

Representation of the Temporal Ego-networks through Graph Evolution Rules: a Tool for Web3 Applications

Alessia Galdeman^{1,2,*}, Matteo Zignani², Christian Quadri² and Sabrina Gaito²

¹*IT University of Copenhagen, Copenhagen, Denmark*

²*Computer Science Department, University of Milan, Milan, Italy*

Abstract

The understanding of how the complex systems governing various domains, from social interactions to financial transactions, is closely connected with our comprehension of their underlying dynamic networks and their evolution patterns. In particular, the evolution of these networks provides insights into the underlying mechanisms driving their changes, which can be pivotal for applications such as node segmentation, prediction of future states, and role discovery. Among the various approaches to studying network evolution, graph evolution rules (GERs) stand out since they produce human-readable outcomes without requiring any pre-assumptions about the underlying evolutionary mechanisms. In this work, we leverage GER to derive evolutionary node profiles (NEPs), capturing the distinct patterns of how nodes change over time within the network. These profiles allow us to identify groups of accounts characterized by similar evolution rules, revealing common interaction patterns. As a case study, we apply our approach to Sarafu, a complementary currency platform following the Web3 paradigm, which offers rich temporal economic data. Sarafu represents a contemporary human complex system that integrates humanitarian aid, collaboration, and financial aspects. By analyzing Sarafu's network using our GER-based method, we identify two distinct evolutionary traits, uncovering significant behaviors that contribute to the platform's operation. Our findings suggest the effectiveness of using graph evolution rules in real-world dynamic networks, showcasing their potential to enhance our understanding of the node-level dynamics of complex systems.

Keywords

graph evolution rules, temporal networks, node representation, Web3

1. Introduction

Studying the evolution of real-world dynamic networks is critical for understanding complex systems that span various domains, from social interactions to financial transactions. The temporal dynamics of these networks can reveal underlying mechanisms driving changes and enabling applications such as anomaly detection, prediction of future states, and role discovery. Traditionally, models, mechanisms, and metrics have been introduced to interpret how dynamic networks grow and evolve, often assuming that their growth is governed by a unique parameterized mechanism, such as preferential attachment or triadic closure [1]. Alternatively, approaches that avoid specific assumptions have been developed, focusing on small substructures such as temporal graphlets or temporal motifs to capture the complex topology of networks. But to fully understand the evolution of dynamic networks, it is essential to move beyond a network-level perspective and focus on node-level temporal patterns. In fact, at the node level, we can capture how individual nodes and their immediate neighbors - the ego-network - interact over time. By examining these local interactions, we can identify recurring patterns and behaviors that are frequent within the network. In systems modeled as temporal networks, this method may enable the detection of interaction patterns among individuals, highlighting the most relevant behaviors - those that are repeatedly exhibited over time by the same or different people - and supporting diverse applications.

ITADATA2025: The 4th Italian Conference on Big Data and Data Science, September 9–11, 2025, Turin, Italy

*Corresponding author.

✉ gald@itu.dk (A. Galdeman); matteo.zignani@unimi.it (M. Zignani); christian.quadri@unimi.it (C. Quadri); sabrina.gaito@unimi.it (S. Gaito)

ORCID 0000-0003-3286-4666 (A. Galdeman); 0000-0002-4808-4106 (M. Zignani); 0000-0002-3608-8142 (C. Quadri); 0000-0003-3779-2809 (S. Gaito)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

In this work, we embrace the ego-networks viewpoint for the analysis of temporal networks but rely on graph evolution rules (GERs) rather than dynamic graphlets [2] or temporal network motifs [3]. As the latter ones, graph evolution rules are based on identifying small relevant temporal subgraphs, but they express the dynamics of the interactions occurring inside ego-networks according to association rules where pre- and postconditions are subgraphs, and encapsulate the transitions between network states over time. Indeed, GERs make the mechanisms driving interactions in ego-networks even more explicit and easily readable (see an example in Figure 1). Based on the GER concept, we developed a methodology that associates an evolutionary vector, namely the node evolutionary profile (NEP), to nodes in a temporal network built from transactional data. A NEP is a vector representation of the rules characterizing the dynamics of the interaction within an ego-network as they are captured by GERs, i.e., it captures the growth patterns of personal interaction networks. The pipeline of the methodology involves two main tasks: the extraction of ego-networks from a temporal network and the identification of graph evolution rules using the EvoMine algorithm, one of the state-of-the-art algorithms for mining graph evolution rules.

We showcase the potential applications of NEPs on transactional data from the Sarafu platform. Sarafu is a digital complementary currency platform developed by Grassroots Economics to facilitate mobile payments and distribute humanitarian aid in Kenya, and is representative of a vast landscape of applications based on the Web3 paradigm [4]. From this data, we construct a temporal transaction network where nodes represent users and directed edges represent timestamped transactions between them. In the Sarafu temporal network, we leverage NEPs to identify groups of accounts - nodes - characterized by similar evolution rules in the transactions occurring within their ego-networks. Indeed, we identify two primary evolutionary traits. The first one is characterized by a predominance of a single rule, which involves the creation of a single link at the next timestamp when the precondition is empty. This rule accounts for approximately 20% of the interactions in these ego-networks. The dynamics in this cluster primarily involve star-like (one central node connecting to multiple others) and chain-like (sequential connections) expansions without a precondition, indicating rapid initiation of transactions. In contrast, the second trait features a more even distribution of various graph evolution rules, suggesting a more diverse set of interaction patterns with no single rule dominating the dynamics. Finally, the assortativity analysis revealed a tendency (assortativity coefficient of 0.59) for accounts to interact with others that exhibit similar ego-network dynamics. This suggests that while there is a certain level of homophily based on transaction patterns, connections are not strictly confined to similar behavior types.

To sum up, the representation of the dynamics within ego-networks using GERs offers a potential tool for uncovering and understanding the intricate patterns of interactions in complex temporal networks. By leveraging GERs, we can gain insights into how individual nodes and their immediate neighborhoods evolve, providing valuable information for applications such as node segmentation, prediction of future network states, and role discovery. This approach not only enhances our comprehension of the underlying mechanisms driving network changes but also supports the development of more accurate models for analyzing and predicting the behavior of real-world dynamic systems.

2. Related work and background

In this section, we provide a brief background on the frequent graph mining approach we use to characterize the dynamics of ego-networks, and we summarize the related works on networks and node representation based on (temporal) subgraphs, as well as works about our case study: Sarafu, the complementary and humanitarian aid crypto-currency.

Graph Evolution Rules - GER. Our approach to describing dynamics in temporal ego-networks is mainly rooted in graph evolution rules. GERs mining is a frequency-based pattern discovery method that allows analyzing the evolution of temporal networks. The goal of graph evolution rules (GERs) is to discover frequent local changes occurring repeatedly throughout the network evolution [5]. Following

the *association rules* concept in the data mining field, GERs are composed of a precondition (called *body*) and a postcondition (called *head*). The rules' interpretation is that a subgraph that matches the body will probably evolve into the head, making the outcomes human-readable and explainable. For instance, Figure 1 shows a representation of a graph evolution rule that indicates the presence of triadic closure.

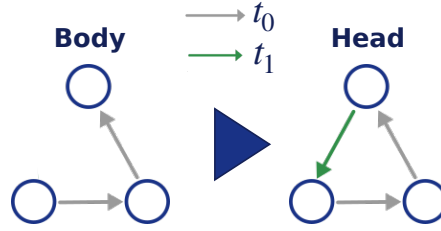


Figure 1: Example of graph evolution rule. On the left graph with two links - grey arrows - created at time t_0 , while on the right, the head of the rule where the graph on the left evolves by adding the new link - green arrow - at the successive timestamp t_1 .

GERs are a powerful method that can enable the development of more accurate network evolution models for predicting future network changes, or be used to distinguish them from other graphs, whose evolution is governed by different mechanisms. State-of-the-art methods focused on detecting the topological evolutionary mechanisms share the same two-step methodology: first, they extract rules via frequent subgraph mining, and then they filter the output using quantities such as the support and/or confidence measures. In the literature about GERs, one of the first methods is GERM, developed by Berlingerio *et al.* [5]. Rules identified by GERM detect undirected edge insertion events, considering the relative time differences. Edge removals and node and edge relabeling are not captured. Another rule mining algorithm was proposed by Leung *et al.* [6] and further adopted by Ozaki *et al.* [7]. The rules detected are called LFR (Link Formation Rule), and they aim to capture how directed links between a source and a destination are created. GERM and LFR algorithms used the minimum image-based support [8] and Gspan Frequent Subgraph mining [9]. Ozaki *et al.* [7] have proposed an undirected version of LFR, together with a method to find relationships between rules. Moreover, Vakulík [10] has developed a method, called DGR miner, whose evolution rules also capture edge deletion and relabeling. Lastly, EvoMine [11] shares the same idea of DGR, allowing more advanced evolution patterns than the simple edge insertion. Furthermore, EvoMine's authors have also proposed a novel type of support measure. In this paper, we chose EvoMine to detect evolution rules because it is the most complete one and offers an alternative type of support measure. Other works on the identification of evolution rules can be found in the literature; however, they focus more on the evolution of attributes, ignoring [12] or giving less importance [13] to the structural evolution of the networks.

Node representation based on (temporal) subgraphs Recent advancements in node vectorial representation have leveraged frequent subgraphs, motifs, and graphlets to enhance the richness of temporal graph embeddings. One notable approach is the Neural Temporal Walks (NeurTWs) [14], which leverages structural and tree traversal properties along with time constraints to capture dynamic patterns in temporal graphs. This method allows for an effective characterization of temporal nodes through representative motifs. Another prominent study embeds nodes based on their structural roles within the network, providing versatile representations for dynamic and evolving graphs [15]. On the other side, among the non-neural approaches, Hulovatyy *et al.* [2] proposed a vectorial representation of nodes using dynamic graphlets. This method adopts a common approach in this context, which is to decompose networks into smaller segments [16, 17, 3] to characterize node behavior over time. Along this line, a strategy proposed by Longa *et al.* [18, 19] suggests adopting an egocentric perspective. This method tracks the evolution of node neighborhoods across temporal layers, collecting egocentric temporal subgraphs at each time step and condensing them into egocentric temporal motifs (ETMs), facilitating efficient identification of recurring interaction patterns in dynamic contexts through comparison against a null model.

Complementary currency. Complementary currencies (CCs) are alternative currencies that supplement national currencies in various geographic contexts. Viewed as fungible vouchers redeemable for goods and services, there have been 3,500 to 4,500 CC projects in over 50 countries since the 1980s. Among these projects, Sarafu is a complementary currency on a blockchain created by the Grassroots Economics (GE) Foundation. Users make payments via mobile phones, transferring Sarafu tokens to other registered users. During the COVID-19 pandemic, the Kenyan Red Cross used Sarafu to distribute humanitarian aid, with new users receiving free tokens backed by donor funds. Sarafu has been the subject of several studies since the GE Foundation provided an anonymized dataset of user transactions spanning a year and a half. For instance, a dataset paper offering context and background of the platform has been provided by Mattsson *et al.* [20], while Ussher *et al.* [21] analyzed the dataset and the Sarafu project’s history. Mqamelo [22] studied the impact on local economic engagement, and Mattsson *et al.* [23] modeled money circulation within Sarafu’s network. Finally, Ba *et al.* [24, 25] analyzed cooperation behaviors within the Sarafu network, highlighting cooperation patterns, the significance of group accounts, and the role of the geographical positions of accounts.

3. Methodology

From a node-centric perspective, our main aim is to represent nodes based on the mechanisms that characterize the evolution of the interactions surrounding them. The methodology to get this kind of representation is based on two main tasks: *a)* the extraction of the ego-networks from the overall temporal network describing the system, i.e., interactions surrounding every single node; and *b)* the identification of the mechanisms/rules driving the evolution of each ego-network through the computation of the graph evolution rules. In this section, we detail these two main tasks and propose a vector-based representation for nodes, rooted in graph evolution rules, namely the *node evolutionary profile - NEP*.

3.1. Ego-networks from temporal networks

In this work, we model the set of interactions or transactions among the members of a networked system following the broad definition of *temporal network* $\mathcal{G} = (V, E)$ proposed in [26, 27], where:

- V is the set of users in the system; and
- $E = \{(u, v, t) \mid t \in [1, T], (u, v) \in V \times V\}$ is a set of timestamped directed links (u, v, t) . Each link corresponds to an interaction/transaction from node u to user v that occurs at time t .

By the temporal network definition, we cover both repeated interactions between pairs of nodes, or unique interactions, leading to the formation of growing networks, i.e., a first approximation of the evolution of online social networks, for instance.

To accomplish the first task, we extract the temporal ego-network of a generic node u from the temporal graph \mathcal{G} . For each node u , its ego-network $S(u) = (V_u, E_u)$ corresponds to the temporal subgraph induced by u ’s neighborhood, including u itself. Formally, the set of nodes is defined as $V_u = u \cup \Gamma(u)$ where $\Gamma(u)$ is the neighborhood of node u . The set of edges is defined as $E_u = \{(v, w, t) \mid (v, w, t) \in E, v \in V_u, w \in V_u\}$, i.e. all temporal links whose endpoints are in V_u . The extraction of temporal ego-networks leads to the creation of a set of small temporal subgraphs that can facilitate a parallelization of node-centric analyses.

3.2. EvoMine

To deal with the second task, i.e., identifying the graph evolution rules characterizing a temporal ego-network of a node, we rely on the EvoMine algorithm [11] since it offers a richer set of link/node events for rule extraction, including edge deletion and the relabeling of nodes and edges. Moreover, the EvoMine approach based on consecutive snapshots is suitable for temporal networks based on interaction/transaction data, where a link can appear and disappear many times. On the contrary, other

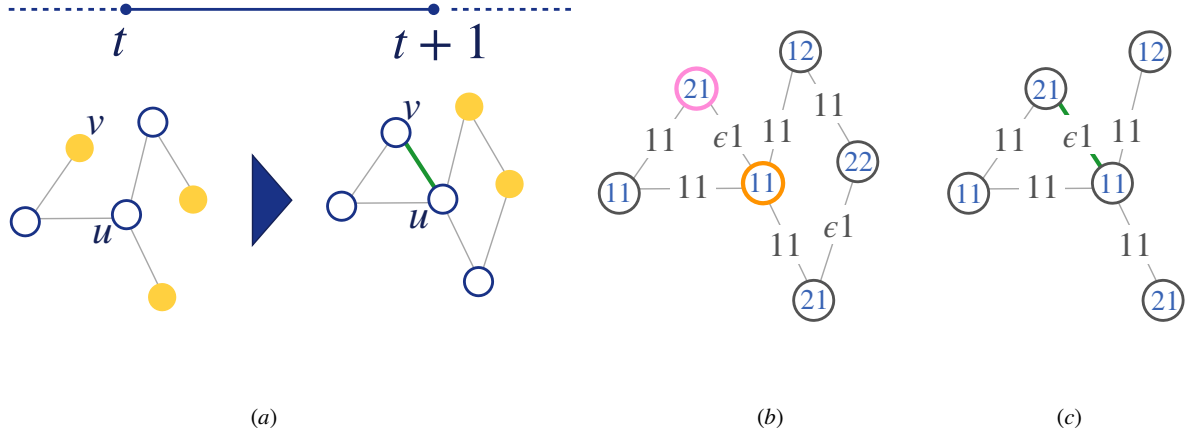


Figure 2: Union and event graph - toy example. a) shows a two-timestamp graph sequence, while b) represents its union graph, where the evolution of edges and nodes is encoded in edges/nodes labels. In the edge labels, the number of characters indicates the length of the graph sequence. As for the node label, each element in position i indicates the attribute of the node in timestamp i (in this case 1 or 2, indicating the node being white or yellow, respectively). Meanwhile, in the edge labels, ϵ or 1 indicate whether the edge is missing or not in the timestamp corresponding to the position of the character in the label. For instance, the pink circled node has label 21 because at timestamp t the node is yellow while in timestamp $t + 1$ it is white. Finally, c) shows the event graph of the insertion of the edge between u and v (the pink and orange circled nodes). It is the subgraph of b) induced by the neighboring nodes of u and v .

state-of-the-art algorithms for GER extraction, such as GERM [5] and LFR [6] - which only capture edge insertions - are not able to deal with multiple link insertions and deletions occurring between two nodes. In the following, we briefly summarize the main aspects of EvoMine to better understand which kind of graph evolution we are able to identify through it.

Topological constraint of rules. The main characteristic of the evolution rules returned by Evomine is their focus on changes between two consecutive timestamps, i.e., rules describing how a precondition subgraph will evolve into a postcondition subgraph in the very next future (next timestamp or snapshot), ensuring that any modifications specified in the postcondition occur immediately, following the timestamp of the precondition's edges. In EvoMine a rule $r : (G_{pre}, G_{post})$ is *valid* whether: *i*) V_{pre} , i.e. the set of nodes of G_{pre} , matches V_{post} ; *ii*) $E_{pre} \neq E_{post}$ or $\ell_{pre} \neq \ell_{post}$, ensuring evolution - at least a change between consecutive timestamps; and *iii*) the union graph of (G_{pre}, G_{post}) must be connected, to guarantee the rule captures a localized process. In particular, the *union graph* $\mathcal{G}_U(G_1^T)$ corresponds to a merging of the graphs in a graph sequence $G_1^T = (G_1, \dots, G_T)$ with $G_t = (V, E_t, l_t)$. The merging results in a labeled static graph that includes the same set of nodes V and the union of all edge sets E_t , $\forall t = 1 \dots T$. Through the merging, we assign labels to nodes and edges by concatenating the labels of all timestamps included, thus encoding the temporal evolution. An example of a union graph obtained from the graph sequence in Figure 2a is reported in Figure 2b. Here, the label 21 for the pink node indicates that the node had label 2 in t , and label 1 in $t + 1$. Similarly, the label assigned to the link connecting the pink and orange circled nodes - $\epsilon 1$ - indicates that the link was not in E_t , but was present at E_{t+1} .

The algorithm. The union graph is the main input of the EvoMine algorithm. Indeed, the algorithm applies a frequent connected subgraph mining method (Gspan [9]) to the sequence of union graphs for consecutive snapshots, i.e. $\mathcal{G}_U(G_i^{t+1})$. By employing this mining algorithm, EvoMine ensures that the node set and connectivity properties (constraints *i*) and *iii*) described above) are valid by definition. Moreover, to not violate the constraint regarding edge or label changes, the algorithm filters the resulting patterns based on the specified properties.

Event-based support. A further advantage of Evomine is that it provides two types of support measures: (i) an embedding-based support, i.e., the minimum image-based support, and (ii) an event-based support. In our pipeline, we adopt the second type of support, which states that the support of a rule is determined by the total number of change events that include the rule. In the event-based setting, the input for Gspan is a set of *event graphs*, i.e., a subgraph of the union graph $\mathcal{G}_U(G_i^{t+1})$ induced by the event neighborhood. The *event neighborhood* is defined as the neighborhood of the node(s) involved in the node or edge event¹. Finally, the event-based support $\sigma_{event}(r)$ of a rule r is the number of event graphs in the overall set of event graphs that contains the union graph of r as a subgraph.

For example, given the snapshots t and $t + 1$ of a temporal graph \mathcal{G} as the one depicted in Figure 2a, we obtain its union graph $\mathcal{G}_U(G_i^{t+1})$ reported in Figure 2b. If we consider the creation of the edge between u and v as the target event, the corresponding event graph is the subgraph induced by u , v , and its neighbors (depicted in Figure 2c).

Algorithms parameters. The outcomes of EvoMine are mainly influenced by two parameters. The first one is the support s , which specifies the support threshold for rules to be included in the output, and allows us to discard rare evolution patterns. The second crucial parameter is the maximum number of edges allowed in the union graph of any rule in the output, and impacts the complexity of the evolution rules. Details about the selection of the parameter for the specific case study are reported in Section 5.

Scaling strategy. We adopt a few parallelization strategies to scale the method to relatively large temporal networks that may span long periods. First, the computation for each node can be parallelized since each ego-network can be extracted and treated independently. However, this may not be enough since computational times increase with the length of the sequence graph. To cope with this issue, we rely on how EvoMine computes the event-based support: the support for each rule is computed across consecutive timestamps (event graphs) in a pairwise manner, and then the results for each pair of consecutive snapshots are aggregated. This approach is not available through the original implementation, but it can be run in parallel by applying EvoMine on each pair of consecutive timestamps in the timespan $\{t_1, t_2, \dots, t_n\}$. This parallelization strategy asks for a precaution since each application of EvoMine independently generates rules (with edge lists and support) whose identifiers are not unique across different pairs of snapshots - a canonical form to identify isomorphic patterns across the various outputs is missing. To deal with this issue, and effectively aggregate the supports of a rule across timestamps, we integrate into the algorithm an isomorphism check which assigns a unique identifier to classes of isomorphic temporal pattern [28]. In this way, despite the external parallel and independent execution, the results remain comparable in the aggregation. Finally, we introduce a further check in the parallelization strategy to apply the computation of the support only on pairs of consecutive snapshots that contain at least a link insertion event. For example, if in a temporal network, only the timestamps $t = [1, 2, 5, 7, 8]$ contain link events, EvoMine is only run on the pairs (1, 2) and (7, 8). The cost reduction resulting from applying this filter depends on the frequency of the events and how constant the frequency is.

3.3. Node Evolutionary Profile

In static and temporal networks, the distribution of measures based on static and/or temporal subgraphs of different order and size has been used for encapsulating the static and dynamical signature from both a network- and node-level perspective. In this sense, the graphlet-degree vector in [29] and its extension to the temporal setting given by the dynamic graphlet degree vector [2] represent some of the extents to derive a network or node representation expressing the (temporal) subgraphs a node is involved in. Here we proposed a similar representation for nodes that relies on the graph evolution rules characterizing the evolution of nodes' ego-network.

¹In our case, we focus on edge events only.

We denote the vector representation as *Node Evolutionary Profile* - *NEP*, and it represents the distribution of the graph evolution rules for the ego-network $S(u)$ of the node u . The construction of the node evolutionary profile is based on the graph evolution rules and their supports computed by EvoMine on each ego-network $S(u)$; while a vector representation common to all nodes is supported by the unique and common identifiers for rules based on the canonical form (see the paragraph on "Scaling Strategy" in Section 3.2). Specifically, each element of the Node Evolutionary Profile $nep(u)$ is defined as follows:

$$nep(u)_i = \frac{\sigma_{event}(r_i)}{\sum_{j=1}^n \sigma_{event}(r_j)} \quad (1)$$

where $\sigma_{event}(r_i)$ is the event-based support of the rule r_i in the u 's ego-network and n is the number of distinct GERs identified by EvoMine on the whole set of nodes. In short, given an ego-network of a node u , its NEP represents a signature of its evolution as well as a compact representation based on the dynamics of the interactions among the neighbors of u and with the neighbors and u .

4. Dataset

Our work on node temporal behavioral representation using graph evolution rules considers as a case study a transaction network in the Web3 landscape: the Sarafu network. Sarafu² is a digital complementary currency token developed by the Grassroots Economics (GE) foundation³. This platform enables users to make payments via mobile phones by transferring Sarafu tokens to other registered users. The Kenyan Red Cross leveraged Sarafu tokens to distribute humanitarian aid during the COVID-19 pandemic [21]. New users received free Sarafu tokens, backed by donor funds, to help maintain the system's operation. The Sarafu dataset represents a unique collection of transactions enriched by elements of humanitarian aid and collaboration, facilitated by the involvement of the Red Cross and the presence of group accounts, i.e. special wallets in the aid network managed by local communities to support micro-loans and the economy of the local communities.

Data Collection and features. As collected by [30], the Sarafu dataset provides comprehensive and anonymized data on token transactions and user characteristics from January 2020 to June 2021. The dataset – a quite unique playground in the field of temporal graph analysis, mining, and machine learning – encompasses a total of 412,050 economic transactions involving 40,343 users. Each transaction includes anonymized IDs for its source and target, representing the sender and receiver of cryptocurrency tokens. Additionally, the transaction weight, indicating the number of tokens transferred from source to target, is recorded. Critical for this study is the timestamp, detailing the precise date and time of each transaction down to milliseconds. Since the millisecond temporal granularity would result in a huge set of extremely sparse snapshots, making the application of EvoMine unfeasible, we aggregate timestamps into a daily granularity. As for information on user profiles, data include attributes such as business type (user-provided standard categorization), geographical information about the user's residence, and a distinction between beneficiaries (regular users) and group accounts. This latter distinction between regular and group accounts is fundamental in the remainder of the paper since it allows focusing only on token transactions capturing value exchanges for the communities and not transactions useful only for the functioning of the platform.

Relevance to temporal behavioral representation. The rich temporal data contained within the Sarafu dataset makes it an ideal case study for applying our dynamic graph evolution rules approach. The dataset represents a contemporary human complex system that integrates humanitarian aid, collaboration, and financial aspects. By analyzing this dataset, we aim to capture and characterize the temporal behavioral patterns of nodes within this transaction network, providing insights into the dynamics of digital currency exchanges in a humanitarian context.

²Sarafu means "currency" in Swahili

³<https://www.grassrootseconomics.org/pages/about-us.html>

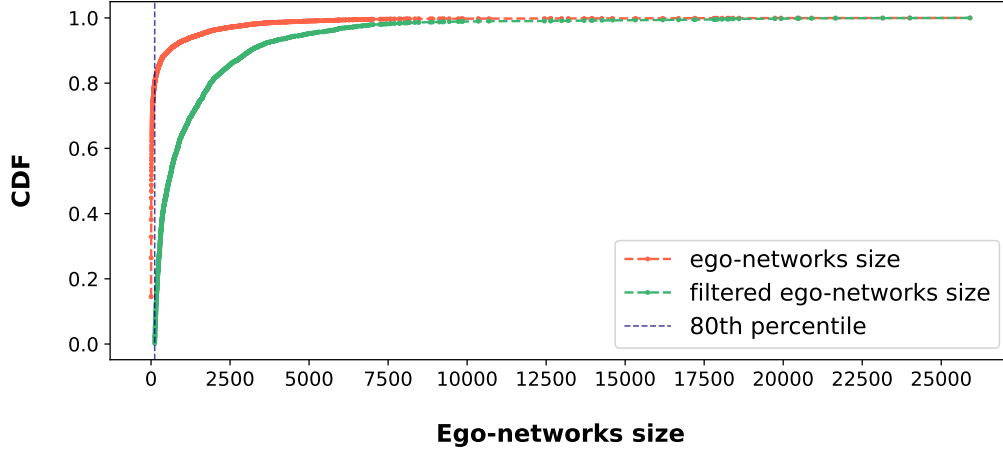


Figure 3: Ego-network size distribution. The red curve displays the Cumulative Distribution Function – CDF — of ego-network size on the 16030 ego-networks obtained after applying the filter on consecutive snapshots. The vertical blue dotted line indicates the 80th percentile (116 transactions) of the distribution. The green curve shows the distribution (CDF) of ego-network size for the most active ego-networks (3207 elements).

5. Results

We applied the described methodology to the Sarafu transaction network, limiting our attention to economical exchanges among regular users and between regular users and group accounts. Through Node Evolutionary Profiles (NEPs) we point out interesting insights into the dynamics of ego-networks. By identifying distinct interaction patterns and traits, we showcase the efficacy of NEPs in capturing temporal behaviors. The results highlight two primary evolutionary traits within the network that can be extracted by clustering NEPs. This analysis of Sarafu serves as a showcase of the potential applications of NEPs on temporal networks — especially networks reconstructed from Web3 platform data — demonstrating their capability to uncover the dynamics of ego-networks in complex transactional networks, which can be extended to various other domains and contexts.

5.1. Preprocessing and filtering

The extraction of the ego-networks from the original transaction network of Sarafu has returned 40343 ego-networks, which were reduced to 16030 after applying the filter on consecutive snapshots (see Section 3.2). This important reduction in the number of valid ego-networks indicates that more than half of the accounts do not show interactions with and among their neighbors in consecutive timestamps⁴. First, we investigate, for each ego-network, the number of included interactions to assess if, even in this case, this quantity shows a heavy-tail trait like most of the phenomena concerning real-world networks and human dynamics [31]. To this aim, in Figure 3 we report the cumulative distribution of the number of interactions (ego-network size) in each ego-network. The distribution of the ego-network size (red curve) is skewed, with the majority of nodes having very few interactions within their ego-networks. This observation may impact the outcomes of our analysis since if the temporal subgraph from which we identify the evolutionary profile is too small, it does not have enough data to actually describe the temporal behavior of a node. For this reason, based on the distribution, we only consider nodes whose ego-network presents at least 116 interactions, corresponding to the 80th percentile of the ego-network size distribution, shown in Figure 3 by the blue dotted line. Thus, we obtain 3207 ego-networks, whose size distribution is depicted in Figure 3 with a green line. In short, the analysis of ego-network sizes allowed us to identify the most significant accounts in the Sarafu networks in terms of transaction activity, and at the same time, it highlighted that most economic activities are handled by a small

⁴We are aware that the reduction of ego-network is quite important, but not applying the filter on consecutive timestamps would have altered the dynamics within ego-networks.

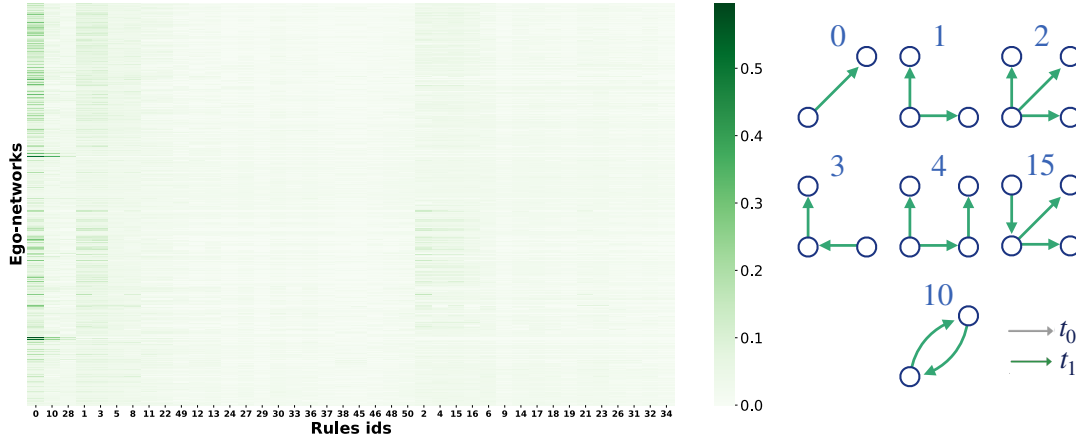


Figure 4: NEPs. Visualization as a heatmap of the matrix obtained by stacking all the NEPs. Each row corresponds to an ego-network while the dimensions of NEPs, i.e., the evolution rule, have been ordered by multiple criteria which capture the increasing complexity of the pattern: the first criterium is the order - number of nodes - of the rule, then we consider the size (number of edges) and finally the number of timestamps (1 or 2). In the heatmap, the intensity of the green color is proportional to the frequency of the GER. On the right, the most common GERs are represented through the compact visualization described in Figure 1.

portion of the accounts in the system.

5.2. NEPs

After the preprocessing and filtering phase, we apply the EvoMine algorithm to the 3207 ego-networks. In particular, we fixed 1 as the minimum support σ — the algorithm returns all the graph evolution rules in the graph — and a maximum number of three edges per pattern, using the event-based support. As a result, after applying the general mapping procedure, we obtained 40 different graph evolution patterns, which correspond to the dimensions of the node evolutionary profiles. We collect all the NEPs by stacking them into a matrix, and we visualize them through a heatmap, where rows represent ego-networks and columns indicate the IDs of the graph evolution rules. The matrix of all the NEPs is displayed in Figure 4. From a column-wise inspection, we observe that in general, only a limited set of graph evolution rules characterizes the dynamics of the transactions in ego-networks. Indeed, only rules 0, 1, 2, 3, 4, 10 and 15 show high frequency values, with the first rule (0) being the most frequent for many ego-networks.

We report these frequent GERs on the right side of Figure 4. We note that six out of seven rules (0, 1, 2, 3, 4, 15) describe star- or chain-like expansions starting from an empty precondition, while rule 10 expresses transactions that become reciprocal in just one day. The *reciprocal rule may represent a distinctive trait of the Sarafu*, since it mirrors the cooperation among the members of local communities. In this case, it is worth noting *this cooperative behavior actualizes in only a day*. Moreover, from a data perspective, the skewed distribution of rules points out that it is very likely that if we reduce the dimensionality of NEPs, most of the information in the data will remain. On the other side, from a row-wise inspection, we observe a certain level of variability in NEPs, so *there is not a common trait characterizing the dynamics of the interactions in ego-networks*, even if a few representative traits are identifiable.

5.3. NEPs clustering

The analysis of NEPs has pointed out two principal observations that are fundamental for showcasing how NEPs can be exploited for data discovery tasks: *i)* only a few GERs are frequent in NEPs; and *ii)* NEPs are varied but with a limited level of variability. Based on these observations, we focus on identifying a few classes that may represent dynamic traits of ego-networks' evolution in Sarafu. We apply a clustering pipeline on the NEP matrix to identify these possible different traits. Taking advantage

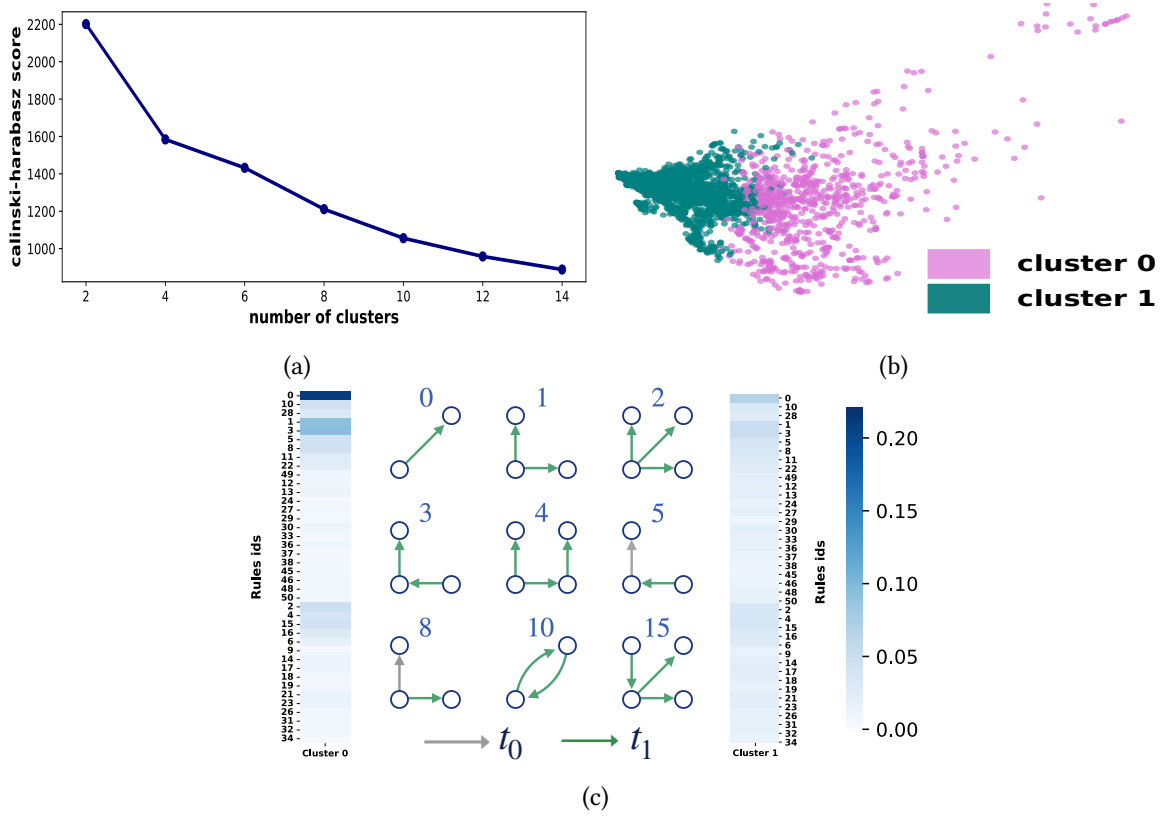


Figure 5: In a) The Calinski-Harabasz score as a function of the number of clusters k returned by the agglomerative clustering algorithm. The highest CH-score corresponds to $k = 2$, i.e., two clusters. In b) NEPs are visualized in 2 dimensions with points and colored according to the membership of one of the two clusters. To visualize NEPs in a 2D space we ran dimensional reduction (PCA) on the two principal components. In c) the NEP centroids of the two clusters, along with the most frequent, on average, GERs. GERs are represented through the compact visualization in Figure 1.

of the first observation on NEP dimensions, first, we performed dimensionality reduction by Principal Component Analysis (PCA) to reduce NEP dimensions while preserving most of the information in the NEP matrix; then we ran a hierarchical clustering algorithm on the transformed NEPs to identify groups of ego-networks showing the same evolutionary trait. As for the PCA algorithm, we selected the number of components using the explained variance ratio: fixing a cumulative percentage of explained variance to 0.99, we still halve the dimensions from 40 to 24, finding a lower-dimensional representation of NEPs that is still informative as the original one.

The transformed NEPs feed an agglomerative clustering algorithm using Ward as the linkage strategy since it is less sensitive to noise and should return more even clusters in terms of size. When using agglomerative clustering, one has to select the best value for the number of clusters k , a fundamental parameter not known a priori that must be fixed before running the algorithm. In this case, we select k as the value that maximizes the Calinski-Harabasz (CH) score of the agglomerative clustering by varying k from 2 to 14 with step 2. The Calinski-Harabasz (CH) index assesses clustering performance by comparing the variance between clusters to the variance within clusters: a higher CH score indicates compact and well-separated clusters, suggesting a more optimal clustering configuration. According to the trend of the CH scores reported in Figure 5a, we choose $k = 2$. Thus, *there are two main traits characterizing the dynamics of the transactions occurring within ego-networks*, and consequently, two groups of accounts. In Figure 5b we display these two groups of accounts in a 2D representation returned by applying PCA, while in Figure 5c we report the centroids of the two clusters to highlight the difference between the two prototypical behaviors. In particular, we note that the average behavior characterizing the dynamics of the interactions in ego-networks of accounts belonging to the cluster 0 is dominated by the rule 0 - the creation of a single link at time $t + 1$ when the precondition is empty

- which on average accounts for the 20% of the rules involved in the dynamics of ego-network. The remaining rules also characterize the cluster 1, but on average, they are spread more uniformly than in cluster 0. In short, the *dynamics of transactions in the ego-networks of the accounts in the first clusters are mainly driven by star- and chain-like expansion rules* that appear in a successive timestamp without a precondition, where the appearance of a single link is dominant; on the contrary, *the dynamics in the second cluster are more homogeneous even if they lead to the same kind of expansion rules*. The most important graph evolution rules are the same, while it is the rule frequency that differentiates the two dynamic traits.

Given these two traits and the networked nature of our dataset, we finally wonder if accounts that usually interact by exchanging transactions are characterized by the same rules describing the dynamics of their ego-networks. To cope with this question, we first proceed by computing a static projection – graph flattening – of the graph sequence describing the Sarafu temporal network, then we compute the assortativity of the network using the clusters returned by the clustering algorithm as a categorical attribute. In detail, in the static projection, two accounts are connected if they interact at least once during the observation period, and the directed links are weighted according to the number of transactions sent by the source node toward the target node. Moreover, the construction of this graph is limited to the 3027 accounts in the analysis since the cluster attribute is missing for the remaining accounts. In this setting, the attribute assortativity is 0.590 and indicates a *tendency for accounts to interact with other accounts that have a similar ego-network dynamic*. In general, we stress the fact that by utilizing node evolutionary profiles, it is possible to develop applications that highlight properties and relationships between accounts based on the dynamics of the extracted ego-networks.

6. Conclusions

In this study, we introduced a method for representing ego-network dynamics through subgraph-based evolution rules, enabling a nuanced analysis of temporal behaviors within networks. Applying Node Evolutionary Profiles (NEPs) to the Sarafu transaction network revealed significant insights, identifying two main interaction traits: one dominated by the single-link expansion over other star- and chain-like expansions, and another with a more homogeneous distribution among the same expansion rules.

These findings underscore the potential of NEP-based representations to reveal underlying behavioral patterns in complex networks. Analyzing ego-network dynamics with graph evolution rules supports various applications, such as distinguishing user behaviors in financial transaction networks like Sarafu, crucial for operational improvements and strategic decision-making. Beyond identifying behavioral traits, the NEP-based approach enhances the understanding of interaction dynamics, aiding in the development of applications that highlight the properties and relationships between accounts.

Acknowledgments

MZ and SG are partially supported by PRIN 2022 Project "AWESOME: Analysis framework for Web3 Social Media" (2022MAWEZA - H53D23003550006, G53D23002900006). MZ, SG and CQ are partially supported by the project SERICS (PE00000014), under the MUR National Recovery and Resilience Plan funded by the EU - NextGenerationEU.

Declaration on Generative AI

During the preparation of this work, the author(s) used Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

- [1] A. Galdeman, C. T. Ba, M. Zignani, S. Gaito, Exploring time-ordered triadic closure in online social networks, *ACM Trans. Web* 19 (2025). URL: <https://doi.org/10.1145/3732295>. doi:10.1145/3732295.
- [2] Y. Hulovatyy, H. Chen, T. Milenković, Exploring the structure and function of temporal networks with dynamic graphlets, *Bioinformatics* 31 (2015) i171–i180.
- [3] A. Paranjape, A. R. Benson, J. Leskovec, Motifs in temporal networks, in: *Proceedings of the tenth ACM international conference on web search and data mining*, 2017, pp. 601–610.
- [4] C. T. Ba, A. Galdeman, M. Dileo, C. Quadri, M. Zignani, S. Gaito, Web3 social platforms: Modeling, mining and evolution, in: *Proc. of the First Italian Conference on Big Data and Data Science 2022 - ITADATA 2022*, volume 3340, CEUR-WS, 2022, pp. 168–179.
- [5] M. Berlingerio, F. Bonchi, B. Bringmann, A. Gionis, Mining graph evolution rules, in: *Joint European conference on Machine Learning and Knowledge discovery in databases*, Springer, 2009, pp. 115–130.
- [6] C. Leung, E.-P. Lim, D. Lo, J. Weng, Mining interesting link formation rules in social networks, 2010, pp. 209–218. doi:10.1145/1871437.1871468.
- [7] M. Yuuki, T. Ozaki, O. Takenao, Mining interesting patterns and rules in a time-evolving graph, *Lecture Notes in Engineering and Computer Science* 2188 (2011).
- [8] B. Bringmann, S. Nijssen, What is frequent in a single graph?, in: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Springer, 2008, pp. 858–863.
- [9] X. Yan, J. Han, gspan: Graph-based substructure pattern mining, in: *2002 IEEE International Conference on Data Mining*, 2002. *Proceedings.*, IEEE, 2002, pp. 721–724.
- [10] K. Vaculík, A versatile algorithm for predictive graph rule mining., in: *ITAT*, 2015, pp. 51–58.
- [11] E. Scharwächter, E. Müller, J. Donges, M. Hassani, T. Seidl, Detecting change processes in dynamic networks by frequent graph evolution rule mining, in: *2016 IEEE 16th International Conference on Data Mining (ICDM)*, IEEE, 2016, pp. 1191–1196.
- [12] P. Fournier-Viger, G. He, J. C.-W. Lin, H. M. Gomes, Mining attribute evolution rules in dynamic attributed graphs, in: *International Conference on Big Data Analytics and Knowledge Discovery*, Springer, 2020, pp. 167–182.
- [13] K.-N. T. Nguyen, L. Cerf, M. Plantevit, J.-F. Boulicaut, Discovering inter-dimensional rules in dynamic graphs, in: *Proceedings of the 1st International Conference on Dynamic Networks and Knowledge Discovery-Volume 655*, 2010, pp. 5–16.
- [14] M. Jin, Y.-F. Li, S. Pan, Neural temporal walks: Motif-aware representation learning on continuous-time dynamic graphs, *Advances in Neural Information Processing Systems* 35 (2022) 19874–19886.
- [15] R. A. Rossi, N. K. Ahmed, E. Koh, S. Kim, A. Rao, Y. Abbasi-Yadkori, A structural graph representation learning framework, in: *Proceedings of the 13th international conference on web search and data mining*, 2020, pp. 483–491.
- [16] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, U. Alon, Network motifs: simple building blocks of complex networks, *Science* 298 (2002) 824–827.
- [17] L. Kovanen, M. Karsai, K. Kaski, J. Kertész, J. Saramäki, Temporal motifs in time-dependent networks, *Journal of Statistical Mechanics: Theory and Experiment* 2011 (2011) P11005.
- [18] A. Longa, G. Cencetti, B. Lepri, A. Passerini, An efficient procedure for mining egocentric temporal motifs, *Data Mining and Knowledge Discovery* 36 (2022) 355–378.
- [19] B. Arregui-García, A. Longa, Q. F. Lotito, S. Meloni, G. Cencetti, Patterns in temporal networks with higher-order egocentric structures, *Entropy* 26 (2024) 256.
- [20] C. E. Mattsson, T. Criscione, W. O. Ruddick, Sarafu community inclusion currency 2020–2021, *Scientific data* 9 (2022) 426.
- [21] L. Ussher, L. Ebert, G. M. Gómez, W. O. Ruddick, Complementary currencies for humanitarian aid, *Journal of Risk and Financial Management* 14 (2021) 557.
- [22] R. Mqamelo, Community currencies as crisis response: Results from a randomized control trial in kenya, *Frontiers in Blockchain* (2021) 44.
- [23] C. E. Mattsson, T. Criscione, F. W. Takes, Circulation of a digital community currency, *arXiv*

preprint arXiv:2207.08941 (2022).

- [24] C. T. Ba, A. Galdeman, M. Zignani, S. Gaito, Temporal analysis of cooperative behaviour in a blockchain for humanitarian aid during the covid-19 pandemic, in: Proceedings of the 2022 ACM Conference on Information Technology for Social Good, 2022, pp. 292–299.
- [25] C. T. Ba, M. Zignani, S. Gaito, Cooperative behavior in blockchain-based complementary currency networks through time: The sarafu case study, Future Generation Computer Systems 148 (2023) 266–279.
- [26] X.-X. Zhan, C. Liu, Z. Wang, H. Wang, P. Holme, Z.-K. Zhang, Measuring and utilizing temporal network dissimilarity, arXiv preprint arXiv:2111.01334 (2021).
- [27] P. Holme, J. Saramäki, Temporal network theory, volume 2, Springer, New York City, NY, 2019.
- [28] A. Galdeman, M. Zignani, S. Gaito, Unfolding temporal networks through statistically significant graph evolution rules, in: 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2023, pp. 1–10.
- [29] T. Milenković, N. Pržulj, Uncovering biological network function via graphlet degree signatures, Cancer informatics 6 (2008) CIN–S680.
- [30] W. O. Ruddick, Sarafu Community Inclusion Currency, 2020-2021 uk data service reshare, 2021. URL: <https://reshare.ukdataservice.ac.uk/855142/>. doi:10.5255/UKDA-SN-855142.
- [31] M. Karsai, H.-H. Jo, K. Kaski, et al., Bursty human dynamics, Springer, New York, NY, 2018.