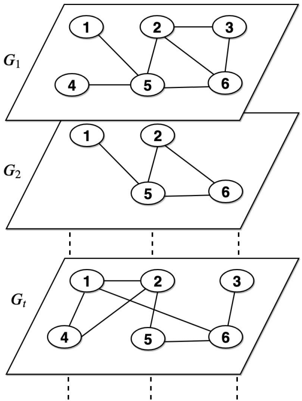
Dynamic Consensus Community Detection and Combinatorial Multi-Armed Bandit

Domenico Mandaglio and Andrea Tagarelli
d.mandaglio@dimes.unical.it, *andrea.tagarelli@unical.it*

DIMES, University of Calabria, Italy

29th August 2019

Community Detection (CD) in Temporal Networks



temporal sequence
$\mathcal{G}_{\leq t}$

- need for modeling the change events in the communities and tracking their evolution.
- **Challenges**: choice of timestep width, different type and occurring rates of community evolution events.
- incremental nature of the problem

Applications

- enhanced group-recommendation
- user behavior prediction
- evolution of user interaction patterns in relation to real-world events

Existing works on CD in Temporal Networks

Existing approaches:

- try to discover a sequence of mappings for the community structures independently derived at each time step
- detect a community structure for the current topology as dependent on the structure(s) from prior time step(s)
- update a community structure in order to reflect newly observed changes
- aggregate the various snapshots of the network in order to enable a static community detection method

Limitations of existing approaches

- require to match and/or track the evolution of communities over time
- depend on specific community-change events (merge, split, etc.)
- depend on restricted graph models
- assume the same nodes and number of communities for each snapshot

CD in Temporal Networks as Consensus problem

Requirement: balancing over time between the need for embedding long-term changes observed in the community formation and the need for capturing short-term effects and newly observed community structures.

- give more importance to the more recent community structures in the consensus generation

Dynamic consensus community structure

A community structure that encompasses the knowledge about newly observed as well as the previously detected communities in a temporal network

Dynamic consensus community detection problem (DCCD)

Input: Given the temporal graph sequence $\mathcal{G}_{\leq t}$ (undirected and unweighted graphs) and associated set of detected community structures (non-overlapping communities)
Output: for any time $1 \leq t \leq T$, compute a dynamic consensus community structure $\mathcal{C}_{\leq t}$ such as to maximize:

$$R(T) = \sum_{t=1}^{T} Q_t(\mathcal{C}_{\leq t})$$

where Q_t is a chosen quality criterion for a community structure, over the history (before t) of the network (e.g. multilayer modularity [1]).

[1] A. Tagarelli et al. "*Ensemble-based community detection in multilayer networks*". Data Min. Knowl. Discov. (2017)

Dynamic Consensus representation model

Dynamic co-association matrix (DCM) $\mathbf{M}$: the (i, j)-th entry of $\mathbf{M}$, denoted as m_{ij}, stores the probability of co-association for entities $v_i, v_j \in V$, i.e., the probability that v_i and v_j are assigned to the same community, in the observed timespan

Computing meaningful co-associations and properly maintaining and updating the consensus community structure over time is challenging:

- avoid (re)computation of the consensus from scratch
- avoid to depend on any mechanism of tracking of the evolution of communities
- density of $\mathbf{M}$

Reinforcement Learning approach to DCCD problem

Reinforcement Learning:

- interrelated actions with unknown "rewards" ahead of time
- choose which actions to take in order to maximize the reward
- exploitation/exploration trade-off
  - exploitation: make the best decision given current information
  - exploration: gather more information

CD in temporal networks

- uncertainty about the temporal network system, and the structural changes and consequent decisions to take about the node memberships and structure of the communities
- balancing between relying on older community structures and newly observed ones

Multi-Armed Bandit (MAB)

- $\mathcal{A}$ is a set of m slot-machines/arms to choose from
- each arm is associated with a set of random variables $\{X_i^t \mid 1 \leq i \leq m, t \geq 1\}$, $X_i^t \in [0, 1]$
- At each step t the agent selects/plays an arm $a_t \in \mathcal{A}$ and the reward $X_{a_t}^t$ is revealed
- The goal is to maximise the cumulative reward $R(T) = \sum_{t=1}^{T} X_{a_t}^t$
- The goal is pursued through an exploration/exploitation trade-off

Combinatorial Multi-Armed Bandit (CMAB)[2]

- At each step t the agent selects a **subset of base arms** (super arm) $A_t \subseteq \mathcal{A}$ and the rewards $X_{a_t}^t$ for all $a_t \in A_t$ are revealed
- the base arms belonging to A_t may probabilistically trigger other base arms not in A_t
- The reward of playing A_t, $R(A_t)$, is a linear/non-linear combination of the rewards of the selected and triggered base arms
- The goal is to maximise the cumulative reward $R(T) = \sum_{t=1}^{T} R(A_t)$

[2] Wei Chen et al. *"Combinatorial multi-armed bandit and its extension to probabilistically triggered arms."* The Journal of Machine Learning Research 17.1 (2016)

Translating the DCCD problem into CMAB (I)

- each pair $\langle v_i, v_j \rangle$ is a base arm and its semantics is assigning the nodes to the same community at a given time
- each pair $\langle v_i, v_j \rangle$ is associated with an unknown distribution (with unknown mean μ_{ij}) for the probabilities of co-association over time (they may change), whose mean estimate is the entry m_{ij} in DCM
- A superarm A_t is a set of base arms which corresponds to a dynamic consensus community structure i.e., a set of pairs $\langle v_i, v_j \rangle$ such that $c_i^{(t)} = c_j^{(t)}$

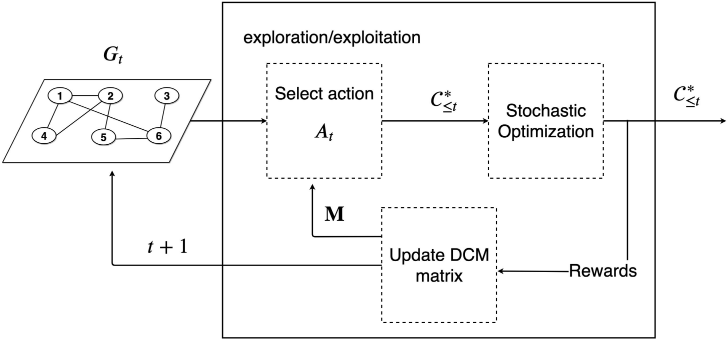
Translating the DCCD problem into CMAB (II)

- Playing a superarm A_t corresponds to:
  1. inducing a community structure from the played superarm
  2. performing stochastic relocation of nodes to neighbor communities (trigger base arms)
- the rewards associated to the entity pairs (base arms) are revealed after the relocation phase thus **M** can be updated
- $R(A_t)$ corresponds to the quality of the solution after the relocation phase, e.g. modularity is a non-linear combination of rewards X_{ij}^t

$$R(A_t) = \frac{1}{d(\mathcal{V}_{[1..t]})} \sum_{i,j} \sum_{\ell=1}^{t} \beta^{t-\ell} \Big(A_{ij}^{l} - \frac{k_i^{\ell} k_j^{\ell}}{d(\mathcal{V}_{[1..t]})} \Big) \delta(X_{ij}^t)$$

$$X_{ij}^t = 0 \text{ if } c_i^{(t)} \neq c_j^{(t)},\ 1/|c_i^{(t)}| \text{ otherwise}$$
$$\delta(X_{ij}^t) = 1 \text{ if } X_{ij}^t > 0,\ 0 \text{ otherwise}$$

Overview of the framework



- updates: $newEstimate \leftarrow oldEstimate + \alpha(newValue - oldEstimate)$
- exploitation/exploration: partition the DCM-graph/current snapshot
- versatile in terms of bandit strategy and the static community detection algorithm

Algorithmic scheme

Algorithm 1 General scheme of CMAB algorithm for Dynamic Consensus Community Detection

Input: Temporal graph sequence $\mathcal{G}_{\leq T}$ ($T \geq 1$), bandit strategy $\mathcal{B}$, (static) community detection method $\mathcal{A}$.

Output: Dynamic consensus community structure $\mathcal{C}^*_{\leq T}$.

Initialize the dynamic consensus matrix $\mathbf{M}$
for $t = 1$ **to** T **do**
  if $\mathcal{B}$ decides for EXPLORATION **then**
    Find a community structure $\mathcal{C}^{(t)}$ on G_t using $\mathcal{A}$
  else {EXPLOITATION}
    Partition the DCM-graph using $\mathcal{A}$
    Infer a community structure $\mathcal{C}^{(t)}$ on G_t based on the DCM-graph partitioning
  end if
  Project the community memberships from $\mathcal{C}^{(t)}$ onto $\mathcal{G}_{\leq t}$
  Stochastic optimization of $\mathcal{C}^*_{\leq t}$
  Update the DCM matrix $\mathbf{M}$ based on $\mathcal{C}^*_{\leq t}$
end for
return $\mathcal{C}^*_{\leq T}$

Evaluation

Data:

- 5 real-world datasets and 1 synthetic network[3]

Evaluation goals:

- Impact of learning rate α
- Efficiency evaluation
- Comparison with competing methods:
  - DynLouvain [4]
  - M-EMCD*[5]
  - EvoAutoLeaders [6]

[3] G. Rossetti. *RDyn: graph benchmark handling community dynamics.* Journal of Complex Networks, 2017.

[4] J. He and D. Chen. *A fast algorithm for community detection in temporal network.* Physica A: Stat. Mech. Appl., 429:87–94, 2015

[5] D. Mandaglio, A. Amelio, and A. Tagarelli. *Consensus Community Detection in Multilayer Networks Using Parameter-Free Graph Pruning.* In Proc. PAKDD, pages 193–205, 2018.

[6] W. Gao, W. Luo, and C. Bu. *Adapting the TopLeaders algorithm for dynamic social networks.* The Journal of Supercomputing, 2017.

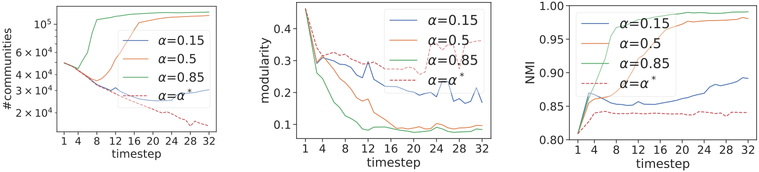
Datasets

Table: Main characteristics of our evaluation data. Mean $\pm$ standard deviation values refer to all snapshots in a network.

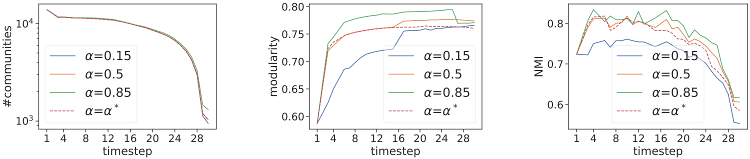
	#entities ($\lvert\mathcal{V}\rvert$)	#edges	#time steps	node set coverage	edge semantics	% static (nodes, edges)	% hapax (nodes, edges)	% dynamic (nodes, edges)
Epinions	131 828	727 344	32	0.05	trust/distrust	(0.1, 0)	(80.8, 95.6)	(19, 2.2)
Facebook	63 731	17 676 817	30	0.87	friendship birth	(82.9, 2.7)	(0.2, 0)	(16.9, 1.9)
Wiki-Conflict	118 100	2 272 276	82	0.05	wikipage editing	(0, 0)	(60.1, 83.4)	(38.9, 5.8)
Wiki-Election	7 118	102 906	44	0.08	vote assignment	(0, 0)	(49.7, 95.7)	(50.3, 2.2)
YouTube	3 223 589	41 955 741	8	0.62	friendship birth	(33.4, 6.7)	(12.4, 4)	(54.2, 11.6)

	network evolution rate			
	$e_t^+ = \frac{\lvert E_t \setminus E_{t-1}\rvert}{\lvert E_t\rvert}$	$e_t^- = \frac{\lvert E_{t-1} \setminus E_t\rvert}{\lvert E_{t-1}\rvert}$	$v_t^+ = \frac{\lvert V_t \setminus V_{t-1}\rvert}{\lvert V_t\rvert}$	$v_t^- = \frac{\lvert V_{t-1} \setminus V_t\rvert}{\lvert V_{t-1}\rvert}$
Epinions	0.97 ± 0.007	0.98 ± 0.008	0.65 ± 0.08	0.69 ± 0.06
Facebook	0.02 ± 0.01	0	0.006 ± 0.006	0
Wiki-Conflict	0.95 ± 0.02	0.95 ± 0.02	0.52 ± 0.1	0.51 ± 0.12
Wiki-Election	0.99 ± 0.004	0.99 ± 0.005	0.5 ± 0.07	0.49 ± 0.08
YouTube	0.16 ± 0.06	0	0.14 ± 0.06	0

Impact of learning rate (I)



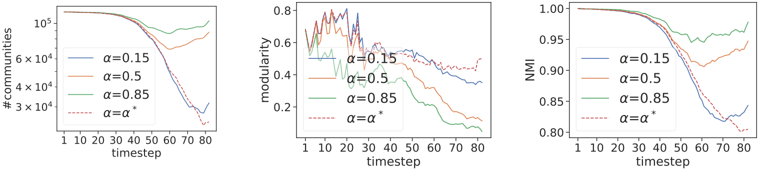
(a) Epinions



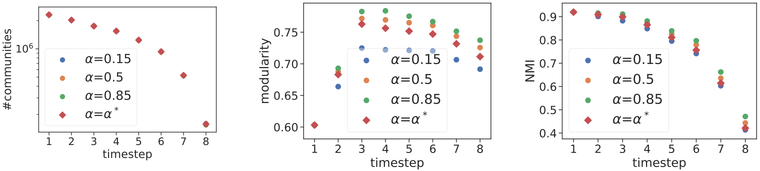
(b) Facebook

Figure: Size of the dynamic consensus by CreDENCE (left), multilayer modularity of the CreDENCE solutions (mid), and NMI between the CreDENCE consensus community structure and the snapshot's community structure, at each t (right).

Impact of learning rate (II)



(c) Wiki-Conflict



(d) YouTube

Figure: Size of the dynamic consensus by CreDENCE (left), multilayer modularity of the CreDENCE solutions (mid), and NMI between the CreDENCE consensus community structure and the snapshot's community structure, at each t (right).

CreDENCE vs competing methods

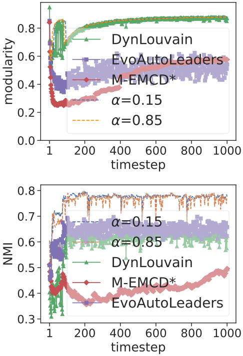


Fig. 4: Competitors vs. CreDENCE on RDyn: modularity (top), NMI (bottom).

Table: Increment percentages of CreDENCE w.r.t. DynLouvain[7] and M-EMCD*[8]. Values correspond to the increment percentages averaged over all snapshots in a network, using the average best-performing α.

	DynLouvain		M-EMCD*	
	Modularity	NMI	Modularity	NMI
Epinions	1789.0 %	-2.2 %	13.9 %	37.6%
Facebook	3.5 %	9.4 %	60.0 %	37.5 %
Wiki-Conflict	$>$ 1.0 E+05 %	-1.8 %	-6.8 %	37.6 %
Wiki-Election	660.5 %	-2.1 %	32.0 %	58.5 %
YouTube	-0.1 %	8.4 %	21.1 %	11.6 %
RDyn	2.0 %	24.97 %	103.22 %	81.1 %

Summary of findings

- ability of dealing with temporal networks with different evolution rate
- scales linearly with the number of timesteps
- outperforms competing methods in terms of Normalized Mutual Informationand Multilayer Modularity.

Conclusion & Future Works

Summary:

- CMAB paradigm for CD in temporal networks
- novel problem of dynamic consensus community detection
- general algorithmic scheme to solve DCCD problem

Future Works:

- the impact of different bandit strategies (e.g., UCB, Thompson sampling)
- learning the model parameters to best fit the community structure and evolution in a given temporal network

Thank you for your attention.