D-Map: Visual Analysis of Ego-centric Information Diffusion Patterns in Social Media

Siming Chen[1]* Shuai Chen[1] Zhenhuang Wang[1] Jie Liang[2†] Xiaoru Yuan[1] Nan Cao[3‡]
Yadong Wu[4§]

1) Key Laboratory of Machine Perception (Ministry of Education), and School of EECS, Peking University, China
2) Faculty of Engineer and Information Technology, The University of Technology, Sydney, Australia
3) New York University, Shanghai, China
4) Southwest University of Science and Technology, China

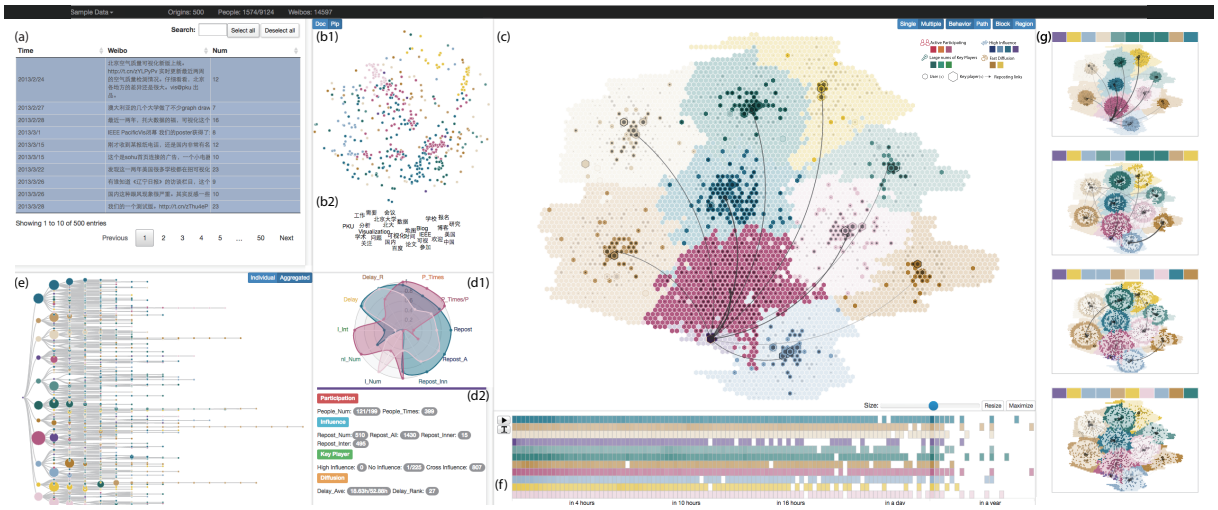

Figure 1: System Interface: Source Weibo Table View (a), for selecting different groups of source weibos; Source Weibo Distribution View (b), including Documents View (b1) and Keywords View (b2); D-Map View (c), summarizing the social interaction among participating people of a central user; Community Radar View (d), showing the high dimensional features of communities with a Radar View (d1) and a Statistics Information Window (d2); Hierarchical View (e), illustrating the reposting structures; Timeline View (f), highlighting the temporal trends of the diffusion; Small Multiple View (g), identifying key time frames of D-Map's snapshots.

Abstract

Popular social media platforms could rapidly propagate vital information over social networks among a significant number of people. In this work we present D-Map (Diffusion Map), a novel visualization method to support exploration and analysis of social behaviors during such information diffusion and propagation on typical social media through a map metaphor. In D-Map, users who participated in reposting (i.e., resending a message initially posted by others) one central user's posts (i.e., a series of original tweets) are collected and mapped to a hexagonal grid based on their behavior similarities and in chronological order of the repostings. With additional interaction and linking, D-Map is capable of providing visual portraits of the influential users and describing their social behaviors. A comprehensive visual analysis system is developed to support interactive exploration with D-Map. We evaluate our work with real world social media data and find interesting patterns among users. Key players, important information diffusion paths, and interactions among social communities can be identified.

Keywords: Social Media, Map, Information Diffusion

1 Introduction

Social media has become an important part of our daily life and has significantly influenced our ways of communication. Every day, millions and even billions of people from all over the world interact with each other across space and time via posting or replying, producing a large number of messages that spread in social media platforms. The richness of social media data provides great opportunities for understanding the process of information diffusion and social communication behaviors of people, where identifying key players (e.g., opinion leaders) and understanding their influences are two critical tasks.

Existing visualization techniques mostly focus on illustrating how social objects (e.g., a message, a topic, or an opinion extracted from messages) spread over space and time [7, 48, 52]. Little research has focused on revealing how people get involved in the diffusion process and get influenced by a central user who initiates the process with multiple original microblogs, which is the focus of

*e-mail: {siming.chen, shuai.chen, zhenhuang.wang, xiaoru.yuan}@pku.edu.cn. Xiaoru Yuan is the corresponding author.
†e-mail: christy.jie@gmail.com
‡e-mail: nan.cao@gmail.com
§e-mail: wyd028@163.com

2016 IEEE Conference on Visual Analytics Science and Technology (VAST)
23-28 October, Baltimore, Maryland, USA
978-1-5090-5661-3/16/$31.00 ©2016 IEEE

this paper. Moreover, how one original microblog gets reposted can be visualized with a reposting tree [36], but understanding many reposting trees and revealing the social interactions among influenced users requires mentally merging the reposting trees, which can be difficult. Therefore, there is an urgent need for producing a clear and intuitive summarization of the diffusion process to illustrate the spreading of messages across different groups of people and reveal the social impact of a central user.

There are challenges to designing a visualization fulfilling the above requirements. First, social media data are usually very complex. To be more specific, they are heterogeneous, big, and dynamic, containing both structured and non-structured data, making the summarization of information spreading structures among communities difficult. Second, capturing a user's influence requires an in-depth understanding of his/her social behaviors and a detailed analysis of the user's historical communication records. Such kind of analysis is usually difficult, as a user's behavior patterns are complicated in the real world and may change frequently, making capturing the diffusion dynamics and revealing regular diffusion patterns challenging tasks. Third, the visualization of information diffusion processes and patterns among different groups of people requires to encode multiple types of information such as relationships between the users, their roles, the messages that they are involved with, and the entire message-spreading process. Meanwhile, it is important to avoid clutter, such as overlapping nodes and edge crossings in the visualization.

To address the above challenges, we introduce D-Map, an interactive information Diffusion Map that can summarize the historical information diffusion processes initiated by a central user in a social space context considering the communities of the influenced users. Specifically, we produce the map based on the hexagonal tessellation to reduce visual clutter by eliminating node overlaps. In our design, social media users are visualized as hex nodes with color and size encoding their behaviors and roles. These people are grouped into different regions as communities on the map based on their behaviors, forming the social portrait of the central user. In this way, the central user's social influence is visually summarized.

In particular, the paper makes the following contributions:

- **Visual Metaphor Design.** We introduce a novel dynamic information map design to reveal the dynamic patterns of how people are involved in diffusion processes and influenced by a central user. The techniques ensure a clear and intuitive visual representation of an aggregated ego-centric diffusion process, thus forming a social portrait of a central user.
- **Visual Analytics System.** We develop a comprehensive visual analytics system (Figure 1) incorporating advanced community detection techniques and multiple coordinated visualization views. It provides a solution for understanding the influence of a central user and the social interactions in a diffusion process. We evaluate our system with the data collected from Weibo, the biggest microblog platform in China and reveal many interesting real-world patterns that have, to the best of our knowledge, never been visualized before.

2 Related Works

2.1 Social Network Visualization

The extensive studies on social network cover a broad range of topics, including community detection [18], role identification [29], and, most recently, information diffusion and influence anaysis [27, 38]. Visualization techniques play a major role in analyzing the social network [23, 24, 25]. Most of the existing techniques focus on capturing the structure of social network, which are visualized using node-link diagrams [23], an adjacency matrix [24], or a combination of both methods [25]. However, none of the existing techniques are developed for producing a network map to illustrate the diffusion pathway among different people and across diverse communities. This is the focus of our paper.

2.2 Information Diffusion Analysis and Visualization

Information diffusion has become an important research area in the domain of social media analysis in recent years [22]. Studies cover a wide range of topics including showing the evolution of topics [15], influence analysis [42], and visualizing and analyzing of diffusion process [7]. Many visual analysis techniques have been developed to help users better understand the diffusion process via interactive exploration and analysis. For example, Marcus et al. [31] introduced TweetInfo for a flexible aggregation of tweets from spatial, temporal, and event dimensions, thus supporting an accessible exploration of the event propagation process. Viegas et al. [47] introduced Google+ Ripples, which employed a hierarchical circular packing schema to illustrate the re-sharing behaviors and the message-spreading process. Cao et al. [7] introduced Whisper, a flower-like visualization designed for monitoring the information diffusion of a given topic in real time. Ren et al. [36] proposed WeiboEvents which enabled flexible annotations of an information diffusion process based on crowdsourcing. All these techniques are successful designs illustrating the information diffusion process from different aspects, but none of them produce a static summarization of the diffusion process in forms of a map, by which diffusion patterns can be revealed at a first glance. Recently, more studies have focused on exploring the collective topic or opinion diffusion dynamics [41, 48, 50]. Multiple visual analysis techniques have been developed to detect anomalous spreading of messages [52], opinions [10], and user accounts with suspicious behaviors [11]. These are visual analysis techniques that focus on the problems in different application domains, instead of summarizing the diffusion process, so they are different from our work.

2.3 Dynamic Network and Ego-Centric Visualization

Researchers proposed advanced visualization methods for the dynamic network [1]. Animation and small-multiples are two general approaches [3]. Recently, to reveal more insights of relationship evolution, researchers proposed timeline-based approaches for dynamic network visualization [2, 16, 46]. In the dynamic network, identifying the key players and their influences is another critical analysis task for understanding information propagation [42]. Classification and cluster analysis are widely used for role identification [35, 44]. These techniques group users into categories of different roles based on their behavior features. Cha et al. [14] measured a user's influence on Twitter based on the indegree, the number of retweets and mentions. These problems also attracted attention in the visualization field. In particular, an ego-centric view enables a closer look at individual behaviors, thus providing more detailed behavior patterns [8, 39, 49]. For example, Brandes et al. proposed a ripple metaphor to display the passing of time and the biography of the movie actor [5]. Shi et al. chose 1.5D form to embed the network along the time axis, revealing both the temporal and ego-network structures [39]. Cao et al. [8] developed Episogram that discussed the data model in ego-centric social interactions. Different from these techniques, D-Map introduces a novel diffusion map design that illustrates how people across the various communities are influenced by a central user. The proposed method captures both topological and content information of an ego-centric social interaction network, producing a social portrait of the central user, which has not been done before.

2.4 Map oriented Graph Visualization

There are earlier works on representing network data with map-like visualizations. Gansner et al. [20] introduced GMap, an interactive visualization design to transform a social network into a

map view to highlight the boundaries between different communities. Further, they proposed a stable layout of such map views for dynamic network data [27, 32]. They applied the dynamic-maps generation techniques in Twitter data [21] and computer science literature [19]. Though these works have done an excellent job in preserving humans mental map in analyzing the dynamic data, their focus is not the ego-centric users' social connections. Cao et al. [12] introduced FacetAtlas based on a node-link graph visualization and bundling techniques to represent a multifaceted atlas of a text corpus. Following a similar idea, Nachmanson et al. [33] introduced GraphMaps, which also applied edge bundling in a node-link diagram to help with the exploration of large graph. Yang et al. [51] proposed the hexagon-tilling algorithm to visualize hierarchical data. Recently, Cao et al. [11] visualized a social interaction graph on a triangle map for multidimensional data [9]. However, none of these techniques produce a compact visualization as the portrait of the centric social media user to illustrate his/her influence regarding spreading messages, which is the focus of this paper.

3 Data Description

In this study, the data is extracted from Sina Weibo, whose primary services are very similar to those of Twitter. Each weibo is a micro blog, as a tweet on Twitter. We aim to evaluate the social impact of an influential person in the social network. Thus, our target data are the series of weibos from one single user, with all the reposting weibos originating from these source weibos. We extracted the weibo content, timestamp, id and the pid, which is the id of its parent weibo. According to the pids and ids of the selected weibos, we built up a hierarchical reposting tree to show the information diffusion of a single weibo. We merged all the reposting trees with the super center node. This data constructs the social network rooted from one user (Figure 2).

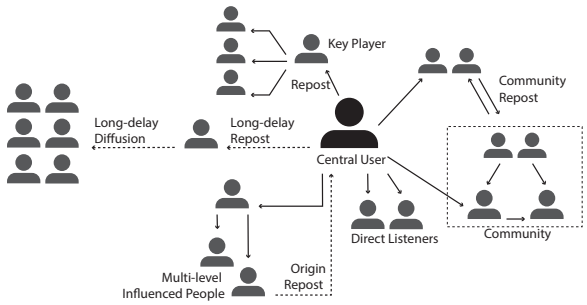

Figure 2: An illustration of the weibo data. People play different roles in one central user's reposting network with multiple behaviors.

Based on observation of the data, we summarized the characteristics of ego-centric weibo data from four perspectives:

- **Participating Features** Central users attract different numbers of participants. Among them, Active people repost weibos frequently and inactive ones repost once.
- **Influence of the Participants** Participants' reposts lead to different times of multi-level reposts. Both direct reposts and total reposts count that one user attracts indicate the impact.
- **Key Player Distribution** We can define people whose weibos are largely reposted as key players. Key players could have impacts on different groups or types of people.
- **Dynamic Diffusion** The life cycle of the diffusion of social media consists of multiple stages, including beginning, bursting and dying. Reposting frequency, latency, impacts and involved people are different in each stage.

Our design consideration is to explore the diffusion process and user relationships based on these features, for a deeper understanding of users' social behaviors.

4 D-Map

In this section, we provide a conceptual model for designing D-Map, and detail the visual design and construction process.

4.1 Conceptual Model

We aim to evaluate the social impact of one central user from multiple aspects. Specifically, we are interested in how source weibos are diffused among multiple groups of people. In this process, key players and important diffusion paths should be pointed out. Furthermore, the interaction patterns among different people could reflect the social structure of the central user (Figure 3).

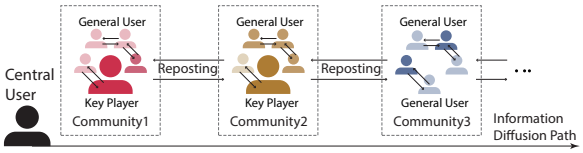

Figure 3: Conceptual model illustrating the diffusion process. Starting from a central user, information diffuses within and across multiple communities through a series of reposting behaviors.

To achieve these goals, we need to merge all the reposting weibos and conduct the analysis. A direct node-link graph visualization of users' communication records might be helpful, but it usually leads to cluttered visualization, which fails to reveal data insights effectively [40]. The hair-ball clutter prevents users from making sense of the group distribution, and makes it hard to select individuals. The links add too much interference for analysis and the visualization wastes space with a large amount of blank space. Moreover, it lacks temporal information to investigate the diffusion process further. Thus, considering both the limitations of the force-directed graph and the characteristics of reposting behaviors, we summarize the design requirements.

- **Show uncluttered participants' community distribution** To investigate the participating people, we need to categorize and group people with similar reposting behaviors.
- **Understand social interactions among people** Repostings lead to the message diffusion, reflecting the social interactions. We need to compare users' reposting patterns
- **Understand reposting features of people** Central user's social portraits are built with the reposting people's features. Key players and their connections should be highlighted.
- **Tell stories of dynamic diffusion** Understanding the diffusion process allows users to recall the past. We should allow users to select diffusion states and individual paths for details.

To fulfill the above requirements, we propose the D-Map design to generate a social user's portrait with explorable features.

4.2 Community Detection

A community is a set of nodes which are densely and internally connected while having sparse connections with other sets. People who often repost the same individual's weibos and have similar behaviors can be considered as a community. As the basis of D-Map design, we need to detect communities of all the participants based on their reposting behaviors. The input graph of the map is the multi-edge reposting network of social media users, merged from all the reposting trees of the source weibos (Figure 6a). The dashed link connecting nodes in different trees indicates the connected nodes represent the same person. After the merging process, each node is one social media user and each edge represents that user A reposts from user B one time. There could be multiple edges between two nodes. To find the community structure of the multi-edge graph, we use degree-corrected stochastic blockmodels [30].

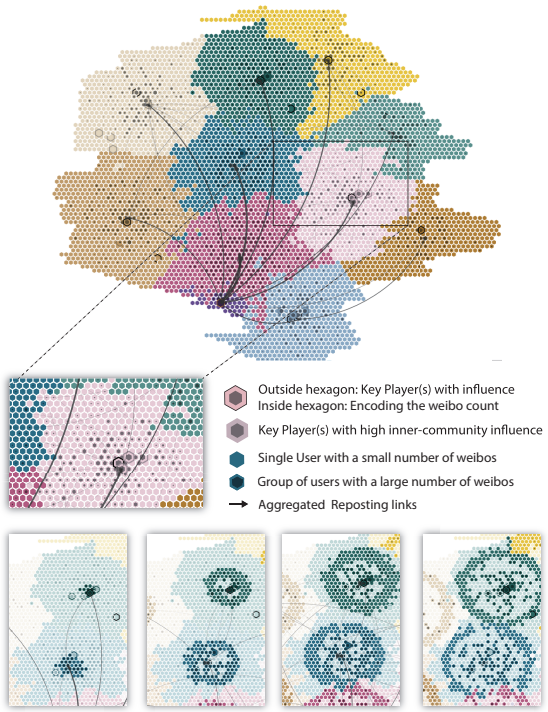

Figure 4: Visual encodings of D-Map of a central user. Each node represents a group of people, participating in the diffusion process by reposting weibos.

The advantage of this method, which fits our design goal, is that it can not only identify the node's community assignment but also find the interactions between the communities. In the other aspect, we do not rule out possibilities of using other algorithms. Let G be an undirected multi-edge graph on n nodes. It is assumed that there are K groups, and g_i is the group assignment of node i. Here we give the unnormalized log-likelihood function:

$$L(G|g) = \sum_{rs} m_{rs} log \frac{m_{rs}}{k_r k_s} \quad (1)$$

m_{rs} is the total number of edges between group r and group s. k_r, k_s are the sums of the degrees in group r and s, respectively. The goal is to maximize the probability on the group assignments of the nodes. The network is divided into an initial random set of k communities. By repeatedly moving a vertex from one group to another, the method will find a state with the highest score L.

K is determined when L is maximized. Following the works of [34], we can set a minimal and maximal range for K calculation. For large groups of people, e.g. more than 10,000, we set the range of K from 5 to 30. In our test, most users' community results fell in this range. Users can also adjust the range in different scenarios.

4.3 Visual Encoding

To avoid clutter, we choose the compact layout, with the candidates of mosaic map and voronoi-based tessellation. We choose the mosaic cartograms because they communicate data with countable integer units, which is easy for visual comparison [6]. We would like to choose a shape to minimize wasted space among items and maximize the area inside them. The triangle grid introduces two types of triangles - regular and inverted triangles, which may introduce ambiguities in visual representation. Square binning appears stretched out in the vertical and horizontal directions [13]. Other shapes with larger numbers of edges are too complex. The point and circle grid are not compact. Hexagons, which are widely used in the cartographic domain [26], are common in nature, further enhancing the

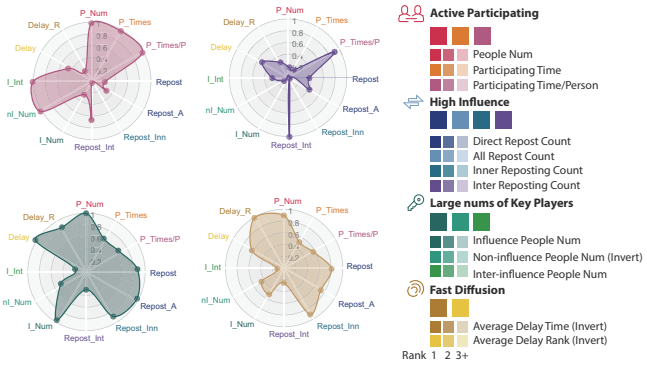

Figure 5: Color decision for D-Map. Four series of color encode the high-dimensional features of the community. The radar visualization shows the distribution of each dimension of selected communities.

aesthetic quality, familiarity, and acceptance from users [17]. Considering these factors, we finally choose to use a hexagon grid as the basis of D-Map.

In the map design, each node represents one person or a group of people with similar behaviors. Each color region with multiple nodes indicates a community (Figure 4). The central user is drawn with a highlighting orange stroke. The key players are determined by a threshold value of the number of reposting people. In our experiment, we set the threshold as the square root of the total amount of people in consideration. The key players are highlighted with an enlarged hexagon with a black stroke to indicate they have a stronger influence on others. Within each hexagon, there is a small hexagon, the size of which shows how many weibos these people have reposted. To avoid clutter, we show the aggregated links among communities by default and show individual links of selected people on demand. The width of a link encodes the number of all the repostings between two communities. The repostings include both the direct and indirect repostings. Users can control a threshold to filter the numbers of repostings to reduce the clutter. The nodes in each community are laid out from the inside out based on the relative time, indicating the dynamic diffusion process for each community (Figure 4). There is a design trade-off for such reordering. To gain the awareness of the critical temporal relationship, we may lose the topological relationship in the local cluster. To compensate for that, users can perceive the relationship by multiple interactions. Furthermore, users can still perceive the distances as relationships among different communities.

We use color regions to encode different communities. Region size represents the number of people in each community. We aim to generate a unique map for each central user showing the properties of his/her social networks. One important feature is to make the map comparable among multiple central users. We provide a color mapping scheme and size mapping functions to achieve this goal. As discussed in Section 3, there are four categories of important features, including participating, influence, key player, and diffusion process statistics. We designed four color series with different levels of details (Figure 5). In the design process, we consider both balanced perceptual properties of color [43] and the characteristics of the data. After setting the color, there are two approaches for mapping color to each community:

- **Project feature-vectors into the RGB space** We define the distance among high dimensions and adopt dimension reduction methods to obtain the color. The advantage is that the projection considers all the attributes. However, the drawback is that the color could be random and not easy to compare.
- **Select the pre-defined color of the most representative feature** We calculate the rankings of all features in all communi-

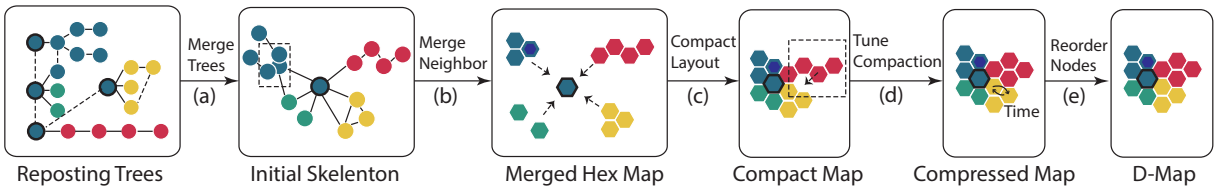

Figure 6: D-Map construction process. With the input of a series of source weibos from a central user and the reposting trees of these source weibos, we can merge them into a graph, compact the layout and reorder nodes in each community based on chronological order.

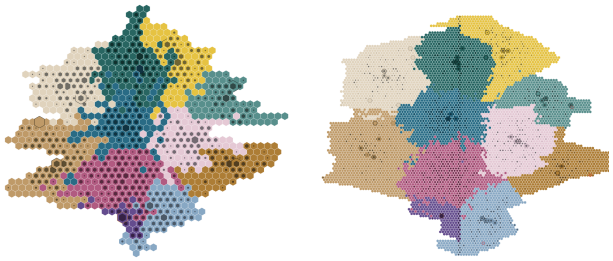

Figure 7: Merging thresholds for the size control. (a) Set the hex-bin number as 1000. (b) Maximize the number (9,000 in this case).

ties. For each community, we choose the dimension with the highest ranking among all the features of the community as the most representative feature. The drawback is the loss of information. However, we can get a comparable and carefully designed aesthetic color scheme to rectify that.

Considering the design trade-off between two methods, we choose the second one because being comparable and understandable is one of the most important goals for the map. For each community, we choose the corresponding ranked color in the color series of the most representative feature (Figure 5). To compensate the lost details, we provide an interactive radar-style visualization to illustrate the normalized distribution of each dimension. Each axis is a sub-category of feature, derived from data characteristics we discussed in Section 3. Each axis's name is the abbreviation of the corresponding sub-category with the same color on the right (Figure 5). Thus, users can understand why the color is chosen. Other candidates of representative features are also perceived.

4.4 Map Construction

With the input of the multi-edge network with detected communities (Figure 6a), the map construction procedure includes customized force-directed layout, nodes merging, layout compacting, layout tuning and reordering based on time (Algorithm 1).

To make people in the same community positioned together, we choose a force-directed layout with customized link settings [28]. Besides the original links between people, we add an artificial type of link in the graph. As illustrated before, we have a series of source weibos from a central user. We add edges among participants who repost the same source weibo. The edge-adding process makes people who repost the same weibo stay nearer, which may indicate they share the similar interest. Moreover, it adds the forces inside each community, which contributes to better separating communities in the final D-Map. In the next step, we merge the nodes within a distance threshold to reduce the visual complexity (Figure 6b). These nodes usually have similar behaviors, so they are forced together. We apply a hierarchical merge operation in each community. We calculate the pair-wised distances of each node. After sorting the distance values, we start merging two nodes with least distance value. Through repeatedly merging, we can get the merged nodes with expected granularity of hexagons. Users can adjust the granularity adapting to different scenarios (Figure 7). To reduce the computational complexity, we divide each community into multiple blocks equally and run the merging process for each of the block. Finally, we merge all the nodes in each block to get the final ones (Algorithm 1-Step 2).

After the merging process, we need to delete the blanks among nodes and make the layout compact (Figure 6c). We attract each node from different directions with the force strength to the central user node's position. Users can apply different dividing values of 360°. With an experienced value of 45°, we could attract the

Algorithm 1 D-Map Layout Algorithm

Input:
A list of people nodes V_i with initialized force-directed layout position (**step 1**, omitted) V_i.pos, i = 1, 2 ... n;
Detected community number C;
Expected output hexagon map size (hexagon number) S_h;
Dividing parameter for compacting the layout N, making 360° into N pieces;
Output:
A list of hexagon map points V_i, i = 1, 2 ...m;
1: **//Step 2: Merge the nodes that are close**
2: Calculate the $minX$, $maxX$, $minY$ and $maxY$ of V
3: Collecting nodes from V into each community V_c, c = 1, 2,..C
4: Equally divide V_c into M blocks
5: **for** $i = 0; i < M; i$++ **do**
6: Calculate the pair-wised Distance Matrix $DisMatrix_c$ of block M_i
7: $MergeNum = (V_c.length * S_h)/(V.length * M)$
8: Sort nodes in M_i of V_c, and merge the nearest $MergeNum$ nodes
9: **end for**
10: Merge all the nodes in M blocks of each community V_c and get S_h hexagons
11: **//Step 3: Compacting hexagons into the center**
12: Build N direction histograms $histogramDir$, each with direction range $[h/360°, h + 1/360°]$, h = 0, 1, ... N-1
13: Get the $centerNode$ position
14: Push each hexagon V_i (i = 1, 2 ..m) into each $histogramDir$ bin based on direction, sorting based on the nearer distance to the $centerNode$
15: Initialize a candidate position $queue$; $queue$.enqueue($centerNode$); $count$=0;
16: **while** $queue.length > 0$ and $count < V.length$ **do**
17: $currentNode$ = $queue$.dequeue()
18: **for** $i = 0; i <$ $currentNode.neighbors.length; i++$ **do**
19: **if** $currentNode.neighbors[i].notOccupied$ **then**
20: $queue$.enqueue($currentNode.neighbors[i]$)
21: $currentNode.neighbors[i].notOccupied$ = false
22: **end if**
23: **end for**
24: dir = DIR($currentNode$,$centerNode$)
25: $hexgon$=histogramDir[dir].top();
26: **if** $hexgon != NULL$ **then**
27: histogramDir.pop(); Set position of $hexgon$ as $currentNode$; $count$++
28: **end if**
29: **end while**
30: **//Step 4: Compact into rectangle**
31: Move hexagons to the center horizontally and vertically (repeat step 3, N = 4).
32: **//Step 5: Reorder based on time**
33: **for** V_c in each V **do**
34: Sort V_c based on the relative time of its source weibo
35: Calculate the geometry center of nodes in V_c
36: Map each hexagon V_{ci} from center out
37: **end for**

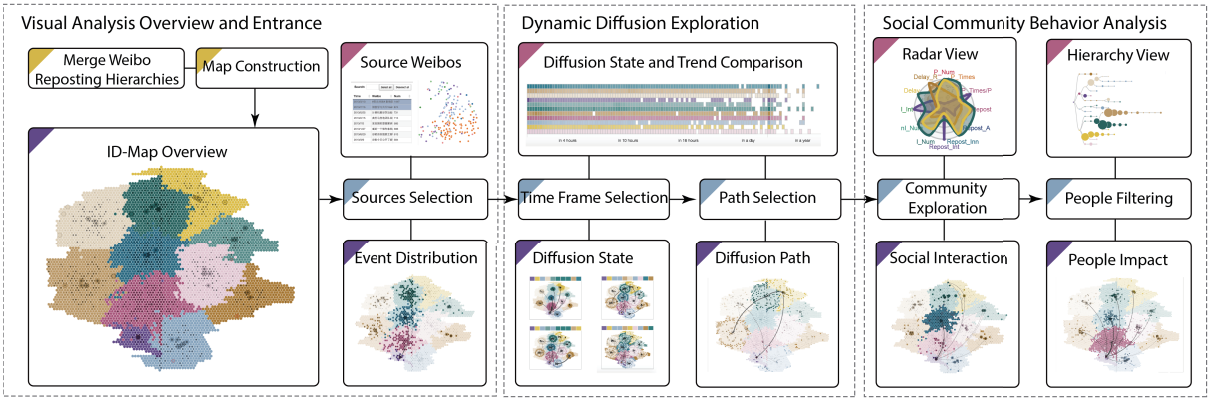

Figure 8: Visual analysis pipeline, illustrating how to explore users' D-Map with the augmented trend, hierarchy and high dimensional analysis. Diffusion path, social interactions and people impact can be found.

nodes while keeping the relative positions of the neighborhood. To achieve the attraction process, we use eight direction histograms storing nodes in each 45° range and pop up the nearest nodes to the center one by one (Algorithm 1-Step 3). After the attraction, sometimes a large number of nodes would be packed in a particular direction. To solve the problem, we apply a second round compacting process to make the layout into a rectangle, which saves spaces and increases the data-ink ratio (Figure 6d, Algorithm 1-Step 4). In each community, we calculate the relative time of each weibo compared with its source weibo. We set the minimal time for the node if it contains multiple weibos. We calculate the center of each community and map the nodes from the inside out according to their relative time (Figure 6e, Algorithm 1-Step 5).

D-Map is a customized visualization to represent the people participations in the social communities and describe the diffusion process. To enhance the analytical capability of D-Map from multiple aspects, we propose an interactive visual analytics system.

5 Visual Analytics Procedure

The visual analytics system combines D-Map, Souce Weibo Table View, Community Radar View, Hierarchical View, Timeline View and Small Multiple View (Figure 1). By analyzing multiple aspects of the weibo data, users can explore the diffusion process among communities systematically (Figure 8). The color is coherent in the system and mapped to the detected community (Figure 5).

5.1 Visual Analysis Overview and Entrance

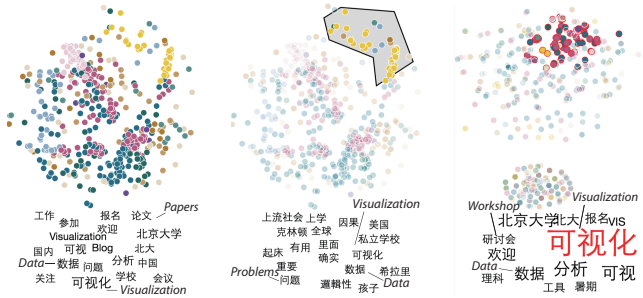

Figure 9: Weibo sources analysis with (a) reposting people distribution distance and (c) document distance. Interactions including (b) brushing and (c) keyword filtering are supported.

By projecting the source weibos from a central user into a 2D space, we provide a starting point for analysis. One of the research goals is to understand the characteristics of communities and which types of information some communities prefer. There are two key points - participating people and weibo content. In one aspect, users can analyze people relationships, which are reflected by the distributions of participants in reposting each source weibo. In the other aspect, users can explore the participants' preference for different keywords and content, to further understand the characteristics of communities. Therefore, we enable users to analyze the weibo sources from these two perspectives. By default, we construct a high-dimension vector for each source weibo. Each dimension is the people count of each community. Being consistent with Section 4.3, we choose the color of the community with the largest participating number to encode the source weibo. With the calculated high dimensional distance, we project the documents into a 2D space with t-SNE [45] (Figure 9a). From the content perspective, we first process the text of source weibo by word segmentation and remove the stop words. Stop words include the standard terms without specific meaning. To get the distance matrix, we adopt Term Frequency-Inverse Document Frequency (TF-IDF) [37] to create a weighted vector and measure the similarity of each source weibo based on the cosine distance between the vectors. Finally, we project the source weibos to the 2D space based on the content similarity with t-SNE (Figure 9c).

Interactions such as clicking and brushing selection (Figure 9b) are supported, and users can also click the keywords to select related weibos and reposting people (Figure 9c). Moreover, we provide a table view of source weibos with sorting, keyword searching, and filtering functions (Figure 1a). The selected source weibos will pop up for highlighting. Participants of the selected source weibos are highlighted on D-Map for the further exploration.

5.2 Dynamic Diffusion Exploration

We apply a Timeline View (Figure 1f) with a Small Multiple View (Figure 1g) to support the exploration of the dynamic diffusion process with D-Map. In the Timeline View, the y-axis is the detected community and the x-axis is the aligned relative timeline. Considering the short live span of each weibo, we show the first 24 hours of the reposting weibos with 80 percent of the time line width. We provide an animation function to fast-forward the diffusion process. We propose two methods to split the timeline and show the critical period range in the small multiples, based on percentile division and entropy-based division. We can use the percentile group to have an overview of information diffusion among communities (Figure 1g). We split the data with 25%, 50% and 75% amount threshold. For the entropy-based division, we use the Shannon entropy which measures the distribution's degree of dispersal or concentration of communities. For a given histogram $X = \{n_i, i = 1, ..., N\}$, community

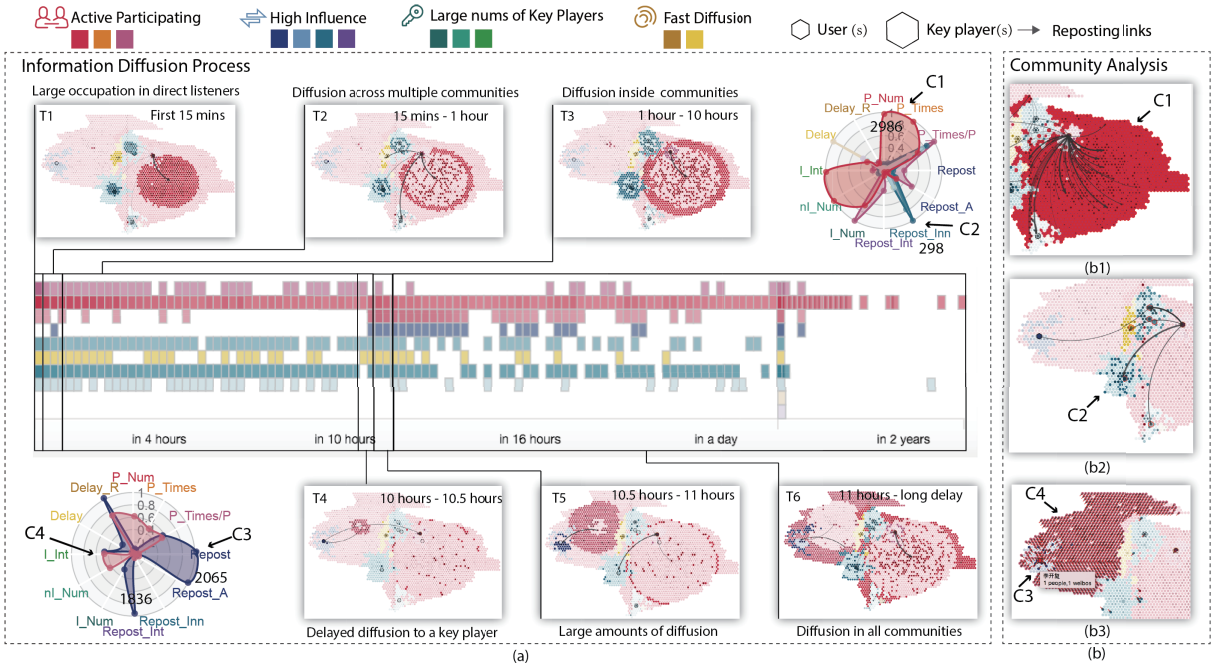

Figure 10: Dynamic diffusion pattern analysis. (a) Multiple diffusion stages are shown in the small multiples and the timeline. (b) By interactively exploring the communities, we find key players and summarize diffusion patterns.

i occurs n_i times in the sample. $S = \sum_{i=1}^{n}(n_i)$ is the total number of the community observations. H(X) is defined as the following:

$$H(X) = -\sum_{i=1}^{n}(n_i/S)log_2(n_i/S) \quad (2)$$

We aim to find time periods with low entropy values and large entropy changes. It is likely to have concentrated community distribution of repostings inside the community with low entropy values. The change of entropy indicates the source weibos are diversely reposted to others from few communities or vice versa. In the Small Multiple View, key players are shown as rectangles in the order of influenced people number (Figure 1g). When we click the thumbnail, the corresponding D-Map would be shown in the main window. On the selected D-Map, users can explore the particular diffusion paths of different people . People in the highlighted key players' diffusion path can be found as important participants.

5.3 Social Community Analysis

A series of reposting behaviors lead to information diffusion, which reflects social interactions. Specifically, our system supports the investigation of the characteristics of each community, inter-community diffusion and people impact.

First, in Community Radar View (Figure 1d1), the high dimensional features reflect the community characteristics. When users select nodes inside a community, the selected people number will be shown (Figure 1d2). Besides the statistics, an overview of inner community behaviors can be perceived as the arrow glyph design. These behaviors usually include single center diffusion (Figure 11e), or strong connections among community members (Figure 11f). The glyph design can also reduce the clutter by reducing the length of links. Second, by selecting a community on the map, the correlated communities would be highlighted . Thus, we can infer how much influence the community has, and how diverse the users' influence is. Also, the Hierarchical View aggregates the nodes of the same communities in the diffusion process, which helps users understand the position of a selected community in the hierarchical reposting tree (Figure 11c). Moreover, when users select multiple communities, features of communities can be compared interactively in the Community Radar View (Figure 11b). Third, by selecting the nodes on the map, we can investigate the people's direct reposting and reposted nodes . Diffusion path and key players can reflect the central users' impact (Figure 1c).

The visual analytics system is built with HTML5/Javascript, and the server-side processing is with Python and MongoDB. The client uses SVG with D3.js [4]. We crawled weibo data through the open APIs by Sina Weibo and constructed the reposting tree for each source data with Weibo Events Crawler [36]. The data is stored in MongoDB and provided with customized API for fetching data.

6 Case Study

We present three cases showing different functional aspects of our system. The addressed topics are of interest to sociology experts.

6.1 Case 1: Dynamic Diffusion Pattern Analysis

In this case, we explored the diffusion patterns among communities. We selected 300 weibos of one influential person and constructed a D-Map with 7,694 reposting weibos from 5,917 unique users (Figure 10). There were two largest groups, with 2,986 (C1) and 1,811 people (C4), shown in red color. By exploring the diffusion process, we can have a better understanding of how these communities formed and what their behavior patterns were.

There were two main diffusion states (Figure 10a). The first state included three stages (T1 - T3). In the first 15 minutes (T1), the central user posted weibos and mainly affected the direct listeners group C1. Later in one hour (T2), people in the surrounding communities reposted more weibos, while the weibos kept spreading inside C1. By selecting C2 (Figure 10b2) in T3, we found that it had the most inner spreading counts as 298. It indicated that people in C2 were active. In the later stage (in 10 hours), the reposts lasted and spread mainly within each community. Afterward, it transited to the second main state, which was also segmented into three important stages (T4 - T6). The purple community C3 with high influence reposted weibos from C1, and shortly had burst diffusions in C3 and C4 (T5). Further the information spread in all the communities (T6).

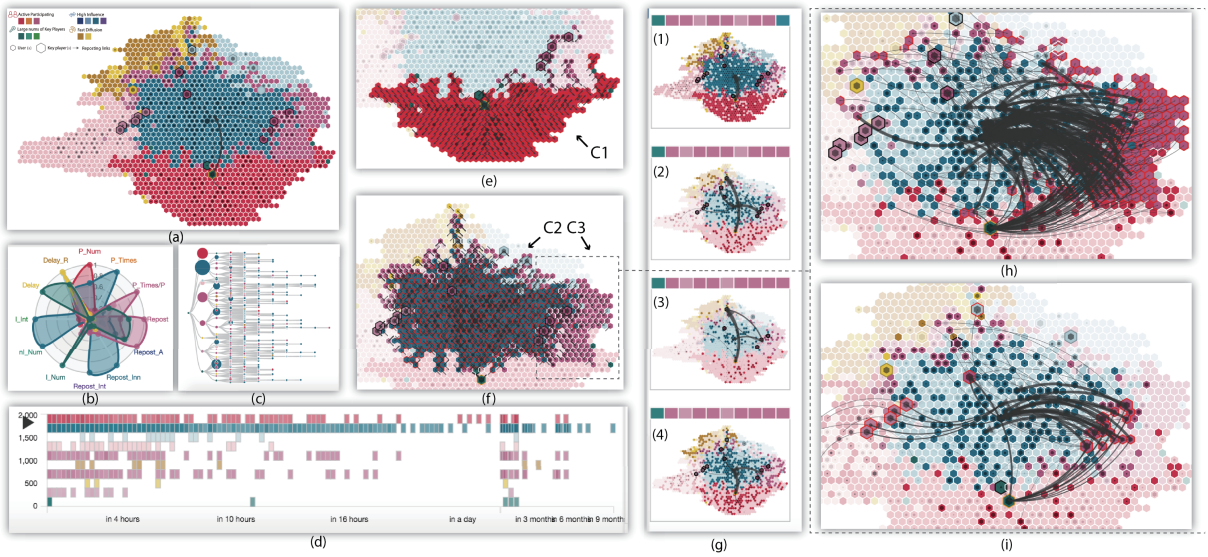

Figure 11: Community analysis on D-Map. Users can have an overview on the (a) D-Map and (b) Community Radar View. By selecting the community, users can explore the characteristics (e, f). (d) Timeline View, (c) Hierarchical View and (g) Small Multiple View are provided to investigate different aspects of the diffusion process. Users can explore the detailed (h) inter-community behaviors and (i) key player behaviors.

By further investigating the community property, we could tell the reasons of the communities' separation and behaviors. Besides the large common first-listeners (C1, Figure 10b1), the central user had another long-delayed reposting community (C3). By clicking the key player in C3 (Figure 10b3), we found he was one of the most influential persons on Sina Weibo, who had much more followers than the central user. Thus, we can be aware of the prominent level of different people, as well as the diffusion state changes over time.

6.2 Case 2: Social Community Behaviors Analysis

We studied one "We Media" who shared technique information on Sina Weibo. We extracted 79,013 weibos from 10,209 users, who participated in spreading his 500 source weibos in three months (Figure 11).

Seven of the ten communities had remarkable number of people (Figure 11a). The largest two communities consisted of 4,391 (red, C1) and 2913 (blue, C2) participants. We first investigated the community details by selecting all people in each group. In C1, people tended to repost mainly the weibos directly from the central user (Figure 11e), as the arrow glyphs' directions are uniform. Reposting behaviors in C2 were more irregular. They had multiple sources and strong communications inside the community. More interestingly, we found this community had large amounts of interactions with C3 community (pink) (Figure 11f). By exploring the timeline, we found that the central user posted weibos and mainly affected people in C1 and C2 in the first 8 hours. In 8 - 16 hours, people in C2 largely reposted C3's weibo, which indicated a second round bursting. The Timeline View and Hierarchical View also confirmed this phenomenon - people in C2 kept active for long (Figure 11d) and participated in second-round reposting (Figure 11c). In all, C2 people reposted 53,542 times, more active than people in C1 with 10,237 repostings. C3 had a significant influence on others and led to the highest reposting count - 10,310 (Figure 11b). Specifically, we concluded people in C3 had a larger influence on C2 (Figure 11h). To further investigate human behaviors, we could identify the key players who impacted people in C3 (Figure 11i).

In this case, we conclude the features of three unique communities - direct reposting people (C1), high influence people (C3) and active reposting people (C2). We also demonstrate the capability of investigating dynamic diffusion patterns among each community and telling stories about social interactions.

6.3 Case 3: People Portrait

To further evaluate D-Map, we tested more cases of influential accounts in Sina Weibo. We crawled 34 influential accounts from a wide range of fields. There are around 500,000 weibos originated from these influential accounts in five years. Two million people participated in all these weibos at least once.

We can select one central user and load all or a part of the weibos and their reposting weibos. Due to crawling limitations of APIs, we ran a filtered set of weibos to portrait each user. The controlled size of the source weibos was 500, with a range of 10,000 to 50,000 participating people. The parameters for running each case were the same, with the community range from 5 to 30, and the number of hexagon to be around 3,000. We find many patterns and select nine of them for illustration (Figure 12). From left to right, the number of detected communities increases. From top to bottom, the influences among communities become larger. Central users have a large amount of people belonging to the "first-listener" community, which is indicated by the red color. More interestingly, we can find different patterns among people based on their reposting structures.

First, central users with a few communities and weak inter-community influences usually turn out to be service accounts, which are not operating well (Figure 12a). Although they have a lot of followers and repostings, they cannot actively engage people or followers in further step repostings. Usually, they are some related service accounts reposting each other's weibos and some accounts are even bot accounts. Second, central users with larger communities but low inter-community influence are more likely to be businessmen in the social media (Figure 12c). They are good at creating topics and attract different kinds of people to repost. However, these people play a substantial central role, and there are weak inter-community influences among other communities. Third, central users with few number of communities but strong inter-community influences usually have one or some influential key players. Key players build up their "territory" on the central user's map, which forms a dual-center pattern (Figure 12d, g). Lastly, central users with active inter-community behaviors tend to have more equal-size community (Figure 12f, i). Each community has its key players, and they have connections with each other. These central users are likely to be domain experts in particular fields, with many followers in the same area and actively reposting each others' weibos.

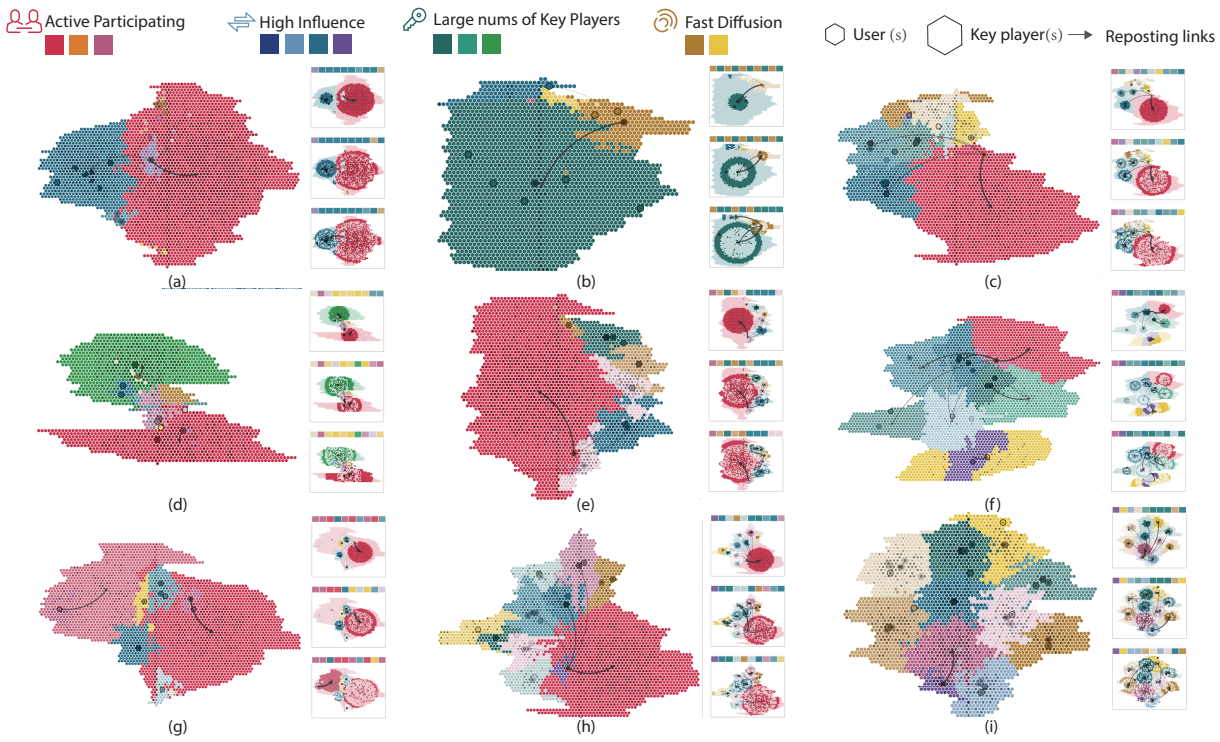

Figure 12: People portrait visualization from D-Map. We can differentiate people from multiple dimensions. Two dimensions showing here are the split community number and inter-community influence. We can see from the top-left, where there is merely communications among a few communities, to the bottom-right with the influence pattern in multiple, evenly-distributed communities.

The trend of diffusion pattern along the time is also different. For example, the businessman's diffusion pattern keeps almost the same as the beginning along the diffusion time. We build the hypothesis that the businessman keeps influencing the other communities, by reposting his previous weibos to gain the public attention again. Different from the businessman, the domain expert's weibos are reposted by the key players in different communities (likely to be other experts in the same field), and diffuse across multiple communities in later stages.

This case confirms the capability of portraiting and comparing social patterns of central users. We can check whether an account is influential or badly operated. We find interesting patterns such as dual-center diffusion (Figure 12d, g), strong-center role (Figure 12c) and strong interactions (Figure 12f, i).

7 Discussion

In this section, we discuss the pros and cons of the proposed D-Map visualization. In particular, D-Map, by placing nodes into a hexagon grid, eliminates node overlaps, thus producing a clear summarization of diffusion processes initiated by a central user. The visualization forms a diffusion map that portraits user's social behaviors and reveals his/her influence regarding spreading information in the social space. This visualization enables a dynamic exploration of the historical diffusion processes and facilitates a fast comparison of diffusion patterns.

Although novel and powerful, the current implementation of the D-Map design still has room for improvement. In particular, both the force-oriented initial layout algorithm and community detection method may introduce randomness to the final results, making the resulting map of the same data appear differently sometimes. There are two approaches to address this problem: (1) precisely controlling the initial parameters used in the force directed layout and community detection to reduce randomness and (2) employing optimization instead of heuristic algorithm for the layout. The other issue is that the link overlay introduces the clutter. We provide the filtering threshold and arrow glyph design to reduce the clutter.

We envision extending D-Map in several ways. First, we can consider multiple central users and build up a larger D-map, enabling the crowdsourcing participation for the event analysis. Second, we can combine more thematic information with the diffusion structure in both map construction and analysis process, which may provide more semantic-rich results. Third, based on the interesting patterns of different users, we can further extend D-Map with a prediction model. With a real time feed data source, we can predict the diffusion paths and targeted communities of people. Furthermore, we would conduct a systematical evaluation of the proposed method from sociology experts.

8 Conclusion

We propose a novel visualization method, D-Map, to visually summarize and explore central users' social networks. We map all the people reposting a central user's weibos to a hexagon map. Diffusion patterns and community interactions can be detected with a focus on key players and important diffusion paths. With a comprehensive visual analytics system, we evaluate our work with real-world social media data and find interesting patterns on understanding the unique features of individual's social impact.

Acknowledgements

The authors wish to thank Xiaolong Zhang, Ying Zhao, Chen Guo, Bowen Yu and the anonymous reviewers for their valuable suggestions and comments. This work is funded by NSFC Key Project No. 61232012 and the National Program on Key Basic Research Project (973 Program) No.2015CB352503. It is a continuous effort from previous funding of NSFC No. 61170204. This work is also supported by PKU-Qihoo Joint Data Visual Analytics Research Center.

REFERENCES

[1] D. Archambault, J. Abello, J. Kennedy, S. Kobourov, K.-L. Ma, S. Miksch, C. Muelder, and A. C. Telea. *Temporal Multivariate Networks*, pages 151–174. Springer International Publishing, 2014.

[2] D. L. Arendt and L. M. Blaha. SVEN: informative visual representation of complex dynamic structure. *CoRR*, abs/1412.6706, 2014.

[3] F. Beck, M. Burch, S. Diehl, and D. Weiskopf. A taxonomy and survey of dynamic graph visualization. *Comput. Graph. Forum*, 2016.

[4] M. Bostock, V. Ogievetsky, and J. Heer. D^3 data-driven documents. *IEEE Trans. Vis. Comput. Graph.*, 17(12):2301–2309, 2011.

[5] U. Brandes, M. Hoefer, and C. Pich. Affiliation dynamics with an application to movie-actor biographies. In *Proceedings of EuroVis*, pages 179–186, 2006.

[6] R. G. Cano, K. Buchin, T. Castermans, A. Pieterse, W. Sonke, and B. Speckmann. Mosaic drawings and cartograms. *Comput. Graph. Forum*, 34(3):361–370, 2015.

[7] N. Cao, Y. Lin, X. Sun, D. Lazer, S. Liu, and H. Qu. Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2649–2658, 2012.

[8] N. Cao, Y. R. Lin, F. Du, and D. Wang. Episogram: Visual summarization of egocentric social interactions. *IEEE Computer Graphics and Applications*, PP(99):1–1, 2015.

[9] N. Cao, Y.-R. Lin, and D. Gotz. Untangle map: Visual analysis of probabilistic multi-label data. *IEEE Trans. Vis. Comput. Graph.*, 22(2):1149 – 1163, 2016.

[10] N. Cao, L. Lu, Y.-R. Lin, F. Wang, and Z. Wen. Socialhelix: visual analysis of sentiment divergence in social media. *Journal of Visualization*, 18(2):221–235, 2015.

[11] N. Cao, C. Shi, W. S. Lin, J. Lu, Y. Lin, and C. Lin. Targetvue: Visual analysis of anomalous user behaviors in online communication systems. *IEEE Trans. Vis. Comput. Graph.*, 22(1):280–289, 2016.

[12] N. Cao, J. Sun, Y. R. Lin, D. Gotz, S. Liu, and H. Qu. Facetatlas: Multifaceted visualization for rich text corpora. *IEEE Trans. Vis. Comput. Graph.*, 16(6):1172–1181, Nov 2010.

[13] D. B. Carr, R. J. Littlefield, and W. L. Nichloson. Scatterplot matrix techniques for large n. In *JASA*, pages 297–306, 1986.

[14] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi. Measuring user influence in Twitter: The million follower fallacy. In *Proceedings of ICWSM*, 2010.

[15] W. Cui, S. Liu, Z. Wu, and H. Wei. How hierarchical topics evolve in large text corpora. *IEEE Trans. Vis. Comput. Graph.*, 20(12):2281–2290, 2014.

[16] T. N. Dang, N. Pendar, and A. Forbes. Timearcs: Visualizing fluctuations in dynamic networks. *Comput. Graph. Forum*, 2016.

[17] M. Emmer. *The Visual Mind II (Leonardo Books)*. The MIT Press, 2005.

[18] S. Fortunato. Community detection in graphs. *Physics reports*, 486(3):75–174, 2010.

[19] D. Fried and S. G. Kobourov. Maps of computer science. In *2014 IEEE Pacific Visualization Symposium*, pages 113–120, March 2014.

[20] E. R. Gansner, Y. Hu, and S. G. Kobourov. GMap: Drawing graphs as maps. In *Proceedings of the 17th International Conference on Graph Drawing*, pages 405–407, 2010.

[21] E. R. Gansner, Y. Hu, and S. C. North. Interactive visualization of streaming text data with dynamic maps. *J. Graph Algorithms Appl.*, 17(4):515–540, 2013.

[22] A. Guille, H. Hacid, C. Favre, and D. A. Zighed. Information diffusion in online social networks: a survey. *SIGMOD Record*, 42(2):17–28, 2013.

[23] J. Heer and D. Boyd. Vizster: Visualizing online social networks. In *IEEE Symposium on Information Visualization.*, pages 32–39, 2005.

[24] N. Henry and J.-D. Fekete. Matlink: Enhanced matrix visualization for analyzing social networks. In *IFIP Conference on Human-Computer Interaction*, pages 288–302. 2007.

[25] N. Henry, J.-D. Fekete, and M. J. McGuffin. Nodetrix: a hybrid visualization of social networks. *IEEE Trans. Vis. Comput. Graph.*, 13(6):1302–1309, 2007.

[26] Hexagonal maps. http://cartonerd.blogspot.com/2015/05/helecxagon-mapping.html.

[27] M. Hu, S. Liu, F. Wei, Y. Wu, J. Stasko, and K.-L. Ma. Breaking news on Twitter. In *Proceedings of the SIGCHI*, pages 2751–2754, 2012.

[28] Y. Hu. Efficient and high quality force-directed graph drawing. *The Mathematica Journal*, 10:37–71, 2005.

[29] A. Java, X. Song, T. Finin, and B. Tseng. Why we Twitter: Understanding microblogging usage and communities. In *Proceedings of the WebKDD/SNA-KDD*, pages 56–65, 2007.

[30] B. Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *CoRR*, abs/1008.3926, 2010.

[31] A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller. Twitinfo: Aggregating and visualizing microblogs for event exploration. In *Proceedings of the SIGCHI*, pages 227–236, 2011.

[32] D. Mashima, S. Kobourov, and Y. Hu. Visualizing dynamic data with maps. *IEEE Trans. Vis. Comput. Graph.*, 18(9):1424–1437, 2012.

[33] L. Nachmanson, R. Prutkin, B. Lee, N. H. Riche, A. E. Holroyd, and X. Chen. GraphMaps: Browsing large graphs as interactive maps. *CoRR*, abs/1506.06745, 2015.

[34] T. P. Peixoto. The graph-tool python library. *figshare*, 2014.

[35] M. Pennacchiotti and A. Popescu. A machine learning approach to Twitter user classification. In *Proceedings of ICWSM*, 2011.

[36] D. Ren, X. Zhang, Z. Wang, J. Li, and X. Yuan. Weiboevents: A crowd sourcing weibo visual analytic system. In *Pacific Visualization Symposium (PacificVis) Notes,*, pages 330–334, March 2014.

[37] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., 1986.

[38] T. Schreck and D. A. Keim. Visual analysis of social media data. *IEEE Computer*, 46(5):68–75, 2013.

[39] L. Shi, H. Tong, J. Tang, and C. Lin. Flow-based influence graph visual summarization. *CoRR*, abs/1408.2401, 2014.

[40] M. A. Smith, B. Shneiderman, N. Milic-Frayling, E. Mendes Rodrigues, V. Barash, C. Dunne, T. Capone, A. Perer, and E. Gleave. Analyzing social media networks with nodexl. In *Proceedings of the ICCT*, pages 255–264, 2009.

[41] G. Sun, Y. Wu, S. Liu, T.-Q. Peng, J. Zhu, and R. Liang. Evoriver: Visual analysis of topic coopetition on social media. *IEEE Trans. Vis. Comput. Graph.*, 20(12):1753–1762, Dec 2014.

[42] J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD*, pages 807–816, 2009.

[43] M. Tennekes and E. de Jonge. Tree colors: Color schemes for tree-structured data. *IEEE Trans. Vis. Comput. Graph.*, 20(12):2072–2081, Dec 2014.

[44] R. Tinati, L. Carr, W. Hall, and J. Bentwood. Identifying communicator roles in Twitter. In *Proc. of WWW*, pages 1161–1168, 2012.

[45] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

[46] C. Vehlow, F. Beck, P. Auwärter, and D. Weiskopf. Visualizing the evolution of communities in dynamic graphs. *Comput. Graph. Forum*, 34(1):277–288, Feb. 2015.

[47] F. Viégas, M. Wattenberg, J. Hebert, G. Borggaard, A. Cichowlas, J. Feinberg, J. Orwant, and C. Wren. Google+Ripples: A native visualization of information flow. In *Proceedings of WWW*, pages 1389–1398, 2013.

[48] Y. Wu, S. Liu, K. Yan, M. Liu, and F. Wu. Opinionflow: Visual analysis of opinion diffusion on social media. *IEEE Trans. Vis. Comput. Graph.*, 20(12):1763–1772, 2014.

[49] R. Xiong and J. Donath. Peoplegarden: Creating datnulla portraits for users. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*, UIST '99, pages 37–44, 1999.

[50] P. Xu, Y. Wu, E. Wei, T. Peng, S. Liu, J. J. H. Zhu, and H. Qu. Visual analysis of topic competition on social media. *IEEE Trans. Vis. Comput. Graph.*, 19(12):2012–2021, 2013.

[51] M. Yang and R. P. Biuk-Aghai. Enhanced hexagon-tiling algorithm for map-like information visualisation. In *Proceedings of VINCI*, pages 137–142, 2015.

[52] J. Zhao, N. Cao, Z. Wen, Y. Song, Y. Lin, and C. Collins. Fluxflow: Visual analysis of anomalous information spreading on social media. *IEEE Trans. Vis. Comput. Graph.*, 20(12):1773–1782, 2014.