RESEARCH

Spatiotemporal Graph k-means

Devavrat Vivek Dabke[1*†] and Olga Dorabiala[2†]

*Correspondence:
ddabke@princeton.edu
[1]Program in Applied and Computational Mathematics, Princeton University, Fine Hall, Princeton, NJ, USA
Full list of author information is available at the end of the article
†Equal contributor

Keywords: community detection; dynamic graphs; graph clustering

Introduction

Dynamic graphs are becoming increasingly prevalent mathematical structures as we collect more detailed data on the world around us. Though graphs have traditionally been studied as static objects, the dynamic setting better captures the many systems that evolve over time. Also called "time-varying" or spatiotemporal graphs, they extend static graphs by permitting edges to change over time, and they inherently reflect many systems, e.g., road networks, online communities, and epidemic spread. Since they are much less understood than their static counterparts, they pose an exciting and rich area of study.

Much of the literature on dynamic graphs focuses on extending well-known concepts from the static case like connectivity [1], optimal routing [2], induced dynamical systems [3], and more. Our work fits into this foundational literature by extending the notion of vertex clustering for the purpose of community detection. Graph clustering is a fundamental tool for network analysis, with applications across the social and natural sciences, and we seek to bring this tool to the dynamic setting. In dynamic graph clustering, we find a partition of graph vertices that takes into account both spatial similarity—so that there are many edges within a cluster and relatively few between clusters—and temporal similarity so clusters stay consistent over time. These partitions help us detect latent community structures.

In this paper, we propose a method we call spatio-temporal graph k-means (STGkM) that is able to track the multi-scale relationships between graph vertices. STGkM applies a two-phase clustering approach, wherein the first phase outputs an assignment for each vertex at every time step and the second phase produces a single, long-term partition of vertices based on historical cluster membership. STGkM allows us to identify communities of interest, as well as automatically track their evolution over time. To validate our method, we provide certain theoretical guarantees and showcase the utility of STGkM on synthetic and real-world datasets.

Related Work

The closest related work directly extends static techniques using online algorithms [4] or using machine learning [5]. Both of these papers make important contributions: dynamic graphs can be staggeringly large structures and an online algorithm can assist with parsing them, especially if data is collected in real time; graph neural networks are also an important and active area of research and intersecting them with recurrent networks is a natural extension of deep learning techniques. Other approaches, which use aggregation to associate clustering snapshots from static methods, are unable to achieve temporal smoothness and inevitably do

not capture the dynamics of the network [6]. Therefore, we seek a clustering method that is unified in both space and time.

Spatiotemporal Graph k-means

Given a dynamic graph, we want to perform community detection on its nodes. More precisely, given a set of vertices, we want to cluster or partition them in a way that leverages an underlying dynamic network structure. Mathematically, we say that we have a vertex set V of size n and a dynamic graph $\mathcal{G} = (V, E^t)_{t\in\mathbb{T}}$ where $E^t \in V \times V$. This graph is indexed with respect to some ordered time set $\mathbb{T}$, e.g., some subset of the natural numbers. We want to find a partition of the vertex set $\mathcal{V} = (V_j)_{j\in[k]}$ such that $V_j \subseteq V$, no subset in the partition is empty, and the subsets are disjoint. Note: we use the notation $[c]$ to mean the set $\{1, \ldots, c\} \subset \mathbb{N}$ throughout this paper.

In STGkM, we define partitions by finding *central* nodes. More precisely, we select k nodes that each represent a cluster and then assign the remaining vertices to the "closest" cluster under some notion of distance. Just as with k-means, we can define the problem of finding good clusters as a minimization problem; our novel objective has a unified formulation over space and time that predicts a partition for each vertex at every time step. By building upon the k-means algorithm, cluster centers are explicitly tracked and there are fewer parameters to tune, in contrast to other community detection models. We have a two-phase algorithm: in a single pass of Phase 1, vertex membership and dynamic cluster center journeys are output; in Phase 2, we extract the long-lived communities from the graph.

Phase 1

The first phase of STGkM assigns each vertex to a partition at every time step. Vertices have the flexibility to change clusters between time steps. In particular, we write $c_j^t \in V$ to be the node that is the center of cluster j at time t. Our objective is shown in Equation (1).

$$\min_{c,W} \sum_{u\in V} \sum_{j\in[k]} \sum_{t} \left[W_{u,j}^t \cdot \delta^t(u, c_j^t) + \frac{\lambda}{n} \cdot \delta^t\left(c_j^t, c_j^{t+1}\right) \right] \tag{1}$$

The objective in Equation (1) has two parts. The first term forces vertices to be assigned to closer cluster centers by penalizing the distance between a vertex and its assigned cluster. The second term penalizes cluster centers if they drift over time and so performs temporal smoothing and regularization. Note that the tensor W has entries in $[0, 1]$ and each matrix W^t must be stochastic; it is minimized over all such tensors. Also, c is minimized over all possible $\binom{n}{k}$ cluster assignments over all time steps. The distance function $\delta^t(\cdot, \cdot)$ denotes the shortest journey between two vertices in the dynamic graph at time t as described previously [1]. Finally, note that λ is a tuning parameter that controls the influence of temporal regularization.

Phase 2

Phase 2 of STGkM aims to identify the long-lived partitions of graph vertices. The output is an assignment of communities containing vertices with the most similar

spatiotemporal characteristics. Phase 2 builds upon Phase 1, taking into account short-term information in making decisions about long-term behavior. Intuitively, we expect vertices with similar partitioning histories to belong to the same community in the long run. We first extract the assignment histories from the weight tensor W. We define the assignment histories $a_u \in [k]^{|\mathbb{T}|}$, which is a vector where each entry contains the cluster assignment of vertex u at time t where $a_u^t = \text{argmax}_j \, W_{u,j}^t$.

We can then leverage the Hamming distance to define a similarity score between two vectors a_u, a_v. Recall that the Hamming distance is defined by counting the number of entries where two vectors disagree, i.e., $H(u,v) \triangleq |\{t : a_u^t \neq a_v^t\}|$. Using this distance, we say that the similarity between two vertices is $\text{sim}(u,v) \triangleq 1 - \frac{H(u,v)}{|\mathbb{T}|}$. We can then use agglomerative clustering to partition the vertices based on their associated assignment history vectors.

Theoretical Analysis

Especially for statistical analysis, it is important to understand the "control case" or null hypothesis; we provide an exposition of clustering behavior when there are no true underlying clusters. To do this, we use the definition of dynamic connectivity freely [7, 1] and define the Discrete Dynamic Erdős-Rényi graph: it is a dynamic graph over discrete time, where we sample a new Erdős-Rényi graph at each time step. More precisely, if we have N vertices, T total time steps (T may be infinite), and $p \in (0,1)$, we define

$$A_{uv}^t \overset{iid}{\sim} \text{Ber}(p)$$

where $u, v \in [N], t \in [T]$, and $\text{Ber}(p)$ is a standard Bernoulli distribution that is 1 with probability p and 0 otherwise. We can observe that A^t is the adjacency matrix at time t for our graph.

Definition 1 (Dynamic Connected Component) We say that distinct vertices u, v are *(dynamically) connected* if there exists a finite journey from u to v and from v to u for all time steps. By definition, we say a vertex is connected to itself, regardless of the existence of journeys.[1] We say a set of vertices U (where $U \subseteq V$) is a *(dynamic) connected component* if all vertices in this set are connected and there is no vertex in $V \setminus U$ that is connected to a vertex in U.

Proposition 1 *Dynamic connectivity is an equivalence relation. In particular, two vertices are in a connected component if and only if they are connected.*

Theorem 2 *A discrete dynamic Erdős-Rényi graph almost surely (as time goes to infinity) has only the trivial connected components, i.e., the connected components are exactly the vertices in their own respective singleton sets.*

[1] We do not need to impose this condition, but it makes our exposition much simpler for the purposes of this paper.

Experiments

In order to test performance, we propose to evaluate STGkM on a handful of datasets. We begin with small networks representing basketball and sheep herding data and culminate in large networks from the SNAP library [8]. All of our datasets contain ground-truth communities, allowing us to quantify the validity of our results. Our first experiment will explore basketball data [9]. We assign our players to the vertices of our graph and define an edge as existing between players if the ball can be passed between them. We aim to identify the underlying offense-defense structure within each team. Our next experiment looks at a synthetic sheep herding dataset, and the goal is to identify the sheepdog that is driving the dynamics of the sheep herd. Finally, we intend to explore increasingly larger networks from the SNAP database. We also hope to use additional datasets if practicable.

Conclusion and Future Work

We introduce spatiotemporal graph k-means (STGkM) as an approach for community detection through vertex clustering on dynamic graphs. Existing work treats dynamic graph clustering as an online problem, iteratively updating existing graphs or through machine learning, which has extensive parameters. Instead, we provide an approach that is unified over space and time and provides us the ability to analyze both the short-and long-term partitions of graph vertices, monitor the multi-scale relationships between communities, and has just two explainable parameters.

We also provide some theoretical guarantees that explain clustering behavior under a null hypothesis, which further reinforces any clustering results found with our method. Finally, we propose experiments on a handful of datasets that will empirically validate STGkM.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Program in Applied and Computational Mathematics, Princeton University, Fine Hall, Princeton, NJ, USA.
[2]Applied Mathematics Department, University of Washington, Lewis Hall, Seattle, WA, USA.

References

1. Hylton, A., Short, R., Cleveland, J., Freides, O., Memon, Z., Cardona, R., Green, R., Curry, J., Gopalakrishnan, S., Dabke, D.V., Story, B., Moy, M., Mallery, B.: A survey of mathematical structures for lunar networks. In: 2022 IEEE Aerospace Conference (AERO), pp. 1–17 (2022). doi:10.1109/AERO53065.2022.9843305
2. Cleveland, J., Hylton, A., Short, R., Mallery, B., Green, R., Curry, J., Dabke, D.V., Freides, O.: Introducing tropical geometric approaches to delay tolerant networking optimization. In: 2022 IEEE Aerospace Conference (AERO), pp. 1–11 (2022). doi:10.1109/AERO53065.2022.9843242
3. Dabke, D.V., Karntikoon, K., Aluru, C., Singh, M., Chazelle, B.: Network-augmented compartmental models to track asymptomatic disease spread. Bioinformatics Advances (2023). pre-print.
4. Ruan, B., Gan, J., Wu, H., Wirth, A.: Dynamic structural clustering on graphs. In: Proceedings of the 2021 International Conference on Management of Data, pp. 1491–1503 (2021)
5. Yao, Y., Joe-Wong, C.: Interpretable clustering on dynamic graphs with recurrent graph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 4608–4616 (2021)
6. Görke, R., Maillard, P., Schumm, A., Staudt, C., Wagner, D.: Dynamic graph clustering combining modularity and smoothness. Journal of Experimental Algorithmics (JEA) **18**, 1–1 (2013)
7. Dabke, D.V.: On systems of dynamic graphs: Theory and applications. PhD thesis, Princeton University (May 2023)
8. Leskovec, J., Krevl, A.: SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data (2014)
9. Dabke, D.V., Chazelle, B.: Extracting semantic information from dynamic graphs of geometric data. In: Benito, R.M., Cherifi, C., Cherifi, H., Moro, E., Rocha, L.M., Sales-Pardo, M. (eds.) Complex Networks & Their Applications X - Volume 2, Proceedings of the Tenth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2021, Madrid, Spain, November 30 - December 2, 2021. Studies in Computational Intelligence, vol. 1016, pp. 474–485. Springer, ??? (2021). doi:10.1007/978-3-030-93413-2_40. https://doi.org/10.1007/978-3-030-93413-2_40