Mining and analysing temporal networks with graph evolution rules

Alessia Galdeman

Connets LAB
Computer Science
UNIVERSITÀ DEGLI STUDI
DI MILANO

Outline

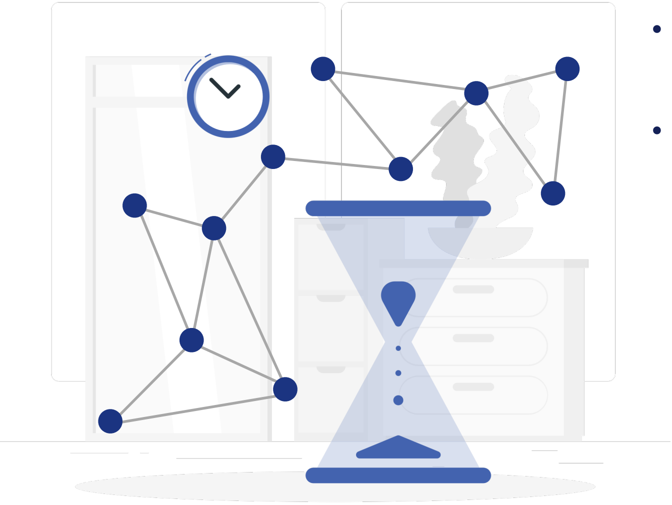
1. **Temporal and evolving networks**
   Definitions, Formalisms, and Open Questions
2. **Background**
   frequent subgraph based methods and reasons to choose GERs
3. **Graph evolution rules**
   definitions, formalisms, and visualization
4. **Algorithms and extensions**
   Existing algorithms and a null model extension
5. **Real-world case studies**
   Examples of application to social, communication and Web3 networks

Temporal Networks

DEFINITIONS, FORMALISMS AND OPEN QUESTIONS

Temporal networks

CHALLENGES AND DATA SOURCES



- An interesting but yet not fully explored field, mainly due to the lack of temporal data
- Thanks to the web3 development, we have enough data to develop solid temporal methodologies

WEB3 data

Blockchain-based online social netoworks

Social networks based on a reward-system for content creator and curators
Examples: Steemit, Hive, and Galxe

Non-fungible tokens

Networks of NFT trades on different markets Examples: CRyptokitties, OpenSea, and Decentraland

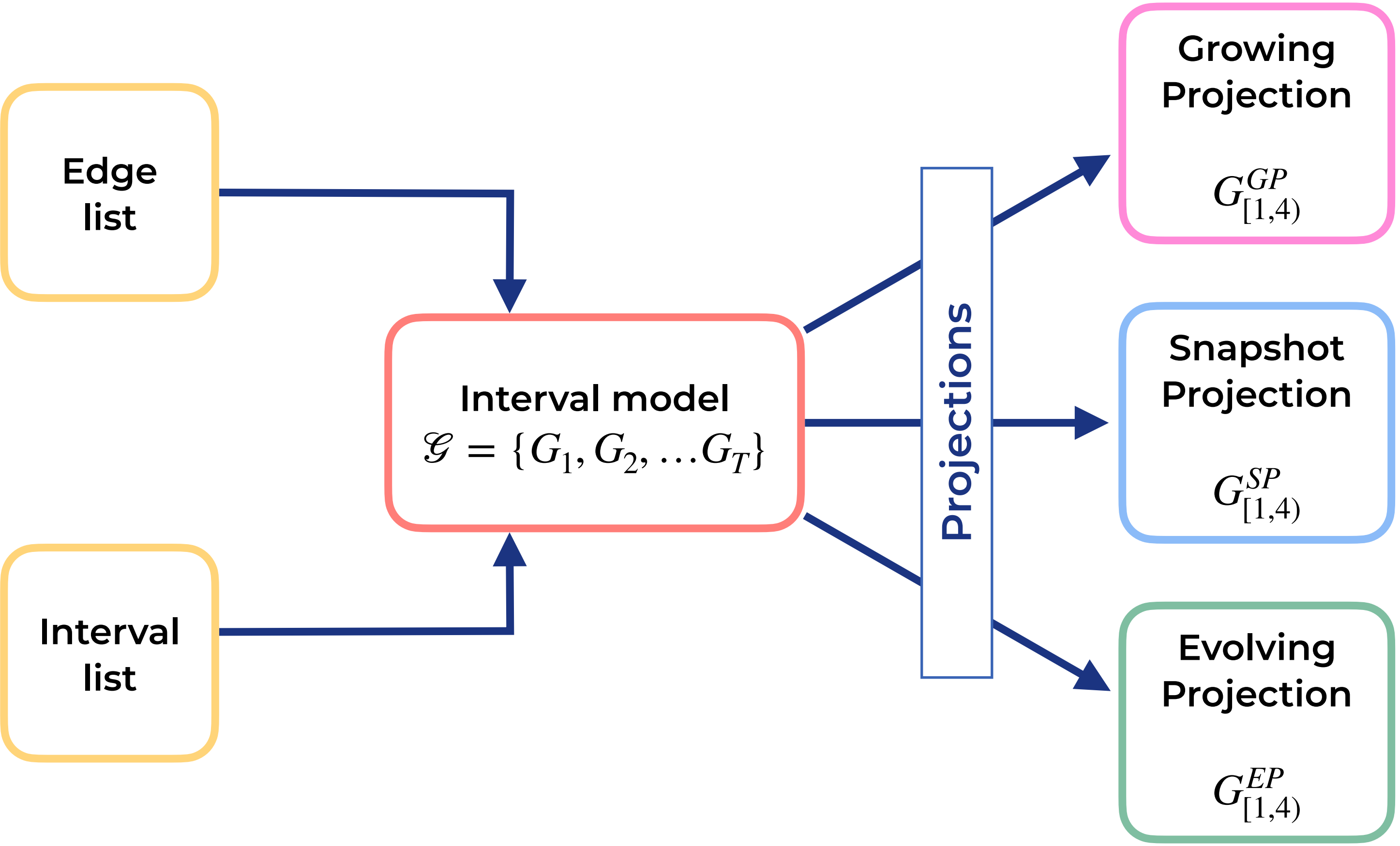
Complementary currency

Exchange of a complementary currency through the blockchain technology. Examples: Sarafu, and Circle

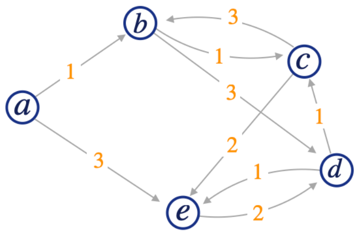
Stable coins

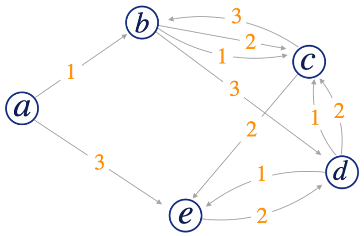
Temporal networks modeling

A COMPREHENSIVE TAXONOMY



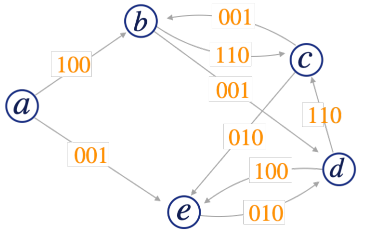
The growing projection considers maximum one edge for each couple of nodes, with the label and timestamp corresponding to the most recent (max ts) or the least recent (min ts) one, within the selected interval (for instance $[1,4)$ like in the figure).





The snapshot projection considers all inserted edge for each couple of nodes, with the relative timestamps, within the selected interval.

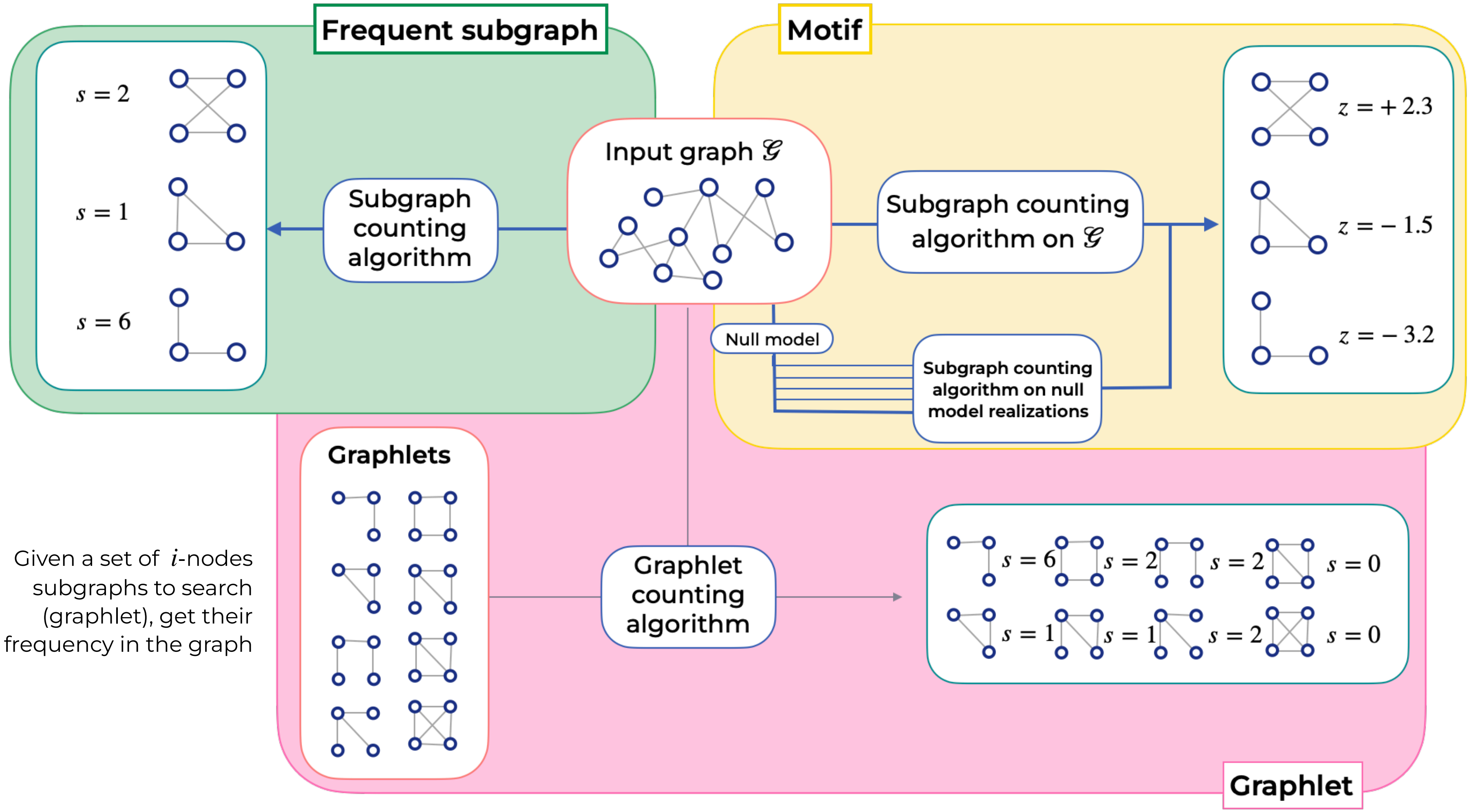
The evolving projection embeds in a single edge between couple of nodes all potential multiple edges happened in the selected interval, with a label that encodes the evolution.



Background

Frequent subgraph based methods

Search all frequent subgraphs

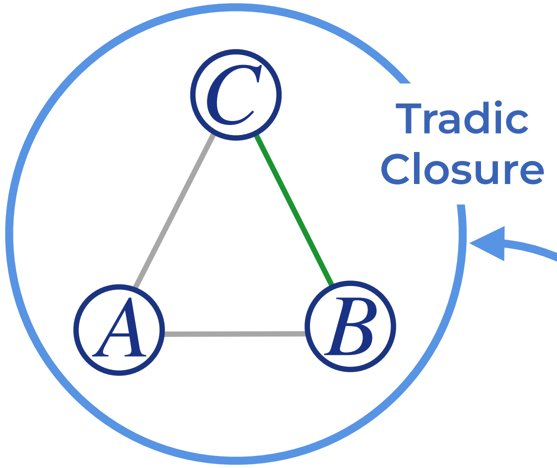


Given a set of i-nodes subgraphs to search (graphlet), get their frequency in the graph

- Search for frequent subgraphs in the real graph;
- Shuffle the real graph into n null realizations
- Search for frequent subgraph in each realization
- Use for instance the z-score to get the motifs (subgraphs that are more frequent in the real graph wrt to the null model realizations)

Graph evolution rules

REASONS WHY



Several models, mechanisms and measures have been proposed to describe the network growth

BUT

- They assume that the growth is guided by a single parameterized mechanism
- Identifying which mechanism plays a more important role is challenging

Graph evolution rules mining can detect evolutionary behaviors, while avoiding any a-priori mechanism

Graph Evolution rules

DEFINITIONS, FORMALISMS AND VISUALIZATION

Rules

COMPOSITION AND MEANING

Precondition

Body

Antecedent

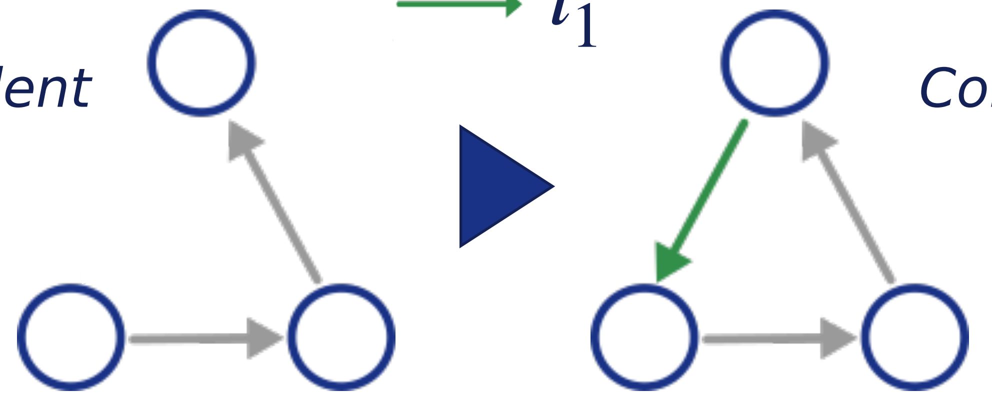
$\longrightarrow t_0$

$\longrightarrow t_1$

Postcondition

Head

Consequent

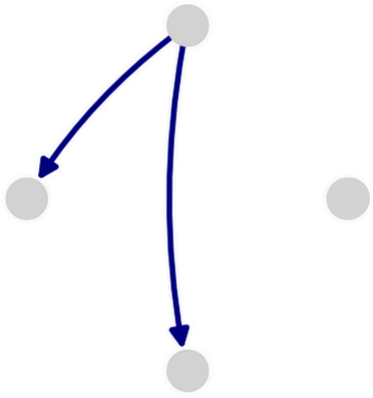


A rule matching (being isomorphic) to the precondition → will probably (frequently) evolve into one matching the postcondition

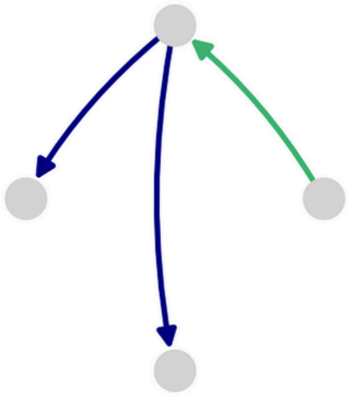
GER visualizations

TWO ALTERNATIVES

Precondition



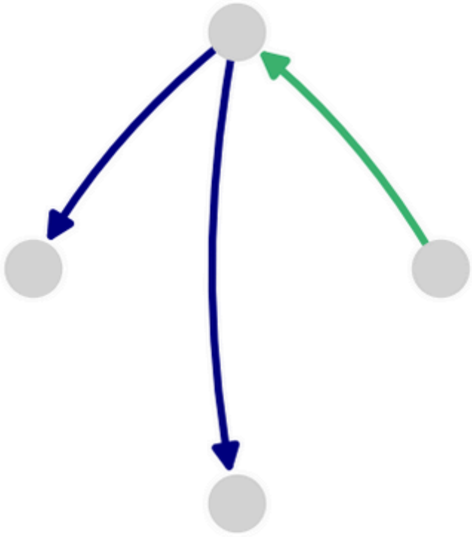
Postcondition



Thanks to the antimonotonicity property, there's one pre-condition for each post-condition

So we can just visualize the post condition (with colored edges)

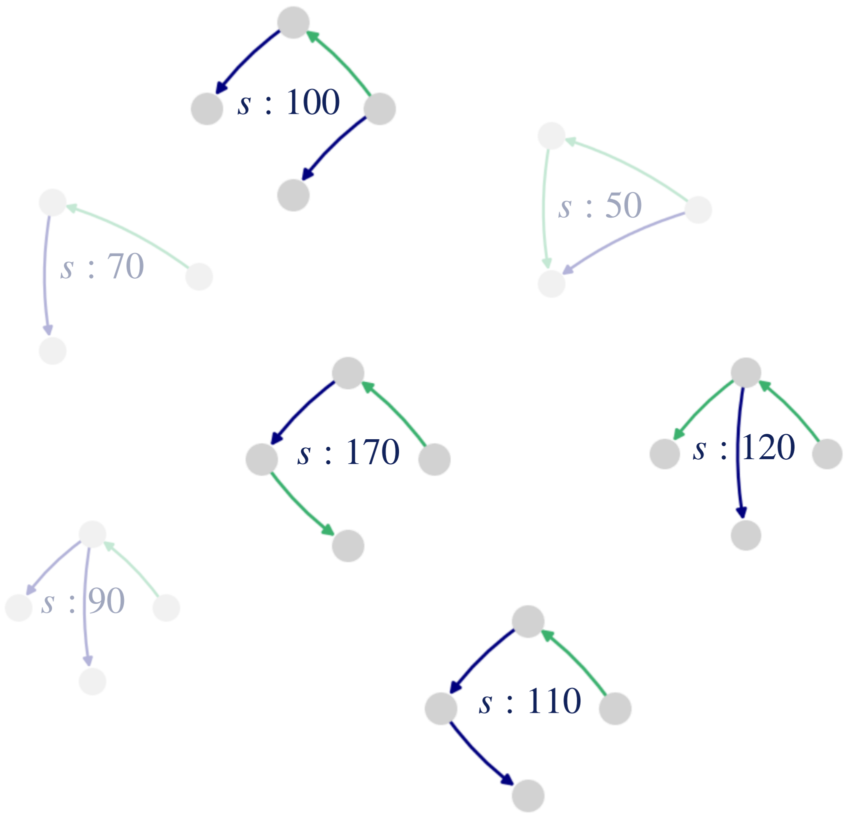
Compact version



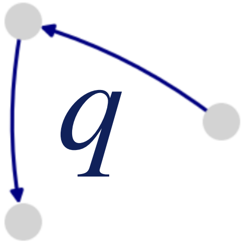
Support

AKA FREQUENCY OF A RULE

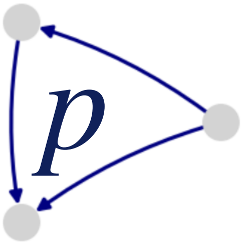
Support threshold $\sigma \geq 100$



- The **support** is a fundamental parameter in ger mining algorithms because it **filters** the patterns to determine which are **frequent**, and so can be considered as rules
- In the data mining field, it correspond to the **frequency** of the pattern
- In graphs, it can't be simply the number of occurrences of the pattern because it should satisfy the **anti-monotonicity property**



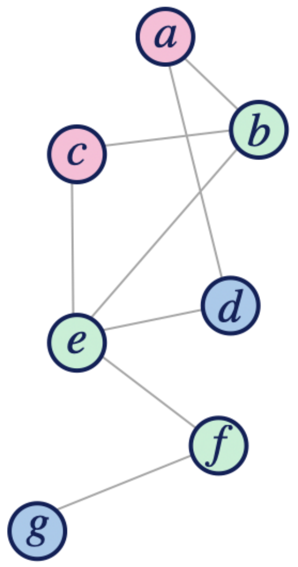
$$q \subset p$$
$$\sigma(q) \geq \sigma(p)$$



Intuitively, everytime we see a pattern matching p, there is also q because q is a subset of p, so q's support should be higher

Support

MINIMUM IMAGE BASED



(a) Input graph G

(b) Subgraph p

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	$\lvert\Phi(v_i)\rvert$
v_1	a	c	c	c	2
v_2	b	b	e	e	2
v_3	e	e	b	g	3

(c) Four isomorphisms (columns) and unique mappings (rows)

MIB SUPPORT:

$$\sigma(p, G) = 2$$

minimum of the number of unique mappings for the nodes in the pattern

GSpan

MINIMUM DFS CODE

A graph (or subgraph) can be described through a list of 5-tuple, called DFS code:

$$(i, j, l(i), l(i,j), l(j))$$

source node

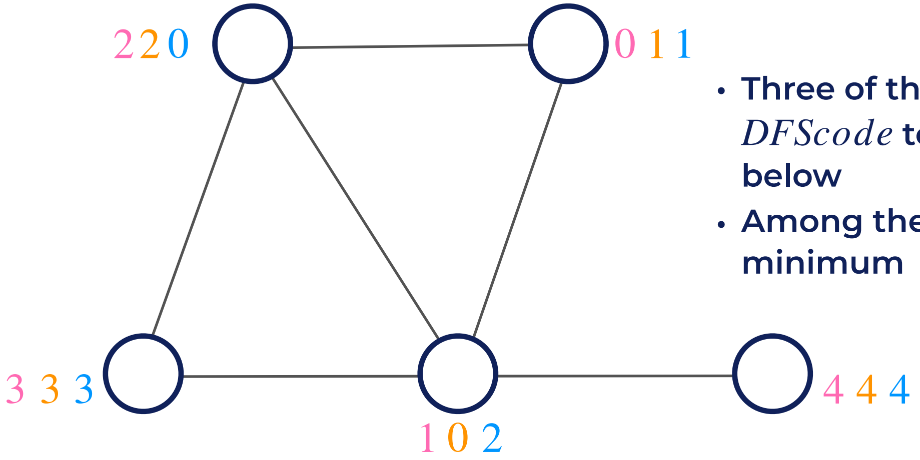
destination node

label of node i

label of edge between i and j

label of node j

The multiple DFS code for a graph can be lexicographically ordered to obtain the minimum DFS code



- Three of the many possible $DFScode$ to describe G are listed below
- Among the listed, $DFScode_2$ is the minimum

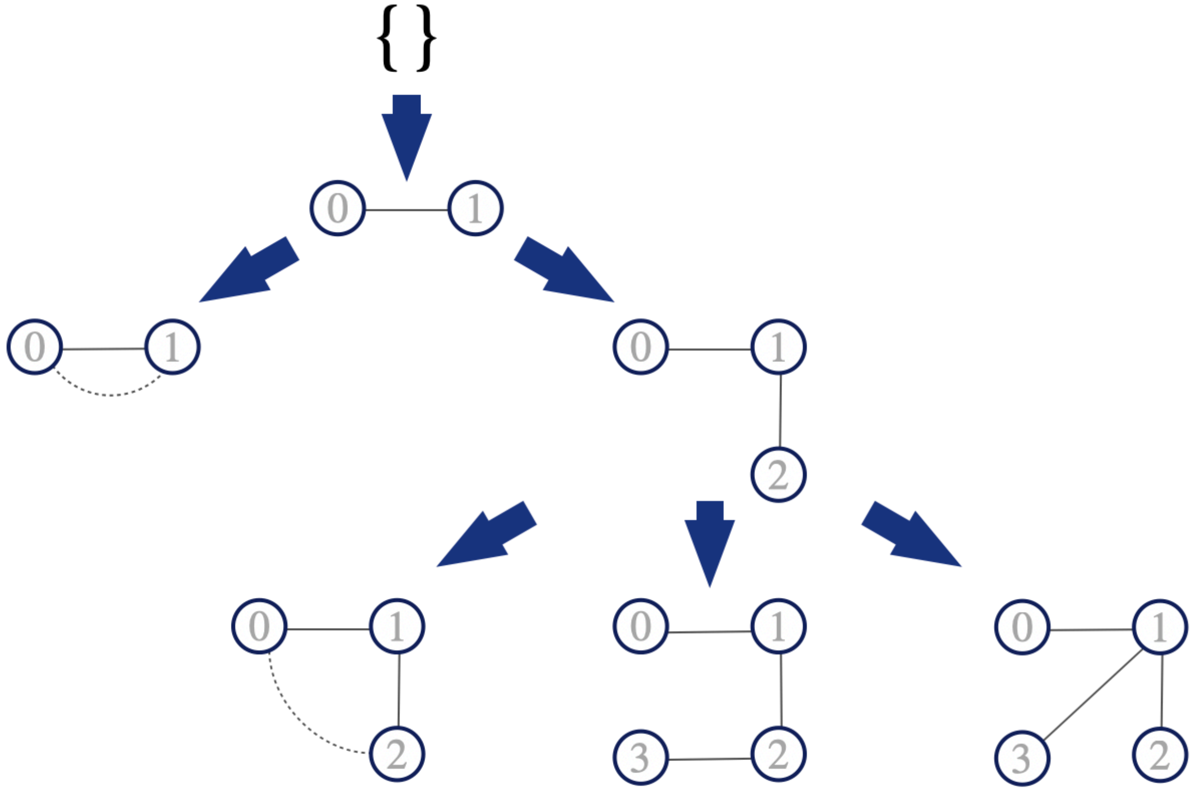
$DFScode_1 = (0,1), (1,2), (2,0), (2,3), (3,1), (1,4)$

MIN $DFScode_2 = (0,1), (1,2), (2,0), (2,3), (3,0), (0,4)$

$DFScode_3 = (0,1), (1,2), (2,0), (2,3), (3,0), (2,4)$

GSpan

DFS TREE

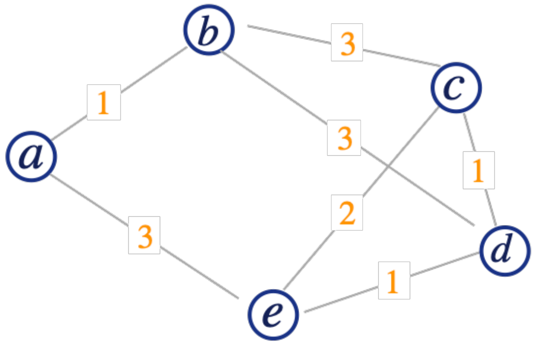


- Each node of the DFS Tree is a DFScode;
- The n^{th} level contains DFS codes for graphs with $n-1$ edges
- The n^{th} level is obtained through rightmost-extension of the parent node
- If a DFScode is not minimum or not frequent, the tree is pruned on that node (nothing will be frequent coming from that branch)
- Setting a maximum of edges (levels of the tree), the DFS tree is expanded up to the specified level and all the subgraphs in the tree are frequent

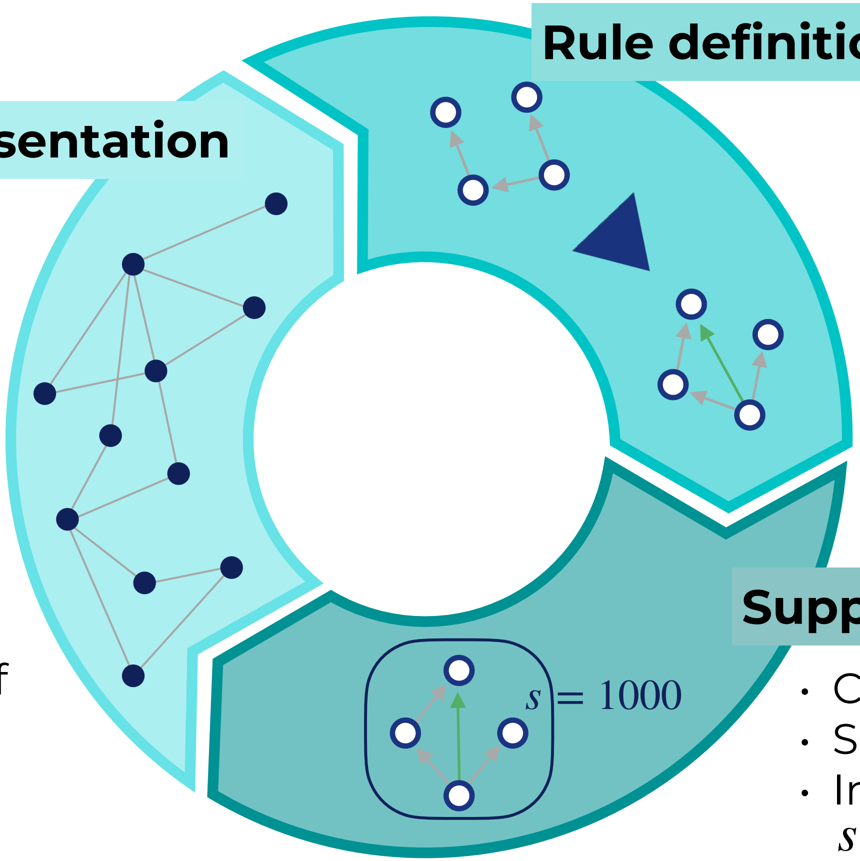
Algorithms

GERM [1]

Graph Representation



- Algorithm applied to the last graph of a **growing projection sequence**: a single edge per couple of nodes, if multiple exists choose the one with minimum timestamp (first interaction)
- Can be applied to **undirected** graphs only
- The evolution is tracked along the **whole timespan**



Rule definition

- The body is extracted from the head removing the edges with maximum timestamp
- The head and the body must be connected graphs

Support

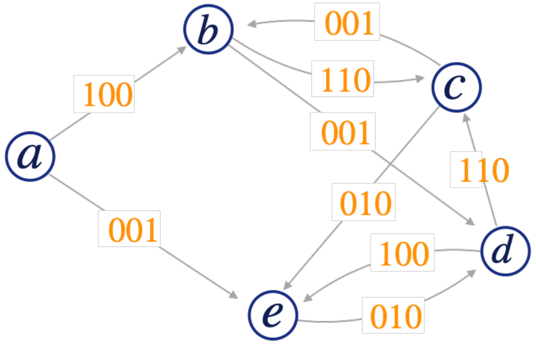
- Classical **MIB** support
- Support of a rule = support of the head
- Include also a confidence measure:

$$\frac{sup(head)}{sup(body)}$$

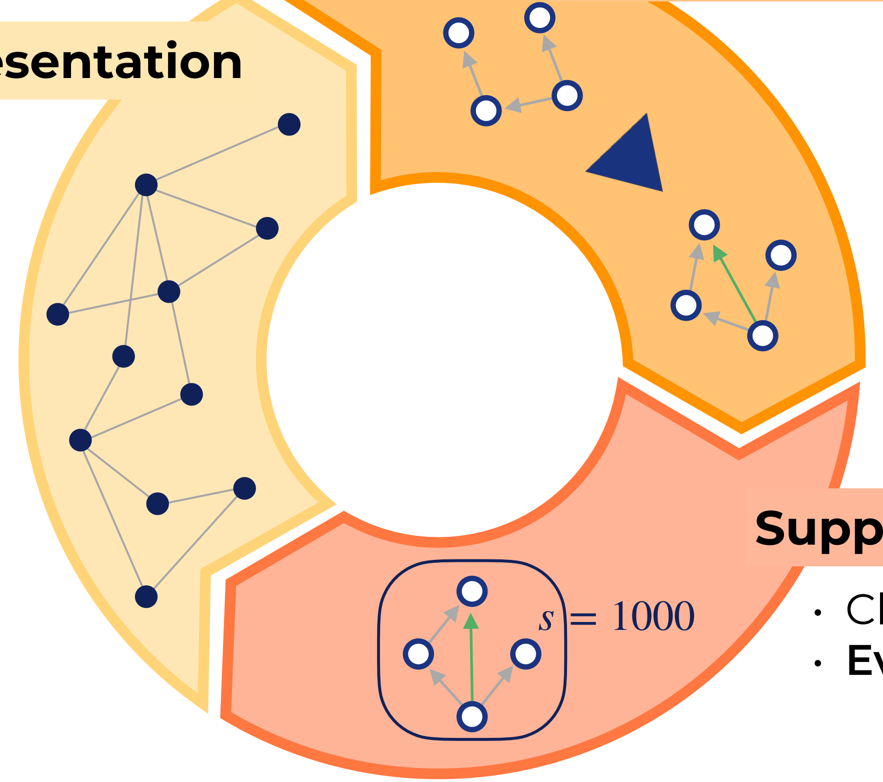
[1] Berlingerio, M., Bonchi, F., Bringmann, B., and Gionis, A. Mining graph evolution rules. In joint European conference on machine learning and knowledge discovery in databases (2009), Springer, pp. 115–130.

Evomine [2]

Graph Representation



- **Union Graph (Evolving projection):** the label on the edges encodes the evolution
- Can be applied to **directed** and **undirected** graphs
- The evolution is tracked within **consecutive timestamps**



Rule definition

- The only timestamps on the edges are t_0, t_1
- The body (pattern without the edges with t_1 timestamp) must have the **same nodes** as the head
- From body to head **something must change**, labels or edges
- The union graph of the rule must be **connected**

Support

- Classical **MIB** support
- **Event-based** support:
  - creates event graphs: subgraphs including the neighborhood of each event (edge insertion, node relabeling and so on)
  - count the event graphs in which a rule appears

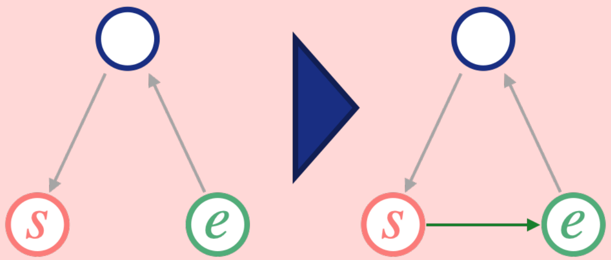
[2] Scharwächter, E., Müller, E., Donges, J., Hassani, M., and Seidl, T. Detecting change processes in dynamic networks by frequent graph evolution rule mining. In 2016 IEEE 16th International Conference on Data Mining (ICDM) (2016), IEEE, pp. 1191–1196.

Comparison

	GERM	EvoMine
Mining algorithm	Extended gSpan	Extended gSpan
Graph representation	Last graph of a growing projection sequence	• Pairwise union graph sequence (evolving projection) • Event graphs
Support	MIB	MIB + Event graph
Confidence	$\frac{sup(head)}{sup(body)}$	not defined
Type of graph	undirected	directed and undirected
Type of evolution	spanning all timestamp, relative-time rules	consecutive timestamps only
Evolutionary constraints	head and body must be connected	• union graph of the rule must be connected, • head and body has the same node set • from body to head something must evolve
Examples of use	• When the whole temporal span is important, • it makes possible to study the speed of evolution too	• When the graph is directed, • we have relabeling and edge deletion too, • when we're interested in more close evolution (consecutive timestamps only), • it can be applied for anomaly detection

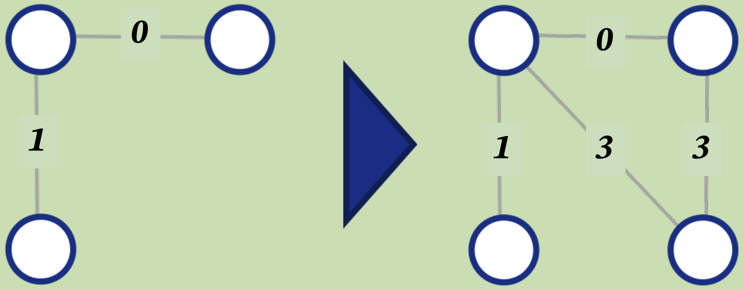
LFR [3]

- The focus is on the process that drives single links formation;
- For this reason, LF rules are more restrictive with respect to the others, but the mining time descreases;
- A null model is integrated to extract meaningful rules;
- They have a tailored support measure and also consider a confidence measure



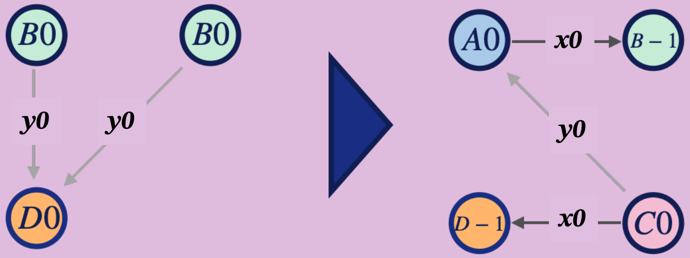
TP-MINER [4]

- It proposes the idea of representative time pattern;
- The algorithm extract the body from the head in the same way as the other ones;
- Builds a DAG from graph evolution rules
- The confidence measure takes into consideration the evolution from body to head



DGR-MINER [5]

- It is designed for labeled multigraph, both directed and undirected
- Proposes its own graph representations and support measures



[3] Leung, C., Lim, E.-P., Lo, D., and Weng, J. Mining interesting link formation rules in social networks. pp. 209–218.
[4] Yuuki, M., Ozaki, T., and Takenao, O. Mining interesting patterns and rules in a time-evolving graph. Lecture Notes in Engineering and Computer Science 2188 (03 2011)
[5] Vaculík, K. A versatile algorithm for predictive graph rule mining. In ITAT (2015), pp. 51–58

Support is not all you need

Null model

Problem

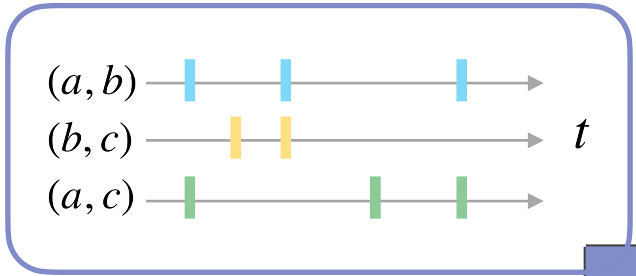
The support alone is **not enough** to measure if a pattern (rule) is **representative** of the evolution of the graph:
A pattern can be frequent as a consequence of a general process of a dynamic network, not telling anything on how the network we're studying is evolving

Solution

Apply a **null model** on the graph evolution rules algorithm

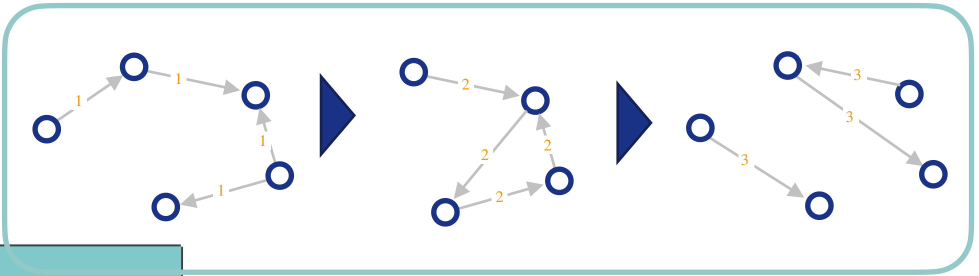
- Apply the graph evolution rules algorithm on the **real graph**
- Apply the graph evolution rules algorithm on a **randomized version** of the graph
- The rules whose support is higher in the real graph are **significative**

Microcanonical Randomized Reference Models [6]



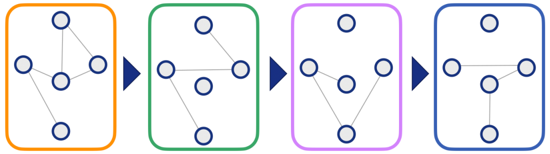
MRRM

Graph representation

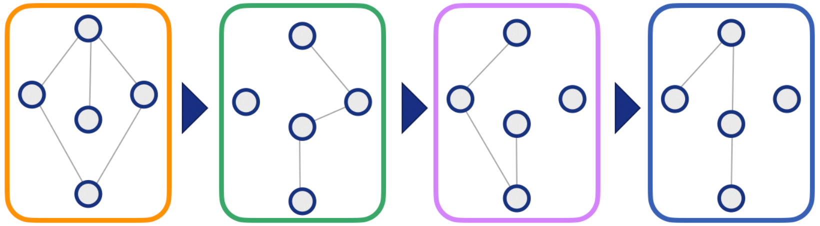


MRMMs categories		Timeline representation	Snapshot representation
What's preserving	Topology	*Timeline shuffling*	*Sequence shuffling*
	Temporal distribution	*Link shuffling*	*Snapshot shuffling*

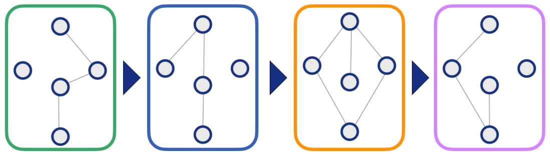
Preserving Temp. Distribution



Original graph



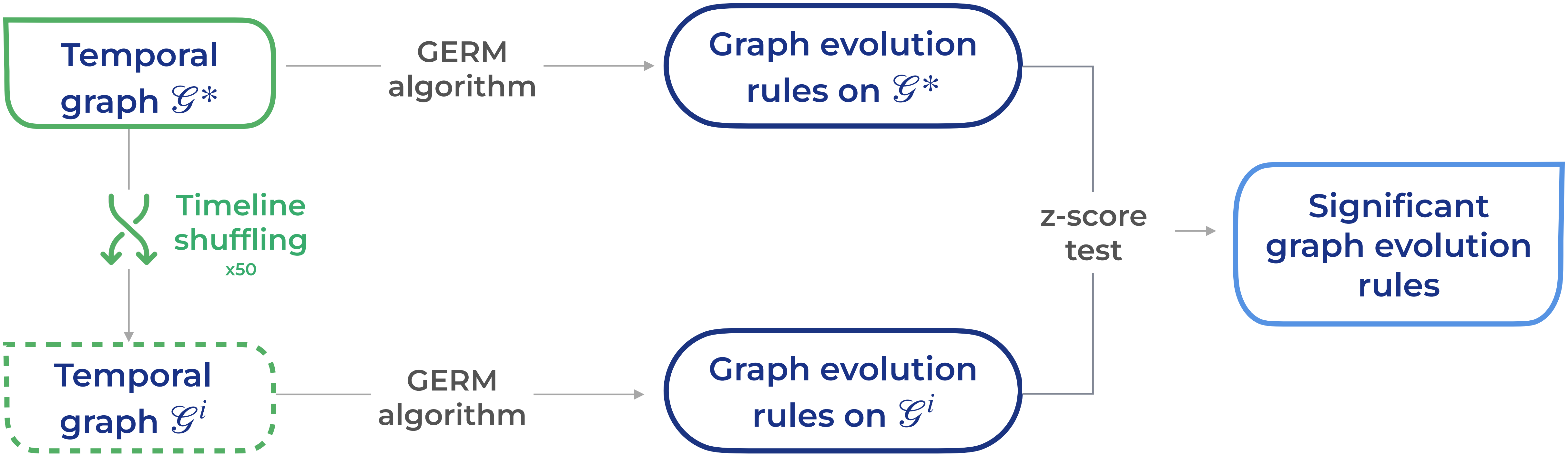
Preserving Topology



[6] Gauvin, Laetitia, et al. "Randomized reference models for temporal networks." SIAM Review 64.4 (2022): 763-830.

Methodology

PIPELINE



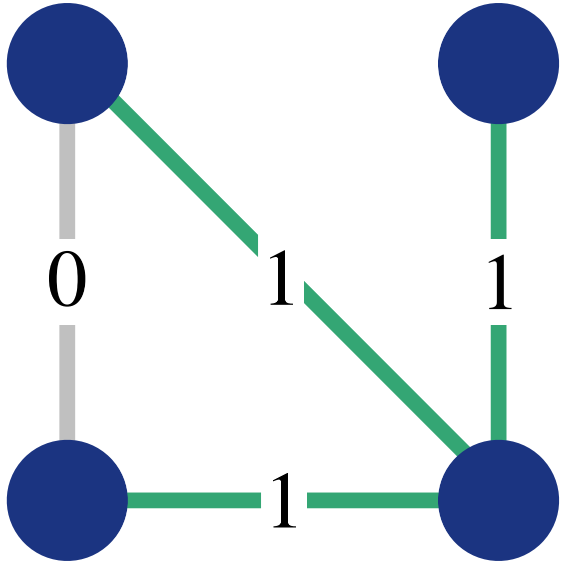
Real world case studies

IN SOCIAL, COMMUNICATION AND WEB3 BLOCKCHAIN-BASED NETWORKS

DBLP co-citation network [7]

EXAMPLE OF THE IMPACT OF THE NULL MODEL ON THE GERM ALGORITHM

(a)

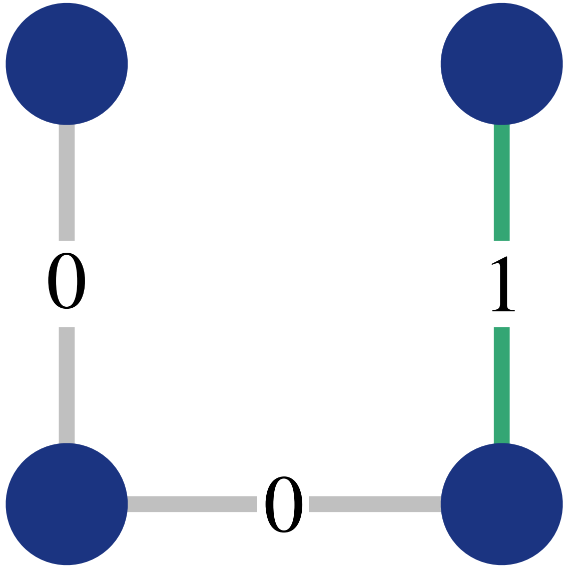


Low support
(over 100th)

High z-score

High support
(14th)

Low z-score



(b)

Rule apparently ***NOT*** important but actually ***worthy*** of attention

Rule apparently ***important*** but actually ***NOT worthy*** of attention

[7] Galdeman, Alessia, Matteo Zignani, and Sabrina Gaito. "Unfolding temporal networks through statistically significant graph evolution rules." 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2023.

EvoMine [2]

Comparing web3 platforms through GER [8]

THE PIPELINE

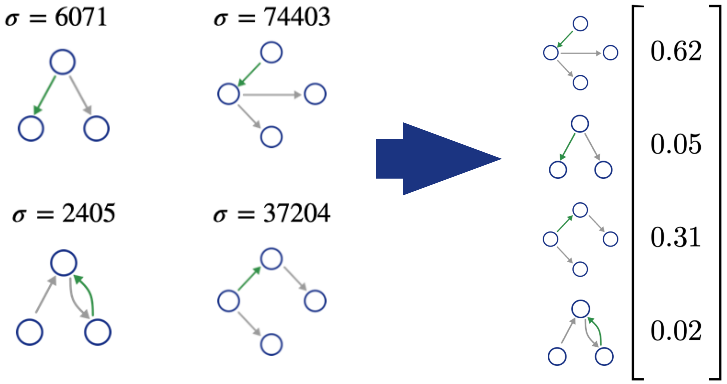
1 Web3 data modeled as temporal networks

2 Graph evolution rule mining with EvoMine

3 GER PROFILE

GER with supports

GER profile

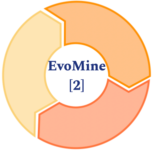


- GER profiles show the distribution over types of evolution rules for a given dynamic graph
- The comparison of the GER profiles for different graphs makes possible to find similar evolutionary behaviors

Specifically we worked on

- two networks extracted from operations (transfer and follow) on Steemit, that is a blockchain-based online social network
- two networks from NFT exchanged on two different markets (Cryptokitties and OpenSea)

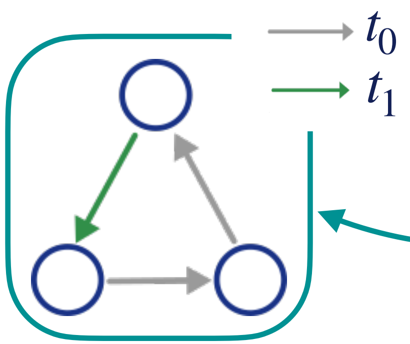
[8] Galdeman, Alessia, Matteo Zignani, and Sabrina Gaito. "Disentangling the Growth of Blockchain-based Networks by Graph Evolution Rule Mining." 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2022.

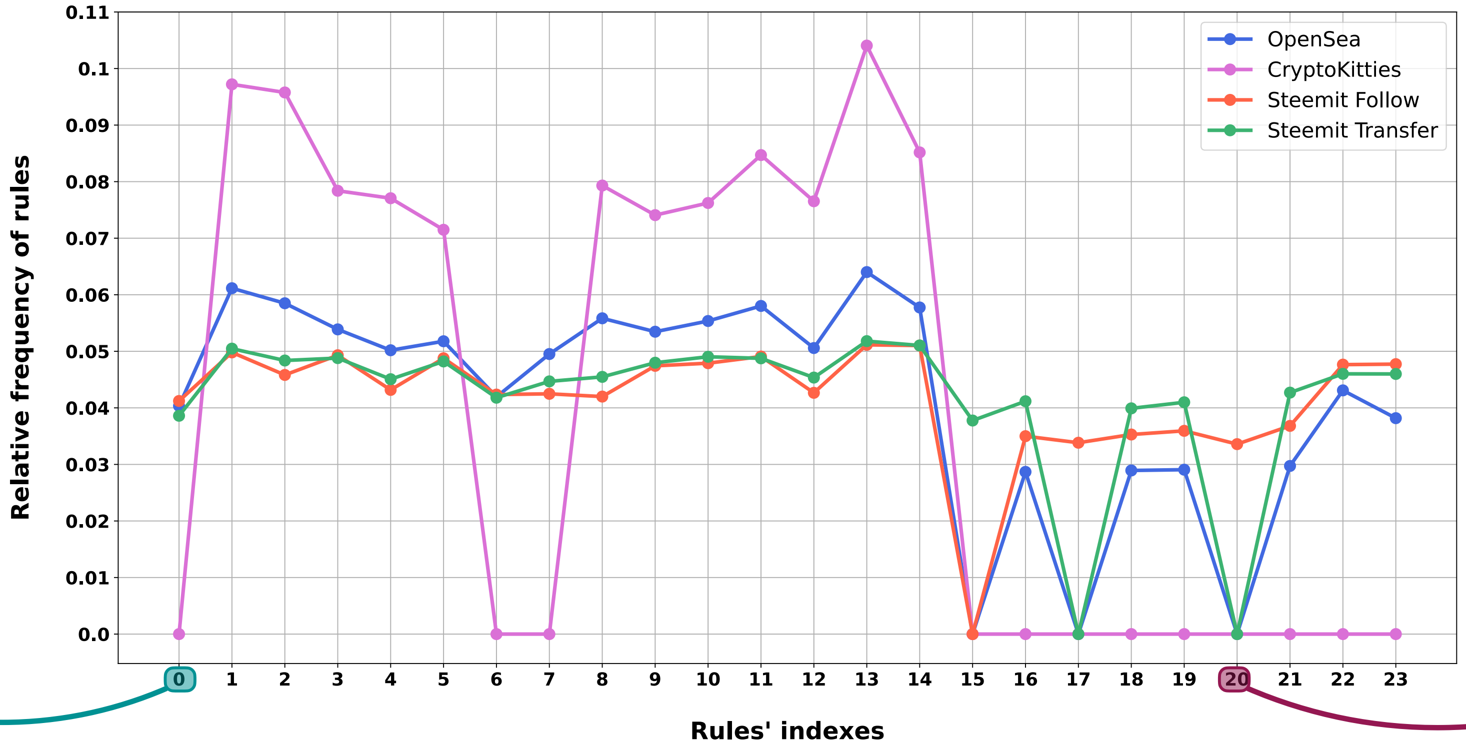


Comparing web3 platforms through GER³

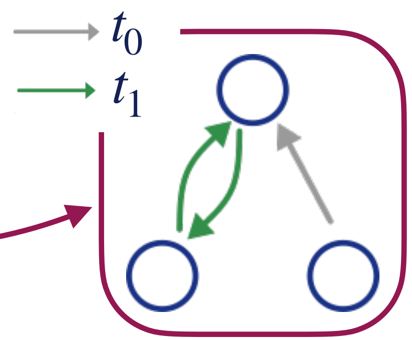
THE GER PROFILE

Not in the frequent GER set for the cryptokitties market





Frequent only in Steemit follow (the only social network)



Both cases are explainable with the nature of the network itself

Now let's play

https://github.com/alessiaatunimi/geranio

GERANIO

Graph Evolution Rules ANalytics vIsualization tOolkit

Deepnote

GitHub

Alessia Galdeman

Phd Student

alessia.galdeman@unimi.it

@AlessiaGaldeman

Alessia Galdeman

https://alessiaatunimi.github.io/

Thanks for your attention

References

[1] Berlingerio, M., Bonchi, F., Bringmann, B., and Gionis, A. Mining graph evolution rules. In joint European conference on machine learning and knowledge discovery in databases (2009), Springer, pp. 115–130.

[2] Scharwächter, E., Müller, E., Donges, J., Hassani, M., and Seidl, T. Detecting change processes in dynamic networks by frequent graph evolution rule mining. In 2016 IEEE 16th International Conference on Data Mining (ICDM) (2016), IEEE, pp. 1191–1196.

[3] Leung, C., Lim, E.-P., Lo, D., and Weng, J. Mining interesting link formation rules in social networks. pp. 209–218.

[4] Yuuki, M., Ozaki, T., and Takenao, O. Mining interesting patterns and rules in a time-evolving graph. Lecture Notes in Engineering and Computer Science 2188 (03 2011)

[5] Vaculík, K. A versatile algorithm for predictive graph rule mining. In ITAT (2015), pp. 51–58

[6] Gauvin, Laetitia, et al. "Randomized reference models for temporal networks." *SIAM Review* 64.4 (2022): 763-830.

[7] Galdeman, Alessia, Matteo Zignani, and Sabrina Gaito. "Unfolding temporal networks through statistically significant graph evolution rules." 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2023.

[8] Galdeman, Alessia, Matteo Zignani, and Sabrina Gaito. "Disentangling the Growth of Blockchain-based Networks by Graph Evolution Rule Mining." 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2022.