Disentangling the Growth of Blockchain-based Networks by Graph Evolution Rule Mining

Alessia Galdeman
Department of Computer Science
University of Milan
Milan, IT
alessia.galdeman@unimi.it

Matteo Zignani
Department of Computer Science
University of Milan
Milan, IT
matteo.zignani@unimi.it

Sabrina Gaito
Department of Computer Science
University of Milan
Milan, IT
sabrina.gaito@unimi.it

***Abstract*—Web3, one of the novel paradigms which may drive the evolution of the future Web, is offering an invaluable volume of data stored in the supporting blockchains. Researchers from different fields such as network science, computational social science and data mining, might benefit from these large collections of temporal and heterogeneous data capturing different kinds of interaction among people and between people and the platforms. In this study we focus on a specific issue related to these modern techno-social systems, i.e. the understanding of the rules driving their growth. To reach this goal, we performed an analysis based on graph evolution rules - GERs - on different networks gathered from Web3 platforms such as Steemit or OpenSea. Graph evolution rules mining is a frequency-based method for evaluating network evolution which does not require any prior growth process for disentangling how networks evolve. By comparing the evolution rules of social network platforms and asset trading services through GER profiles, we observe that some evolution rules are common to all Web3 platforms, regardless of the system specificity. On the other hand, in specific cases, the frequency of graph evolution rules is influenced by the nature of the platform: whereas social and token-transfer networks are characterized by rules which increase network transitivity and reciprocity, NFT trading networks, especially those specialized in a specific type of digital asset, are driven by rules which form trading chains. These findings suggest that the GER approach and the GER profiles are a good starting point to get insights into the evolutionary behavior of a network and to define a classification of graph evolution rules.**

***Index Terms*—graph evolution mining, blockchain-based platforms, subgraph mining, evolution profile, graph evolution rule, NFTs**

I. Introduction

In the last years, we witnessed the emergence of novel paradigms which are attempting to replace the current organization of Web 2.0, considered by many as over-centralized around a few big companies. Among such paradigms, the idea of platforms and software systems built on blockchain technologies - namely Web3 - is gaining momentum so much so that many online services have their own decentralized counterpart in the Web3 world. For instance, blockchain online social networks, such as Hive, Mind, or Steemit, are proposing services similar to Twitter or Reddit; online games, cloud storage systems, and gambling and betting platforms are a few of the decentralized applications - DApps - currently implemented on the most important blockchains. Moreover, the Web3 ecosystem is also characterized by peculiar services, such as Decentralized Finance (DeFi), where currencies are exchanged without institutional intermediaries; Decentralized Autonomous Organizations (DAOs), organizations that are completely compliant with their smart contracts which can be updated through the voting of the community; and non-fungible token (NFT), a financial asset, linked to data stored in the blockchain, that can be traded.

Despite the great debate on Web3 between enthusiasts and skeptics, there is no doubt that all these kinds of services and platforms supported by blockchain technologies offer a great opportunity to researchers in different fields. In fact, through the underlying blockchains, one can easily access a broad set of data about modern techno-social systems; and, despite the main online social platforms, data are publicly available, validated, and accessible by interfacing with the blockchain. Moreover, data from Web3 systems have two further characteristics: i) each block of the blockchain has a validation timestamp, so each record has temporal information associated with it; and ii) each block may contain heterogeneous types of information which capture the different way - social, economical, financial - people interact through Web3 systems. So, having a large volume of temporal and heterogeneous data describing the networked structure of the interactions among platforms' users is crucial for facing tasks and issues related to modern techno-social networks. Specifically, here we focus on the growth of such networks from the perspective of the link creation process, and we exploit the high-resolution temporal data different Web3 systems provide.

Understanding and inferring how the networks behind large techno-social systems form and grow are fundamental elements for the comprehension of the main processes driving the evolution of such systems and for the identification of specific patterns of growth which are consequences of the platform design or of the users' behavior. To reach these goals, in the past years, many models, mechanisms, and measures describing the network growth from a link formation perspective have been proposed, including preferential attachment, node fitness [1], triadic closure [2], homophily [3], or node latent features [4]. Most of these approaches rely on the assumption that the growth is guided by a single parameterized mechanism, but current techno-social networks are the result of different and heterogeneous behaviors where different choices and mechanisms occur [5]. For these reasons, for a better understanding of which mechanisms are driving the network evolution and avoiding a-priori mechanism, methods which rely only on the identification of small frequent subgraphs evolving in time [6] can be more effective in capturing the rules of growth since they do not require specific assumptions on the processes governing the growth.

In this paper, we adopt the latter approach based on the mining of subgraph evolution rules to study the fundamental growth patterns characterizing different Web3 platforms. Specifically, we leverage a state-of-art algorithm - EvoMine - to extract graph evolution rules - GERs - from a collection of high-resolution temporal annotated datasets gathered from Web3 platforms. In short, a graph evolution rule is a pair of small subgraphs where the second element - head - represents the evolution of the first element - body. The main goal of the algorithm is to compute how frequently the rule occurs in the growth of the network. We apply the above method to different Web3 services: a collection of trade networks where the traded assets are NFTs, and social and transfer networks extracted from the blockchain online social media Steemit. The findings resulting from the analysis of the most frequent graph evolution rules characterizing each platform have highlighted two main aspects of the Web3 techno-social networks:

- the nature and the design of the different platforms impact on the frequency of the graph evolution rules. In fact, by comparing how evolution rules are distributed in the platforms, we find that some rules are more frequent than others, and those rules can be explained by taking into account the type of the network. For instance, in an NTF trading network specialized in a single type of NTF - images of cats - rules based on the triadic closure process are completely missing;
- despite the specificity of the systems - trading, transfer and social networks - some evolution rules are common to all the systems. In particular, GERs related to the expansion of the neighborhood of a node are frequent in all datasets. In fact, all Web3 systems are in an expansion phase, as a consequence of their novelty;

To the best of our knowledge, this paper stands as the first study on the growth of temporal networks coming from modern Web3 platforms. In doing so, for describing the growth of the networks, we have adopted an approach for the identification of graph evolution rules which i) supports an analysis without a prior assumption about the growth processes; and ii) maintains a certain degree of readability and explainability of the results since it returns rules based on small subgraphs. Moreover, from a methodological viewpoint, our approach allows generating a footprint of the network growth - the GER profile - that can be used to compare how two o more temporal networks evolve.

The paper is organized as follows. Section II provides a brief introduction of Web3 platforms we hereby investigate: blockchain online social networks and NFT trade networks; and a review of methods for the extraction of graph evolution rules. In Section III we describe the datasets collected from Web3 platforms along with their peculiarities. The approach for modelling, extracting and analyzing graph evolution rules is presented in Section IV. Sections V and VI report the main findings of the growth of the Web3 platforms and a discussion about similarities and differences of the evolution of the networks. Finally, Section VII concludes the paper, pointing out possible future works.

II. Background and Related Works

The Web3 paradigm is quite a new framework in the Web landscape, especially as far as regards the aspects directly related to the blockchain technology supporting platforms and services. To introduce this paradigm, first, we provide the reader a brief overview of the Web3 platforms we treated in our analysis: blockchain online social networks and NTF trades. Then, we also review the main methods to extract graph evolution rules from large-scale temporal networks.

A. Blockchain online social networks

Blockchain technology has enabled the development of blockchain online social networks (BOSNs), providing data storage and validation for these platforms. In their core BOSNs replicate the main user experience of the main micro-blogging and social media platforms such as Twitter, Reddit or Medium, but they introduce token-economy aspects, such as a reward system based on cryptocurrencies that promote high-quality content. In fact, in these systems cryptocurrencies can be created, exchanged, and used for validating both social operations (follow, vote, comment) and economical transactions (transfer, borrow tokens).

In this context, one of the most attractive and spread platforms is Steemit [7]. Steemit is a blockchain social network launched in March 2016, hosted on the Steem blockchain. Steemit users can exchange goods and services using the dedicated cryptocurrency, called STEEM. Furthermore, the cryptocurrency powers a reward system that encourages network growth by compensating users for their participation on the platform. Web3 platforms such as Steemit offer a rich data source for understanding the system's dynamics and the networked structure of its components, so much so that the literature about BOSNs analysis is growing. Some works leverage user content for bot detection [8] or text mining tasks [9]. Other works focus on the relationship between blockchain technology and social networks [10]–[12]. For example, Chonan [13] and Kim *et al.* [14] have analyzed the social network structure of the Steemit platform, while Guidi *et al.* [15] have studied the graph of follow operations, and then focus on other operation types [16]. When studying dynamic systems BOSNs, temporal information plays an essential role, so it is important to model the data as dynamic graphs and study its temporal aspects. For example, Ba *et al.* [17] have studied how cryptocurrency and graph evolution are related to each other. The same authors have also conducted an analysis on the network burstiness [18], focusing on the link creation process and the claiming of rewards. Finally, the interplay between social and economical network layers has been investigated in [19] to cope with user migration across Web3 platforms.

B. Non-fungible tokens - NFTs

An NFT is a blockchain-based data unit with a double goal: first, it provides a unique certificate of ownership of a digital object. Second, it attests to the uniqueness and non-transferability of a digital asset. Thanks to this technology, it is possible to track down the complete history of ownership of an object and check its authenticity. In concrete terms, an NFT can represent a variety of digital items, including photographs, movies, and audio. As a consequence, several contexts, such as art, gaming, and sports collectibles, utilize NFTs to regulate and control digital objects. The birth of the NFT market can be traced back to late 2017 when the blockchain game Cryptokitties gained popularity. However, the market remained dominated only by Crypokitties until July 2020 when it started to grow and in March 2021 reached a peak of popularity, due to the selling of an artwork's NFT for $69.3 million. This purchase allowed the author, Beeple, to reach one of the highest auction prices for a living artist.

The peculiar growth history of the NFTs market can explain why the literature on them is currently in rapid growth. Nadini *et al.* [20] conducted the first comprehensive quantitative overview of the NFTs market, including the overall statistical properties, its evolution over time, a network-based analysis, and a study about the predictability of NFT sales. Other works present a more focalized analysis, for example, Vasan *et al.* [21] analyzed the cryptoart ecosystem, while Franceschet [22] focused on the creators-collectors network. Other research studied the role of social media attention on NFT trends [23], and the financial advantage that experienced users gain in the NFT trading context [24].

C. Graph Evolution Rules - GERs

Our approach for describing the growth of networks related to Web3 systems is mainly rooted in graph evolution rules. Graph evolution rules mining is a frequency-based pattern discovery method that allows analyzing the evolution of temporal networks over time. The goal of graph evolution rules (GERs) is to discover frequent local changes occurring repeatedly throughout the network evolution [25]. Following the *association rules* concept belonging to the data mining field, GERs are composed of a precondition (called *body*) and a postcondition (called *head*). The rules' interpretation is that a subgraph that matches the body will probably evolve into the head, making the outcomes human-readable and explainable. For instance, Fig. 1 shows a representation of a graph evolution rule that indicates the presence of triadic closure.

Graph evolution rules are a powerful method that can help reveal complex mechanisms in temporal networks. Moreover, they enable the development of more accurate network evolution models for predicting future network changes. The set of graph evolution rules of a network can also be used to distinguish it from other graphs, whose evolution is governed by different mechanisms. The state-of-the-art methods that are focused on detecting the topological evolutionary mechanisms of a network share the same two steps methodology: first, they extract rules via frequent subgraph mining, and then they filter the output using quantities such as the support and/or confidence measures. In the literature about graph evolution rules, one of the first methods is GERM, developed by Berlingerio *et al.* [25]. Rules identified by GERM detect undirected edge insertion events, considering the relative time differences. Edge removals and node and edge relabeling are not captured. Another rule mining algorithm was proposed by Leung *et al.* [26] and further adopted by Ozaki *et al.* [27]. The rules detected are called LFR (Link Formation Rule), and their aim is to capture how directed links between a source and a destination create. Both GERM and LFR algorithms used the minimum image-based support [28] and Gspan Frequent Subgraph mining [29]. Ozaki *et al.* [27] have proposed an undirected version of LFR, along with a method to find relationships between rules. Moreover, Vakulík [30] has developed a method, called DGR miner, whose evolution rules capture also edge deletion and relabeling. Lastly, EvomMine [31] shares the same idea of DGR, allowing more advanced evolution patterns than the simple edge insertion. Furthermore, EvoMine's authors have also proposed a novel type of support measure. In this paper, we chose EvoMine to detect evolution rules because it is the most complete one, and offer an alternative type of support measure. Other works on the identification of evolution rules can be found in the literature, however, they focus more on the evolution of attributes, ignoring [32] or giving less importance [33] to the structural evolution of the networks and to the rules driving their growth.

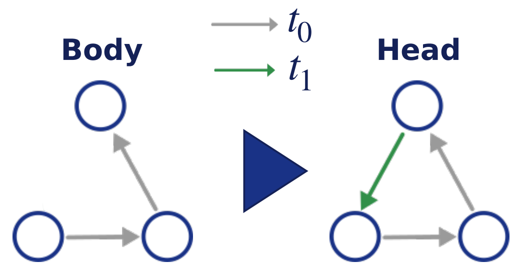


Fig. 1. Example of graph evolution rule. On the left a graph with two links - grey arrows - created at time t_0, while on the right the head of the rule where the graph on the left evolves by adding the new link - green arrow - at the successive timestamp t_1.

III. Datasets

We conducted our analysis on datasets that represent the two main trends in Web3 platforms. On one side we deal with the blockchain online social network Steemit, an example of converting the online services of the current social media into applications for the Web3 world. On the other side, we analyzed an example of platforms and assets which are peculiar to the Web3 paradigm, since they required characteristics of the blockchain technology: NFTs. In the following, we illustrate the details of the datasets chosen for the analysis.

A. Steemit dataset

The Steemit dataset records every user's actions, called operations, with a three-second granularity. The specific APIs allow the gathering of all operations as transactions, organized into three macro categories, as detailed in Guidi *et al.* [7]: *social, financial and management*. However, here we focus on two specific types: *follow* and *transfer*, which represent respectively the most common type of social and financial operations. Intuitively, each follow operation (u, v, t) records that user u has started to follow user v at timestamp t. On the other hand, every transfer operation (u, v, a, t) states that user u transferred an amount a of cryptocurrency to user v at timestamp t. In this study, we considered follow and transfer operations of the first 3-month period, due to computational constraints. Specifically, transactions present a daily timestamp from December, 1 2016 to March, 1 2017. The data extraction results in a total of 92803 follow operations and 42452 transfer operations.

B. NFTs sales dataset

The NFTs sales dataset aggregates NFTs trades from different marketplaces (APIs): *Cryptokitties, Atomic, Opensea, Gods-unchained*, and *Decentraland*. The data collection is composed of 6.1 million trades of 4.7 million NFTs in 160 cryptocurrencies, primarily Ethereum and WAX, covering the period from June 23, 2017 to April 27, 2021. In this study we restrict the dataset for computational constraints to a 50-days period. For the same reason, we select transactions within two markets only: OpenSea and Cryptokitties, which were the most active in the initial period. For each market, we focus on the first period with transactions: from December 1, 2017 to January 19, 2018 for Cryptokitties market and from February 4, 2018 to March 26, 2018 for OpenSea. In this way, we collected a total of 255947 transactions on the Cryptokitties market, and 23251 on OpenSea. **CryptoKitties** is one of the world's first blockchain games, where users can collect and breed cats with unique characteristics that define their appearance and traits. Born in November 2017, it is based on the Ethereum network and it registered a really quick and high peak in popularity when kitties start being sold at extremely high prices (more than $\$100,000$). The NFT exchange in this market concerns just a type of digital object, i.e. photos of kitties. **OpenSea** is the largest marketplace for NFTs, where users can exchange several types of rare digital objects, such as art, music, sports, games, and so on. It was founded in December 2017 and, like CryptoKitties, it relies on the **Ethereum** blockchain technology.

IV. METHODOLOGY

To analyze the evolution of the Web3 networks, we applied a graph evolution rules algorithm called *EvoMine* [31]. In the next sections, we are going to define how we model transactions and social operations into a graph representation. Then, we detail the EvoMine method and specifically its event-based support version.

TABLE I
DATASETS METRICS: SECOND AND THIRD COLUMNS INDICATE THE ORDER AND THE SIZE OF EACH GRAPH, RESPECTIVELY; WHILE THE LAST COLUMN SPECIFIES THE LENGTH OF THE GRAPH SEQUENCE.

Dataset	Nodes	Edges	Timestamps
Steemit Follow	11004	92803	90
Steemit Transfer	2815	42452	90
NFT Cryptokitties	58906	255947	50
NFT OpenSea	4870	23251	50

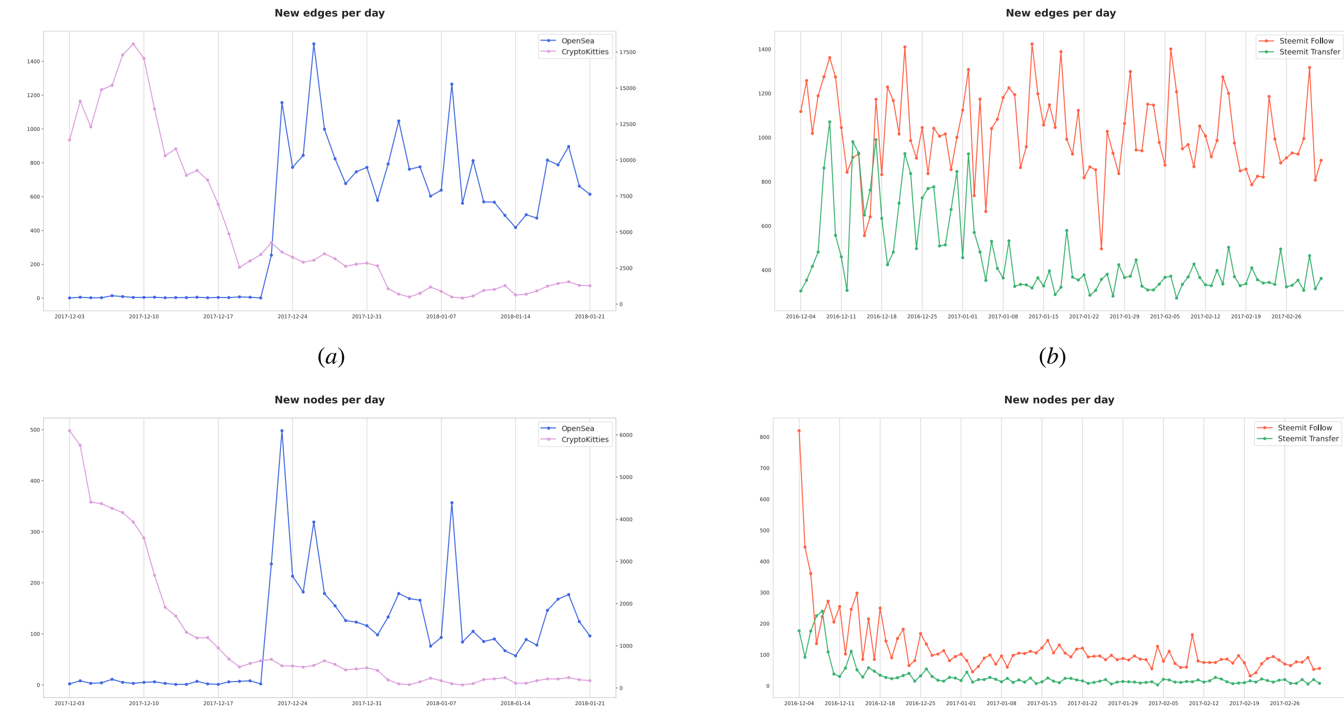
A. *Representation and modeling*

We model the transactional data gathered from Web3 platforms into directed, temporal graphs. All the four different datasets share the same operation structure: every transaction is a tuple (s, d, t), composed of a source s that performs an operation (follow, money transfer, or NFT exchange) towards a destination d at timestamp t. In the directed-temporal graphs, each transaction is translated into a directed link from source node s to destination node d with timestamp, or edge label, t. Note that the temporal graphs are modeled with a sequence of snapshots and not as a sequence of incremental graphs, to satisfy the algorithm input format requirements. Specifically, a snapshot graph for a time interval $[t_0, t_1]$ $(t_1 > t_0)$ is made up by all the links whose timestamp is between t_0 and t_1. The modeling results in four different graphs, whose size and order are reported in Table I.

As a first step to understanding the evolution of these networks, we observe the number of daily new nodes and edges, depicted in the plots of Fig. 2. Specifically, Fig. 2a and Fig. 2b show how the number of new edges changes over time; while Fig. 2c and Fig. 2d depict the growth of emerging nodes. The plot related to the NFT sales datasets (2a) highlights a change in popularity of the two markets: Cryptokitties has an initial peak, but then the number of new edges rapidly decreases, on the other hand, OpenSea presents the opposite behavior: a fast increase of the activities after the middle of December 2017. The same observations stand for the trend of daily new nodes, depicted in Fig. 2c. As regards the Steemit follow (social) and transfer graphs, they share a common trait: even if values are lower in the case of the transfer graph, the two networks show a similar trend in the number of new nodes, which reaches a certain degree of stability after initial oscillations. A difference between the evolution of the graphs emerges when observing Fig. 2b, approximately after one month, the number of new transfer edges starts a decreasing trend, while the trend of the follow graph is characterized by a higher volume of operations while keeping wide fluctuations.

B. *EvoMine*

Here we illustrate the methodological and implementation aspects of EvoMine, the algorithm for the extraction of graph evolution rules.



(*a*) (*b*) (*c*) (*d*)

Fig. 2. **Number of daily new nodes and edges.** Plots on the same column represent the same dataset (NFTs - first column, Steemit - second column), while plots on the same row represent either new edges or new nodes. In (a) and (b) the plots show how the number of new edges on a daily basis, respectively in the NFTs and Steemit datasets. On the other hand, (c) and (d) depict the number of new nodes per day, respectively in the NFTs and Steemit datasets.

Topological constraint of rules. While other state-of-the-art methods like GERM and LFR can capture only edge insertions, EvoMine rules are richer because they consider also edge deletions and node and edge relabeling. One peculiarity of EvoMine rules is that the captured evolution takes place in two consecutive timestamps so that every change developed by the postcondition occurs the immediate next timestamp with respect to the timestamp of the precondition's edges.

In the context of graph evolution rule mining, it is common to specify some constraints that describe the changes that the methods promise to detect. An EvoMine rule is described as $r : (G_{pre}, G_{post})$, and the first topological constraint states that V_{pre}, i.e. the set of nodes of G_{pre}, must correspond to V_{post}. The second condition is needed to ensure an evolution: a rule is valid if $E_{pre} \neq E_{post}$ or $\ell_{pre} \neq \ell_{post}$. In order to explain the last condition, a preliminary definition is needed. Given a sequence of graphs $G_1^T = (G_1, \ldots, G_T)$ with $G_t = (V, E_t, l_t)$, a union graph $\mathcal{G}_U(G_1^T)$ is a condensed representation of a graph sequence, composed by the same set of nodes (that remains constant in the sequence), and the union of all edges sets E_t, $\forall t = 1 \ldots T$. The labels of nodes and edges encode the temporal evolution by concatenating the labels of all timestamps included. Let us consider an example: Fig. 3b represents the union graph of the graph sequence in Fig. 3a. The encoded label of upper-left node 122 tells that the node had label 1 in the first timestamp, and label 2 in the next two snapshots. In the same way, to the edge between the two left nodes (nodes with labels 122 and 112) has been assigned the label $1\epsilon1$, which indicates that the link that was in E_1, disappeared in the second timestamp and appeared again in the third one. Once the concept of union graph is defined, the third topological constraint is easily explained. In fact, the union graph of (G_{pre}, G_{post}) must be connected, to ensure that the evolution rule captures a localized process.

The algorithm. EvoMine relies on a frequent connected subgraph mining method (Gspan [29]), applied to a specific mapping of the input graph sequence. The use of a frequent connected subgraph mining algorithm ensures that the node set and connectivity properties are achieved. In order to guarantee the constraint about edge or label changes, the method filters the resulting patterns according to the desired property. Moreover, two types of support measure can be used: *(i)* an embedding-based support (the well-known minimum image-based support), and *(ii)* an event-based support. Here, we focus on the second type of support.

Event-based support. The idea behind this novel definition is that the support of a rule corresponds to the total number of change events containing the rule itself. Before defining properly the event-based support, it is necessary to explain some preliminary concepts. In the event-based perspective, the input of the frequent subgraph mining (FSM) algorithm is a set of event graphs. Basically, an event graph is a subgraph of the

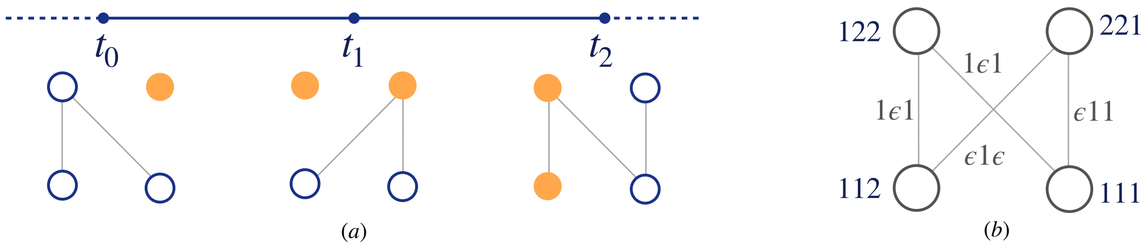


Fig. 3. **Union graph - toy example.** (a) shows a three-timestamps graph sequence, while (b) represents its union graph, where the evolution of edges and nodes is encoded in edges/nodes labels. In the edge labels, the number of characters indicates the length of the graph sequence. As for node label, each element in position i indicates the attribute of the node in timestamp i (in this case 1 or 2, indicating the node being blue or yellow respectively). Meanwhile, in the edge labels, ϵ or 1 indicate whether the edge is missing or not in the timestamp corresponding to the position of the character in the label.

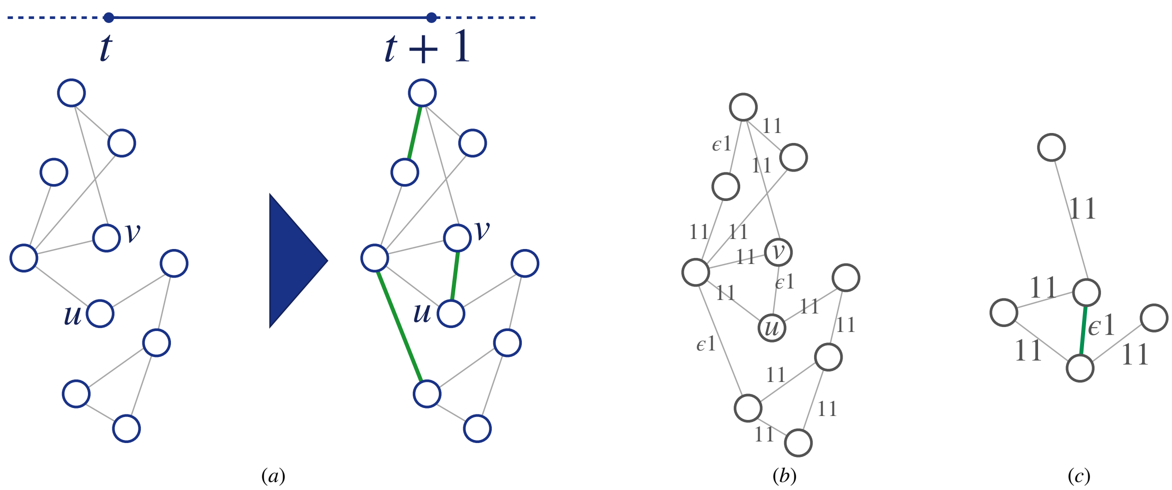


Fig. 4. **Event graph - toy example.** (a) shows two timestamps of a graph sequence, where the creation of the edge between nodes u and v is highlighted, and (b) represents its union graph. (c) is the event graph associated with the creation of (u, v), that is the subgraph of (b) induced by the neighboring nodes of u and v.

union graph $\mathcal{G}_U(G_t^{t+1})$ induced by the event neighborhood, i.e. the neighborhood of the node(s) involved in the node or edge event. For example, given the snapshots t and $t+1$ of a temporal graph $\mathcal{G}$ as the one depicted in Fig. 4a, we obtain its union graph $\mathcal{G}_U(G_t^{t+1})$ reported in Fig. 4b. If we consider the creation of the edge between u and v as the considered event, the corresponding event graph is the subgraph induced by u, v and its neighbors (depicted in Fig. 4c). The event-based support $\sigma_{event}(r)$ of a rule r is the number of event graphs in the event graph database that contains the union graph of r as subgraph.

C. GER Profiles

The goal of this study is to give a thorough analysis of the evolution rules obtained, rather than just focusing on numerical observations (like the number of rules found). To do so, we define a vector-based representation of the graph evolution rules by which we can summarize the evolutionary behavior of a network. The vector representation is called *GER profile*. This vector indicates the distribution of each kind of rule, so we first identify all the temporal subgraph isomorphism classes. Note that we worked on the union graphs of the resulting rules, so the isomorphism classes consider the topological structure but also the temporal information. After the identification of all the subgraph isomorphism classes, the vector $v(a)$ is computed for each application a of the EvoMine algorithm. Specifically, each element of the GER profile is defined as follows:

$$v_i = \frac{\sigma_{event}(r_i)}{\sum_{j=1}^{n} \sigma_{event}(r_j)} \tag{1}$$

where $\sigma_{event}(r_i)$ is the event-based support of the rule r_i and n is the number of distinct GERs identified by EvoMine. Given a temporal network, its GER profile represents a footprint of its evolution as well as a compact representation of its growth.

Distance. As an application of the GER profiles, we can exploit them to assess how the growths of two different networks follow similar evolution rules. It is possible since the GER profile is essentially a probability distribution over the space of the graph evolution rules. In this case, to measure how dissimilar the distributions are, we compute a pairwise distance for all the applications, i.e. for all the temporal networks gathered from Web3 platforms. We use the *Wasserstein* distance [34], also known as *Kantorovich–Rubinstein metric* or *Earth mover's distance*. The last name is related to the analogy that sees each distribution as a unit amount of earth and the metric as the minimum cost of turning one pile into the other (amount of earth that needs to be moved multiplied by the mean distance). Formally, the Wasserstein distance W_p of two distributions u, v is defined as follows:

$$W_p(u,v) = \left(\inf_{\pi \in \Gamma(u,v)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x-y\|^p d\pi \right)^{\frac{1}{p}}$$

where $\Gamma(u,v)$ is the set of all joint probability measures on $\mathbb{R}^d \times \mathbb{R}^d$ whose marginals are u, v.

V. Findings

We apply the EvoMine algorithm described in Section IV to the datasets described in Section III. In this section we analyze the results, giving a quantitative description of the graph evolution rules found, studying the results on the single datasets, and then comparing their GER profiles to highlight common aspects and differences.

A. Quantitative descriptions of results

EvoMine was applied to the four datasets specifying the directed nature of the graphs, a maximum number of 3 edges per evolution rule, and the absence of edge/node colors/labels. As regards the minimum support of patterns, we tried different steps of support, starting by $s = 150000$ and then decreasing until reaching a non-empty output. The chosen support values are shown in Table II, that report also the number of rules returned by the algorithm. Note that the output has been filtered in order to obtain just *meaningful rules*, namely, rules whose union graphs present edges with two distinct timestamps, so that they really describe an evolution or a growth process.

TABLE II
Support and number of rules obtained for each network.

Graph	Support	Number of GERs
Steemit Follow	50000	23
Steemit Transfer	30000	22
NFT Cryptokitties	150000	12
NFT OpenSea	10000	21

From this pure numerical observation of the number of rules obtained, we can observe that the dataset about Cryptokitties NFT market stands out with respect to the others with a lower value of rules. In general, given the thresholds chosen for the support, the number of rules describing the evolution of the Web3 networks is relatively low.

We investigate more about the difference in the different outputs by analyzing the intersections of the four sets of rules. Fig. 5 shows a graphic representation of the four sets of results, divided into two Venn diagrams to give a more intuitive idea. From the diagram on the left, we deduct that Cryptokitties results set is a subset of OpenSea one, that recursively is a subset of the Steemit Follow network. However, Steemit Transfer - a transfer network - shares almost the entire output with OpenSea - a trading network, except for one pattern that is not present in Steemit Follow either. Fig. 5 also illustrates the patterns in the difference sets between Steemit Follow and Steemit Transfer, i.e. $(steemit\_follow \setminus steemit\_transfer)$, and $(steemit\_transfer \setminus steemit\_follow)$. Note that in this work, we represent graph evolution rules as a unique temporal pattern, discerning the timestamps of edges with a gray (for timestamp t_0) or green (for timestamp t_1) color. From the figure, we can observe that the rules present only in the Steemit Follow graph (GERs in the orange rounded rectangle) suggest an instant reciprocal behavior, while the pattern in the Steemit Transfer graph (GER in the rightmost green circle) can represent an expansion-oriented pattern. For a more detailed explanation of the rule interpretation see Section VI.

B. GER Profiles

We apply the method described in Section IVC to get values that can measure the differences between evolutionary behaviors in the four different datasets. We apply the Wasserstein distance between pairs of GER profiles, obtaining the values shown in the distance matrix depicted in Fig. 6. First, the values are generally low, meaning that the distribution over the different kinds of rules is rather homogeneous. This suggests a first finding about the overall trait of the evolution processes characterizing Web3 networks: the types and the frequency of the evolution rules are quite uniform across the platforms. That indicates that in our set of Web3 platforms, there is not a manifest outlier that is driven by special evolution rules. Second, a more specific analysis of the distance matrix shows that Cryptokitties network is the one that differs the most from the other ones, a further insight that has deserved a more detailed discussion in Section VI.

VI. Discussion

In this section, we are going to propose semantic interpretations of some rules, starting from the distribution given by GER profiles described in Section IV(C). Our discussion mainly relies on Fig. 7, which represents a graphical representation of the GER profiles of the four temporal networks. The first evident feature concerns the distance from Cryptokitties GER profile with respect to others, which confirms the results highlighted by the computation of the Wasserstein distance. The other three vectors, especially the Steemit ones, are very similar, with a few exceptions. In the following paragraph, we deepen these differences and we propose a graphical representation of the patterns that correspond to the indexes where the distributions present noticeable differences.

A first evident difference concerns the first rule. In fact, rule 0 is not present in the Cryptokitties results set, and it corresponds to the closed triangle depicted in Fig. 8c and in Fig. 1. This rule expresses the classical triadic closure process typical of social networks where in the body we have an open directed triad and in the head the formation of link between the extremes of the open triad closes the triad, forming a directed triangle. Its absence can be explained by the nature of the network, i.e a trade network, and by the fact that only a single

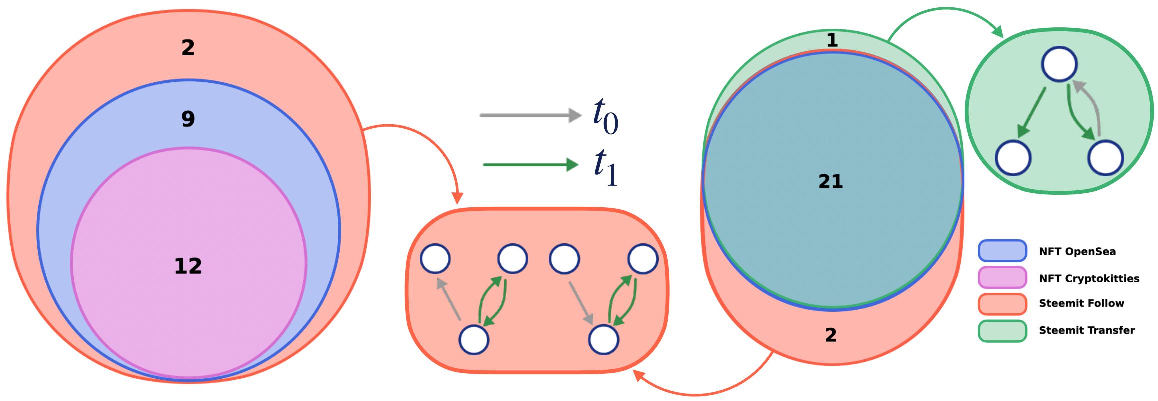


Fig. 5. Venn diagrams that show how the sets of graph evolution rules found in the four different graphs intersect with each other. From the left, the diagram shows how Cryptokitties results are a subset of OpenSea ones that in turn are completely covered by Steemit follow output. The graphic in the middle shows the two evolution rules that are present only in the Steemit follow set. The following diagram depicts the relation between OpenSea, Steemit transfer and Steemit follow. Finally, the rule on the right represents the one present only in the Steemit transfer set.

Fig. 6. Wasserstein distance matrix between GER profiles of each graph. The distance matrix has been depicted as a heatmap, where the green intensity is proportional to the distance.

type of NFT can be exchanged on this market, i.e. cats. In fact, if there is only one type of object to sell/buy, it is very uncommon to create a closed directed triangle of sales. On the contrary, it is more likely to create chains of sell operations or to have expansion-oriented behaviors, i.e. an account buys more digital assets of the same type from different sellers. This intuition is confirmed by the higher values associated with rules from 8 to 13, they all embed an expansion mechanism of the source node (node with zero in-degree). Fig. 8f shows an example of this kind of expansion, where the most left node, first creates a link towards the bottom node, and in the next timestamp (green arrow), it expands to a third node, that in turn expands to a fourth node. In this case, the rule indicates that a new chain is creating in a new direction starting from the source node. As for the triadic closure rule (Fig. 8c), it is worth noting that, while in a social network as Steemit Follow it is a quite expected rule, in trade and transfer networks such as Steemit Transfer and NFT OpenSea is an unexpected trait which, especially in a transfer network, may deserve further investigations as it might be linked to malicious actions.

Even rules 6, 7, corresponding to Fig. 8d-e, present support equal to zero in the Cryptokitties scenario. Here, the head of the rule is chain of transfers between wallets or users which originates from a single link (gray arrow) making the resulting chain. The absence of this type of chains might be related to a problem in the time granularity. Note that the graphs are built aggregating all transactions performed on the same day, and EvoMine algorithm can only catch evolution rules between consecutive timestamps. So, the evolution rules discovered highlight the evolution that happens on consecutive days. These chains may happen and be frequent in the graph but within the same day or on non-consecutive days. In fact, rules 6 and 7 generate chains of sell actions, which are likely in trade networks but they may actualize in more than a day, especially when there is only a type of object to be sold.

Finally, rule 15 marks a checkpoint from which the distribution of all graphs but Cryptokitties lose their common trend, up to rule 20. For example, rules 17 and 20 (respectively Fig. 8a-b) reflect an instant reciprocal behavior, because in both cases there is an initial link between two nodes (gray edge), and one of them creates a link with another node, that reciprocates it in the same timestamp (two green arrows). The cited rules are not frequent in the three economical/trading networks (NFTs sales and Steemit Transfer), but are present in the most common rules for the only social network considered (Steemit Follow). This suggests that reciprocal behavior is less common in transfer and economical networks with respect to social networks, especially considering the daily granularity of the outcomes.

VII. Conclusions

Blockchain-based platforms and online services are the backbone of the Web3 paradigm, one of the candidates for guiding the evolution of the future Web. Given the important

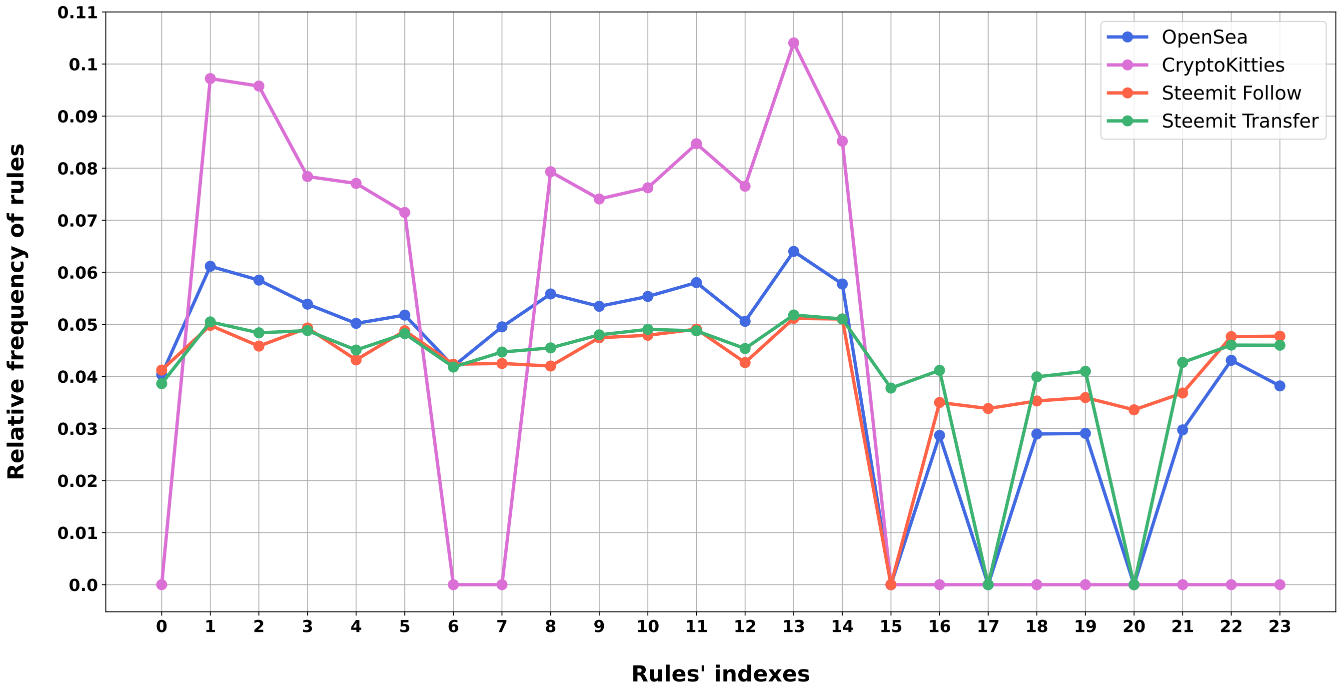


Fig. 7. **GER profiles for each Web3 network.** The plot shows, for each graph, the relative frequency of each kind of rule identified. Specifically, the x-axis specifies the rules indexes (from 0 to 23), while on the y-axis the relative frequency of each rule over the entire graph is specified. In Section VI we discuss rules with indexes 0, 6, 7, from 8 to 13, 17, and 20.

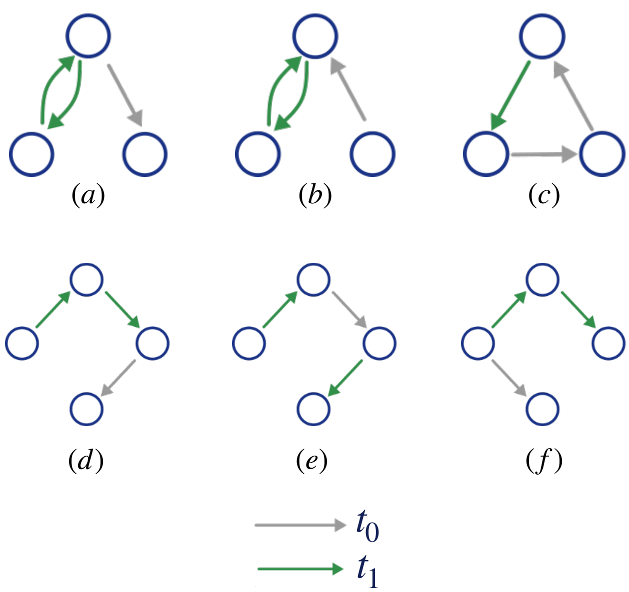


Fig. 8. Examples of graph evolution rules found. Each graph is a condensed representation of a graph evolution rule, where the gray edges belong to the *body* of the rule (precondition) and the green edges belong to the *head* (postcondition).

role Web3 platforms might have in the future, it is crucial to understand which are their specific properties, how people behave within these platforms, and how design principles inspired by decentralization and token-based economy may influence how people interact with each other and with the platform functionalities. In the case of Web3 paradigm, researchers may be facilitated in coping with these issues, since the underlying blockchains publicly offer a large volume of temporal and heterogeneous data capturing interactions that occur in these techno-social systems. In this paper we have dealt with a few types of the temporal networks generated by Web3 platforms by disentangling their rapid growth. Our methodology is based on methods rooted in frequent subgraph mining. Specifically, by the state-of-art algorithm EvoMine, we identify the most frequent graph evolution rules which capture the essential paths of growth of different blockchain-based platforms. In fact, methods based on subgraph counting are mechanism-agnostic, i.e. they do not make any assumption on the process generating the links, and return human-readable and explainable description of the network evolution, w.r.t. methods for dynamic graph representation learning. By comparing the evolution rules of social network platforms and asset trading services, we observe that GER profiles - a vector-based representation of the network evolution - are able to identify evolution mechanisms strictly related to the nature of every single platform: whereas social and token-transfer networks are characterized by rules which increase network transitivity and reciprocity, NFT trading networks, especially those specialized on a specific type of digital asset, are driven by rules which form trading chains or expand node neighborhood. From this perspective, an approach based on GER profiles may be adopted to characterize the nature of new Web3 networks, so to identify which kind of network we are observing.

The findings and the methodology presented in this work open up a few research directions which might be explored in future works. From a methodological viewpoint, the graph evolution rules returned by EvoMine are constrained to the choice of the cut-point timestamp, making the tuning of this parameter an important element for correctly identifying significant evolution rules when the link formation process is not stationary. Moreover, most of the methods for extraction of evolution rules do not provide a statistical significance of the outcomes. So, in this context, a definition of a proper null model is mandatory to evaluate the significance of the rules. On the other hand, as for the characterization of the growth of Web3 platforms, future research directions may regard the creation of an extensive dataset repository collecting temporal and heterogeneous networks from blockchain-based platforms, or a special focus on the stationarity of the evolution rules along with the entire growth of the networks.

Moreover, results suggest that the methodology explained can be leveraged in models that aim at studying the evolutionary behaviors of dynamic networks. For instance, graph evolution rules can be embedded to predict how a network will evolve or can be adopted to inform data-driven models for network evolution. Another possible employment of GERs could concern the identification of change points in the temporal version of the GER profile so as to identify whether changes in the growth dynamics and in the mechanisms leading it are occurring.

References

[1] G. Bianconi and A.-L. Barabási, "Competition and multiscaling in evolving networks," *EPL (Europhysics Letters)*, vol. 54, no. 4, p. 436, 2001.

[2] G. Bianconi, R. K. Darst, J. Iacovacci, and S. Fortunato, "Triadic closure as a basic generating mechanism of communities in complex networks," *Physical Review E*, vol. 90, no. 4, p. 042806, 2014.

[3] F. Papadopoulos, M. Kitsak, M. Serrano, M. Boguná, and D. Krioukov, "Popularity versus similarity in growing networks," *Nature*, vol. 489, no. 7417, pp. 537–540, 2012.

[4] P. Boldi, I. Crimaldi, and C. Monti, "A network model characterized by a latent attribute structure with competition," *Information Sciences*, vol. 354, pp. 236–256, 2016.

[5] J. Overgoor, A. Benson, and J. Ugander, "Choosing to grow a graph: Modeling network formation as discrete choice," in *The World Wide Web Conference*, ser. WWW '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1409–1420. [Online]. Available: https://doi.org/10.1145/3308558.3313662

[6] M. Zignani, C. Quadri, M. D. Vicario, S. Gaito, and G. P. Rossi, "Temporal communication motifs in mobile cohesive groups," in *International Conference on Complex Networks and their Applications*. Springer, 2017, pp. 490–501.

[7] B. Guidi, A. Michienzi, and L. Ricci, "Steem blockchain: Mining the inner structure of the graph," *IEEE Access*, vol. 8, pp. 210 251–210 266, 2020.

[8] T.-H. Kim, H. min Shin, H. Hwang, and S. Jeong, "Posting bot detection on blockchain-based social media platform using machine learning techniques," *ArXiv*, vol. abs/2008.12471, 2020.

[9] K. Kapanova, B. Guidi, A. Michienzi, and K. Koidl, "Evaluating posts on the steemit blockchain: Analysis on topics based on textual cues," in *Proceedings of the 6th EAI International Conference on Smart Objects and Technologies for Social Good*. EAI, 2020.

[10] R. Ciriello, R. Beck, and J. Thatcher, "The paradoxical effects of blockchain technology on social networking practices," in *Proceedings of the Thirty Ninth International Conference on Information Systems*. AIS, 12 2018.

[11] B. Guidi, "When blockchain meets online social networks," *Pervasive and Mobile Computing*, vol. 62, p. 101131, 2020.

[12] A. Kiayias, B. Livshits, A. M. Mosteiro, and O. Litos, "A puff of steem: Security analysis of decentralized content curation," *ArXiv*, vol. abs/1810.01719, 2019.

[13] U. W. Chohan, "The concept and criticisms of steemit," CBRI Working Papers: Notes on the 21st Century, Available at SSRN: http://dx.doi.org/10.2139/ssrn.3129410, 2018.

[14] M. S. Kim and J. Y. Chung, "Sustainable growth and token economy design: The case of steemit," *Sustainability*, vol. 11, no. 1, p. 167, 2019.

[15] B. Guidi, A. Michienzi, and L. Ricci, "A graph-based socioeconomic analysis of steemit," *IEEE Transactions on Computational Social Systems*, vol. PP, pp. 1–12, 12 2020.

[16] ——, "Steem blockchain: Mining the inner structure of the graph," *IEEE Access*, vol. 8, 11 2020.

[17] C. T. Ba, M. Zignani, and S. Gaito, "The role of cryptocurrency in the dynamics of blockchain-based social networks: The case of steemit," *PloS one*, vol. 17, no. 6, p. e0267612, 2022.

[18] ——, "Social and rewarding microscopical dynamics in blockchain-based online social networks," in *Proceedings of the Conference on Information Technology for Social Good*, 2021, pp. 127–132.

[19] C. T. Ba, A. Michienzi, B. Guidi, M. Zignani, L. Ricci, and S. Gaito, "Fork-based user migration in blockchain online social media," in *Proceedings of the 14th ACM conference on web science*, 2022.

[20] M. Nadini, L. Alessandretti, F. Di Giacinto, M. Martino, L. M. Aiello, and A. Baronchelli, "Mapping the nft revolution: market trends, trade networks, and visual features," *Scientific reports*, vol. 11, no. 1, pp. 1–11, 2021.

[21] K. Vasan, M. Janosov, and A.-L. Barabási, "Quantifying nft-driven networks in crypto art," *Scientific reports*, vol. 12, no. 1, pp. 1–11, 2022.

[22] M. Franceschet, "Hits hits art," *Blockchain: Research and Applications*, vol. 2, no. 4, p. 100038, 2021.

[23] A. Kapoor, D. Guhathakurta, M. Mathur, R. Yadav, M. Gupta, and P. Kumaraguru, "Tweetboost: Influence of social media on nft valuation," *arXiv preprint arXiv:2201.08373*, 2022.

[24] S. Oh, S. Rosen, and A. L. Zhang, "Investor experience matters: Evidence from generative art collections on the blockchain," *Available at SSRN*, 2022.

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

[26] C. Leung, E.-P. Lim, D. Lo, and J. Weng, "Mining interesting link formation rules in social networks," 01 2010, pp. 209–218.

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

[28] B. Bringmann and S. Nijssen, "What is frequent in a single graph?" in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2008, pp. 858–863.

[29] X. Yan and J. Han, "gspan: Graph-based substructure pattern mining," in *2002 IEEE International Conference on Data Mining, 2002. Proceedings.* IEEE, 2002, pp. 721–724.

[30] K. Vaculík, "A versatile algorithm for predictive graph rule mining." in *ITAT*, 2015, pp. 51–58.

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

[32] P. Fournier-Viger, G. He, J. C.-W. Lin, and H. M. Gomes, "Mining attribute evolution rules in dynamic attributed graphs," in *International Conference on Big Data Analytics and Knowledge Discovery*. Springer, 2020, pp. 167–182.

[33] K.-N. T. Nguyen, L. Cerf, M. Plantevit, and J.-F. Boulicaut, "Discovering inter-dimensional rules in dynamic graphs," in *Proceedings of the 1st International Conference on Dynamic Networks and Knowledge Discovery-Volume 655*, 2010, pp. 5–16.

[34] A. Ramdas, N. G. Trillos, and M. Cuturi, "On wasserstein two-sample testing and related families of nonparametric tests," *Entropy*, vol. 19, no. 2, p. 47, 2017.