TimesliceVis: an online system for analyzing temporal features in dynamic visualizations

Song Wang
College of Computer Science and Technology, Southwest University of Science and Technology,
Mianyang China
wangsong@swust.edu.cn

Yifan Zhang
College of Computer Science and Technology, Southwest University of Science and Technology,
Mianyang China
as466931001@qq.com

Yonghui Chen
College of Computer Science and Technology, Southwest University of Science and Technology,
Mianyang China
chenyonghui@swust.edu.cn

Abstract

To support stable dynamic graph layout and reveal time features hidden in the data, we propose a joint discovery method that extracts the skeleton of a dynamic graph by combining graph structural information and clustering algorithms, and explores time-sequence data features mainly based on graph community patterns. In addition, we propose a discrete exploration method that helps users better understand roles, connections, and interactions in the network, through dimensionality reduction views. We integrate these methods to design a visualization ana-lysis system called TimesliceVis, which helps users construct online dynamic graphs and explore and understand the temporal features of network graphs. Finally, through case studies on real-world data, we demonstrate the practicality and effectiveness of TimesliceVis in perceiving and interpreting network based on structural information.

Keywords: Network visualization, temporal features, dynamic-Graph layout, LDA dimensionality reduction, HAC clustering

1. INTRODUCTION

Dynamic graph visualization preserves structure during animation. Methods include animation, timeline, and mixed approaches[1]. Online methods, a type of animation, use previous layout information to position new nodes, with stronger expressiveness. However, limitations exist in expressing complex information, data representation, and inter-action methods.

To visualize complex Twitter network data, compression methods like feature extraction are needed. Our paper proposes a multidimensional exploration method that incorporates time-series information into online incremental dynamic graphs. We use clustering algorithms to group nodes into topic communities, embed time-informed graphs into topic network graphs, and visualize changes and correlations using dimensionality reduction algorithms. We demonstrate the feasibility and effectiveness of our method through real data case experiments.The main contributions of this paper are: proposing an online incremental graph layout

algorithm that maintains user mental maps, designing a feature-based community classi-fication time-series demonstration method, proposing a multidimensional joint explo-ration method for visualizing time-series features of dynamic graphs.

2. RELATED WORK

2.1 Visualization Methods for Temporal Data Feature

Online methods use initial layouts to create new layouts at the next time step and enhance aesthetics through computation. These layout problems arise from interactive and real-time monitoring of dynamic graphs with unknown time series. Online dynamic graph layout methods include Bayesian decision theory combined with force-directed layout by Brandes et al. [2], global structure preservation with user-modifiable continuous layouts by Frishman et al.[3] Lin et al. [4] proposed an online algorithm with a costly cost function considering aesthetics and stability, old and stable structure preservation using vertex age by Gorochowski et al. [5], improved drawing efficiency through initial vertex displacement by Hayashi et al. [6], node similarity query method for large-scale dynamic graphs proposed by Wang et al. [7], and movement simulation as an inverse Markov process for convergence distance constraint by Sheng et al. [8].Specialized visualization tools are needed for temporal data, including History Flow [9] for text editing, time curves [10], and TextFlow [11] for topic mining. General techniques such as small multiples [12], time planes, animation, and 3D cubes can be used, but complex visualization may suffer from data overload. Nakazawa et al. [13] used clustering and different colors for time, while Jingming et al. [14] proposed an interactive display method based on collaborative network visualization, mapping nodes and edges onto the time axis. Cakmak E et al. [15] proposed a multiscale snapshot approach for handling large-scale, high-dimensional dynamic graph data, along with a multiscale visual summarization technique for simplifying the visualization and analysis of dynamic graph data. Federico P et al. [16] proposes the VATSON to integrates visualization and analysis to aid understanding of complex temporal relationships in social networks. Ahn J et al. [17] propose a novel method for temporal visualization that aids researchers in gaining deeper insights into social interactions and relationship dynamics in social networks.

2.2 Graph feature extraction methods

Hierarchical Agglomerative Clustering (HAC) merges the closest clusters until a termination criterion is met. Jafarzadegan et al. [18] combined different levels of clustering using PCA. HAC is flexible, as it doesn't require a fixed K value like K-MEANS and KNN. Dimensionality reduction is often necessary to reduce computational complexity. Principal Component Analysis (PCA)[19] maps high-dimensional data to a low-dimensional space, while Linear Discriminant Analysis (LDA) extracts topics from a document set. Liu et al. [20] used graph signal processing and semi-supervised learning for feature extraction, while Xiao et al. [21] used convolutional neural networks. Wang Y et al. [22] propose a novel non-uniform time-slicing approach for dynamic graphs based on visual complexity to aid users in analyzing and interpreting graph evolution.

To summarize, displaying relationships between temporal data in incremental layouts can be challenging for maintaining users' mental maps. Analyzing dynamics of temporal node-link diagrams and integrating temporal features into analysis, as well as designing visualization modules for dynamic graph temporal features, are all challenges. Existing solutions include heat maps, network diagrams, and changing node sizes/colors to display changes in link strength or number.

3. METHOD

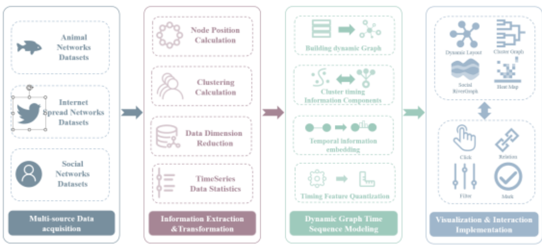

Figure 1 : System Flowchart

This paper proposes a collaborative analysis process As shown in Figure 1. in four steps for online dynamic graph visualization and exploring temporal features of network data: (1) Collect data from various sources such as social network data, animal relationship network, information propagation network, etc. (2) Extract and transform information using clustering and dimension reduction algorithms to explore network graph features and extract temporal features. (3) Construct a model linking network flow data and dynamic graph, embed time information into the network graph, and link clusters and time features. (4) Interactive visualization demonstration using various components such as dynamic graphs, clustering diagrams, theme river graphs, heat maps, etc., to discover temporal features and mark changes between related nodes or clusters through click events.

3.1 Online Dynamic Graph Layout Algorithm

We propose a constrained range for new node convergence and a synchronized constraint edge layout algorithm for online dynamic graphs to address time performance issues in incremental layouts while preserving the user's mental map. Inspired by Gorochowski[5] and others, we introduce the concept of levels to restrict node motion in dynamic graphs, reducing convergence time. At time t, a node's level is set, with the initial level of new nodes being 1. After dynamic graph data iteration, each node's level is reassessed.

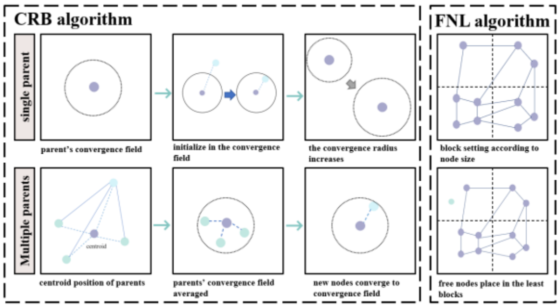

Figure 2 : Node-Adding algorithm

This article proposes a centroid region binding (CRB) algorithm for incremental layouts, specifically for single-parent nodes in new nodes. Position data is randomly generated on the canvas based on the parent node's location in different block areas. The new node then converges to the region divided by energy level according to its parent node, with a certain convergence radius and positioning direction to reduce total layout energy. For multiple parent nodes, the centroid position of the combined polygon formed by the parent node group and the new node group is determined using the polygon centroid formula. The centroid position is treated as a pseudo-parent node, and the new node converges towards the centroid in the direction of the pseudo-parent node to achieve local convergence of parent nodes towards the centroid. To improve view readability, the FNL (Free Node Layout) algorithm is proposed for free nodes, which are not connected to their parent nodes or any reserved nodes from the previous time step. The FNL algorithm optimizes the distance between new nodes to prevent node stacking and edge crossing. The canvas area is divided into K equal regions based on data scale, and free nodes are placed in the least populated block. If free nodes are added in pairs, the position of the preceding node is randomly generated, and the CRB algorithm is used to converge the succeeding node to the convergence domain of the preceding node, ensuring that attributes of newly added nodes are related to the nodes in the view.

In this paper, "additional edges" are newly added edges in the graph data. However, generating these edges can create many crossings, which reduces readability. To address this, the paper proposes the Community Node Drift (CND) algorithm (Figure 3). CND considers the community and position changes of related points to minimize changes to the community layout while increasing additional edge connections. We use the CCBD algorithm (Figure 4) to partition nodes into communities based on their degree, avoiding giant and discrete communities. CND's design philosophy is to reduce edge crossings by adjusting community positions. If additional edges are within the same community, the community is subdivided and relocation convergence is calculated.

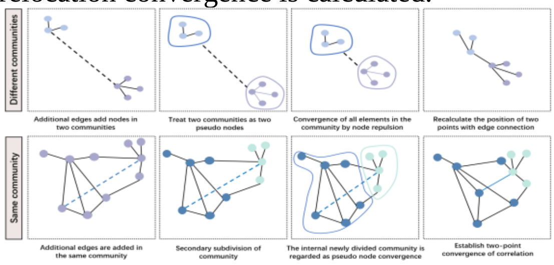

Figure 3: CND Algorithm

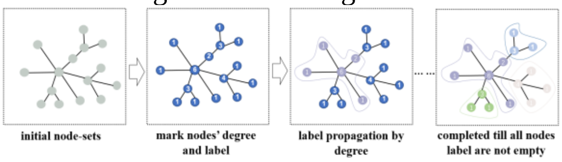

Figure 4: CCBD

3.2 Dynamic Graph Temporal Feature Extraction Method

The HACed algorithm (Figure 5), which combines HAC with edge propagation algorithms, improves clustering results, computational efficiency, scalability, and flexibility. It can be customized for specific datasets and applications. However, the algorithm still faces the issue of different initial node selections for community classification. To address this, we introduced degree sorting rules into the clustering method and combined them with node degree sorting HAC, resulting in improved accuracy, reduced computational complexity, and increased interpretability. The HACed algorithm considers both Euclidean distance and node degree as joint decision criteria, selecting nodes with larger degrees as initial values when Euclidean distance is

similar, resulting in more consistent communities and facilitating user judgment of different themes.
The challenge in representing temporal data in a dynamic network topology graph is designing a clear and intuitive graphic that can depict node relationships and changes over time. Current methods, such as color saturation and text labels, have limitations and may cause misunderstandings. To address this, we propose using a discrete bar graph to visualize time attributes and using different color saturation to distinguish time steps. Our design embeds attribute value changes between nodes into edge links(figure 6), and the color changes based on the magnitude of the attribute change. Saturation values for the segments are determined by sample variance.

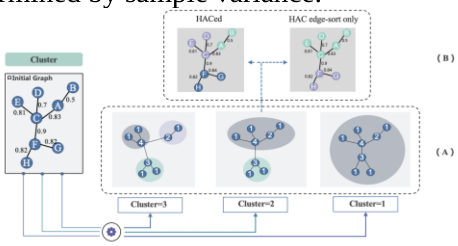

Figure 5: HACed

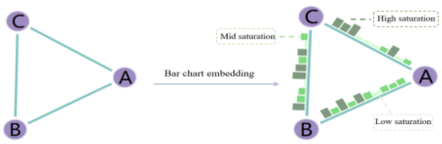

Figure 6: Embedded representation

We designed a 1.5-dimensional collision river map as a more effective visualization method than a two-dimensional dynamic node-link diagram to illustrate changes in network information after dynamic increments. This representation captures time information and changes in node relationships more effectively than one-dimensional or two-dimensional representations. The 1.5-dimensional repress-entation combines time information and changes in node relationships, making it easier for users to understand time patterns and trends of the graph. This visualization method is especially useful for analyzing dynamic graphs and identifying potential relationships.
To explore position data changes of the community-based two-dimensional node graph at different time points in combination with the HACed algorithm, we introduced the Latent Dirichlet Allocation (LDA) technique to reduce the dimensionality of the network topology graph. LDA dimensionality reduction technology can help users more effectively analyze text data, reduce data dimensions while retaining the main information, and improve interpretability of the results.
Our visualization module uses the y-axis for node projection on the LDA plane and the x-axis for displaying time step data. LDA classifies nodes based on topic attributes, as shown in Figure 7. The perpendicular projection of all nodes in a node cluster onto the LDA line represents the community's relative position at that time step. Community positions change on the y-axis as time steps increase. We calculate "steady-state" or "active" communities by measuring displacement at different time steps..

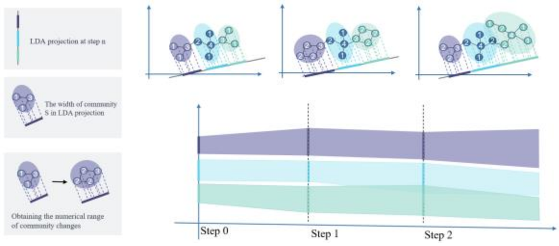

Figure 7: LDA dimensionality river map.

4. SYSTEM AND CASE STUDY

The TimesliceVis system is an online tool for dynamic graph temporal feature analysis. It uses a top-down approach with a comprehensive temporal feature analysis interpreter to help users obtain temporal feature information from their target network interactively and progressively. The system has six main components: (Figure 8A) a dynamic node-link diagram displaying information from the target dynamic network, ((Figure 8B) views visualizing the backbone of the diagram obtained through graph clustering, (Figure 8C) showing the closeness of inter-cluster connections over time, (Figure 8D) providing a reduced-dimensional representation of communities in the target dynamic network, (Figure 8E) serving as an information and control panel for obtaining global and single-node information based on user-set parameters, and (Figure 8F) displaying dynamic value variables and other network-related information.

4.1 MVBT datasets

The MVBT dataset, collected by Davis et al. [23] at BHP, is a dynamic network data model of mammal information (voles). It was observed multiple times from March to November, with data size increasing from 25 nodes and 33 edges to a peak of 163 nodes and 253 edges. We analyzed the temporal features of the MVBT dataset, shown in Figure 8. The visual representations reveal changes in relationships between voles over time. The community network (Figure 8B) indicates high genetic correlation among most voles, possibly forming a subspecies, with some isolated communities suggesting population diversity. The individual relationship between vole communities (Figure 8C) shows low coupling relationships, indicating breeding cycles. The temporal process of vole groups on the LDA dimensionality reduction graph (Figure 8D) reveals relatively high genetic correlation among vole groups in the BHP generation. Changes in group composition over time suggest group recombination, with some vole groups disappearing while others expanding on the river map.

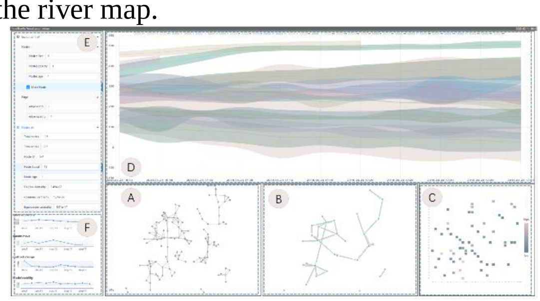

Figure 8：MVBT dataset visualization

4.2 Newcomb datasets

Newcomb Datasets contain data on Newcomb's personal social life, represented as a network with nodes for individuals and edges for social relationships [24]. In Figure 9(A), node color changes from step 5 to step 15 indicate stable social circles around Newcomb in terms of people and connections. Figure 9(B-a) shows a giant community connected to almost all sub-communities, except for one sub-community with no social relationship with the main circle. However, Figure 9(B-b) shows a connection between two closely related communities. We selected the number 3 community, which has a relationship with the main circle, as shown in Figure 9(C-a). In the third time step, the river projection width of the two selected communities has reduced, possibly due to members joining other communities. According to Figure 9(C-b), connectivity between members within the communities remains high and stable, indicating maintained social relationships between the two communities.

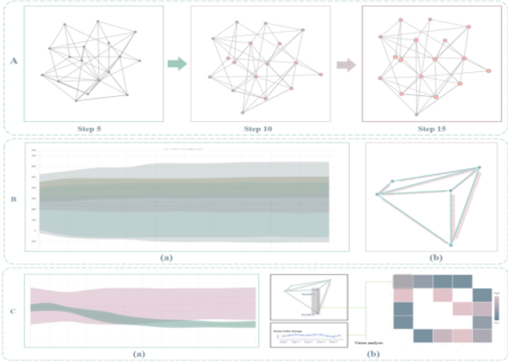

Figure 9：Newcomb datasets

5. CONCLUSION

In this paper, we propose a dynamic graph layout method that restricts node and edge movements. We introduce a visualization approach called TimesliceVis, which explores the temporal features of dynamic graphs based on embedded temporal connectivity patterns and reduced-dimensional community distribution encoding. Our system integrates dynamic data visualization and temporal feature analysis capabilities. Through interactive exploration, users can gradually gain insights into the structural information of nodes and analyze changes in node communities in dynamic networks. In the future, we plan to adapt our system for large-scale datasets to broaden its applicability, and also consider incorporating rich contextual information for each node.

References

[1] Beck F., Burch M., Diehl S., et al. A taxonomy and survey of dynamic graph visualization. Computer Graphics Forum 36, 1 (2017), 133–159.

[2] Brandes U, Wagner D. A Bayesian paradigm for dynamic graph layout[M] //Graph Drawing, Heidelberg: Springer, 1997: 236-247

[3] Frishman Y, Tal A. Online dynamic graph drawing[J]. IEEE Transactions on Visualization and Computer Graphics, 2008, 14(4): 727-740

[4] Lin C C, Lee Y Y, Yen H C. Mental map preserving graph drawing using simulated

annealing[J]. Information Science, 2011, 181(19): 4253-4272
[5] Gorochowski T E, Dibernardo M, Grierson C S. Using aging to visually uncover evolutionary processes on networks[J]. IEEE Transactions on Visualization and Computer Graphics, 2012, 18(8): 1343-1352
[6] Hayashi A, Matsubayashi T, Hoshide TI, et al. Initial positioning method for online and real time dynamic graph drawing for time varying data[C] //Proceedings of the 17th International Conference on Information Visualisation. Washington D C: IEEE Computer Society Press, 2013: 435-444
[7] Junlu W, Yulong S, Baoyan S. Node Similarity Top-k Query of the Large-Scale Dynamic Graph With Weak Repeated Path Constraint[J]. IEEE Access, 2019, 7: 64431-64441
[8] Sheng S Y, Chen S T, Dong X J, et al. Inverse Markov Process Based Constrained Dynamic Graph Layout[J]. Journal of Computer Science and Technology, 2021, 36(3): 707-718.
[9] Viégas F B, Wattenberg M, Dave K. Studying cooperation and conflict between authors with history flow visualizations[C] //Proceedings of the SIGCHI conference on Human factors in computing systems. 2004: 575-582.
[10] Bach B, Shi C, Heulot N, et al. Time curves: folding time to visualize patterns of temporal evolution in data. IEEE T rans Visualizat Comp Graph 2015; 22(1): 559–568.
[11] Cui W , Liu S, T an L, et al. TextFlow: towards better understanding of evolving topics in text. IEEE T rans Visualizat Comp Graph 2011; 17(12): 2412–2421.
[12] Bach B, Dragicevic P, Archambault D, et al. A review of temporal data visualizations based on space-time cube operations. In: Borgo R, Maciejewski R and Viola I (eds) Eurographics Conference on Visualization (EuroVis). Swansea, Wales, United Kingdom: The Eurographics Association, 2014, pp. 23–41.
[13] Nakazawa R, Itoh T and Saito T. A visualization of research papers based on the topics and citation net- work. In: 2015 19th international conference on information visualisation (IV), Barcelona, 22–24 July, pp. 283–289. New York: IEEE.
[14] Jing M, Li X, Zhang L. Interactive temporal display through collaboration networks visualization[J]. Information Visualization, 2019, 18(2): 268-280.
[15] Cakmak E, Schlegel U, Jäckle D, et al. Multiscale snapshots: Visual analysis of temporal summaries in dynamic graphs[J]. IEEE Transactions on Visualization and Computer Graphics, 2020, 27(2): 517-527.
[16] Federico P, Aigner W, Miksch S, et al. A visual analytics approach to dynamic social networks[C]//Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies. 2011: 1-8.
[17] Ahn J, Taieb-Maimon M, Sopan A, et al. Temporal visualization of social network dynamics: Prototypes for nation of neighbors[C]//Social Computing, Behavioral-Cultural Modeling and Prediction: 4th International Conference, SBP 2011, College Park, MD, USA, March 29-31, 2011. Proceedings 4. Springer Berlin Heidelberg, 2011: 309-316.
[18] Jafarzadegan M, Safi-Esfahani F, Beheshti Z. Combining hierarchical clustering approaches using the PCA method[J]. Expert Systems with Applications, 2019, 137: 1-10
[19] Dunteman G H. Principal components analysis[M]. Sage, 1989.
[20] Liu Z, Lai Z, Ou W, et al. Structured optimal graph based sparse feature extraction for semi-supervised learning[J]. Signal Processing, 2020, 170: 107456.
[21] Xiao G, Li J, Chen Y, et al. MalFCS: An effective malware classification fram-ework with automated feature extr-action based on deep convolutional neural netw-orks[J]. Journal of Parallel and Distributed Computing, 2020, 141: 49-58.
[22] Wang Y, Archambault D, Haleem H, et al. Nonuniform timeslicing of dynamic graphs based on visual complexity[C]//2019 IEEE Visualization Conference (VIS). IEEE, 2019: 1-5.
[23] Davis, Stephen, et al. "Spatial analyses of wildlife contact networks." Journal of the Royal Society Interface 12.102 (2015): 2014100
[24] Newcomb T M. The acquaintance process: Looking mainly backward[J]. Journal of Personality and Social Psychology, 1978, 36(10): 1075.