Discover Data

Perspective

A powerful lens for temporal network analysis: temporal motifs

Ahmet Erdem Sarıyüce[1]

Received: 3 November 2024 / Accepted: 10 April 2025
Published online: 28 April 2025

Abstract
Temporal networks serve as a potent representation framework facilitating the comprehension and characterization of diverse complex systems. Whether examining face-to-face human contacts, financial transactions, or computer communications, these interactions can be conceptualized as temporal networks, wherein activities occur intermittently at specific time points. While significant progress has been made in static network analysis and the handling of temporal/sequential data, the current state-of-the-art techniques for temporal network analysis are not sufficiently mature to effectively address the challenges posed by real-world continuous-time temporal networks, such as their diverse nature, unclear scale, and massive size. In this perspective article, we argue that temporal motifs are a powerful lens and promise potential to be a standard method for temporal network mining. We first summarize various approaches for modeling temporal motifs and discuss the inherent challenges due to the diverse nature of temporal networks. We then explore a multitude of temporal network mining applications that have effectively employed temporal motifs, spanning from financial network analysis to temporal graph generation. This exploration aims to offer practical insights to practitioners on how to effectively model and leverage temporal motifs. Lastly, we present a comprehensive overview of existing challenges that hinder the broader adoption of temporal motifs and highlight several potential future directions for research and development in this domain.

1 Introduction

Unrelenting dynamism and increasingly interconnected nature of data made temporal networks a big part of data science pipeline. In temporal networks edges are active only at certain points in time. For example, interactions are transient in face-to-face contacts, financial transactions, and computer communications—understanding these networks is critically important for preventing pandemics, detecting fraud, and ensuring cyber-secure environments, respectively.

Despite the big advances in static network analysis as well as temporal/sequential data, the state-of-the-art techniques for temporal network analysis are not mature enough to deal with the challenges in real-world temporal networks. One big reason for this is the mismatch between data and model.

1.1 Background and challenges

To better understand the data-model mismatch, one can think about how the temporal network is actually formed. In biological networks, for example, the data is monitored, or pulled, at regular intervals for a time period and the interactions are recorded at a certain resolution. This creates a series of static networks occurring at a given resolution. Note that data representation in this case is restricted by the collection methodology, i.e., interactions within finer

✉ Ahmet Erdem Sarıyüce, erdem@buffalo.edu | [1]University at Buffalo, Buffalo, USA.

resolution are not distinguished. On the other hand, event-based systems permit obtaining a true data representation by pushing the stream of interactions whenever they happen. The exact ordering among the events can be recorded, e.g., call-detail records. Note that one can always convert a stream of interactions to a series of snapshots by decreasing the resolution. We categorize temporal networks into two types, depending on how the data is represented:

- Discrete-time temporal networks: (also known as snapshots, dynamic graphs). The temporal network is a time-series of static networks in which each snapshot has a timestamp and all the events in a snapshot inherent the timestamp of the snapshot. Examples are daily snapshots of face-to-face contacts and coauthorship relations with publication year information. Discrete-time temporal networks are often an artifact of the data collection process, as it is challenging to collect the data, and have a limited view of the temporal dimension. Most biological networks, such as brain scans, protein-protein interaction, animal movements, and ecological networks are discrete-time temporal networks [1].
- Continuous-time temporal networks: (event streams). The network consists of events where each event has an independent timestamp. Examples include interaction networks, communication networks such as messages and call detail records, and financial transactions. Continuous-time temporal networks are obtained in the finest possible resolution and are often described as streams.

Discrete-time temporal networks, by definition, contain less temporal information than the continuous-time temporal networks, but are often considered easier to analyze. One common practice is to treat each snapshot as a static network and then combine the findings from all the snapshots to create a time-series. In general, network is more temporal in continuous-time temporal networks than in discrete-time temporal networks and hence each type requires different types of methods for analysis.

Unfortunately, the literature is abundant with studies where a continuous-time temporal network is converted to a discrete-time temporal network by an arbitrary process in which a custom resolution is chosen to create snapshots with regular intervals [2–11]. The stream of interactions is aggregated over specific periods to create snapshots, and the ordered set of snapshots is considered as an evolving (or dynamic) graph. Even the discrete-time temporal networks are sometimes further aggregated to create coarser networks [12]. Although this aggregation can be advantageous in some scenarios for reducing temporal sparsity and noisy data, it decouples temporality from the network structure and hence hinders an effective analysis. The snapshot-based approach requires finding the proper resolution that will capture the inherent rhythm of the network, which is a substantial challenge given the *diverse nature* of various temporal networks [13, 14]. This is a major issue especially when the network is a *massive* stream of events with a fine resolution, as in communication and financial transaction networks, where each interaction matters and the network is temporally heterogeneous (i.e., inter-event times are varying). Selecting any resolution greater than the finest possible would oversimplify the temporal nature of the network since the fine-grain temporal information is lost in each snapshot. Choosing the finest possible resolution would yield too many and highly sparse snapshots for which the static network analysis is ineffective/inefficient.

In general, there are three main challenges in handling continuous-time temporal networks that the snapshot-based approach cannot address:

- Challenge 1: Diverse nature. Temporal networks vary greatly in various aspects such as how the events are ordered or whether the events are instantaneous or not [15]. For example, the events in a financial transaction network have a total ordering , assuming that they are recorded in the finest possible resolution, but the same cannot be said for an email network in which one-to-many exchanges create partially ordered interactions. Each temporal network has unique characteristics, and any network analysis method should account for this [16–18]. The snapshot-based approach does not account for such key differences, e.g., setting up the snapshots with any resolution greater than the finest possible would oversimplify the temporal nature of the network as the fine-grain temporal information is lost in each snapshot. Customizable solutions that incorporate various characteristics of temporality are essential to address the diversity of temporal networks.
- Challenge 2: Unclear notion of scale. In temporal networks, scale is not only characterized by the number of nodes/events but also by the timespan. It is often not clear at what scale the analysis should be performed. For instance, key characteristics, such as burstiness and periodicity, need to be incorporated to discover mesoscale structures, when needed [19–24]. The snapshot-based approach only considers the global scale by imposing an arbitrary resolution and creating a series of static networks [13, 14]. There is a need for a modular approach to enable microscale, mesoscale, and macroscale analysis in temporal networks.

- Challenge 3: Massive size and timespan. Temporality brings an additional difficulty to the already challenging task of large scale graph processing. It is particularly hard to analyze temporal networks that are a massive stream of events with a wide timespan and fine resolution where each interaction matters, as in communication and financial transaction networks. The snapshot-based approach is ineffective for such data: Choosing a small resolution would yield too many and highly sparse snapshots for which the static network analysis is ineffective and inefficient. It is essential to design scalable techniques that would enable the consideration of the topological and temporal structure of the network in tandem to work on the finest available resolution with no aggregation.

Overall, there is a need for a new way of thinking in continuous-time temporal network analysis that would enable the consideration of the topological and temporal structure of the network in tandem to work on the finest available resolution with no aggregation.

1.2 New perspective

We believe that these challenges can be neatly addressed through the use of temporal network motifs [15, 25–28]. Temporal motifs are defined as a set of nodes which interact with each other in a short time period. Network motifs in static networks are defined as recurrent and statistically significant subgraphs [29–34]. Interestingly, motifs are also crucial (and indeed first proposed) in sequence mining for computational biology [35] and are studied in time-series analysis [36, 37]. Static networks, time-series, and sequences are complex, interdependent structures with no clear notion of scale, and motifs provide a principled way to summarize and interpret these data types [38–42]. Utilizing well-accepted methods from time-series and sequence mining to handle temporal networks is promising but is considered beyond trivial due to the complex inter-connected structure of networks, and thus has been met with little success [43, 44].

Temporal motifs can address the *diverse nature* of temporal networks by encoding the key characteristics at the low level. They can provide a bottom-up scheme to build larger mesoscale structures, and thus enable the analysis in *multiple scales*. They are also suitable to design *scalable algorithms* by means of sampling (for approximate computations) and/or parallel computations. In static networks, motifs are a versatile tool in downstream network analytical tasks such as anomaly detection, community detection/search, and link prediction. Static motif counts have been used to find anomalous nodes in financial transactions [47] and to discover communities [48–50]. Likewise, graph-embedding methods that use static motif counts have been shown to be effective for link prediction [51, 52]. Temporal motifs have more to reveal in all of these tasks, as already demonstrated to some extent [53–55]. For instance, a sequence of three cyclic financial transactions in a short period ($A \rightarrow B \rightarrow C \rightarrow A$) is unexpected — two transactions would be sufficient to balance all the accounts. Even the counts of such specific motifs can offer some insight about the nature of transactions—in a Venmo dataset of person-to-person payments, it is observed that cyclic triangles are very rare and often happen when some friends play poker and send/receive funds after each round [56]. Another advantage of temporal motifs is that they provide an interpretable lens to study various unsupervised applications. Fig. 1 gives a few examples. For instance, in money laundering, bitcoins are transferred through several intermediate addresses by certain patterns [45] (Fig. 1b). A large amount is first split into two uneven transfers and then the smaller amount is further split into transactions of similar amount. Topology, ordering, and timing of these transactions can successfully be expressed as a temporal motif.

Note that use of temporal motifs is not limited to continuous-time temporal networks: there are several studies that successfully utilized temporal motifs for dynamic graphs that appear as a series of snapshots [12, 55, 57–66].

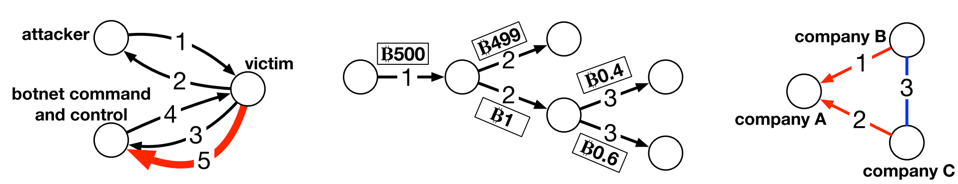


Fig. 1 Examples of temporal motifs. Numbers denote the order of interactions

(a) *Information exfiltration: An attacker first establishes a channel to a victim, then messages are exchanged from botnet command and control to obtain a large amount of data (in red) [44]. Thickness denotes size of interactions.*

(b) *Money-laundering: A large amount of bitcoin is forwarded to two nodes; one transfer is large and the other is very small. The latter is then evenly splited to two other nodes. These motifs are widely used as a peel-chain structure in money-laundering [45].*

(c) *Patent oppositions and collaborations: One indicator of a collaboration (blue) between two companies is the earlier patent oppositions against a common rival [46].*

In this perspective article, we argue that temporal motif-based approaches should be the de-facto method for temporal network mining for continuous-time temporal networks. In the rest, we first give a quick background on fundamental definitions on temporal networks and temporal motifs (Sect. 2). Then we summarize different approaches on how to model temporal motifs and discuss why it is a challenging problem (Sect. 3). Next we discuss numerous temporal network mining applications that successfully used temporal motifs, ranging from financial network analysis to temporal graph generation, to provide guidance to practitioners on how to model and utilize temporal motifs (Sect. 4). Then we give a comprehensive overview of existing challenges for broadening the use of temporal motifs and pinpoint several future directions (Sect. 5). At the end, we conclude our discussion (Sect. 6).

2 Preliminaries

A continuous-time temporal network is typically represented as $G(V, E)$, where V denotes the set of nodes, and E represents the set of events. Each event, denoted as $e_i \in E$, is described as a 4-tuple $(u_i, v_i, t_i, \Delta t_i)$ [18]. Here, u_i and v_i identify the source and target nodes, respectively, forming the node pair where the i-th event takes place. The parameters t_i and Δt_i correspond to the starting time and duration of the i-th event. It is noteworthy that events in continuous-time temporal networks are often directed, reflecting an initiator and a receiver in interactions. The set E constitutes a time-ordered list of m events, with starting times arranged as $t_1 \leq t_2 \leq t_3 \leq \cdots \leq t_m$, and V is the set of nodes appearing in E. In real-world continuous-time temporal networks, it is common for the inter-event time, $t_{i+1} - t_i$, to be significantly larger than Δt_i, especially when the interaction is always an engagement, such as phone calls. Consequently, disregarding event durations is a prevalent practice for the sake of simplicity. It is also important to distinguish between edges and events: an edge (u, v) represents the static projection of an event (u, v, t) and there can be many events on the same edge. A discrete-time temporal network can be defined in the same way as a continuous-time temporal network—all the events in the same snapshot are assigned an identical timestamp. As mentioned before, the opposite is considered harmful as casting an event stream to snapshots either loses information or creates arbitrary snapshots that are too sparse [15].

At minimal, a temporal motif is defined as a directed temporal subgraph with k nodes, l events, and with a total ordering among l events and a timing constraint that limits the time difference between events. An example is given in Fig. 2a to show the instances of a temporal motif in a toy graph. Note that it is typical to consider the entire spectrum of (at most) k nodes, l events motifs with all possible total ordering combinations. Fig. 2 shows the spectrum of 3 event motifs with at most 3 nodes. Additionally, one can define further constraints in temporal motifs such as inducedness, partial orderings, event durations, or node/edge labels [15]. We give more details on different modeling choices in the next section.

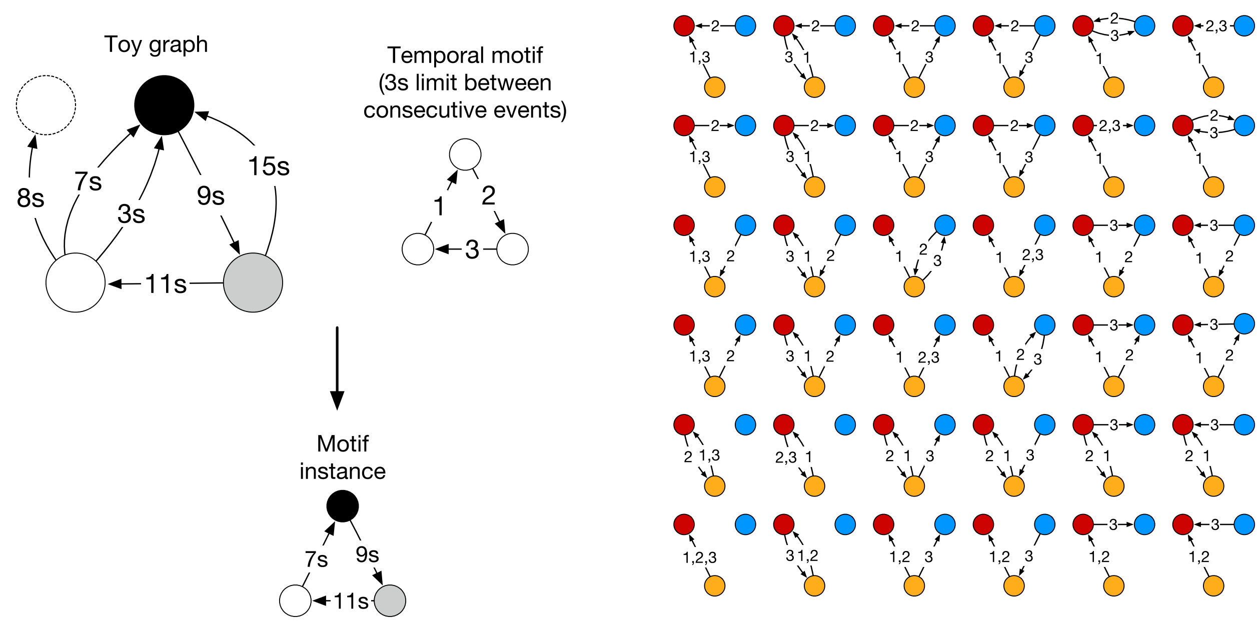
3 Modeling temporal motifs

Temporal motifs have been conceptualized through various models, and historically, four prominent studies providing diverse perspectives are highlighted [15]:

- Kovanen et al. [26] presented the initial model incorporating the concept of temporal adjacency to establish relationships among events within a motif.
- Song et al. [28] introduced another model designed for streaming workloads, where motifs are identified on-the-fly, and events within a motif can be partially ordered.
- Hulovatyy et al. [25] explored new relaxations and constraints to enhance Kovanen et al.'s model, and also considered events with durations.
- Paranjape et al. [27] proposed a practical model with a specified time window to constrain all events within a motif.

Here we explain each paper's main idea and draw some comparisons.

The inaugural temporal network motif model, as introduced by Kovanen et al. [26], aims to create more expressive motifs than classical network motifs in static networks [33], by utilizing edge timestamps. The temporal motif is defined as an ordered set of events, characterized by two key features: (1) the time difference between consecutive events (in the entire set) is less than the threshold Δ_C, an input parameter, and (2)for each node u in the motif, given its adjacent event in the motif with the earliest timestamp, e_e, and the latest timestamp, e_l; all the adjacent events of u in the network with



(a) *A toy graph and a temporal motif—a cycle of 3 events with timing limit of 3s. The only motif instance for the given temporal motif in the toy graph is shown at the bottom.*

(b) *Spectrum of 3 event motifs with at most 3 nodes. 36 motifs in total. 4 motifs have only 2 nodes, shown at the bottom-left, and the rest have 3 nodes.*

Fig. 2 Examples for temporal motifs

a timestamp between e_e and e_l must be in the motif (i.e., no adjacent event of u in the network that happens in between can be left over). This design is intended to consider causality among the events.

Song et al. [28] introduced the event pattern matching problem for real-time graph streams, enhancing traditional complex event processing by incorporating graph structure. Approaching the issue from a time series and stream processing perspective, the authors treated the graph structure as a novel feature. The event pattern is essentially a temporal motif model, accounting for node/edge labels, partial orderings among events, and an input parameter, Δ_W, representing an upper bound for the time difference between the first and last events in the motif.

Hulovatyy et al. [25] proposed another temporal motif model based on graphlets (induced motifs) in static networks [67]. Enhancing Kovanen et al.'s model, the authors considered induced subgraphs, encompassing all interactions among a specific set of nodes, and relaxed the constraint that adjacent events of a node should be consecutive. Additionally, Hulovatyy et al. introduced the use of events with durations in temporal network motifs for the first time. Introducing a further restriction (constrained dynamic graphlets), they aimed to reduce computational complexity while obtaining approximate results. Their model captured diverse temporal motifs from each node's perspective and demonstrated more effective results than previous techniques in predicting aging-related genes in humans.

In the model proposed by Paranjape et al. [27], a relaxation of the first model by Kovanen et al. [26] is considered. The constraint requiring adjacent events of a node to be consecutive, as in [26], is relaxed to capture motifs occurring in short bursts. A time window, Δ_W, is introduced to bound the time difference between the last and first events in a motif. Formally, a k-node, l-event, Δ_W-temporal motif is defined as a sequence of l events such that the time difference between the last (l-th) event and the first event is no greater than Δ_W and the induced static graph from the l events is connected and has k nodes. Paranjape et al.'s model is relatively simpler than the other definitions and has been adopted in many applications in which the temporal networks are used. It is important to note that a subgraph may be considered a valid motif in some models but not in others, owing to specific constraints imposed by different models.

There are several aspects of temporal networks and motifs that differ across the four models, and Table 1 (adapted from [15]) provides a concise comparison. Tailoring temporal motifs to address the diversity in temporal networks involves making distinct choices in these aspects. These choices are vital for adapting temporal motifs to the varied application space of temporal networks. For example, in financial transaction networks where fraudsters disguise their identities through repetitive legal transactions, a strictly induced temporal motif may prove inadequate. This is because it considers

Table 1 Aspects of temporal motif models by Kovanen et al. [26], Song et al. [28], Hulovatyy et al. [25], and Paranjape et al. [27] (obtained from [15])

Article	[26]	[28]	[25]	[27]
Induced subgraph	Node-based temporal	✘	Static only	Static only
Event durations	✘	✘	✓	✘
Partial ordering	✓	✓	✘	✘
Directed edges	✓	✓	✘	✓
Node/Edge labels	✘	✓	✘	✘
Adjacent events in Δ_C	✓	✘	✓	✘
Entire motif in Δ_W	✘	✓	✘	✓

all transactions among a set of entities, potentially overlooking fraudulent transactions. In contrast, in a communication network, it might be more meaningful to use induced motifs to comprehensively understand information dissemination.

It is important to note that these aspects are not exhaustive, and there are further ways to parameterize temporal motifs categorically. For instance, Lehmann proposed considering fundamental structures to study temporal networks. These structures represent temporal-topological network patterns corresponding to single real-world events [68]. Such structures can include a single event (one-to-one), a set of events from the same source (one-to-many, e.g., when a person sends an email to multiple recipients), or a set of events among multiple nodes (many-to-many, e.g., a clique formed among authors of a new paper in a temporal coauthorship network). Defining a temporal motif as an ordered list of these fundamental structures, rather than events, can better capture the semantics of the network. An extended discussion of the aspects in Table 1 along with a comparative experimental evaluation of different choices for those aspects is given by Liu et al. [15].

4 Applications

Temporal motifs have been defined in various ways to accommodate the needs of an array of applications that appear in real-world continuous-time and discrete-time temporal networks. Here we survey the application scenarios that make use of temporal motifs which may offer guidance to practitioners on how to define temporal motifs.

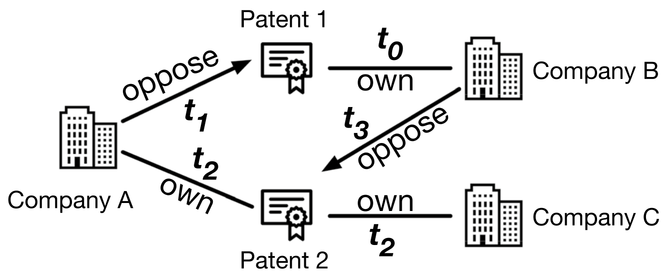
4.1 Communication networks

Communication activities among humans, such as text messages and call detail records, have a temporal nature and are heavily analyzed to understand information propagation as well as causality in human behavior. With the increasingly available online data, communication networks often appear as continuous-time temporal networks. One of the earliest works that consider motifs in temporal networks is by Zhao et al. [69]. In this work, the authors defined the communication motifs to characterize information propagation in synchronous (call detail records) and asynchronous networks (wall-post history of Facebook interactions). For a given temporal network, the communication graph is defined as the subset of events in the temporal network such that each event e_1 in the subset has an adjacent event e_2 such that e_1 and e_2 share a common node (topological proximity) and have occurred close in time (temporal proximity). The communication motifs are then defined as the equivalence classes in the communication graph. As the cardinality of those classes is often too large, the authors used a support threshold to consider most common patterns as the communication motifs. In addition, the durations of the events are used to quantify the flow of information where an event (e.g., call) that happened closer and took more time (duration) is considered to carry more information. The main findings of the study suggest that motif distributions are stable over time and synchronous and asynchronous networks have different motif distributions.

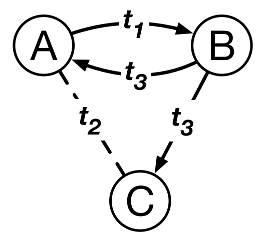
In another study, Kovanen et al. analyzed a mobile phone call network using temporal motifs [70]. The authors relied on their motif model [26], explained in Sect. 3, which uses Δ_C timing threshold. Nodes are marked with the individual attributes of the nodes—sex, age, and payment type (prepaid or postpaid)—and only 2-event motifs are used for analysis. One interesting result in this work is the observation of temporal homophily: Motifs where the participants share all three attributes are significantly more common when compared to randomized networks.

Some studies examined the evolution characteristics of network motifs. Li et al. analyzed the mobile phone call networks from two companies operating in Europe and China [71]. The authors considered 3-node motifs only. Results

Fig. 3 An example for patent oppositions and collaborations (obtained from [46])

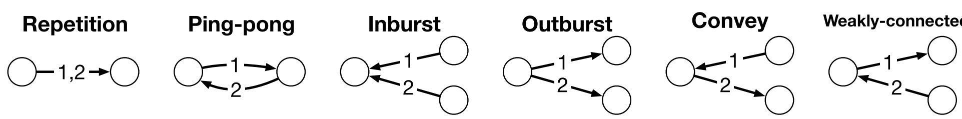


(a) *Company A opposes a patent owned by company B at t_1, and then companies A and C co-own a patent at t_2, which is opposed by company B at t_3.*



(b) *The corresponding patent network. The directed solid lines represent patent oppositions. The undirected dashed line represents the collaboration.*

Fig. 4 All 2-event temporal network motifs, along with the custom names



show that triadic closures form frequently at an intraday time scale and open motifs typically evolve to other open 3-motifs with a higher number of reciprocated calls. In another work, Sarkar et al. considered the transition properties from 4-node motifs to 5-node motifs [72]. One interesting finding authors mention is that during the information cascade lifecycles, the individuals have a tendency to reshare from high degree nodes instead of interacting in a way that would form large cycles.

4.2 Trading and business networks

There are many uses of temporal motifs in trading and business applications where movements of goods and various relationships among the companies are studied. One early work by Bajardi et al. studied the cattle movements in Italy [12]. The data is modeled as a directed discrete-time temporal network as the collected data is a series of daily snapshots. The authors defined causal motifs as a temporal sequence of links such that the destination node of a link at time t_0 is the origin of another link at time $t_0 + \Delta t$. Longer sequences are observed less and a comparison against various null models suggest that the observed frequencies are significant, hence the dataset has an intrinsic time arrow. In another study, Zhang et al. studied two bipartite shipping networks, the world-wide commercial ship chartering (between charters and owners) and build-to-order ships (between shipyards and owners) [57]. The data is a series of monthly snapshots, modeled as an undirected discrete-time bipartite temporal network. The authors studied seven motif evolution patterns such that a motif between 2 x 2 nodes transitions to a 2 x 2 biclique. Comparison against randomly shuffled networks shows that a 3-edge motif is the most significant pattern in both networks.

More recently, Liu et al. analyzed patent opposition and collaboration networks using temporal motifs to understand how corporate innovation relates to rivalry [46]. The authors modeled company relations in a two-layer temporal network—one layer for oppositions (a company opposing another's patent) and one layer for collaborations (companies co-owning a patent). An illustration is provided in Fig. 3. A temporal motif is defined as a subgraph where consecutive event pairs are temporally close (controlled by Δ_C in Sect. 3) and the time between first and last events is limited (controlled by Δ_W in Sect. 3). Authors examined 2-event and 3-event (3-node) motifs between opposition events (Fig. 4 presents the former ones) and considered the collaborations on top of those motifs. The key findings are (1) the oppositions tend to be more repetitive than reciprocal among companies; (2) different timing thresholds (Δ_C and Δ_W) produce consistent motif type rankings; and (3) opposition patterns align with the structural balance theory—i.e., triangles of negative relations are rare. The authors also performed statistical significance evaluation of the opposition motif frequencies using different null models and showed that motifs with repetitions and in-bursts are abundant, while ping-pong and triangle motifs are rare. Regarding company sizes, the opposer companies found to be larger than the opposed companies, and large companies often seem to be opposing multiple others. For the interplay between collaborations and oppositions, (1) collaborations seemed to occur between companies with no opposition; (2) for two collaborators with a common adversary, collaborations tend to predate joint oppositions; and (3) adversaries are more likely to later collaborate than collaborators are to later oppose.

4.3 Financial transaction networks

Financial transaction networks are one big class of temporal networks that are highly dynamic, and hence continuous-time temporal, and also notoriously private (in various ways) due to regulations. Fund transfers among bank accounts or cryptocurrency transfers between addresses can be modeled as a temporal network where each event corresponds to a transfer and also has a numeric label that denotes the monetary amount. Two main applications on financial transaction networks are fraud detection and anti-money laundering, which are highly crucial for the financial institutions and even have implications for national security and economic competitiveness of the countries.

It is challenging to find publicly available datasets as the financial institutions are not able to share the transactions, even in an anonymized way, due to regulations. One interesting remedy for this is the (by definition) publicly available cryptocurrency transfers. All the transactions in cryptocurrency transaction networks are public, in the ledger, but the addresses are pseudo-anonymous. Regardless, there are several studies on figuring out which addresses belong to the same user especially thanks to the use of exchanges [73–75]. This problem is called address clustering.

A study by Kosyfaki et al. proposed flow motifs to study temporal networks where each event has a numerical value that denotes something that is transferred (bitcoins, passengers) or the strength of the event (Facebook interactions) [76]. Flow motif is defined as a subgraph of events such that (1) for any event pair where the target of the first is the source of the second, the former happens before the latter; (2) duration of the entire motif is parameterized (Δ_W in Sect. 3); and (3) the flow of each event has a lower bound. The authors also define maximal flow motifs to include the repeated events that satisfy all the three constraints. Evaluations suggest that the flow motifs that contain cycles, indicating large flow movements that close a cycle, are statistically over-represented in the bitcoin and passenger-flow networks whereas chain motifs are more statistically significant in a Facebook interaction network.

With the increasing popularity of cryptocurrencies, there have been recent studies that used temporal motifs for analysis. One study, by Wu et al., aimed to detect mixing services in Bitcoin, which are originally designed to enhance transaction anonymity, but have been widely employed for money laundering to complicate the process of trailing illicit fund [77]. An example of a mixing service is given in Fig. 5. The authors introduced a temporal motif-based network analysis framework to identify statistical properties of mixing services in network, account, and transaction levels. They used the motif model by Paranjabe et al. [27], only considered the 2-event motifs, and added binary attributes to the nodes to encode the relative amount and timestamp information from the adjacent events. Experimental results on three real Bitcoin datasets showed the effectiveness of the detection model based on temporal-motif features. Another related study is performed by Wang et al. to detect Ethereum phishing scams by using temporal motif features [78]. Blockchain-based phishing scams trick users into transferring money to certain accounts by pretending to be corporate entities with credit. The authors employ graph classification task to detect scammers. For each Ethereum address, they sample a subgraph that preserves the direct neighbors, and more if

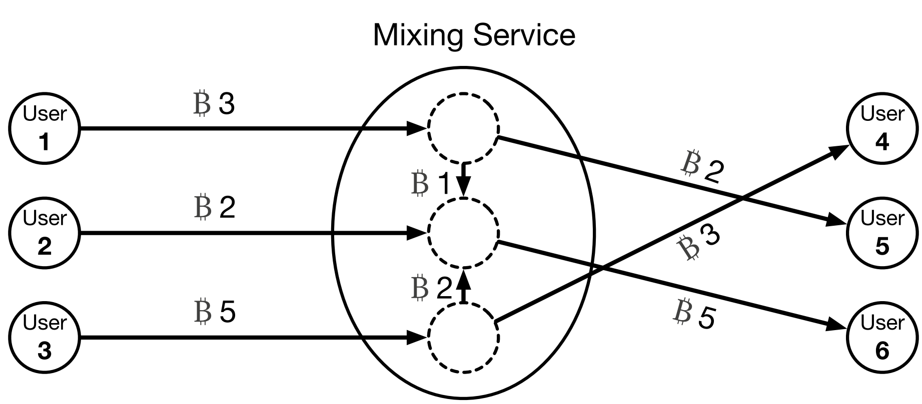


Fig. 5 Examples for bitcoin transactions via mixing service. User 1 sends 3 bitcoins to User 4 by using a mixing service that enhances transactions anonymity. Mixing service receives 3 bitcoins from User 1 and sends 3 bitcoins to User 4, however, the source of the bitcoins is changed. This is done thanks to many transfers to the mixing service and a series of internal transactions within the service. Likewise, User 2 sends 2 bitcoins to User 5, and User 3 sends 5 bitcoins to User 6

there is an exchange involved. Then 3-event temporal motifs, as defined by Paranjabe et al. [27], are used to extract features from each subgraph, some other general graph features are appended, and all are fed to the classifiers such as SVM and XGBoost. Evaluation suggests that the proposed method outperforms established embeddings such as node2vec and graph2vec in detecting scammers. In a similar direction, Liu et al. used the 2-event temporal motifs to analyze phishing scams in Ethereum network and utilized the convey motifs (Fig. 4) to design a model for detecting Ethereum phishing gangs [79].

More recently, Liu et al. studied three real-world financial transaction networks by using temporal motifs: (1) transactions in Mercari, a consumer-to-consumer online marketplace, (2) synthetic transaction network generated by J.P. Morgan Chase (JPMC), and (3) payment and friendship relations among Venmo users [56]. The authors studied fraud detection problem on Mercari and JPMC networks where the ground truth is available, and examined the interplay between financial and social relations among the Venmo users on friendship prediction, vendor identification, and analysis of temporal cycles. They chose to use temporal motifs with inter-event timing constraints (Δ_C in Sect. 3). All the 2- and 3-event motif types are computed in each egocentric network of a user which includes all the events between user and its neighbors as well as the events among its neighbors. To account for the behavior of the user, count of the specific motif type that contains the user is normalized with respect to the entire count (i.e., including those that do not contain the user). Then those features are fed to several types of classifiers. Empirical results on Mercari and JPMC networks showed that all the classifiers trained with temporal motif features consistently outperform the ones trained with simple graph features and with LINE and node2vec embeddings for both fraud and non-fraud classes. Moreover, the runtime performance of the proposed method is also shown to be practical when compared against the costly embedding methods. Regarding the interplay between payments and friendships in Venmo network, temporal motifs-based logistic regression outperforms classical measures, like Jaccard and Adamic-Adar similarity, for friendship prediction. In addition, the authors studied the problem of vendor identification in Venmo which is important for accurate tax verification purposes and currently relies on simple measures like number of transactions or total amount received. A subset of temporal motifs where a specific node is likely to be a vendor as well as another subset of motifs where no node is likely to be a vendor are used to create features to predict vendor users. Evaluation based on manually obtained vendorship information suggests that temporal motifs offer a decent solution for identifying vendors and business owners. Lastly, the authors point out that certain motifs such as three-event cycles are extremely rare in Venmo and are often marked with payment notes regarding gambling activities such as online poker where users perform transactions after each round of play.

4.4 Science of science

Collaborations among researchers have long been studied to understand the innovation dynamics in science and technology [80]. Coauthorship networks are often the main medium of study, which is obtained via publication data that has a year resolution, hence can be considered more of a discrete-time temporal network. Boekhout et al. used multilayer temporal motifs to investigate how scientific collaboration and scientific mobility are related and compare between different fields of science [62]. Authors defined four layers of collaborations, organizational (from the same institution), local (from the same city), national (from the same country), and international, and inferred mobility patterns through the configuration of some multilayer temporal motifs. The multilayer temporal motifs are defined as a sequence of possibly concurrent directed events forming an edge, a star, or a triangle and occurring within a Δ_W time threshold (which limits the time difference between the first and last events). The authors studied 3-node motifs only, 88 in total, without accounting for the possibilities with respect to layers. Experiments on five broad disciplines showed distinct characteristics of collaboration in each. For example, there seems to be strong trend to continue to collaborate within the organization in Mathematics & Computer science whereas collaborations extend outside the same organization in Social sciences & Humanities, more than in any other field. Another interesting work in a similar vein, by Xuan et al., studied software development networks where developers and files are the nodes and interactions occur between developers and files as well as among developers [81]. When compared to null models, temporal motifs are found to be significantly abundant. More central individuals and more cohesive teams, with respect to temporal motifs, are found to be more productive.

4.5 Biological networks

Wet-lab experiments and medical imaging techniques have yielded various discrete-time temporal networks, such as metabolic networks and brain networks, which are studied to better understand the gene expression dynamics and neural activities [82]. Temporal motifs are used as effective tools in some of those studies. One early work by Chechik et al. defined timing activity motifs to study transcriptional control mechanism of the yeast metabolic network [63]. The authors focused on chains and branching motifs where the timing of gene activations follow topological order or are concurrent. Results suggest that cells utilize different timing activity motifs to optimize transcription timing in response to changing conditions, e.g., forward activation motifs are observed for synthesizing metabolic compounds, backward shutoff motifs are abundant for stopping production of a detrimental product, and synchronized activation for co-producing the metabolites required for the same reaction. Another use of temporal motifs is proposed to study aging in protein-protein interaction networks [25]. Hulovatyy et al. defined constrained dynamic graphlets, as explained in Sect. 3, and showed that the prediction quality is superior than the static-motif based approaches. Also, the authors showed that dynamic graphlets can uncover the known aging-related knowledge more precisely.

Another related domain that is highly involved with temporal networks is epidemics. Human contact networks have been at the forefront of COVID research in the last few years [83]. Recently, Oettershagen et al. benefited from temporal motifs for classifying dissemination processes in discrete-time temporal networks [64]. Authors defined the temporal graphlet kernels as the inner product of normalized graph feature vectors counting the occurrences of labeled temporal graphlets, where the node labels are defined as infected or recovered. Experimental evaluation showed that their approach outperforms the state-of-art while running significantly faster on large data sets.

4.6 Graph neural networks for temporal networks

Graph neural networks (GNNs) are the de facto paradigm for the node classification and link prediction tasks in graphs where the nodes and edges have rich features. Temporal graphs, both discrete-time and continuous-time types, are also studied with graph neural networks [84]. Temporal motifs are shown to be helpful in the design of graph neural networks in various studies. The main advantage of using motifs in GNNs is that they help to capture higher-order relations in a structured way which helps with the message passing strategies of the GNNs.

To the best of our knowledge, the first work that utilizes temporal motifs for GNNs is by Wang et al. who proposed Causal Anonymous Walks (CAWs) to inductively represent a temporal network [54]. A CAW starts from a link of interest and go back several adjacent links over time to encode the underlying causality of network dynamics—each walk here gives a temporal network motif. To predict temporal links between two nodes, the model samples a few CAWs from those nodes, and encode and aggregate these CAWs via RNNs. This approach is far more efficient than enumerating and counting the temporal motifs. In addition, authors facilitate set-based anonymization which removes the node identities over the walks to enable inductive learning. Relative node identities are encoded with respect to the counts that they appear at a position in the motif, hence can be used for unseen nodes. Experiments on various continuous-time temporal networks show that CAWs obtain superior performance than the state-of-the-art for transductive and inductive temporal link prediction. In a following study, Jin et al. designed NeurTWs which modifies the sampling strategy of CAWs to give more importance to spatially close nodes [55]. In particular, the authors consider most-recent neighbors and neighbors with higher connectivity during the sampling and further make use of an exploitation and exploration trade-off to regularize the walk sampling based on the traversal times of nodes. Overall, NeurTWs outperforms CAWs on transductive/inductive temporal link prediction tasks and also gives promising results for node classification.

Another work that uses temporal motifs is introduced by Qiao et al. [60]. The authors designed Temporal Network Embedding with Motif Structural Features (MSTNE) model which creates node embeddings by sampling the neighbor nodes based on the temporal triads (motifs with three nodes) and modeling the effects of different temporal triads using the Hawkes process. MSTNE considers both the structural identity and temporal relationship of the triad to generate representative node embeddings. The effect of different temporal triads are taken into account through distinction of open and close triangles, and an attention mechanism. Experiments on discrete-time temporal networks suggest promising results for node classification and temporal link prediction.

4.7 Other use cases

Apart from those mentioned above, there are many other use cases from diverse domains that leveraged temporal motifs. Here we cover each to the best of our knowledge.

4.7.1 Visualization

Graphs are dimensionless and complex structures and hence visualizing them in an interpretable way has been the focus of many studies [85]. Temporality brings additional challenges for visualizing networks. Temporal motifs, which can encode both the structural and temporal characteristics of the network in a systematic manner, have been shown to be helpful in visualizing temporal networks. Recently, Jung et al. introduced a visual analytics system, MoNetExplorer, that utilizes 36 3-event temporal network motifs for analyzing discrete-time temporal networks [61]. Authors used temporal motif similarities and related measures to select and validate proper window sizes which can guide the user to slice the given dynamic network and construct the snapshots. In particular, motif fidelity, motif stability, and motif clusterness are computed in each snapshot for a given window size. Motif fidelity indicates how much information is preserved in each snapshot where high fidelity snapshots are similar to the whole network in terms of motif frequencies and Z-scores. Motif stability measures the changes between consecutive snapshots in terms of motif distributions. Lastly, motif clusterness quantifies how well each snapshot can be clustered where clusters are determined by using HDBSCAN on motif profiles. MoNetExplorer lets the user to choose a window size and then visualizes these three measures accordingly in a longitudinal way so that different states of the network can be chosen.

4.7.2 Synthetic temporal graph generation

Real-world temporal networks pose challenges in collection due to privacy and regulatory constraints, coupled with the high costs associated with large-scale crawling [86]. In response, the generation of synthetic graphs, referred to as "surrogate networks," offers a solution that is both arbitrarily large and anonymized, while still possessing characteristics akin to real-world networks [87]. These synthetic graph generators play a pivotal role in sharing surrogate data, such as computer network traffic or financial networks, and support benchmarking studies, including scalability and versatility tests.

Recent advancements in the field leverage temporal motifs to design synthetic graph generators capable of replicating the features of real-world temporal networks. Purohit et al. introduced the Structural Temporal Modeling (STM), which calculates the frequencies of specific atomic motifs and integrates them into generated temporal graphs through a preferential attachment mechanism [88]. The authors consider a set of easy-to-compute atomic motifs, such as wedges, triangles, and squares, and for each type of atomic motif, the model calculates the independent motif frequencies (ITeM) which prohibits the overlaps between motifs [89]. Nodes and motifs in the output graph are then generated based on the ITeM frequency and a preferential attachment function. Similarly, Zeno et al. proposed a generative model for discrete-time temporal networks, considering dynamic changes in overall graph structure using temporal motif activity and node roles within motifs [58]. The authors observed that the fundamental motifs with three nodes (edge, wedge, triangle) do not change/evolve from one snapshot to another but keep re-appearing in the same configuration. Accordingly, they modeled the active node behavior over time by using those motifs and generate those three types of 3-node motifs with different arrival rates. Empirical evaluation suggests that their model outperforms several baselines including a deep graph generative model for dynamic networks which uses temporal random walks [90].

More recently, Porter et al. introduced the Temporal Activity State Block Model (TASBM) to generate a temporal network featuring temporal motif structures [91]. Temporal graph is divided into multiple time windows and average in-event and out-event arrival rates are computed for each node. Subsequently, nodes are categorized into activity groups based on their activity levels and then events generated between pairs of nodes by a Poisson draw according to the arrival rates between the activity groups. Additionally, the TASBM was employed to analytically compute temporal motif frequencies.

These models have limitations, primarily focusing on a restricted set of temporal motifs, such as wedges and triangles. This approach proves insufficient for capturing more intricate subgraph structures. Another drawback is the reliance on counting temporal motifs without considering correlations between motifs of different sizes. Due to the exponential increase in complexity with motif size, existing studies often overlook temporal motifs with more than three events. Addressing these issues, Liu and Sariyuce introduced the Motif Transition Model (MTM) to practically generate a synthetic temporal network that preserves both global and local features of the input network [92]. Fig. 6 summarizes the MTM. MTM utilizes motif transition properties to model the next event, calculating transition properties from the input graph and simulating stochastic motif transition processes based on transition probabilities and rates. The motif transition process involves a sequence of transitions, limited by transition size and time constraints, categorizing events into cold and hot events. The model uses degree distributions and timestamps of cold events and employs a configuration model to generate new cold events in the synthetic graph. Motif transition probabilities, rates, and the average number of edges contribute to generating hot events through a stochastic process. MTM's extensive experimental evaluation demonstrates its ability to preserve structural and temporal characteristics, surpassing several baselines, including TASBM, STM, and TagGen. Notably, MTM exhibits orders of magnitude faster performance than baselines as it avoids counting motif frequencies.

The concept of motif transitions is also applied by Longa et al. for discrete-time temporal networks, where egocentric temporal neighborhoods of each node are computed, and transitions between these encodings are used to generate new snapshot-based dynamic networks [59].

Lastly, Soliman et al. proposed a Multivariate Community Hawkes Model, a highly flexible continuous-time temporal network model considering dependence between node pairs in a controlled manner [93]. Building on the Stochastic Block Model (SBM), the authors jointly model all node pairs using a multivariate Hawkes process, allowing events between node pairs to influence the probability of events between different pairs. The model ensures reciprocal excitations between blocks, facilitating the creation of realistic higher-order temporal motifs. However, a notable drawback is the computational cost, requiring up to 16 h for a network with fewer than a million events.

5 Future directions and challenges

Despite wide usage in many applications, there remain several challenges that needs to be studied to broaden the applicability of temporal motifs. Here we consider some important challenges along with the opportunities for future directions.

5.1 Random graph models for measuring significance of temporal motifs

Characteristics of real-world (empirical) networks are best understood when compared to null models [34, 94]. Random networks that preserve some features of the empirical network are created for reference, e.g., the configuration model in static networks preserves the node degrees. For temporal networks, the literature is rich with null models, referred to as *randomized reference models* in Gauvin et al.'s survey [95]. Each model is expressed as a shuffling mechanism that preserves certain features in the empirical network. Temporal network features are classified as link-based and time-based. Models that preserve link-based features randomize the timestamps of the events

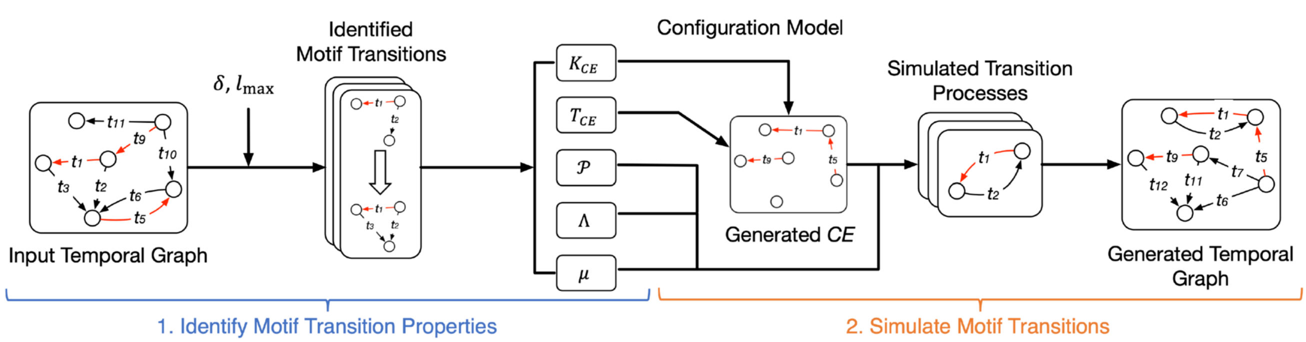


Fig. 6 Motif Transition Model (MTM) by Liu and Sariyuce [92]. Given a time window, δ, and the maximum motif length, l_{max}, the model identifies five key motif transition properties: K_{CE} and T_{CE} are the degree distributions and timestamps of cold events, which are considered to be initial events in motifs, $\mathcal{P}$ is the motif transition probabilities, Λ is the motif transition rates, and μ is the average motif length. Then K_{CE} and T_{CE} are used to generate new cold events, and the rest are used to simulate transition processes to create the final temporal graph (the figure is obtained from [92])

(time-shuffling), whereas models that preserve time-based features randomize the link structure (link-shuffling). Although there are well-established reference models, the proper choice of the null model is non-trivial, even in static networks [96]. The biggest challenge is to find a null model that can preserve both link-based and time-based features, which is also noted by Kovanen et al. [26]. The absolute motif frequencies of the motifs are not an indicator of a significance. Even in static networks certain motif types appear more by definition [67] and hence it is essential to have powerful and representative null models to observe the "expected" motif frequencies. As exemplified below, existing reference models cannot properly measure the significance of temporal motif counts, as they either destroy the link (or time) structure or have no impact at all.

Case study on a messaging network. We consider three random reference models from [95] to measure the significance of 3-event motifs (2 or 3 nodes) in a messaging network (1.9K nodes, 59.8K events). Degree-constrained link shuffling (DLS) shuffles the static edges according to the configuration model, weighted-time shuffling (WTS) randomizes timestamps of the events on each edge, and inter-event shuffling (IS) shuffles the inter-event times among the events on each edge. DLS preserves timestamps, whereas WTS and IS preserve the static projection (link structure). Figure 7 shows the counts of 36 3-event motifs for the original network and randomized versions according to the three models. DLS is loose and yields very different results, with some patterns not observable at all. WTS and IS are very restrictive, with the motif counts (or relative comparison) barely changing. All 36 counts are smaller in the randomized graphs than in the original network. The Z scores are positively large *regardless* of the model used (similar results are observed in [46]). This contradicts the expectation that some motifs would appear more frequently in the original network than in the randomized version, whereas some other motifs would not. Hence, it is essential to design new null models to measure the significance of temporal motif counts.

As discussed in Sect. 3, temporal motifs are a versatile tool and can be parameterized in many dimensions, such as ordering, timing, and inducedness. One promising direction for creating realistic null models for temporal motif analysis is to tailor the reference models to the specific parameters of the temporal motif template. For time-shuffling, one can consider the timing threshold of the motif template to preserve the temporal correlations so that the total count of the motif spectrum is kept the same. For example, if $\Delta_W = 1$ hour, then the randomized network would preserve the number of events occurring each hour. If there is a partial ordering in the motif template, one can ensure that the number of events with the same timestamps is also preserved. For link-shuffling, local event swaps in the ego networks of the nodes can be considered so that the number of motifs in which a node participates in is kept the same. This would restrict the extent of changes in the static projection of the network.

Another interesting future direction can be considering the fundamental structures in temporal networks, defined by Lehmann [68] and discussed in Sect. 3, instead of singular events to develop new reference models. Instead of shuffling individual events, as done in the current reference models, one can shuffle the fundamental structures that form the network. For example, in an email network, one-to-many email interactions can be shuffled while keeping the total number of such emails the same (per each node). One can further control the cardinality constraints in such structures (e.g., number of one-to-three emails) and keep the shufflings independent for each type of one-to-many interaction. We conjecture that this will enable more realistic reference models in which the actual formation process of the temporal network is respected and the atomicity of the interactions is preserved. This is similar to process-based neutral models [97] and considers the topological and temporal properties at the same time.

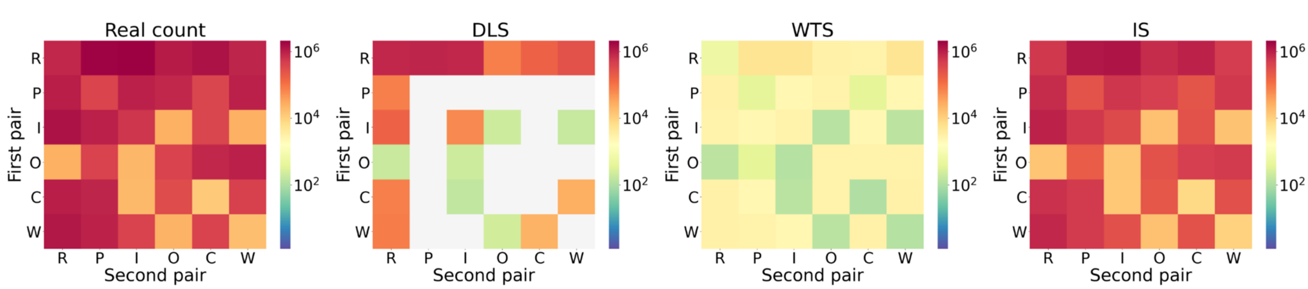


Fig. 7 Counts of 36 3-event motifs in a real and randomized networks according to three models. Each square is a type of 3-event motif; y-axis is the first pair of events (first & second event) and x-axis is the second pair of events (second & third event). There are 6 possible event pairs (Fig. 4); Repetition, Ping-pong, In-burst, Out-burst, Convey, and Weakly-connected (denoted by R P I O C W)

5.2 Subgraph models by using temporal motifs

Subgraph modeling is well studied in temporal networks, although the context is not always truly temporal. Many studies have found temporal dense subgraphs for discrete-time temporal networks [98–100]. The objective is often to find a group of nodes that interact frequently over a short time interval, and the density is defined with respect to the number of interactions over a time period [2, 3, 5, 7, 8, 10, 11, 101, 102]. It is more challenging to characterize and discover subgraphs for continuous-time temporal networks where temporal dimension is dominant [103–106]. Temporal motifs can help in defining new subgraph models that can seamlessly capture both temporal and structural cohesiveness and contain certain higher-order interactions. A few works have sought to adapt core [107] and truss [108] decompositions to temporal networks [109–112]. Although these works could capture dense structures with hierarchical relations, they cannot locate subgraphs where nodes interact in specific ways at higher order, which do not necessarily correspond to cohesive structures.

Example use case. One interesting application for temporal motif based subgraph modeling is the analysis of cryptocurrency networks for anti-money laundering. Peel-chain structure is a common pattern observed in money-laundering in cryptocurrency networks. A large amount of bitcoins are moved between addresses, with a small proportion sent to a destination at each step [73, 75]. For example, the funds stolen from Bitfinex in 2016 were laundered this way [45], as denoted in Fig. 8. Note that the accounts in this example are not densely connected; each node only takes part in a few transactions. Also, the subgraph is rather unstructured and can be best described

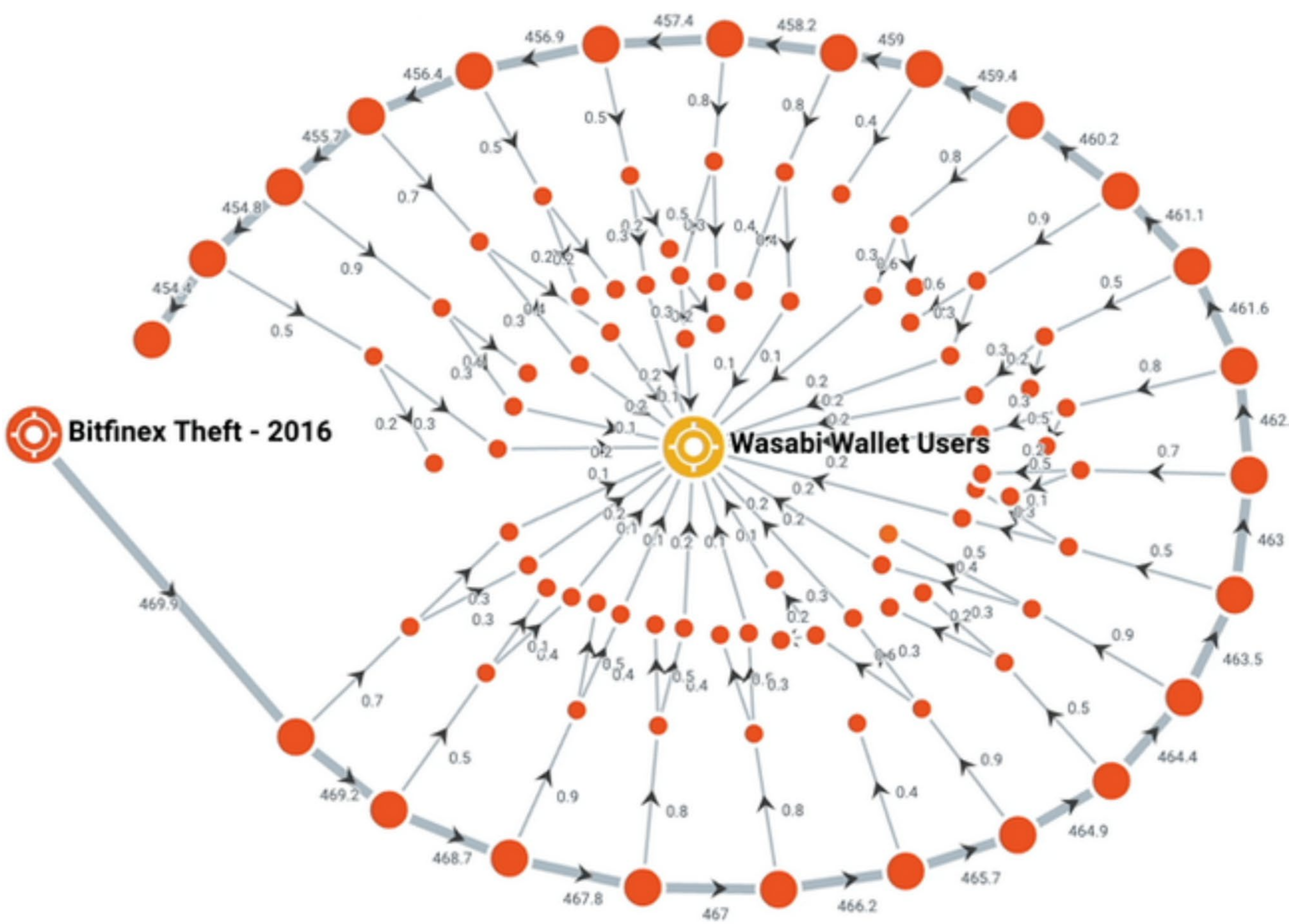


Fig. 8 Obtained from [45]. The stolen fund is laundered through several intermediary addresses in small amounts. The dominant motif in the outer parts is denoted by three events: a large amount is received by a node and forwarded to two other nodes; one transfer is very small and the other is large. In the inner parts a similar motif appears with different weights; received amount is evenly split and forwarded to two other nodes

in terms of certain motifs. This structure can successfully be located by using temporal motifs where each node participates in a certain number of specific types of temporal motifs in the subgraph.

Similar to various subgraph models in static networks, such ask *k*-core and *k*-truss, one can use local temporal motif counts to define subgraphs. Using motifs for graph partitioning/clustering is indeed quite effective in static networks [48–50]. Edge-centric methods have a limited view of the network structure and motifs capture the higher-order relations by considering multiple nodes/edges at the same time. Finding temporal subgraphs where each node participates in a number of specific temporal motif pattern(s) is promising in that sense. Local motif counts can be used to create representative features for nodes/edges and then hierarchical dense subgraph models, such as core/truss decompositions [107, 108], can be adapted to use those counts. Parameterizing the subgraphs with respect to a motif of choice would encode the subgraph in a bottom-up and principled way. The new methods would also inherit the expressive power of the temporal motifs as well as the characteristics of the hierarchical dense subgraph model. The resulting hierarchy can offer a spectrum of subgraphs that have different trade-offs between subgraph size and density.

5.3 Extending temporal motifs to graphs in the wild: weighted, bipartite (hypergraph), and multilayer

Although the fundamental techniques that are targeted for unipartite, unweighted temporal networks form a reliable base, it is important to adapt and extend these to the rich or specific networks observed in real-world, especially the ones that are weighted, bipartite, or multilayer.

For weighted networks, the only related work has a simplistic approach in which the weights of the events in a motif have a lower bound [76]. Basic filtering operations on weights can be helpful to design insightful motifs but also require prior knowledge about the network. Defining relative constraints to incorporate weights in temporal motifs can be promising in that respect, e.g., finding all 3-node 3-event motifs in which the weights of the events increase by time or their standard deviation is bounded with respect to the mean. Such constraints would be more expressive than absolute value-based quantifiers and are expected to be more useful in anomaly detection applications. Another interesting direction could be in subgraph modeling where the subgraphs in which the in-flow or out-flow is maximum/minimum can be targeted. The existence/number of the frequently interacting set of nodes for which the total out-flow/in-flow is high can shed light on the economic stability of the network as well as the groups of sinks/sources that play a significant role in the network.

Regarding bipartite networks, it would be interesting to look at the temporal motifs for which the static projection corresponds to a butterfly (2×2 biclique) [113, 114] or induced 6-cycle (among 6 nodes with 3 on each side and has no butterfly) [115, 116]. This is due to two reasons: (1) the combinatorial explosion in bipartite networks is more serious than in unipartite networks, and thus adhering to certain topologies will enable faster computation; and (2) these two motifs have complementary strengths: butterfly relates two nodes from each side of the network in a fully connected way, whereas induced 6-cycle relates three nodes in the same node set by forming a triangle in the projection in a minimal way. Also, bipartite networks exhibit various types of fundamental structures (defined by Lehmann [68]). A new event may correspond to the formation of a new star if the network is a hypergraph: In an author-paper network, the publication of a new paper creates a new node in the paper set along with the adjacent edges to the authors (which may or may not exist before). On the other hand, a new event in a user-product bipartite network is simply a new single transaction. This distinction is important and must be incorporated when the temporal motifs are defined for bipartite networks. Regarding the subgraph models for bipartite graphs, one challenge is the size imbalance between the two node sets. It has been shown that it is relatively easier to find cohesive subgraphs in which one node set is very small due to the concentrated existence of butterflies [117], which are not helpful to find anomalous nodes. Additional constraints can be incorporated in subgraph definition to balance the sizes of the node sets.

Existing studies on hypergraph motif analysis can also offer interesting insights on designing new bipartite temporal motifs [118–121]. Lee and Shin's definition of TH-motifs is an interesting first step in defining temporal hypergraph motifs [122]. Arregui-García et al.'s adaptation of egocentric temporal motifs to hypergraphs is also promising although the authors solely consider snapshot-based temporal networks [123].

For multilayer networks, although some studies have been conducted on static motif analysis [124–126], temporal networks are largely understudied [127]. One needs to consider the events across all layers when designing multilayer temporal motifs. One challenge in doing this is adjusting the timing parameters. Each layer may have a different rhythm, and thus the same inter-event timing constraint (Δ_C) may not be homogeneously applicable for successive events from different layers. A solution would be to consider a separate Δ_C parameter for each layer, and use the inter-event time distributions of each layer and the entire multilayer network to determine the values. Another challenge is the inducedness constraint; it can be defined separately per layer or collectively for all of the layers. The latter case may be more appropriate for financial transaction networks to consider any type of transaction among the nodes when detecting anomalous groups, whereas the former is better for the analysis of transportation networks where changing the mode of the transportation is costly.

5.4 Scalable algorithms to compute temporal motifs

The general workflow for temporal motif computation has two steps: (1) the static motif computation to locate static projections of a given temporal motif template, and (2) the computation of temporal motifs in each static motif instance. The literature is rich with efficient solutions for step (1) [128, 129]. For step (2), some algorithmic improvements [130–132] and sampling techniques have been proposed [133–136]. In general, scaling motif computations to large networks is challenging. Even in the static case, counting becomes intractable with the motif size. The challenge is more severe in temporal networks due to repetitions and time ordering.

In temporal motif counting studies, motif size is often limited to four nodes as a larger size has a wider spectrum (i.e., too many motif types) for which the computation is also infeasible. Choice of the temporal motif model is an important factor as some, such as Kovanen et al. 's [26], are too restrictive which permit fast algorithms but misses important structures whereas some others, like Paranjape et al.'s [27], result in too many instances which are harder to count—taking hours for large networks with billions of events. In addition, model parameters (shown in Table 1) have a strong impact on the computation runtime—for instance, a large time window may yield few motif instances and relaxed ordering constraints can cause excessively many overlapping, and non-informative, instances.

The state-of-the-art method for (exact) global temporal motif counting, FAST, is proposed by Gao et al. [132]—the implementation is in C++ and the code is available[1]. Authors consider Paranjape et al.'s model (with entire window timing threshold) [27] and facilitates separate custom algorithms to handle star and triangle motifs. For motifs with up to 3 nodes and 3 events, Wang et al.'s algorithm on a single thread takes 1019 s on a network with 613 M events (on a 2.30GHz Intel Xeon E5-2650 v3 processor with 128GB RAM). The algorithm is also parallelized in shared-memory (via OpenMP) on node-level and intra-node parallelism is applied for high-degree nodes, providing 24x speedup on 32 threads (for the network with 613 M events on a 40-core machine). More recently, Li et al. introduced a faster algorithm [137], MoTTo, that makes use of topology- and time-based pruning. For the same motif model [27] and the same motif spectrum (up to 3 nodes, 3 edges), authors report ~2x faster runtimes than FAST [132] on moderate-size networks with up to 27 M events by 32 threads. However, no runtime is reported for large networks with hundreds of millions of events, unlike [132], questioning the scalability of MoTTo. Another recent work, by Yuan et al., proposed a new open-source system in C++ and CUDA, Everest [138], to compute temporal motifs on a (multi) GPU architecture[2]. Everest considers Paranjape et al.'s model too [27]. It makes use of several load balancing optimizations for single GPU, and edge partitioning and scheduling strategies to efficiently map the workloads to multiple GPUs. However, the method takes a single motif as query, rather than computing the entire spectrum. Regardless, Everest is quite efficient and can consider motifs of arbitrary size: in a network with 628 M events, it can compute a 5-node, 4-event motif in 1717 s by a single GPU (NVIDIA A40, 48GB GDDR6 memory). Furthermore, Everest exhibits near-linear scaling with multiple GPUs. Overall, despite promising efforts, scalability of temporal motif computations remains an open challenge that requires further research.

To help with this challenge, two orthogonal techniques can be considered: sampling and parallelization. Sampling techniques are intended for applications that can work with approximate motif counts, e.g., fraud detection based on the node features. Parallelization techniques are applicable as long as a multi/many-core machine with a large memory is available. Sampling and parallelization are not dependent on each other and can be combined when appropriate.

There are three problems with the existing sampling algorithms for temporal motifs [133–136]: (1) they are limited to the motif model in [27], which uses only the entire-motif timing threshold and does not consider any inducedness constraint; (2) they approximate only the global counts, not local; and (3) even the best technique yields up to a 20% accuracy error on large graphs [134]. One open question is to obtain an unbiased estimation of global and local motif counts with provable error bounds for all of the patterns in the motif spectrum of a given motif with specific number of nodes and events. An effective alternative to custom sampling techniques in static networks is color-coding, which is the most scalable technique in terms of graph size and motif size (up to 16 nodes) [139]. In color-coding, nodes are first

[1] https://github.com/steven-ccq/FAST-temporal-motif
[2] https://github.com/yichao-yuan-99/Everest

randomly assigned colors (from a set of k colors, where k is the number of nodes in the motif), after which the colorful treelets, where each node has a distinct color, are counted via dynamic programming (build-up phase), and finally the actual motifs are sampled (sampling phase). Tailoring color-coding for approximate computation of temporal motifs is a promising direction.

For parallelization, one can consider coarse- and fine-grained parallelism. In temporal networks, scale is not only characterized by the number of nodes/events but also by the timespan. The topological scale (i.e., static projection) often increases at a slower rate than the temporal scale [140]. This exposes coarse-grained, pleasingly parallel computations; a network can be sliced into multiple parts with smaller timespans, and each slice can be processed independently—motifs in the intersections can be handled separately. We conjecture that load balancing would be a challenge because events in real-world networks are often not distributed homogeneously over time. One remedy can be work-stealing, where underutilized threads can steal work from other threads' queues. Another challenge would be to determine the optimal size of slices (in terms of time period): A large size may incur load balancing and memory usage issues, whereas a small size may lead to excessive context switching among threads.

Regarding fine-grained parallelism, dynamic programming based approaches that compute sub-motif structures with smaller time windows and reuse those sub-results to compute the temporal motifs are promising. Sub-solutions would be computed along two dimensions: number of nodes/events and time window of the motif. One challenge, however, would be the local temporal heterogeneity: the distribution of the number of events around a node is often skewed (e.g., not all accounts are similarly active).

6 Conclusion

In this perspective piece, we aimed to emphasize a relatively recent but powerful tool, temporal network motifs, for the analysis of temporal networks. Temporal network motifs are a versatile and expressive gadget. They can encode the structural and temporal characteristics of temporal networks at the same time to address the three big challenges in real-world temporal networks: diverse nature, unclear scale, and massive size/timespan. In particular, the analysis of continuous-time temporal networks, where the events happen in a stream, is made simpler through temporal motifs. We covered numerous application areas, including communication networks, trading scenarios, financial transactions, synthetic graph generation, graph neural networks, and more, to exemplify how temporal motifs are successfully deployed for real-world data. Despite several advances in their usage, there are also many challenges that need to be studied to broaden the applicability of temporal motifs. We covered some of those and pinpointed promising future directions. Overall, we believe that temporal motifs are a powerful lens and have potential to be the standard method for temporal network mining. Our hope is this work will prove to be helpful to guide future research on temporal network motifs.

Author contributions A.E.S. conducted the research and wrote the main manuscript.

Funding This research was supported by NSF awards OAC-2107089 and IIS-2236789.

Data availability No datasets were generated or analysed during the current study.

Code availability Not applicable.

Declarations

Ethics approval and consent to participate Not applicable.

Consent for participate Not applicable.

Competing interests Not applicable.

References

1. ...Zitnik M, Li MM, Wells A, Glass K, Gysi DM, Krishnan A, Murali TM, Radivojac P, Roy S, Baudot A, Bozdag S, Chen DZ, Cowen L, Devkota K, Gitter A, Gosline S, Gu P, Guzzi PH, Huang H, Jiang M, Kesimoglu ZN, Koyuturk M, Ma J, Pico AR, Pržulj N, Przytycka TM, Raphael BJ, Ritz A, Sharan R, Shen Y, Singh M, Slonim DK, Tong H, Yang XH, Yoon BJ, Yu H, Milenković T. Current and future directions in network biology. Bioinform Adv. 2024;4(1):vbae099.
2. Bogdanov P, Mongiovì M, & Singh A K. (2011). Mining Heavy Subgraphs in Time-Evolving Networks. In Proceedings of the 11th IEEE International Conference on Data Mining (ICDM) (pp. 81–90).
3. Cadena J, Vullikanti A. (2018). Mining Heavy Temporal Subgraphs: Fast Algorithms and Applications. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI) (Vol. 32, No. 1).
4. Chu L, Zhang Y, Yang Y, Wang L, Pei J. (2019). Online Density Bursting Subgraph Detection from Temporal Graphs. Proceedings of the VLDB Endowment, 12(13), 2353–2365.
5. Gorovits A, Gujral E, Papalexakis E E, Bogdanov P. (2018). LARC: Learning Activity-Regularized Overlapping Communities Across Time. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1465–1474)
6. Jin R, McCalle S, Almaas E. (2007). Trend Motif: A Graph Mining Approach for Analysis of Dynamic Complex Networks. In Proceedings of the 7th IEEE International Conference on Data Mining (ICDM) (pp. 541–546).
7. Ma S, Hu R, Wang L, Lin X, Huai J. (2017). Fast Computation of Dense Temporal Subgraphs. In Proceedings of the 33rd IEEE International Conference on Data Engineering (ICDE) (pp. 361–372).
8. Ma S, Hu R, Wang L, Lin X, Huai J. (2019). An Efficient Approach to Finding Dense Temporal Subgraphs. IEEE Transactions on Knowledge and Data Engineering (TKDE), 32(4), 645–658.
9. Qin H, Li R H, Yuan Y, Wang G, Qin L, Zhang, Z. (2022). Mining Bursting Core in Large Temporal Graphs. Proceedings of the VLDB Endowment, 15(13), 3911-3923.
10. Semertzidis K, Pitoura E, Terzi E, Tsaparas P. (2019). Finding Lasting Dense Subgraphs. Data Mining and Knowledge Discovery (DMKD), 33(5), 1417–1445.
11. Yang Y, Yan D, Wu H, Cheng J, Zhou S, Lui J C. (2016). Diversified Temporal Subgraph Pattern Mining. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1965–1974).
12. Bajardi P, Barrat A, Natale F, Savini L, Colizza V. (2011). Dynamical Patterns of Cattle Trade Movements. PLOS ONE, 6(5), e19869.
13. Caceres R S, Berger-Wolf T. (2013). Temporal Scale of Dynamic Networks. In Temporal Networks (pp. 65–94). Springer.
14. Soundarajan S Tamersoy A, Khalil E B, Eliassi-Rad T, Chau D H, Gallagher B, Roundy K. (2016). Generating Graph Snapshots from Streaming Edge Data. In Proceedings of the 25th International Conference Companion on World Wide Web (pp. 109–110).
15. Liu P, Guarrasi V, Sarıyüce A. E. (2021). Temporal Network Motifs: Models, Limitations, Evaluation. IEEE Transactions on Knowledge and Data Engineering (TKDE), 35(1), 945–957.
16. Holme P. (2015). Modern Temporal Network Theory: A Colloquium. The European Physical Journal B, 88(9), 234.
17. Holme P, Saramäki J. (2012). Temporal Networks. Physics Reports, 519(3), 97–125.
18. Masuda N, Lambiotte R. (2016). A Guide to Temporal Networks. World Scientific.
19. Barabási A L. (2005). The Origin of Bursts and Heavy Tails in Human Dynamics. Nature, 435(7039), 207–211.
20. Lahiri M, Berger-Wolf T Y. (2010). Periodic Subgraph Mining in Dynamic Networks. Knowledge and Information Systems (KAIS, 24(3), 467–497.
21. Clauset A, Eagle N. (2012). Persistence and Periodicity in a Dynamic Proximity Network. arXiv preprint arXiv:1211.7343.
22. Min B, Goh K. (2013). Burstiness: Measures, Models, and Dynamic Consequences. In Temporal Networks (pp. 41–64). Springer.
23. Horváth D X, Kertész J. (2014). Spreading Dynamics on Networks: The Role of Burstiness, Topology and Non-Stationarity. New Journal of Physics, 16(7), 073037.
24. Akbarpour M, Jackson M O. (2018). Diffusion in Networks and the Virtue of Burstiness. Proceedings of the National Academy of Sciences (PNAS), 115(30), 6996–7004.
25. Hulovatyy Y, Chen, H, Milenković T. (2015). Exploring the Structure and Function of Temporal Networks With Dynamic Graphlets. Bioinformatics, 31(12), 171–180.
26. Kovanen L, Karsai M, Kaski K, Kertész J, Saramäki J. (2011). Temporal Motifs in Time-Dependent Networks. Journal of Statistical Mechanics: Theory and Experiment, 2011(11), P11005.
27. Paranjape A. Benson A R, Leskovec J. (2017). Motifs in Temporal Networks. In Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM) (pp. 601–610).
28. Song C, Ge T, Chen C, Wang J. (2014). Event Pattern Matching Over Graph Streams. Proceedings of the VLDB Endowment, 8(4), 413–424.
29. Ahmed N K, Neville J, Rossi R A, Duffield N G, Willke T. L. (2017). Graphlet Decomposition: Framework, Algorithms, and Applications. Knowledge and Information Systems (KAIS), 50, 689–722.
30. Bressan M., Chierichetti, F., Kumar, R., Leucci, S., & Panconesi, A. (2017). Counting Graphlets: Space vs Time. In Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM) (pp. 557–566).
31. Jain S, Seshadhri C. (2017). A Fast and Provable Method for Estimating Clique Counts Using Turán's Theorem. In Proceedings of the International Conference on World Wide Web (pp. 441–449).
32. Jha M, Seshadhri C, Pinar A. (2013). Path Sampling: A Fast and Provable Method for Estimating 4-Vertex Subgraph Counts. In Proceedings of the International Conference on World Wide Web (pp. 495–505).

33. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovski D, Alon U. (2002). Network Motifs: Simple Building Blocks of Complex Networks. Science, 298(5594), 824–827.
34. Milo R, Itzkovitz S, Kashtan N, Levitt R, Shen-Orr S, Ayzenshtat I, Sheffer M, Alon U. (2004). Superfamilies of Evolved and Designed Networks. Science, 303(5663), 1538–1542.
35. Staden R. (1988). Methods to Define and Locate Patterns of Motifs in Sequences. Bioinformatics, 4(1), 53–60.
36. Esling P, Agon C. (2012). Time-Series Data Mining. ACM Computing Surveys, 45(1).
37. Lin J, Keogh E, Lonardi S, Patel, P. (2002). Finding Motifs in Time Series. In Proceedings of the 2nd Workshop on Temporal Data Mining, SIGKDD (pp. 53–68).
38. Gao Y, Lin J. (2019). Discovering Subdimensional Motifs of Different Lengths in Large-Scale Multivariate Time Series. In 2019 IEEE International Conference on Data Mining (ICDM) (pp. 220–229).
39. Mueen A, Keogh E, Zhu Q, Cash S, Westover B. (2009). Exact Discovery of Time Series Motifs. In Proceedings of the 2009 SIAM International Conference on Data Mining (SDM) (pp. 473–484).
40. Yeh C M, Zhu Y, Ulanova L, Begum N, Ding Y, Dau H A., Zimmerman Z, Silva D F, Mueen A., Keogh E J. (2018). Time Series Joins, Motifs, Discords and Shapelets: A Unifying View That Exploits the Matrix Profile. Data Mining and Knowledge Discovery (DMKD), 32(1), 83–123.
41. Bailey, T. L., Williams, N., Misleh, C., & Li, W. W. (2006). MEME: Discovering and Analyzing DNA and Protein Sequence Motifs. Nucleic Acids Research, 34(suppl 2), 369–373.
42. Narasimhan, G., Bu, C., Gao, Y., Wang, X., Xu, N., & Mathee, K. (2002). Mining Protein Sequences for Motifs. Journal of Computational Biology, 9(5), 707–720.
43. Sikdar, S., Ganguly, N., & Mukherjee, A. (2016). Time Series Analysis of Temporal Networks. The European Physical Journal B, 89(1), 11.
44. Kelkar, S., Kraus, T., Morgan, D., Zhang, J., & Dai, R. (2018). Analyzing HTTP-Based Information Exfiltration of Malicious Android Applications. In 2018 IEEE International Conference on Trust, Security and Privacy in Computing and Communications / 12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE) (pp. 1642–1645).
45. Elliptic. (2022). Bitfinex Theft. https://www.elliptic.co/blog/elliptic-analysis-bitcoin-bitfinex-theft [Online; accessed 3-June-2022].
46. Liu, P., Masuda, N., Kito, T., & Sarıyüce, A. E. (2022). Temporal Motifs in Patent Opposition and Collaboration Networks. Scientific Reports, 12(1), 1–11.
47. Kodate, S., Chiba, R., Kimura, S., & Masuda, N. (2020). Detecting Problematic Transactions in a Consumer-to-Consumer E-Commerce Network. Applied Network Science, 5(1), 1–18.
48. Benson, A. R., Gleich, D. F., & Leskovec, J. (2016). Higher-Order Organization of Complex Networks. Science, 353(6295), 163–166.
49. Sarıyüce, A. E. (2021). Motif-Driven Dense Subgraph Discovery in Directed and Labeled Networks. In Proceedings of the Web Conference 2021 (pp. 379–390).
50. Tsourakakis, C. E., Pachocki, J., & Mitzenmacher, M. (2017). Scalable Motif-Aware Graph Clustering. In Proceedings of the 26th International Conference on World Wide Web (pp. 1451–1460).
51. Ahmed, N. K., Rossi, R., Lee, J. B., Willke, T. L., Zhou, R., Kong, X., & Eldardiry, H. (2018). Learning Role-Based Graph Embeddings. Relational AI Workshop IJCAI-18. 2018. 10.48550/arXiv.1802.02896.
52. Rossi, R. A., Ahmed, N. K., & Koh, E. (2018). Higher-Order Network Representation Learning. In Companion Proceedings of The Web Conference 2018 (pp. 3–4).
53. Tu, K., Li, J., Towsley, D., Braines, D., & Turner, L. D. (2018). Network Classification in Temporal Networks Using Motifs. arXiv preprint arXiv:1807.03733.
54. Wang, Y., Chang, Y.-Y., Liu, Y., Leskovec, J., & Li, P. (2021). Inductive Representation Learning in Temporal Networks via Causal Anonymous Walks. In International Conference on Learning Representations (ICLR).
55. Jin, M., Li, Y.-F., & Pan, S. (2022). Neural Temporal Walks: Motif-Aware Representation Learning on Continuous-Time Dynamic Graphs. Advances in Neural Information Processing Systems (NeurIPS), 35, 19874–19886.
56. Liu, P., Acharyya, R., Tillman, R. E., Kimura, S., Masuda, N., & Sarıyüce, A. E. (2023). Temporal Motifs for Financial Networks: A Study on Mercari, JPMC, and Venmo Platforms. arXiv preprint arXiv:2301.07791.
57. Zhang, X., Shao, S., Stanley, H. E., & Havlin, S. (2014). Dynamic Motifs in Socio-Economic Networks. EPL (Europhysics Letters), 108(5), 58001.
58. Zeno, G., La Fond, T., & Neville, J. (2021). DYMOND: Dynamic Motif-Nodes Network Generative Model. In Proceedings of the Web Conference 2021 (pp. 718–729).
59. Longa, A., Cencetti, G., Lehmann, S., Passerini, A., & Lepri, B. (2024). Generating Fine-Grained Surrogate Temporal Networks. Communications Physics, 7(1), 22.
60. Qiao, Z., Li, W., & Li, Y. (2022). Temporal Network Embedding with Motif Structural Features. In Database Systems for Advanced Applications (pp. 665–681).
61. Jung, S., Shin, D., Jeon, H., Choe, K., & Seo, J. (2023). MonetExplorer: A Visual Analytics System for Analyzing Dynamic Networks with Temporal Network Motifs. IEEE Transactions on Visualization and Computer Graphics, 1–15.
62. Boekhout, H. D., Traag, V. A., & Takes, F. W. (2021). Investigating Scientific Mobility in Co-authorship Networks Using Multilayer Temporal Motifs. Network Science, 9(3), 354–386.
63. Chechik, G., Oh, E., Rando, O., Weissman, J., Regev, A., & Koller, D. (2008). Activity Motifs Reveal Principles of Timing in Transcriptional Control of the Yeast Metabolic Network. Nature Biotechnology, 26(11), 1251–1259.
64. Oettershagen, L., Kriege, N. M., Jordan, C., & Mutze, P. (2023). A Temporal Graphlet Kernel for Classifying Dissemination in Evolving Networks. In Proceedings of the 2023 SIAM International Conference on Data Mining (SDM) (pp. 19–27).
65. Yoon, D., Lee, D., Choe, M., & Shin, K. (2023). Graphlets over Time: A New Lens for Temporal Network Analysis. arXiv preprint arXiv:2301.00310.
66. Longa, A., Cencetti, G., Lepri, B., & Passerini, A. (2022). An Efficient Procedure for Mining Egocentric Temporal Motifs. Data Mining and Knowledge Discovery (DMKD), 36(1), 355–378.
67. Pržulj, N. (2007). Biological Network Comparison Using Graphlet Degree Distribution. Bioinformatics, 23(2), 177–183.

68. Lehmann, S. (2019). Fundamental Structures in Temporal Communication Networks. In P. Holme & J. Saramäki (Eds.), Temporal Network Theory (pp. 25–48). Springer, Cham.
69. Zhao, Q., Tian, Y., He, Q., Oliver, N., Jin, R., & Lee, W.-C. (2010). Communication Motifs: A Tool to Characterize Social Communications. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM) (pp. 1645–1648).
70. Kovanen, L., Kaski, K., Kertész, J., & Saramäki, J. (2013). Temporal Motifs Reveal Homophily, Gender-Specific Patterns, and Group Talk in Call Sequences. Proceedings of the National Academy of Sciences, 110(45), 18070–18075.
71. Li, M. X., Palchykov, V., Jiang, Z. Q., Kaski, K., Kert´esz, J., Miccich`e, S., Tumminello, M., Zhou, W.-X., Mantegna, R.N. (2014). Statistically validated mobile communication networks: the evolution of motifs in European and Chinese data. New Journal of Physics, 16(8), 083038.
72. Sarkar, S., Guo, R., & Shakarian, P. (2019). Using Network Motifs to Characterize Temporal Network Evolution Leading to Diffusion Inhibition. Social Network Analysis and Mining, 9(1), 14.
73. Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G. M., & Savage, S. (2016). A Fistful of Bitcoins: Characterizing Payments Among Men with No Names. Communications of the ACM, 59(4), 86–93.
74. Möser, M., & Narayanan, A. (2022). Resurrecting Address Clustering in Bitcoin. In International Conference on Financial Cryptography and Data Security (pp. 386–403). Springer.
75. Kappos, G., Yousaf, H., Stütz, R., Rollet, S., Haslhofer, B., & Meiklejohn, S. (2022). How to Peel a Million: Validating and Expanding Bitcoin Clusters. In 31st USENIX Security Symposium (pp. 2207–2223).
76. Kosyfaki, C., Mamoulis, N., Pitoura, E., & Tsaparas, P. (2019). Flow Motifs in Interaction Networks. In Proceedings of the International Conference on Extending Database Technology (EDBT) (pp. 241–252).
77. Wu, J., Liu, J., Chen, W., Huang, H., Zheng, Z., & Zhang, Y. (2022). Detecting Mixing Services via Mining Bitcoin Transaction Network with Hybrid Motifs. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(4), 2237–2249.
78. Wang, Y., Wang, H., Lu, X., Zhou, L., & Liu, L. (2023). Detecting Ethereum Phishing Scams with Temporal Motif Features of Subgraph. In 2023 IEEE Symposium on Computers and Communications (pp. 631–636).
79. Liu, J., Chen, J., Wu, J., Wu, Z., Fang, J., & Zheng, Z. (2024). Fishing for Fraudsters: Uncovering Ethereum Phishing Gangs with Blockchain Data. IEEE Transactions on Information Forensics and Security, 19, 3038-3050.
80. Fortunato, S., Bergstrom, C. T., Börner, K., Evans, J. A., Helbing, D., Milojević, S., Petersen, A. M., Radicchi, F., Sinatra, R., Uzzi, B., et al. (2018). Science of Science. Science, 359(6379), eaao0185.
81. Xuan, Q., Fang, H., Fu, C., & Filkov, V. (2015). Temporal Motifs Reveal Collaboration Patterns in Online Task-Oriented Networks. Physical Review E, 91(5), 052813.
82. Hosseinzadeh, M. M., Cannataro, M., Guzzi, P. H., & Dondi, R. (2022). Temporal Networks in Biology and Medicine: A Survey on Models, Algorithms, and Tools. Network Modeling Analysis in Health Informatics and Bioinformatics, 12(1), 10.
83. Pung, R., Firth, J. A., Spurgin, L. G., Chang, A., Kong, J., Wong, J., Ooi, J., Selvaraj, D., Yong, D., Lang, J., Sivalingam, A., Lee, V. J., & Kucharski, A. J. (2022). Using High-Resolution Contact Networks to Evaluate SARS-CoV-2 Transmission and Control in Large-Scale Multi-Day Events. Nature Communications, 13(1), 1956.
84. Longa, A., Lachi, V., Santin, G., Bianchini, M., Lepri, B., Lio, P., Scarselli, F., & Passerini, A. (2023). Graph Neural Networks for Temporal Graphs: State of the Art, Open Challenges, and Opportunities. Transactions on Machine Learning Research.
85. Tarawaneh, R. M., Keller, P., & Ebert, A. (2012). A General Introduction to Graph Visualization Techniques. In C. Garth, A. Middel, & H. Hagen (Eds.), Visualization of Large and Unstructured Data Sets: Applications in Geospatial Planning, Modeling and Engineering - Proceedings of IRTG 1131 Workshop 2011 (OASIcs, Vol. 27, pp. 151–164). Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
86. Presigny, C., Holme, P., & Barrat, A. (2021). Building Surrogate Temporal Network Data from Observed Backbones. Physical Review E, 103(5), 052304.
87. Kolda, T. G., Pinar, A., Plantenga, T., & Seshadhri, C. (2014). A Scalable Generative Graph Model with Community Structure. SIAM Journal on Scientific Computing, 36(5), C424–C452.
88. Purohit, S., Holder, L. B., & Chin, G. (2018). Temporal Graph Generation Based on a Distribution of Temporal Motifs. In Proceedings of the 14th International Workshop on Mining and Learning with Graphs (Vol. 7).
89. Purohit, S., Chin, G., & Holder, L. B. (2022). ITEM: Independent Temporal Motifs to Summarize and Compare Temporal Networks. Intelligent Data Analysis, 26(4), 1071–1096.
90. Zhou, D., Zheng, L., Han, J., & He, J. (2020). A Data-Driven Graph Generative Model for Temporal Interaction Networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 401–411).
91. Porter, A., Mirzasoleiman, B., & Leskovec, J. (2022). Analytical Models for Motifs in Temporal Networks. In Companion Proceedings of the Web Conference 2022 (pp. 903–909).
92. Liu, P., & Sariyüce, A. E. (2023). Using Motif Transitions for Temporal Graph Generation. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.
93. Soliman, H., Zhao, L., Huang, Z., Paul, S., & Xu, K. S. (2022). The Multivariate Community Hawkes Model for Dependent Relational Events in Continuous-Time Networks. In Proceedings of the International Conference on Machine Learning (PMLR) (pp. 20329–20346).
94. Newman, M. (2018). Networks. Oxford University Press.
95. Gauvin, L., Génois, M., Karsai, M., Kivelä, M., Takaguchi, T., Valdano, E., & Vestergaard, C. L. (2022). Randomized Reference Models for Temporal Networks. SIAM Review, 64(4), 763–830.
96. Artzy-Randrup, Y. (2004). Comment on Network Motifs: Simple Building Blocks of Complex Networks. Science, 305(5687), 1107c.
97. Gotelli, N. J., & McGill, B. J. (2006). Null Versus Neutral Models: What's the Difference? Ecography, 29(5), 793–800.
98. Aggarwal, C., & Subbian, K. (2014). Evolutionary Network Analysis: A Survey. ACM Computing Surveys, 47(1), 1–36.
99. Bilgin, C. C., & Yener, B. (2006). Dynamic Network Evolution: Models, Clustering, Anomaly Detection. IEEE Networks, 1, 13.
100. Rossetti, G., & Cazabet, R. (2018). Community Discovery in Dynamic Networks: A Survey. ACM Computing Surveys, 51(2), 1–37.
101. Jethava, V., & Beerenwinkel, N. (2015). Finding Dense Subgraphs in Relational Graphs. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 641–654).
102. Rahman, A., Jan, S. T., Kim, H., Prakash, B. A., & Murali, T. (2016). Unstable Communities in Network Ensembles. In Proceedings of the 2016 SIAM International Conference on Data Mining (SDM) (pp. 504–512).

103. Rozenshtein, P., Bonchi, F., Gionis, A., Sozio, M., & Tatti, N. (2020). Finding Events in Temporal Networks: Segmentation Meets Densest Subgraph Discovery. Knowledge and Information Systems (KAIS), 62(4), 1611-1639.
104. Rozenshtein, P., Tatti, N., & Gionis, A. (2014). Discovering Dynamic Communities in Interaction Networks. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 678–693).
105. Rozenshtein, P., Tatti, N., & Gionis, A. (2017). Finding Dynamic Dense Subgraphs. ACM Transactions on Knowledge Discovery from Data (TKDD), 11(3).
106. Viard, T., Latapy, M., & Magnien, C. (2016). Computing Maximal Cliques in Link Streams. Theoretical Computer Science, 609, 245–252.
107. Seidman, S. B. (1983). Network Structure and Minimum Degree. Social Networks, 5(3), 269–287.
108. Cohen, J. (2008). Trusses: Cohesive Subgraphs for Social Network Analysis. National Security Agency Technical Report.
109. Galimberti, E., Ciaperoni, M., Barrat, A., Bonchi, F., Cattuto, C., & Gullo, F. (2020). Span-Core Decomposition for Temporal Networks: Algorithms and Applications. ACM Transactions on Knowledge Discovery from Data (TKDD), 15(1).
110. Li, R.-H., Su, J., Qin, L., Yu, J. X., & Dai, Q. (2018). Persistent Community Search in Temporal Networks. In 2018 IEEE 34th International Conference on Data Engineering (ICDE) (pp. 797–808).
111. Lotito, Q. F., & Montresor, A. (2020). Efficient Algorithms to Mine Maximal Span-Trusses From Temporal Graphs. In Proceeding of the 16th International Workshop on Mining and Learning With Graphs.
112. Wu, H., Cheng, J., Lu, Y., Ke, Y., Huang, Y., Yan, D., & Wu, H. (2015). Core Decomposition in Large Temporal Graphs. In 2015 IEEE International Conference on Big Data (pp. 649–658).
113. Lind, P. G., González, M. C., & Herrmann, H. J. (2005). Cycles and Clustering in Bipartite Networks. Physical Review E, 72, 056127.
114. Robins, G., & Alexander, M. (2004). Small Worlds Among Interlocking Directors: Network Structure and Distance in Bipartite Graphs. Computational & Mathematical Organization Theory, 10(1), 69–94.
115. Opsahl, T. (2013). Triadic Closure in Two-Mode Networks: Redefining the Global and Local Clustering Coefficients. Social Networks, 35(2), 159–167.
116. Niu, J., Zola, J., & Sarıyüce, A. E. (2022, August). Counting Induced 6-cycles in Bipartite Graphs. In Proceedings of the 51st International Conference on Parallel Processing (ICPP) (pp. 1-10).
117. Sarıyüce, A. E., & Pinar, A. (2018). Peeling Bipartite Networks for Dense Subgraph Discovery. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM) (pp. 504–512).
118. Lotito, Q. F., Musciotto, F., Battiston, F., & Montresor, A. (2024). Exact and Sampling Methods for Mining Higher-Order Motifs in Large Hypergraphs. Computing, 106(2), 475–494.
119. Lotito, Q. F., Musciotto, F., Montresor, A., & Battiston, F. (2022). Higher-Order Motif Analysis in Hypergraphs. Communications Physics, 5(1), 79.
120. Lee, G., Ko, J., & Shin, K. (2020). Hypergraph Motifs: Concepts, Algorithms, and Discoveries. Proceedings of the VLDB Endowment, 13(12), 2256–2269.
121. Niu, J., Amburg, I. D., Aksoy, S. G., & Sarıyüce, A. E. (2024). Retrieving Top-k Hyperedge Triplets: Models and Applications. In 2024 IEEE International Conference on Big Data (pp. 630-639).
122. Lee, G., & Shin, K. (2021). THyME+: Temporal Hypergraph Motifs and Fast Algorithms for Exact Counting. In 2021 IEEE International Conference on Data Mining (ICDM) (pp. 310–319).
123. Arregui-García, B., Longa, A., Lotito, Q. F., & Meloni, S., & Cencetti, G. (2024). Patterns in Temporal Networks with Higher-Order Egocentric Structures. Entropy, 26(3), 256.
124. Battiston, F., Nicosia, V., Chavez, M., & Latora, V. (2017). Multilayer Motif Analysis of Brain Networks. Chaos: An Interdisciplinary Journal of Nonlinear Science, 27(4), 047404.
125. Ren, Y., Sarkar, A., Ay, A., Dobra, A., & Kahveci, T. (2019). Finding Conserved Patterns in Multilayer Networks. In Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (pp. 97–102).
126. Zhong, L., Zhang, Q., Yang, D., Chen, G., & Yu, S. (2019). Analysing Motifs in Multilayer Networks. arXiv preprint arXiv:1903.01722.
127. Boekhout, H. D., Kosters, W. A., & Takes, F. W. (2019). Efficiently Counting Complex Multilayer Temporal Motifs in Large-Scale Networks. Computational Social Networks, 6(1), 1–34.
128. Pashanasangi, N., & Seshadhri, C. (2020). Efficiently Counting Vertex Orbits of All 5-Vertex Subgraphs, by EVOKE. In Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM) (pp. 447–455).
129. Seshadhri, C., & Tirthapura, S. (2019). Scalable Subgraph Counting: The Methods Behind the Madness. In Companion Proceedings of The 2019 World Wide Web Conference (pp. 1317-1318).
130. Gurukar, S., Ranu, S., & Ravindran, B. (2015). COMMIT: A scalable approach to mining communication motifs from dynamic networks. In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data (pp. 475-489).
131. Kumar, R., & Calders, T. (2018). 2SCENT: An Efficient Algorithm to Enumerate All Simple Temporal Cycles. Proceedings of the VLDB Endowment, 11(11), 1441–1453.
132. Gao, Z., Cheng, C., Yu, Y., Cao, L., Huang, C., & Dong, J. (2022). Scalable Motif Counting for Large-Scale Temporal Graphs. In Proceedings of the 38th IEEE International Conference on Data Engineering (ICDE) (pp. 2656–2668).
133. Liu, P., Benson, A. R., & Charikar, M. (2019). Sampling Methods for Counting Temporal Motifs. In Proceedings of the 12th ACM International Conference on Web Search and Data Mining (WSDM) (pp. 294–302).
134. Sarpe, I., & Vandin, F. (2021). OdeN: Simultaneous Approximation of Multiple Motif Counts in Large Temporal Networks. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management (CIKM) (pp. 1568-1577).
135. Sarpe, I., & Vandin, F. (2021). Presto: Simple and Scalable Sampling Techniques for the Rigorous Approximation of Temporal Motif Counts. In Proceedings of the 2021 SIAM International Conference on Data Mining (SDM) (pp. 145-153).
136. Wang, J., Wang, Y., Jiang, W., Li, Y., & Tan, K. L. (2020). Efficient Sampling Algorithms for Approximate Temporal Motif Counting. In Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM) (pp. 1505-1514).
137. Li, J., Qi, J., Huang, Y., Cao, L., Yu, Y., & Dong, J. (2024). MoTTo: Scalable Motif Counting with Time-aware Topology Constraint for Large-scale Temporal Graphs. In Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM) (pp. 1195-1204).

138. Yuan, Y., Ye, H., Vedula, S., Kaza, W., & Talati, N. (2023). Everest: GPU-Accelerated System for Mining Temporal Motifs. Proceedings of the VLDB Endowment, 17(2), 162-174.
139. Bressan, M., Leucci, S., & Panconesi, A. (2019). Motivo: Fast Motif Counting via Succinct Color Coding and Adaptive Sampling. Proceedings of the VLDB Endowment, 12(11), 1651-1663.
140. Leskovec, J., Kleinberg, J., & Faloutsos, C. (2007). Graph Evolution: Densification and Shrinking Diameters. ACM Transactions on Knowledge Discovery from Data (TKDD), 1(1), 2-es.