Literature Presentation

Community Detection through Representation learning in Evolving Heterogenous Networks

Egor Dmitriev

Utrecht University

18 November 2021

Surveys

- [Rossetti and Cazabet, 2018] and [Dakiche et al., 2019]
- Categorize methods on problem of tracking community evolution
- Introduce a common definition for evolving communities
- Compile events/properies for analysis of evolving communities

Evolving Communities

- Communities in real world:
  - disjoint (students belonging to different disciplines in an institute)
  - overlapping (person having membership in different social groups on Facebook)
  - hierarchical (cells in human body form tissues that in turn form organs and so on)
- Depend on underlying networks:
  - Time-series of static networks (Snapshots)
  - Real time a stream of edges (Temporal networks)
- Evaluated on synthetic (generated) communities
  - Usually based on a quality score:
    - Normalized Mutual Information score (NMI)
    - Modularity
    - etc.

- Quality function to evaluate algorithms **favors the ones that are designed to optimize it**

Changes in Evolving Communities

- Operations that define community changes

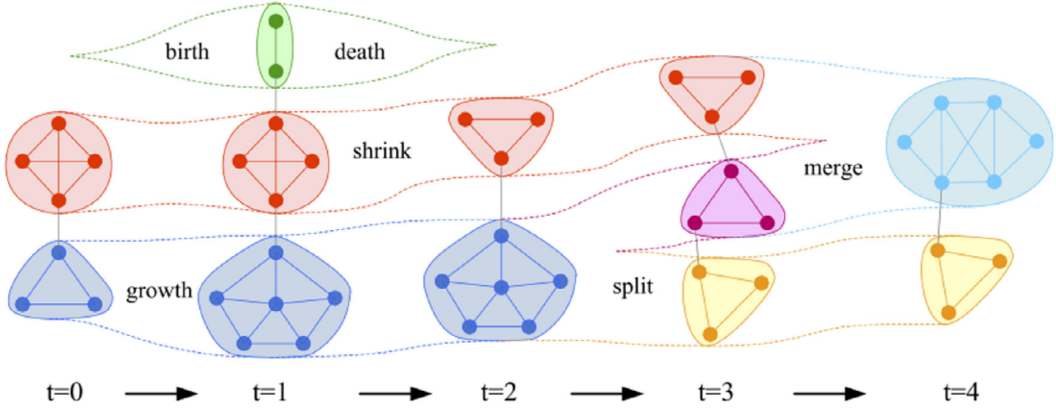

Fig. 3. Community evolution in a dynamic network (Shang, Liu, Li, Xie, & Wu, 2016).

Challenges & Uses

- *deciding if an* ***element composed of several entities*** *at a given instant is the same or not as another one composed of some—or even none—of such entities at a later point in time is* ***necessarily arbitrary and cannot be answered unambiguously***
- Main issues encountered by dynamic community detection approaches is the **instability of solutions**
- Use cases:
  - forecasting emerging market trends in online retail networks
  - characterizing functions of unknown proteins
  - real-time partitioning of web-pages with different topics
  - predicting the emergence and lifetimes of overlapping communities in online social networks

- *deciding if an* ***element composed of several entities*** *at a given instant is the same or not as another one composed of some—or even none—of such entities at a later point in time is* ***necessarily arbitrary and cannot be answered unambiguously***
- Main issues encountered by dynamic community detection approaches is the **instability of solutions**
- Use cases:
  - forecasting emerging market trends in online retail networks
  - characterizing functions of unknown proteins
  - real-time partitioning of web-pages with different topics
  - predicting the emergence and lifetimes of overlapping communities in online social networks

Louvain method (LM)

- Popular clustering algorithm
- Complexity is $n \log n$
- Can be applied to weighted graphs
- Does not require a priori knowledge of the number of partitions

Independent Community Detection and Matching

- **First detect communities** at each time step and **then match** them across different time-steps
- Unmodified traditional community detection methods can be reused
- Parallelism can be used for community detection
- Community detection algorithms are unstable leading to poor matching
- Examples:
  - **Sun, Tang, Pan, and Li (2015)**:
    - Applied the Louvain algorithm to find the communities.
    - Then **built a correlation matrix** to between communities in t and $t+1$
  - **Greene et al. (2010)**:
    - Using the static algorithm MOSES to detect the communities on each snapshot.
    - Then, they described a **weighted bipartite matching to map communities**
  - [Rossetti, 2020]
    - Allows for overlapping communities using modified node labeling algorithm
    -
    - Matching based om multiple labels in t, t-1, t+1

- Independent Community Detection and Matching:
  - Unmodified traditional community detection methods can be reused
  - Parallelism can be used for community detection.
  - Major drawback: Community detection algorithms are unstable.
- **Sun, Tang, Pan, and Li (2015)**:
  - Applied the Louvain algorithm to find the communities.
  - Then **built a correlation matrix** to between communities in t and $t+1$
- **Greene et al. (2010)**:
  - Using the static algorithm MOSES to detect the communities on each snapshot.
  - Then, they described a **weighted bipartite matching to map communities**
- [Rossetti, 2020]
  - Allows for overlapping communities using modified node labeling algorithm
  - Matching based om multiple labels in t, t-1, t+1
  - focuses on lowering the time complexity while at the same time increasing the partition
  - Events are detected and evaluated against ground truth
  - Provides a deterministic output

Dependent Community Detection

- **Detect communities at time t** and then **use them to detect communities at time $t+1$**,
- Reduce computational cost but do not allow parallelism
- Examples:
  - **Gao, Luo, and Bu (2016)**:
    - Evolutionary community discovery algorithm based on **leader nodes**
    - Each **community is considered as a set of follower nodes** congregating close to a potential leader

- Dependent Community Detection
  - Reduce computational cost by reusing much of the previous community
  - Traditional community detection methods are no longer directly applicable
  - Does not allow parallelism in community detection
- **Gao, Luo, and Bu (2016)**:
  - Evolutionary community discovery algorithm based on **leader nodes**
  - Each **community is considered as a set of follower nodes** congregating close to a potential leader

Simultaneous Community Detection on All Snapshots

- **Construct a single graph** and then run a classic community detection
- Solution for the lack of stability of the independent community detection

- Simultaneous Community Detection on All Snapshots
  - **Main advantage**: is providing a solution for the lack of stability of the independent community detection
  - Difficulty to detect complex operations such as merging and splitting

Dynamic Community Detection on Temporal Networks (online approach)

- Update the ones previously found according to network modifications
- Problem: Modifications are done at a **local level**
- Examples:
  - **Shang et al. (2012)**:
    - Update graph real-time, and **locally modify the concerned communities** in a way to increase the modularity
  - **Held and Kruse (2016)**:
    - Assumption that there exist some **highly connected nodes**, called hubs, which will group people around them.
  - [Xu et al., 2020]
    - A dynamic network snapshot is totally re-partitioned once the error accumulation degree of incremental clustering exceeds a pre-defined threshold
    - Use

- Dynamic Community Detection on Temporal Networks (online approach)
  - Since the communities evolve naturally through modifications, there is, **no longer, an instability problem**
  - Advantage: low complexity of tracking communities, since changes can be incremental
  - Problem: Modifications are done at a **local level**, they can **involve drifting towards invalid communities**
- **Shang et al. (2012)**:
  - Method consists in adding (or removing) each new edge as it appears (or disappears), and to **locally modify the concerned communities** in a way to increase the modularity
- **Held and Kruse (2016)**:
  - Based on the assumption that there exist some highly connected nodes, called hubs, which will group people around them.
  - It is based on the assumption that there exist some highly connected nodes, called hubs, which will group people around them. So, in the first step, the proposed algorithm detects these hubs by the node degree and assigns to all non-hub elements the closest hub as a cluster label, then iteratively changes the resultant clustering by applying changes: adding or removing nodes or edges
- [Xu et al., 2020]
  - A dynamic network snapshot is totally re-partitioned once the error accumulation degree of incremental clustering exceeds a pre-defined threshold
  -

User community detection via embedding of social network structure and temporal content

- [Fani et al., 2020]
- Content on the social network are often reflective of issues in the real world **topics discussed on the network constantly change** and hence users' interests towards these topics
- Combine both Temporal Social Content $\mathcal{D} = (\mathbb{U}, \mathbb{M}, \mathrm{T})$ and Social Network Connections $\mathcal{G} = (\mathbb{U}, \mathbb{A})$
  - Model communities **based on topics of interest**
  - Primarily based on the **homophily** principle

- Temporal Social Content: $\mathcal{D} = (\mathbb{U}, \mathbb{M}, \mathrm{T})$
  - $\mathbb{U}$: Users, $\mathbb{M}$: Text content, T time periods
- Social Network Graph: $\mathcal{G} = (\mathbb{U}, \mathbb{A})$
  - $\mathbb{U}$: Users/Nodes, $\mathbb{A}$: Edges
- Homophily: (densely connected groups of users imply a user community)

User community detection via embedding of social network structure and temporal content

- Identify Topics using LDA and Construct Preference Time series
- Learn dense representation of users interests using CBOW model
  - Use “Regions of like-mindedness” for as scoring function
- Topological Embeddings are contructed using a Skip-Gram model
  - Use DFS based random walk over the Social Network

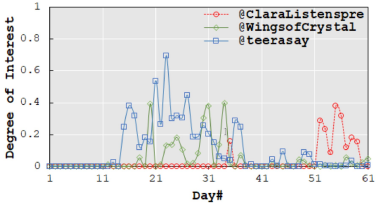

Figure 2: Different temporal behaviour of three Twitter users with respect to the War in Afghanistan topic

- **Region of like-mindedness**: Parts in X where users share interest in same topics given a threshold (for level of interest)
- BFS favours structural equivalence
- DFS in contrast, respects homophily and leads to similar (close) embeddings for densely connected users

User community detection via embedding of social network structure and temporal content

- Finally embeddings are combined into one $h\left(\mathbf{W}_{\mathcal{D}}, \mathbf{W}_{\mathcal{G}}\right) = \alpha \mathbf{W}_{\mathcal{D}} + (1-\alpha)\mathbf{W}_{\mathcal{G}}$
- Community detection:
  - Construct a weighted graph: $\mathrm{G} = (\mathbb{U}, \mathbb{E}, w)$
  - Leverage the Louvain Method (LM)
  -
- Remarks:
  - Users end up in one community per users

- With as weights the user embedding **dot products**

Review Order

- Classical ML Methods:
  - liuMultipleLocalCommunity2021(unread)
- Deep Learning Based Methods
  - faniUserCommunityDetection2020
  - wangVehicleTrajectoryClustering2020
- Related Tasks

This is my note.

- It can contain Markdown
- like this list

References I

N. Dakiche, F. Benbouzid-Si Tayeb, Y. Slimani, and K. Benatchba. Tracking community evolution in social networks: A survey. *Information Processing & Management*, 56(3): 1084–1102, May 2019. ISSN 0306-4573. doi: 10.1016/j.ipm.2018.03.005.

H. Fani, E. Jiang, E. Bagheri, F. Al-Obeidat, W. Du, and M. Kargar. User community detection via embedding of social network structure and temporal content. *Information Processing & Management*, 57(2):102056, Mar. 2020. ISSN 03064573. doi: 10.1016/j.ipm.2019.102056.

G. Rossetti. ANGEL: Efficient, and effective, node-centric community discovery in static and dynamic networks. *Applied Network Science*, 5(1):26, June 2020. ISSN 2364-8228. doi: 10.1007/s41109-020-00270-6.

References II

G. Rossetti and R. Cazabet. Community Discovery in Dynamic Networks: A Survey. *ACM Computing Surveys*, 51(2):35:1–35:37, Feb. 2018. ISSN 0360-0300. doi: 10.1145/3172867.

Z. Xu, X. Rui, J. He, Z. Wang, and T. Hadzibeganovic. Superspreaders and superblockers based community evolution tracking in dynamic social networks. *Knowledge-Based Systems*, 192:105377, Mar. 2020. ISSN 0950-7051. doi: 10.1016/j.knosys.2019.105377.