H. Lyu, P. Carvalho, M. J. Silva, Modelling context with user embeddings for sarcasm detection in social media, in: Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016, 2016, pp. 167–177.
URL http://aclweb.org/anthology/K/K16/K16-1017.pdf

[52] L. Chen, T. Qian, P. Zhu, Z. You, Learning user embedding representation for gender prediction, in: 28th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2016, San Jose, CA, USA, November

6-8, 2016, 2016, pp. 263–269. doi:10.1109/ICTAI.2016.0048.
URL https://doi.org/10.1109/ICTAI.2016.0048

[53] Y. Yu, X. Wan, X. Zhou, User embedding for scholarly microblog recommendation, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers, 2016.
URL http://aclweb.org/anthology/P/P16/P16-2073.pdf

[54] F. Zarrinkalam, H. Fani, E. Bagheri, M. Kahani, Predicting users' future interests on twitter, in: Advances in Information Retrieval - 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings, 2017, pp. 464–476. doi:10.1007/978-3-319-56608-5_36.
URL https://doi.org/10.1007/978-3-319-56608-5_36

[55] F. Zarrinkalam, H. Fani, E. Bagheri, M. Kahani, Inferring implicit topical interests on twitter, in: Advances in Information Retrieval - 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20-23, 2016. Proceedings, 2016, pp. 479–491. doi:10.1007/978-3-319-30671-1_35.
URL https://doi.org/10.1007/978-3-319-30671-1_35

[56] L. Zhao, M. J. Zaki, Tricluster: An effective algorithm for mining coherent clusters in 3d microarray data, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, Baltimore, Maryland, USA, June 14-16, 2005, 2005, pp. 694–705. doi:10.1145/1066157.1066236.
URL http://doi.acm.org/10.1145/1066157.1066236

[57] F. Morin, Y. Bengio, Hierarchical probabilistic neural network language model, in: Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics, AISTATS 2005, Bridgetown, Barbados, January 6-8, 2005, 2005.
URL http://www.gatsby.ucl.ac.uk/aistats/fullpapers/208.pdf

[58] Y. Guan, X. Wei, C. Li, Y. Keller, People identification and tracking through fusion of facial and gait features, in: Biometric Authenti-

cation - First International Workshop, BIOMET 2014, Sofia, Bulgaria, June 23-24, 2014. Revised Selected Papers, 2014, pp. 209–221. doi:10.1007/978-3-319-13386-7_17.
URL https://doi.org/10.1007/978-3-319-13386-7_17

[59] O. Arandjelovic, R. Cipolla, A new look at filtering techniques for illumination invariance in automatic face recognition, in: Seventh IEEE International Conference on Automatic Face and Gesture Recognition (FGR 2006), 10-12 April 2006, Southampton, UK, 2006, pp. 449–454. doi:10.1109/FGR.2006.14.
URL https://doi.org/10.1109/FGR.2006.14

[60] O. Arandjelovic, Weighted linear fusion of multimodal data: A reasonable baseline?, in: Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016, 2016, pp. 851–857. doi:10.1145/2964284.2964304.
URL https://doi.org/10.1145/2964284.2964304

[61] F. Abel, Q. Gao, G. Houben, K. Tao, Analyzing user modeling on twitter for personalized news recommendations, in: User Modeling, Adaption and Personalization - 19th International Conference, UMAP 2011, Girona, Spain, July 11-15, 2011. Proceedings, 2011, pp. 1–12. doi:10.1007/978-3-642-22362-4_1.
URL https://doi.org/10.1007/978-3-642-22362-4_1

[62] J. Qiang, P. Chen, T. Wang, X. Wu, Topic modeling over short texts by incorporating word embeddings, in: Advances in Knowledge Discovery and Data Mining - 21st Pacific-Asia Conference, PAKDD 2017, Jeju, South Korea, May 23-26, 2017, Proceedings, Part II, 2017, pp. 363–374. doi:10.1007/978-3-319-57529-2_29.
URL https://doi.org/10.1007/978-3-319-57529-2_29

[63] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, M. Demirbas, Short text classification in twitter to improve information filtering, in: Proceeding

of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010, 2010, pp. 841–842. doi:10.1145/1835449.1835643.
URL http://doi.acm.org/10.1145/1835449.1835643

[64] P. Ferragina, U. Scaiella, Fast and accurate annotation of short texts with wikipedia pages, IEEE Software 29 (1) (2012) 70–75. doi:10.1109/MS.2011.122.
URL https://doi.org/10.1109/MS.2011.122

[65] G. Petkos, S. Papadopoulos, L. M. Aiello, R. Skraba, Y. Kompatsiaris, A soft frequent pattern mining approach for textual topic detection, in: 4th International Conference on Web Intelligence, Mining and Semantics (WIMS 14), WIMS '14, Thessaloniki, Greece, June 2-4, 2014, 2014, pp. 25:1–25:10. doi:10.1145/2611040.2611068.
URL https://doi.org/10.1145/2611040.2611068

[66] Y. Wu, H. Liu, Y. Yang, Graph convolutional matrix completion for bipartite edge prediction, in: Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2018, Volume 1: KDIR, Seville, Spain, September 18-20, 2018., 2018, pp. 49–58. doi:10.5220/0006900000490058.
URL https://doi.org/10.5220/0006900000490058

[67] T. N. Kipf, M. Welling, Variational graph auto-encoders, CoRR abs/1611.07308. arXiv:1611.07308.
URL http://arxiv.org/abs/1611.07308

[68] T. Chakraborty, Z. Cui, N. Park, Metadata vs. ground-truth: A myth behind the evolution of community detection methods, in: Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018, 2018, pp. 45–46. doi:10.1145/3184558.3186921.
URL http://doi.acm.org/10.1145/3184558.3186921

[69] F. Moradi, T. Olovsson, P. Tsigas, An evaluation of community detection algorithms on large-scale email traffic, in: Experimental Algorithms - 11th International Symposium, SEA 2012, Bordeaux, France, June 7-9, 2012. Proceedings, 2012, pp. 283–294. doi:10.1007/978-3-642-30850-5_25.
URL https://doi.org/10.1007/978-3-642-30850-5_25

[70] Y. Zuo, J. Zhao, K. Xu, Word network topic model: a simple but general solution for short and imbalanced texts, Knowl. Inf. Syst. 48 (2) (2016) 379–398. doi:10.1007/s10115-015-0882-z.
URL https://doi.org/10.1007/s10115-015-0882-z

[71] X. Yan, J. Guo, Y. Lan, X. Cheng, A biterm topic model for short texts, in: 22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013, 2013, pp. 1445–1456. doi:10.1145/2488388.2488514.
URL https://doi.org/10.1145/2488388.2488514

Appendix A. Finding Regions of Like-mindedness

Finding R for time interval t (R_t). The process for finding R_t is dependent on $\mathbf{X}$ and some condition of homogeneity denoted by c. We let $x_{uz_i,t}$ be the extent of u's interest in z_i and define $\mathsf{U}_{z_iz_j,t}(c)$ to be the set of all those users who are interested in both topics z_i and z_j given c. In our definition, $\mathsf{U}_{z_iz_j,t}(c)$ is considered to be maximal if it is not possible to include an additional user while maintaining c. Based on $U_t = \{\mathsf{U}_{z_iz_j,t}(c) : z_i, z_j \in Z\}$, we form a multigraph $\mathrm{G}_t = (Z, U_t)$ whose nodes are the set of topics and for each $\mathsf{U}_{z_iz_j,t}(c) \in U_t$ a directed edge connecting z_i to z_j is added to G_t, which is labeled with the set of users in $\mathsf{U}_{z_iz_j,t}(c)$.

In Figure A.16, we clarify how the multigraph would look like by visualizing it for time interval 22 for the three users introduced earlier. We assume two alternatives for the condition of homogeneity, i) regions that have a value above 0.1 will be considered to be similar, and ii) regions that have a value above 0.1

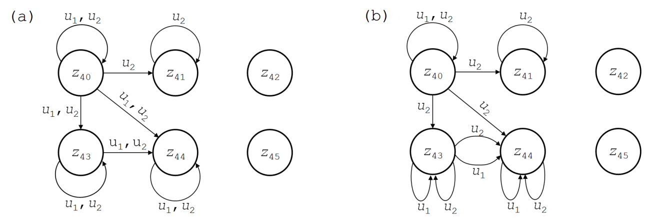

Figure A.16: The Multigraph constructed from the three users introduced in Figure 1 in time interval t_{22} when the condition of homogeneity c is (a) a value above 0.1, and (b) the difference of values above 0.1 falls in the range of [0, 0.1).

and the differences of values fall in the range [0, 0.1) will be considered to be similar. The multigraphs are shown in Figure A.16(a) and A.16(b) respectively.

Once the multigraph has been constructed for time interval t (G_t), we perform a depth first search traversal on G_t in order to find $\boldsymbol{R}_t$, a process which has been outlined in Algorithm 2. We initially commence the process by considering all of the users with an empty set of topics ($r =$ U ?; all users U). The algorithm gradually considers each topic and incrementally add it to the set. In each recursive stage, we have a candidate denoted as $r =$ A B and a set of yet-to-be-processed topics C. The candidate will be added to $\boldsymbol{R}_t$ if it satisfies the condition of homogeneity and is not already subsumed by another region. Since in graph G_t, we only create region of like-mindedness based on a topic (loop) or a pair of topics (directed edge), we need to check condition c as we traverse a DFS path over the directed edges of the graph in order to extend the region of like-mindedness to include more topics and users. Further, We remove all other regions that are subsumed by r when r is added to $\boldsymbol{R}_t$ (Lines 2 to 4). Once r is added, we now expand its topic set to include one of the remaining topics that have not been considered yet as long as there is a directed edge between a topic in r and the new topic in G_t. The algorithm is recursively called on the new candidate that includes a new topic (Lines 5 to 12).

For the sake of further clarification, let us review the process proposed in Algorithm 2 for the multigraph depicted in Figure A.16(a). The algorithm

Algorithm 2 Finding regions of like-mindedness for time interval t (R_t)

Inputs:

c, homogeneity condition;

G_t, multigraph at time interval t;

U, set of users;

Z, set of topics of interest;

Output:

R_t, set of regions of like-mindedness for time interval t

Initialization:

$R_t = \emptyset$;

$\text{find\_r\_t}(r = \mathsf{U} \quad \emptyset, \mathrm{C} = [z_1, z_1, z_2, z_2, ..., z_{|Z|}, z_{|Z|}])$;

1: **procedure find_r_t**$(r = \mathrm{A} \quad \mathrm{B}, \mathrm{C})$

2: **if** $(r \models c) \wedge (\nexists r' \in R_t : r \quad r')$ **then**

3: $\forall r'' \in R_t$ **if** $r'' \quad r$ **then** $R_t \leftarrow R_t \setminus r''$

4: $R_t \leftarrow R_t \cup r$

5: **for all** $z_j \in \mathsf{Z}$ **do**

6: $\mathrm{A} \leftarrow r.\mathrm{A}; \mathrm{B} \leftarrow r.\mathrm{B} \cup z_j; \mathrm{C} \leftarrow \mathrm{C} \setminus z_j$

7: **if** $r.\mathrm{B} = \emptyset$ **then** $\text{find\_r\_t}(\mathrm{A} \quad \mathrm{B}, \mathrm{C})$

8: **else**

9: **for all** $z_i \in r.\mathrm{B}$ **do**

10: **for all** $(z_i \rightarrow z_j) \in U_t$ **do**

11: $\mathrm{A} \leftarrow r.\mathrm{A} \cap \mathsf{U}_{z_i z_j, t}$

12: $\text{find\_r\_t}(\mathrm{A} \quad \mathrm{B}, \mathrm{C})$

---

starts by initializing r to consist of all the three users but an empty set of considered topics and a complete set of unexplored topics ($r = \{u_1, u_2, u_3\} \quad \emptyset, \mathrm{C} = [z_{40}, z_{40}, z_{41}, z_{41}, ..., z_{45}, z_{45}]$). The algorithm then selects the first topic (Topic 40) by removing it from C and adding it to the empty set of topics in r (Line 7). Given the current state of r ($\{u_1, u_2, u_3\} \quad \{z_{40}\}$) does not satisfy the condition for homogeneity, we select the next topic, which is again Topic 40 given the directed looping edge. The new r ($\{u_1, u_2\} \quad \{z_{40}\}$) now satisfies the condition of homogeneity and is hence added to R_t (Line 4). The subsequent step is to consider Topic 41 because there is a direct edge from Topic 40 to Topic 41. Based on this transition, the new r will be $\{u_2\} \quad \{z_{40}, z_{41}\}$, which produces a new element in $\mathcal{R}_t$.

Finding regions of like-mindedness ($\mathcal{R}$). Algorithm 2 identifies $\mathcal{R}_t$ separately for each of the time intervals; however, we will need to identify $\mathcal{R}$ across the whole time period that spans all of the individual time intervals. We adopt a similar strategy for expanding the individual $\mathcal{R}_t$s into $\mathcal{R}$ as explained in Algorithm 3. We build a multigraph G which consists of the time intervals as its nodes and edges representing transitions between time intervals such as i and j only when $\{r.\mathrm{A} \cap r'.\mathrm{A}\} \times \{r.\mathrm{B} \cap r'.\mathrm{B}\} \times \{i, j\}$ satisfies c given two regions $r \in \mathcal{R}_i$ and $r' \in \mathcal{R}_j$.

Algorithm 3 produces $\mathrm{R} = \mathrm{A} \times \mathrm{B} \times \mathrm{C} \in \mathcal{R}$ where A is a set of users who have the similar interests towards topics in B in time intervals in C based on a defined condition of homogeneity. In essence, this provides us with information on which users, when and how, expressed similar preferences towards topics of the social network. This is valuable for determining which users are similar to each other across different time intervals and topics. Those users who are placed together in the same R can be considered to be more similar to each other compared to those users who are not in the same R. We consider regions of like-mindedness such as R to serve as context for each user. Based on such context, we would like to learn user embeddings that maximize the likelihood of users who have been seen together in the same Rs to be close to each other in the embedding space and those who are not seen together to be embedded far apart from each other. Let us first discuss the time complexity of finding regions of like-mindedness.

Time complexity analysis. In each time interval t, it takes $O(|\mathsf{U}| \times |\mathsf{Z}|^2)$ to calculate $\mathsf{U}_{z_i z_j, t}(c)$ for all pairs of z_i and $z_j \in \mathsf{Z}$ and build the multigraph G_t considering the fact that testing the condition of homogeneity can be done in $O(1)$. Furthermore, performing depth-first-search (DFS) on the graph to find regions of like-mindedness $\mathcal{R}_t$ takes $O(|\mathsf{U}|^{|\mathsf{Z}|})$ in the worst case, which happens when there exists an edge between each pair of z_i and z_j associated with $\mathsf{U}_{z_i z_j, t}(c)$ containing only one user. The analysis of the time complexity for finding $\mathcal{R}$ is similar but in the context of the number of time intervals and the

Algorithm 3 Finding regions of like-mindedness (R)

Inputs:

c, homogeneity condition;

U, set of users;

Z, set of topics of interest;

G, multigraph for the whole time intervals;

R_t for each time interval 1 t T;

Output:

R, set of regions of like-mindedness for the whole time intervals

Initialization:

R = ?;

find_r(R = U Z ?, D=[1, 1, 2, 2, ..., T, T]);

```
procedure find_r(R = A  B  C, D)
    if (R j= c) ^ (@R′ 2 R : R  R′ then
        8R″ 2 R if R″  R then R  R n R″
        R  R [ R
    for all j 2 D do
        A  R.A; B  R.B; C  R.C [ j; D  D n j
        if R.C = ? then
            find_r(A  B  C, D)
        else
            for all i 2 R.C do
                for all (i ! j) 2 G do
                    //r 2 R_i, r′ 2 R_j : fr.A \ r′.Ag  fr.B \ r′.Bg  fi, jg
                    A  R.A \ fr.A \ r′.Ag
                    B  R.B \ fr.B \ r′.Bg
                    find_r(A  B  C, D)
```

size of R_t for each time interval. Here, for each pair of time intervals i and j, and a pair of R_i and R_j, we test the condition of homogeneity which takes $O(|r| \; \mathrm{T}^2)$ plus a final DFS in $O(|r|\mathrm{T})$ where $|r|$ is the number of all R_t. As seen, the most expensive parts are the DFS traversal on the multigraphs in the first and second steps which highly depend on the condition for homogeneity c.

We would like to note that the proposed method is efficient in practice because of the following considerations:

1. In the real world, users are only interested in a limited set of topics in each time interval and over the whole time period. For this reason, users' topic preference time series is quite sparse with many topics not even examined or relevant for each user. Therefore, the number of edges in the multigraphs is quite small. Recall that one of the major components of the time complexity of the method was due to the DFS traversal, which will be quite small given the sparsity of the multigraphs in practice.
2. In addition, the depth of the DFS traversal is quite shallow given the fact that the number of users is far larger than the number of topics and time intervals. When compared to the number of users, the number of topics and time intervals can be considered to be constant values.
3. Algorithms 2 and 3 can be easily parallelized across different time intervals.