Machine Learning (2025) 114:272
https://doi.org/10.1007/s10994-025-06905-y

User migration in blockchain-based online social networks through the lens of temporal node representation shift

Manuel Dileo[1] · Matteo Zignani[1]

Received: 2 April 2025 / Revised: 18 August 2025 / Accepted: 23 September 2025

Abstract
User migration in online social networks represents a critical phenomenon that can reshape platform dynamics and lead to abrupt structural changes, often triggered by technical, social, or competitive factors. In the context of blockchain-based online social networks, migration events can be particularly disruptive when triggered by hard forks - fundamental splits in the underlying blockchain protocol that create two incompatible versions of the platform. However, understanding how users adapt their behavior before, during, and after such events remains a challenging research question. To address this challenge, we rely on the framework of graph representation learning, with a particular focus on Temporal Graph Neural Networks (TGNNs). In particular, we analyze how node representations returned by TGNNs evolve during the migration event and examine how representation shifts can mirror changes in users' behavioral patterns and platform interactions. Our study focuses on Steemit, a blockchain-based social network that experienced a significant user migration following a hard fork in its supporting blockchain infrastructure. Our findings highlight that both the prediction performance and node representation are influenced by the occurrence of the migration event. We detect shifts in node representations that correspond to changes in individual user behavior throughout the event. Furthermore, group-centric analysis reveals changes in behavior and memberships among similar users during different transition periods. Additionally, we find a level of polarization in node representations caused by the migration event, which gradually diminishes over time, resulting in more evenly distributed dimensions of node representations months after the first migration. We compare our approach against two baselines based on network statistics and pre-trained LLM embeddings, showing that TGNNs better capture the distribution shift derived by the migration. To summarize, this work offers valuable insights into user behavior dynamics during platform migrations, demonstrating the effectiveness of temporal graph learning approaches in analyzing such transitions in an automated manner.

Keywords Online social networks · Blockchain-data · Temporal networks · Graph neural networks

Editors: Riccardo Guidotti, Anna Monreale, Dino Pedreschi.

Extended author information available on the last page of the article

Published online: 10 November 2025

1 Introduction

Shocking events refer to the introduction of new information originating from external phenomena or internal system changes, often causing abrupt deviations in temporal patterns (Aminikhanghahi and Cook, 2017). Examples include stock market crashes, traffic breakdowns, or the sudden outbreak of an infectious disease. In the context of online social networks (OSNs), user migration represents a particular kind of shocking event, characterized by the mass movement of users from one platform to another or between different versions of the same platform. These events, referred to as network-wide shocking events (Dileo and Zignani, 2025), can significantly influence both the characteristics of individual users and their interactions (Huang et al., 2020).

This paper focuses on blockchain-based online social networks (BOSNs), a class of web platforms where core social functions, such as posting, following, or voting, are recorded directly on-chain. While BOSNs share core social features with traditional Online Social Networks (OSNs), they differ in several key aspects. First, data availability and granularity are significantly higher, thanks to the transparent and immutable nature of blockchain ledgers. Second, platform governance and evolution are frequently decentralized, allowing for collective decisions that can fundamentally reshape the online social network, leading to shocking events. Instances of such events encompass scenarios such as hard forks, i.e., irreversible splits in the blockchain that result in the creation of a parallel platform (Ba et al., 2022), significant modifications to the consensus mechanisms or reward systems, the financial collapse of a cryptocurrency coupled with the BOSN application, and controversies involving blockchain validators or witnesses.

In particular, we investigate the user migration between two BOSNs, from Steemit to Hive, triggered by one of these shocking events: the blockchain hard fork following the acquisition of Steemit, one of the most well-known and used BOSNs. The advantages of this case study are manifold: *(i)* the exact time of the event is publicly recorded on-chain; *(ii)* user identities can be deterministically matched across the two platforms due to shared cryptographic keys; and *(iii)* rich, timestamped social interactions and textual content are available with high resolution temporal information. These conditions contrast sharply with studies on traditional OSNs, where migration must often be inferred through heuristics and incomplete data due to restricted API access (Newell et al., 2016; Senaweera et al., 2018).

While prior work has analyzed this migration event using network statistics and text analysis (Ba et al., 2022; Galdeman et al., 2023; Ba et al., 2024; Ba et al., 2023), showing changes in graph properties and user behavior, we approach the problem from a complementary perspective. Our goal is to understand, without any prior knowledge of the event and in a fully automatic manner (i.e., without relying on handcrafted metrics, data analysis, or feature engineering steps), how user interactions and discussions evolve throughout the migration. To this end, we turn to temporal graph representation learning (Kazemi et al., 2020) as a suitable framework for unfolding the dynamics of user behavior and capturing possible shifts in their latent representations to mirror the effect of the migration. Temporal Graph Neural Networks (TGNNs), in particular, extend message-passing paradigms to time-evolving graphs, allowing node representations to adapt over time, possibly reflecting changes in both topology and user features (Longa et al., 2023; You et al., 2022). In our setup, we train a state-of-the-art TGNN model to predict future links across a sequence of temporal snapshots extracted from Steemit data around the migration period, using as input both structural and textual features of users. Since the model is unaware of the shocking event, the resulting node embeddings evolve purely in response to the observed network dynamics.

This approach allows us to investigate whether, to what extent, and how the latent space learned by the TGNN mirrors the impact of the migration event, thereby assessing the feasibility of representation learning as a means to automatically uncover behavioral and structural changes that may not be evident through hand-crafted features or predefined statistics. The main findings demonstrate that the occurrence of a shocking event influences both the prediction performance and the node representation of the TGNN model. Specifically, by measuring whether users engage in discussions on the same topics and maintain consistent relationships, we observe a shift in the distributions related to the similarity of node representation toward lower values as the event occurred. This change of representation over time suggests that single users change their behavior during the fork. Moreover, from a group-centric standpoint, we observe that groups of similar users change their behavior and memberships in different transition periods. In addition, we find a certain level of polarization in the node representation caused by the shocking event, but such an effect rapidly vanishes over time, and the dimensions of the node representation tend to be more equally distributed after a few months from the user migration event. We summarize our main contributions and key findings as follows:

- **Empirical analysis of node representation shifts** the study put forward different approaches inspired by research on semantic shift detection and linguistic change to track how node embeddings learned by Temporal Graph Neural Networks (TGNNs) evolve, revealing that the node representation reflects changes in the behavior of individual users, in the diversification of behavior among previously similar users during the event, and the impact of the event on users' features.
- **Evaluation of shift detection** the performance of TGNNs in detecting the shift is benchmarked against two baselines: *i)* traditional network statistics and *ii)* pre-trained language model (LLM) embeddings. The results show that TGNNs better capture the distribution shifts caused by the migration.
- **Understanding the impact of the event on the TGNN dynamics** the study observes that the migration event affects the predictive capabilities of the TGNN model on link prediction, reinforcing the idea that structural and behavioral shifts in the network significantly alter user dynamics. Moreover, the approach is validated on a period without a shocking event, highlighting that the occurrence of the user migration within the platform has consistently amplified the shift of a natural evolution process of the network. Furthermore, we conduct an ablation study to identify which TGNN component plays the greatest role in detecting the shift. Finally, we leverage surrogate explainability models to understand the importance of features to predict the evolution of the network, and so to adapt to the event.

This work extends our previous publications in the proceedings of Discovery Science Late Breaking Contributions 2024 (Dileo and Zignani, 2025). Specifically, we investigate the same case study and leverage the same machine learning pipeline to detect shifts in the user migration event. However, our previous study reports only preliminary results on the user-centric analysis. This work extends our previous contribution in several directions, including the evaluation of shift detection and baseline comparison, the validation and the ablation study of the model, the evaluation of the impact of the event on the link prediction performance, and the addition of a group-centric analysis of node representation shift. Code, data, and additional information about our approach can be found in our GitHub repository[1].

This paper is structured as follows. Section 2 reviews the main temporal graph learning methods, provides a brief introduction to the nature of blockchain-based online social networks, and discusses the related works on the user migration case study. Section 3 introduces the methodology for acquiring and analyzing a dynamic representation of nodes within an online social network affected by a shocking event. Section 4 describes the main characteristics of the case study and provides the key statistics of the acquired datasets. Section 5 describes the experimental setup and discusses the main findings of our experiments, including the baselines' comparison and the ablation study, while Section 6 outlines potential future works.

2 Background

The comprehension of the effects of user migration on the behaviors of the users - nodes - involves different methodologies and concepts from different research areas. In the following, we review the main methods of temporal graph modelling and learning. Then, we describe works related to representation or structural shift. Finally, we present related works on the platforms our social data comes from and on the phenomena under study.

Temporal graph modelling. In recent years, a growing body of work has focused on modeling dynamic or temporal graphs, particularly in domains such as online social networks or financial networks, where entities' behavior and interactions evolve continuously over time. Different approaches have emerged for representing such dynamics and have been extensively described in recent surveys (Kazemi et al., 2020; Longa et al., 2023). One common distinction lies in how time is treated: event-based (continuous-time) models process each interaction as an individual timestamped event, capturing fine-grained temporal patterns (Rossi et al., 2020); in contrast, discrete-time models aggregate interactions into snapshots over fixed time intervals (e.g., daily or monthly), allowing for a more tractable representation of the network's evolution (You et al., 2022). Another dimension concerns how the model generalizes: transductive settings assume a fixed set of nodes known during training (Pareja et al., 2019; Seo et al., 2018), while inductive models support the arrival of new nodes and connections (Rossi et al., 2020). Finally, discrete-time temporal graph models can either follow a growing approach (Goglia and Vega, 2024), where new snapshots contain all the interactions accumulated up to a certain time, or a windowed approach, where new snapshots contain only new interactions occuring in them (You et al., 2022), emphasizing short-term trends and reducing model complexity. In this work, we adopt a discrete-time, windowed approach, as it is the most commonly used and well-studied setting in the temporal graph literature (Kazemi et al., 2020). Moreover, user behavior on social platforms typically evolves smoothly over short time intervals, making fine-grained temporal resolution unnecessary (Fabbri et al., 2017). Finally, given that the migration event we study has a known timeline (Ba et al., 2022), interactions can be naturally grouped into regular snapshots that align with the structure of the disruption.

[1] https://github.com/manuel-dileo/nodeshift

Temporal graph learning. The field of temporal graph learning (TGL) aims to extract, learn, and predict information from temporal networks (Kazemi et al., 2020). One of the most studied and well-known problems in this field is the future link prediction task, which is meaningful for solving numerous issues in many application domains like fraud detection (Khazane et al., 2019), recommender systems (You et al., 2019), or online social networks (Dileo et al., 2023), and can be leveraged in a self-supervised setting to extract meaningful euclidean representation for nodes (Liu et al., 2023). The state-of-the-art solutions for future link prediction are represented by Temporal Graph Neural Networks (TGNNs), which generalize message passing to temporal graphs. Based on the way temporal networks are modeled and on the different strategies to handle temporal information on nodes and edges, several TGNNs have been proposed in the literature. The main proposed TGNNs are covered in a recent survey (Longa et al., 2023). In this work, we use state-of-the-art TGNNs to obtain self-supervised node embeddings over time to analyze a possible shift in their representation during the user migration event.

Detecting and representing changes in temporal networks. Detecting and interpreting structural or behavioral changes in evolving graphs is a longstanding challenge in the analysis of temporal networks. Early approaches focused on detecting a change point in time when an abrupt variation in temporal data happened using statistical tests on topological features or matrix factorization techniques (Peel and Clauset, 2015; Akoglu et al., 2015). More recent methods leverage graph machine learning to encode dynamic structure and detect synthetic anomalies or shifts implicitly through embedding dynamics (Ekle and Eberle, 2024). While prior work has primarily focused on detecting the timing of change points or anomalies in temporal networks, our focus is slightly different. In this work, since the event we study —the user migration on Steemit—is a well-documented event and tied to a blockchain hard fork whose exact timestamp is publicly known, we do not aim to identify when a shift occurs; rather, we seek to understand whether, to what extent, and how such an event translates into a shift in the latent space learned by a temporal GNN trained to predict network evolution, without any direct knowledge of the event itself. In this sense, our contribution emphasizes the analysis of representation dynamics and the interpretation of embedding adjustments of the model over time, an aspect that has received comparatively less attention in the existing literature.

Blockchain-based OSNs. Blockchain-based Online Social Networks (BOSNs) are web applications whose core social functionalities–such as posting, voting, and following– are supported by an underlying blockchain infrastructure that ensures transparency, persistence, and tamper-resistance of user actions. Each social operation is recorded on-chain with a high-resolution timestamp, providing a fine-grained and immutable history of interactions. While BOSNs share many similarities with traditional Online Social Networks (OSNs), including user-generated content, social links, and community dynamics, they differ in several important respects beyond the technical architecture. First, BOSNs inherently couple social behavior with economic activity, as users interact not only via content but also through native token transfers, tipping, or staking mechanisms, which are publicly traceable. Second, due to the blockchain ledger, data availability and completeness are substantially higher than in most mainstream OSNs, where API access is often limited or selectively curated. Third, the governance structure and platform evolution in BOSNs are often driven by decentralized decisions, as exemplified by fork events like the one analyzed in this study, wherein governance disputes resulted in a platform-wide user migration. These unique characteristics make BOSNs an increasingly valuable context for studying user behavior, disruption dynamics, and socio-economic interactions at scale. Among BOSNs, Steemit has emerged as one of the most prominent platforms and is widely used in recent literature (Guidi, 2021; Dileo et al., 2022; Dileo and Zignani, 2024), due to its long-standing activity, accessible blockchain data, and clear historical events, such as the fork and subsequent user migration we investigate in this work.

User migration across BOSNs. In this work, we focus on a hard fork leading to a user migration from Steemit to another blockchain-based OSN called Hive (Ba et al., 2022). The event related to the hard fork was the acquisition of Steemit by Justin Sun. This case study's main advantage is that Hive was generated through a blockchain hard fork event, so we know exactly when the user migration starts. Furthermore, user matching across the two platforms is given by design thanks to the underlying blockchain technology. This is possible because users on both platforms retain control over their cryptographic keys, and user identities are directly tied to their public blockchain accounts. After the fork, many users replicated their usernames and account histories on the new chain, enabling deterministic matching based on public keys and transaction histories recorded on both Steem and Hive blockchains. This level of user traceability is not achievable in centralized OSNs or most decentralized platforms, where user migration must be inferred through heuristics (e.g., matching usernames or activity patterns) (Newell et al., 2016) and is often limited by restricted API access. In addition, it is known when the news about Steemit's sale was made public (about one month before the fork). Finally, the publicly available high-resolution temporal data offered by the blockchain allows us to study the event in detail. This event was presented and studied in a few recent works. Ba *et al.* (Ba et al., 2022) conducted the first study on fork-based user migration, analyzing some network-based statistics before and after the event. Subsequent works focus on the role of specific users (Galdeman et al., 2023) or communities(Ba et al., 2024) on the event or predict whether a user migrates or not during the fork (Ba et al., 2023). These works have demonstrated, through data analysis techniques grounded in network science and natural language processing, that structural and textual properties of the platform exhibit changes before, during, and after the migration. In this work, we investigate this disruptive event from a complementary perspective by leveraging advanced deep learning tools, such as temporal graph representation learning and textual content embeddings. Our goal is to understand, without any prior knowledge of the event and in a fully automatic way (i.e., without relying on ad-hoc metrics or feature engineering), how user interactions and discussions evolve throughout the migration, and to what extent the learned representations reflect and mirror these dynamics.

3 Methodology

In the context of an online social network, with access to the social relationships among users and the textual content they generate before, during, and after an event such as the user migration, our objective is to address the following research question: *How are users' behavior, habits, platform discussions, and interactions influenced by disrupting events leading to user migration to other social platforms?*. In particular, we focus on node representations built by temporal graph neural networks to mirror and digest users' behaviors and evaluate the relation between the user migration phenomenon and such representations.

This section introduces the methodology for acquiring and analyzing a dynamic representation of nodes within an online social network affected by a shocking event, i.e., user migration. In short, we model social relationships and text information as a dynamic attributed graph, and we employ a temporal graph learning model, utilizing future link prediction as a self-supervised task, to derive informative node representations. These embeddings capture the node states before, during, and after the shocking event. Finally, we examine the temporal evolution of node states over time.

3.1 Data modeling and preprocessing

Graph modeling and sequence-based framework. Since we are interested in analyzing the node embeddings before, during, and after an event, i.e. in specific discrete-time intervals, we model "follow" links and text information as a dynamic attributed graph using a snapshot-based representation (You et al., 2022). Formally, we model our data as a temporal graph $\mathcal{G} = \{G_t\}_{t=1}^{T}$ where T is the number of snapshots, $G_t = (V_t, E_t, X_t)$ is a monthly snapshot graph corresponding to month t, V_t is the set of nodes, $E_t \subseteq V_t \times V_t$ is the set of interactions happened during month t, $X_t \in \mathbb{R}^{N \times d}$ is the node feature matrix related to textual content posted by users during month t, with N number of nodes, and d number of features. Finally, we cope with the future link prediction task to derive node representations, connecting them to the capability of reconstructing the network evolution. Given a snapshot graph G_t, the purpose of future link prediction is to predict which edges will appear in a successive snapshot graph G_{t+1}. Given the sequence of graph snapshots $\{G\}_{t=1}^{T}$, we rely on the experimental setting for temporal link prediction presented in (Liu et al., 2016).

- G_1 is used to retrieve the list of edges, their relative nodes, and the textual features related to the collection of documents (posts or comments) posted in $t = 1$.
- $G_{i>1}$ is obtained as an induced subgraph constrained around the nodes of G_1. Hence, new users in the system are not considered. This limitation allows for a comprehensive understanding of the evolution of a graph and its connections. Consequently, only the links formed at time i and not at time $i - 1$ are taken into account to form the positive set. Simultaneously, a set of randomly selected non-existing edges is considered, using the same seed of nodes, to form the negative set. The final dataset is obtained by combining the positive and negative sets. If $i < T$, textual features related to the collection of documents posted in the snapshot i are computed.*Initial node representation.* For constructing node feature matrix X_t, we gather a collection of posts and comments made by each user within the period t, and we employ a pre-trained large language model (LLM) to derive their Euclidean representations. Due to the inherent complexity of texts compared to short texts found on microblogging platforms, we opt for using Sentence-BERT (SBERT) (Reimers and Gurevych, 2020), which can generate semantically meaningful sentence embeddings[2]. Consequently, it becomes feasible to adhere to the homophily principle concerning the textual content (Khanam et al., 2022), as sentences that exhibit semantic similarity will be closely positioned within the vector space. To obtain the initial features for each node, we calculate the average of the document embeddings (posts and comments) on each user's collection. Formally, for each time

[2] We use the paraphrase-multilingual-MiniLM-L12-v2 SBERT multilingual model (Reimers and Gurevych, 2020) as the number of non-English content is not negligible.

interval *t*, we denote as $D_t[u]$ the set of documents (posts and comments) published by user *u* during time interval *t*. The initial node features $X_t[u]$ of *u* at time *t* corresponds to the average of its document embeddings, i.e.

$$X_t[u] = \frac{1}{|D_t[u]|} \sum_{d \in D_t[u]} \mathrm{SBERT(d)}$$

using the element-wise sum. Users with no published textual content —missing node features— in one or more time intervals have a zeros vector as initial features for the specific time intervals. At the same time, we define an alternative initial node representation based only on the "follow" interaction patterns: for each node, we compute some well-known structural and centrality indices, such as PageRank, in-degree, out-degree, average neighbor degree, and clustering coefficient, replacing them as node features instead of textual content. We consider this alternative representation to evaluate the significance of textual information in addressing the future link prediction task, as well as to emphasize the distinct contributions of relationships and text, whereas users' behavior is influenced by an important event for the social platform.

3.2 Node embedding shift detection

Discrete-time graph neural networks. We obtain temporal node representations relying on ROLAND (You et al., 2022), the state-of-the-art model for discrete-time dynamic graph learning. We summarize the forward computation of ROLAND in Algorithm 1. The two main ROLAND innovations are *a*) the introduction of hierarchical node states for node embeddings at different GNN layers; and *b*) the recurrent update over time of the node states through an embedding module. The recurrent update is executed by the *UPDATE* function at line (4). It can be any function that takes the previous ($t-1$) and current (*t*) node state at a certain layer and produces a new current node state mixing the two pieces of information. The DECODER function - line (6) - is specific to the task we have to solve. In our case, it takes any candidate pair of the current snapshot and emits in output a score *s*: the more likely the link between two nodes, the higher its score[3].

Input: Dynamic graph snapshot $\mathcal{G}_t = (X_t, E_t)$, hierarchical node states H_{t-1}.
Output: Prediction y_{t+1}, updated node representation $H_t = \{H_t^{(l)}\}_{l=1}^{L}$

```
1: H_t^(0) ← X_t                                          ▷ Initialize node embeddings from G_t
2: for l = 1, ..., L do                                   ▷ L is the number of GNN layers
3:     Ĥ^(l) = GNN^(l)(X_t, E_t, H_t^(l-1))
4:     H_t^(l) = UPDATE^(l)(H_{t-1}^(l), Ĥ_t^(l))         ▷ Embedding update module
5: end for
6: y_{t+1} = DECODER(h_u^(L), h_v^(L)), ∀(u, v, ts) ∈ E, ts ∈ t
```

Algorithm 1 ROLAND - forward computation

Pipeline and model architecture. We integrate the creation of the graph snapshots, the initial node representation based on text and/or structure, and the ROLAND framework into the pipeline depicted in Fig. 1. From online social network data (details in Section 4), we gather both "follow" links and documents (posts and comments) authored by users within a

[3] For simplicity, the algorithm only reports the notation to compute the scores on the positive set.

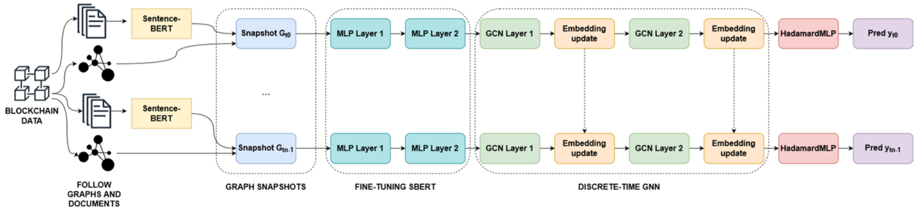


Fig. 1 Pipeline of our methodology to obtain snapshot-based node embeddings on online social network data gathered from a blockchain. At the leftmost level, we pre-process the blockchain data to obtain graph snapshots and SBERT node representations for each snapshot. Moving to the right, we employ the ROLAND framework to obtain node embeddings and incorporate a final layer specifically designed for the self-supervised link prediction task

specified period. For each snapshot, we generate by Sentence-BERT an embedding for the set of documents posted by each user. Subsequently, we construct a sequence of snapshot graphs, which are then fed into the ROLAND framework to obtain node representations for each snapshot by training the model on predicting future links. In the pipeline node states $H_t^{(l)}$ (see Algorithm 1) are fundamental since they mirror the dynamics of users' behaviors. In detail, the model includes: *i)* two MLP layers for fine-tuning the high-dimensional SBERT representations on our dataset (mint layers in Fig. 1), *ii)* a dynamic two-layer GCN based on ROLAND model design; and *iii)* an HadamardMLP as the decoder (red boxes in Fig. 1). Node embedding updates over time snapshots are implemented by GRU cells, the yellow rectangles in Fig. 1. Since in the architectural design we have stacked two GNN layer and embedding update modules, once we have trained our model, we can obtain both node states $H_t^{(0)}$ and $H_t^{(1)}$: the node embeddings obtained by combining the initial node features with the features of the nodes in the one or two-hop neighborhood, respectively. This enables us to analyze the representation shift by separately evaluating the contribution of direct neighbors (1-hop) and extended neighborhoods (2-hop), thereby gaining a more fine-grained understanding of how local and broader network contexts influence the learned representations across time. Notice that, by using a temporal GNN where initial node features are derived from textual content, the model inherently combines both semantic and structural information: the message-passing mechanism propagates and refines textual embeddings based on the graph topology, effectively aligning content-based similarity with the network's structural dynamics. This allows the model to capture how users with similar content behave differently depending on their position in the network, and vice versa.

Analyzing node embeddings over time. Once we obtain node representations for each snapshot, we are interested in analyzing how they change over time. To accomplish this, we put forward different approaches inspired by research on semantic shift detection and linguistic change (Kutuzov et al., 2018; Montariol et al., 2021; Tsakalidis et al., 2019). It is worth noting that in our scenario, the data points are not word representations but rather users' representations in an Euclidean space, which combine users' interests (expressed by the semantic embedding) and users' connectivity. Furthermore, since the node embeddings over time are obtained using the same fixed pre-trained BERT language and GNN model, they exist in the same space and thus they do not require a comparison of alignment (Kulkarni et al., 2015). Starting from the sequences of node embeddings associated with each user in the list of graph snapshots, we deal with the analysis of the node representation dynamics from two different perspectives:

1. **User-centric approach** we calculate the cosine similarity between the node representation from two months before the migration event and all subsequent node representations obtained from the monthly snapshots following the event. In this manner, we evaluate whether a user discusses the same topics and maintains consistent interactions with others before and after the user migration, i.e. whether the event influences users' behavior and habits. Moreover, to understand if the representation of a user tends to concentrate over time on certain dimensions - interests, topics, or vocabularies - or not, we compute the distributions of the interquartile range (IQR) on node embeddings for each snapshot. A polarization of the representation on a few dimensions corresponds with a high IQR, while an equally distributed representation over all the dimensions is characterized by a low IQR.
2. **Group-centric approach** for each snapshot, we cluster users according to the similarity (cosine) among their representations. To assess how groups of similar users change between two consecutive snapshots, we compute the normalized mutual information (NMI) of the partitions of the snapshots. In this manner, we obtain an overall evaluation of the influence of the network's event on posts/comments and "follow" relationships. If NMI over time is low, it implies not only a change in habits among certain users but also that users who were once considered similar are no longer deemed similar in the future, leading to divergent behaviors over time due to the emergence of new discussions or changes in the "follow" relationships. If the NMI is high, it indicates that similar users remain similar or that there is a common change in users' habits, i.e., previously similar users changed their habits together.*Quantifying the node representation shift.* Lastly, we aim to quantify the node representation shift by measuring the distance between two cosine similarity distributions, where each distribution represents the cosine similarities between the user's temporal node embeddings at two different timestamps. We chose the Earth Mover's Distance (EMD) because it provides an intuitive and stable measure of the difference between distributions (Rubner et al., 2000). It is a special case of the 1st-order Wasserstein Distance and it measures the minimum cost required to transform one distribution into another, where cost is defined as the amount of probability mass that needs to be moved, multiplied by the distance it is moved (Peyré and Cuturi, 2019). Unlike KL divergence, which is sensitive to small variations, EMD provides a more robust measure of distributional shift, making it particularly useful for comparing continuous distributions, even when they have different support or slight misalignments. Moreover, it is a metric distance, and it can work with distributions in any range of values without strictly requiring positive probability distributions. Furthermore, Wasserstein-based distance have been leveraged in previous works for comparing distributions derived from sequential and graph embeddings (Abdelwahab and Landwehr, 2022; Xu et al., 2019).

Interpreting the representation shift. Once the presence of a representation shift induced by the migration event has been established and quantitatively measured through the methods described in the previous two paragraphs, we are interested in interpreting how the model has adapted over time. In particular, we aim to uncover the mechanisms through which the temporal GNN captures the effects of the disruption and how the learned representations evolve. To this end, we focus on two complementary aspects:

- **Identifying the key architectural component in adapting to the shift** We perform an ablation study on the temporal GNN to identify which components of the architecture are most critical for detecting and adapting to the disruption. Specifically, we compare the Earth Movers Distances between the similarity distributions over the snapshots obtained when selectively removing architectural modules (e.g., temporal encoder, message-passing, features), thus isolating their contribution to capturing the shift.
- **Temporal evolution of feature importance** To assess whether the relevance of input features changes before, during, and after the event, we employ a surrogate explainability model trained to approximate the GNN's output for each snapshot (Ribeiro et al., 2016). This model takes as input the original structural and textual features and is trained to predict the GNN's output, allowing us to monitor how the contribution of different features evolves. We illustrate the methodology in Fig. 2. For each candidate pair of nodes, the surrogate model takes as input the textual and structural features of both the source and destination nodes, and is trained to reproduce the link prediction made by the temporal GNN for that pair. This process is repeated across all candidate pairs in the test set. We chose LASSO (Tibshirani, 1996) as the surrogate model due to its inherent sparsity, which enables straightforward feature selection and interpretability. Its formulation as a one-layer neural network with a linear structure and L1 regularization aligns with the architecture of decoder-only models commonly used in link prediction (Wang et al., 2022; Ruffinelli et al., 2020). This makes it a natural and interpretable choice, favoring simplicity and feature attribution over more complex surrogate models such as Random Forests. After training, we group the LASSO model's non-zero coefficients into four categories —textual and structural features for both source and destination

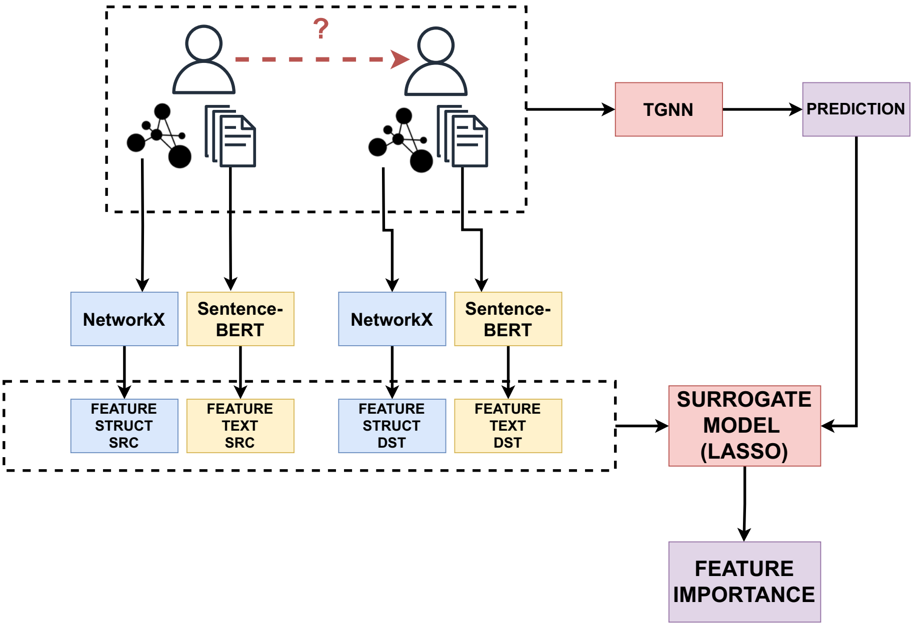


Fig. 2 Surrogate model for temporal GNN on link prediction task. For each candidate pair of nodes, the surrogate model takes as input the textual and structural features of both the source and destination nodes, and is trained to reproduce the link prediction made by the temporal GNN for that pair

nodes— and compute feature importance as the sum of the magnitudes of the learned weights in each group.

4 Dataset

We apply our methodology to data collected from Steemit, one of the widely adopted blogging social media platforms utilizing blockchain technology, which provides two types of information: *i)* the “follow” relationships (`custom_json` transactions), and *ii)* posts and comments (`comment` transactions). The former enables the extraction of timestamped links, while the latter provides access to the textual content generated by users. Data have been obtained using official APIs spanning five years from March 24, 2016, to January 21, 2021. Of particular significance is the date March 20, 2020, which marks a notable event in the form of a hard fork in the underlying blockchain. This hard fork represents an irreversible bifurcation of the blockchain resulting from a modification in the consensus protocol, triggering a substantial debate within the Steemit community, and leading to a user migration from Steemit to the new blockchain/platform Hive. Consequently, we denote this hard fork as the beginning of the user migration. Additionally, there were speculations regarding the acquisition of Steemit approximately one month before the hard fork. In blockchain systems, a block refers to a batch of transactions that are confirmed and added to the distributed ledger at regular time intervals. Each block has a unique identifier (its block number) and a timestamp. In our case, the hard fork that led to the Hive network occurred at block number 41,818,753 on the Steem blockchain. Consequently, we denote this hard fork as the beginning of the user migration. Considering this sequence of events, we focus on four specific snapshots:

1. First month (M1 or “before”): from *2020-01-20* to *2020-02-20*. It represents a “stable” period without disrupting events or rumors;
2. Second month (M2 or “during”): from *2020-02-20* to the block before the fork event. A period with rumors about the acquisition of Steemit, where the discussion on the fork started;
3. Third month (M3 or “after”): from *2020-03-20* to *2020-04-20*. A new platform - Hive - was created through the hard fork and the user could decide to migrate to the new platform, or stay in the original one, i.e. Steemit; and
4. Fourth month (M4 or “after2”): from *2020-04-20* to *2020-05-20*. Two months after the discussion about the fork started, and the shocking event on Steemit moved towards its conclusion.

Given these time windows, the event is related to the period straddling “during” (M2) and “after” (M3). In the next section, we will distinguish the two months involved in the event by referring to the period where the discussion began (M2) or during the user migration (M3). Overall, we processed 301, 653 new “follow” operations and 2, 155, 551 comment operations during the four months.

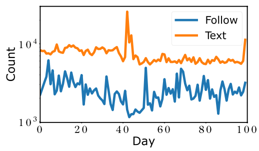
As for the textual content, we focus on the linguistic characteristics. In particular, English posts predominate the platform, accounting for approximately 75% of the total collection. However, languages such as Korean, Spanish, and Italian collectively constitute approximately 12% of the content. It is crucial to acknowledge the potential impact of these languages on the computation of sentence embeddings, as they introduce a level of noise that cannot be disregarded.

Regarding the rate of link creation and production of textual content, Fig. 3a illustrates the daily count of newly formed links and published posts throughout the observation period. While the number of new links remains relatively stable for several months, there is a significant surge in the volume of newly published posts at the onset of the discussion (M2), which occurs approximately one month after the observation period starts. An overview of a few aspects related to the dynamics of the network is reported in Table3b. The fourth months are very similar in terms of edge formation, structural measures, and content length. Indeed, the number of new links, the average degree and clustering coefficient, and the length of comments/posts in the four snapshots are almost equal. Finally, to validate our results, we also consider a "stable" period on Steemit - a period without a network shocking event - which refers to monthly snapshots from August to November 2016 (Dileo et al., 2023). We selected this interval as our control period because it corresponds to Steemit's early phase, during which daily operations (e.g. transactions, post content, social interactions) were relatively stable and not affected by abrupt variations over time.

5 Results

In this section, we present the experimental setup and the evaluation protocol employed to assess our methodology and the main findings related to the user migration event. The main objectives of our experimental evaluation are the following: *i)* understanding which behavioral and structural shifts caused by the hard fork in Steemit are mirrored in the dynamics of node embeddings returned by a TGNN, and if its performance is affected by the migration event; *ii)* validating our approach on a different period and against common baselines such as leveraging network statistics; and *iii)* identifying the key factors that allow a temporal graph learning-based solution to detect the event. To achieve this, we obtain node representations in several different ways, using the temporal GNN or simple pre-computed statistics or embeddings, including structural and/or textual signals, and considering 1 or 2-hop neighbors. We summarize labels and descriptions for the different node representations in Table 1.

Experimental setup. We evaluate our TGNN method over the future link prediction task. We use the area under the precision-recall curve (AUPRC) to evaluate the model, as suggested in (Yang et al., 2014) and adopted in prior works as well (Rossi et al., 2020). As a standard practice, we perform random negative sampling to obtain the same number of



a)

Snapshot	New links	Avg degree	Avg cluster	#Content	Avg length
M1	74,306	1.93	0.0116	415,082	79.12
M2	69,540	1.81	0.019	831,403	82.55
M3	75,241	1.95	0.017	494,195	85.75
M4	82,566	2.1	0.0188	414,874	95.5

b)

Fig. 3 In **a** the daily number of new links and published posts over time. y-axis has a logarithmic scale. We can note the peak in content production after 40 days, i.e. March 1, 2020. In Table **b** an overview of the dataset statistics

Table 1 Description of the different node representations leveraged for the experimental evaluation

Node representation	Features	Description
TGNN struct (1-hop)	Network stats. + topology	It utilizes network statistics as initial features, which are refined by a TGNN leveraging the graph topology over the 1-hop neighbourhood.
TGNN struct (2-hop)	Network stats. + topology	It utilizes network statistics as initial features, which are refined by a TGNN leveraging the graph topology over the 2-hop neighbourhood.
TGNN text (1-hop)	SBERT + topology	It utilizes the pre-computed SBERT embeddings from textual content as initial features, which are refined by a TGNN leveraging the graph topology over the 1-hop neighbourhood.
TGNN text (2-hop)	SBERT + topology	It utilizes the pre-computed SBERT embeddings from textual content as initial features, which are refined by a TGNN leveraging the graph topology over the 2-hop neighbourhood.
Baseline struct	Network stats.	Node representation given by the concatenation of network statistics.
Baseline text	SBERT	Node representation given by the pre-computed embeddings obtained through the SBERT model.

positive and negative examples in the dataset. We consider the *live-update* train-test split method (You et al., 2022), which evaluates model performance over all the available snapshots. We randomly choose 25% of edges in each snapshot to construct the validation set and determine the early-stopping condition. Code, data, and all the information about the model architecture and hyperparameters can be found in our GitHub repository. For users' clustering, we adopt the BIRCH algorithm (Zhang et al., 1996) since it efficiently scales on our dataset. In fact, the dataset consists of about 80k high-dimensional monthly points, which cause scalability issues for standard in-memory clustering algorithms. Moreover, BIRCH does not make any assumptions regarding the shapes of clusters, nor does it require prior knowledge of the number of clusters, which is particularly relevant in our case, where we do not know in advance how many clusters we need. We use the implementation available in Scikit-learn[4].

Future link prediction performance. Even if performances are not the main goal of this work in Table 2 we report the results of the TGL model for future link prediction tasks considering two GNN layers, both on a period potentially affected by the hard fork and migration event (*shock*) and on a "quiet" period (*stable*), using the structural properties - *struct* - of the network only, or the combination of structure and textual content - *text*. In this manner, we highlight the joint contribution of social relationships and textual content in enhancing the performance of the classifier.

[4] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.Birch.html, June 2023

Table 2 AUPRC of TGNN models with different combinations of features (*struct* and *text*) for future link prediction over different snapshots and averaged over time

Snapshot	shock_struct	shock_text	stable_text
Second month	0.944 ± 0.003	0.977 ± 0.004	0.915 ± 0.002
Third month	0.744 ± 0.012	0.817 ± 0.010	0.906 ± 0.005
Fourth month	0.831 ± 0.008	0.889 ± 0.007	0.868 ± 0.006
Over time	0.840 ± 0.007	0.872 ± 0.006	0.890 ± 0.003

Results are reported over 5 different runs. The word "shock" refers to models trained using data related to the user migration/fork event

In the table, we can observe how the *prediction performances are heavily influenced by the presence of the fork event*. In fact, in the third month (M3) —when the user migration occurred— the performances decreased up to 20%, while they remained pretty stable in the control period, i.e. the stable period. Furthermore, the uncertainty on the results, represented by the standard deviation, increases during the third month, and is double the one observed during the stable period. So, the *user migration has certainly changed the behavior, the representation, and the properties of nodes*, worsening the performance of the graph neural network: textual content and "follow" interactions made by users may have changed along with and after the shocking event. Finally, the outcomes in Table 2 also show that the node representation based on both relations and textual information returns better performances w.r.t. structural information only, showing it as a suitable choice to study the disruption, as it is the model that best adapts to the event. Overall, the performances are good, and they show the effectiveness of the TGNN model for the task.

Main findings First, we deal with the user-centric approach by evaluating whether a user engages in discussions on the same topics and maintains consistent interactions with others over time, as captured by its vector-based representation. In Fig. 4 we display the distributions of the cosine similarity between the node representations/embeddings over time, obtained by TGNN models. We repeat the computation considering both structural and textual information, and the 1 or 2-hop neighborhood ($l = 1$ or $l = 2$ in Algorithm 1). In the case of a representation combining social relationships and textual content, the cosine similarity between the representations of the same node decreases over time, especially between *before* and *during* the periods of the event. In fact, we observe a shift of the similarity distributions toward lower values as we move ahead from the first stable period. This change of representation over time suggests that *single users change their behavior during the fork*. The results are more evident considering 1-hop neighbors only, since textual content is directly read by followers, while reaching also the 2-hop neighbors is more difficult. On the other side, *taking into account structural information only, the distributions are quite comparable, so the similarity among the node representations over time does not decrease so much*. This evidence aligns with the findings presented in (Ba et al., 2022), where authors demonstrate that the variations of a few structural features of the social graph between the pre-fork and post-fork periods are minimal. Our results confirm that structural signals alone do not fully capture the behavioral impact of the migration event, supporting the interpretation in (Ba et al., 2022) that the overall network topology remained relatively stable, despite significant changes in user interactions and platform usage, which become more apparent when exploring meso-scale properties such as community structure (Ba et al., 2024) or focusing on individual influential users, such as hubs and opinion leaders (Galdeman et al., 2023).

Then, we look at node representation from a group-centric viewpoint. For each snapshot, we cluster users according to the similarity (cosine) among their representations through the

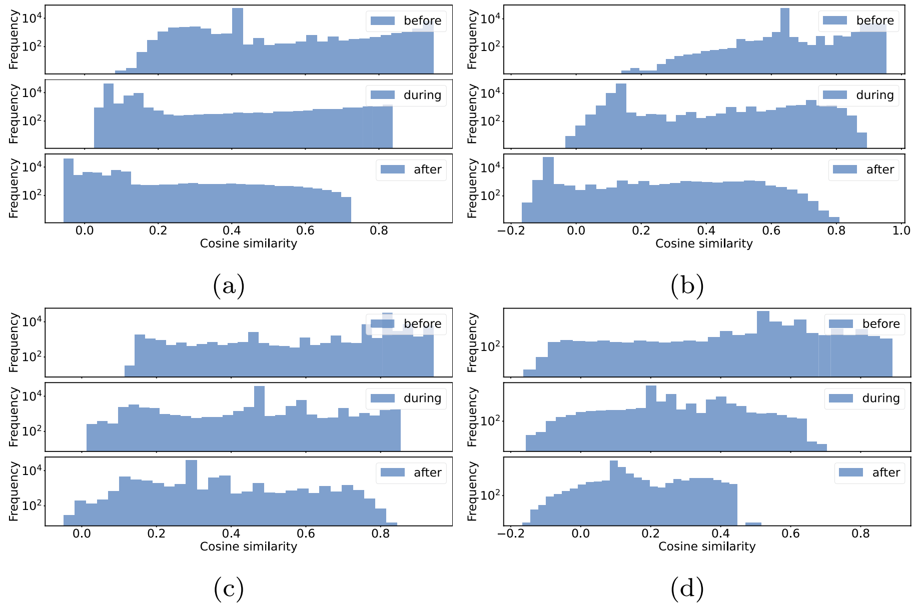


Fig. 4 Distributions of cosine similarity between node representations two months before the hard fork (stable month) and i) one month before (blue), ii) during the user migration (orange), and iii) two months after (green). **a-b** TGNN text (1-hop) and TGNN text (2-hop) node representations. **c-d** TGNN struct (1-hop) and TGNN struct (2-hop) node representations

BIRCH algorithm. Users' clustering with BIRCH returned from 7 to 10 groups depending on the type of representation and/or the neighborhood considered (1-hop or 2-hop). The worst average silhouette score over all the experiments is 0.834, so we can assess that the quality of the partition is good for all cases. Moreover, since we noted that node representations based on 2-hop neighbors are scarcely influenced by the event, we restrict the analysis to the 1-hop neighborhood. After clustering, we assess how groups of similar users change between two consecutive snapshots through the analysis of the normalized mutual information NMI. In Fig. 5 we report the heatmaps of NMI between clustering partitions over time. Each row in the matrix corresponds to the clustering in one of the periods and indicates how groups are similar w.r.t. the other periods. For instance, the first row in Fig. 5a shows that the groups of similar users in the first period - *before* - are less similar to the groups in the period during the hard fork than in periods after the shocking event. In general, when we consider a representation capturing both social and textual information, we observe that *groups of similar users change their behavior and memberships in different transition periods*: from *before* to *during* but also from *during* to *after* (the month after the beginning of the discussion on the hard fork), adopting potentially different strategies/opinions on the shocking event. On the other hand, the NMI values for the clustering on the representation based on structural information only confirm that *groups of similar users from a structural viewpoint have been less affected by the network-wide shocking event*, as previously observed in the analysis of the single user similarity. This final result offers a compelling insight into the significance of textual content in the shift of node representation. It is reasonable to assume

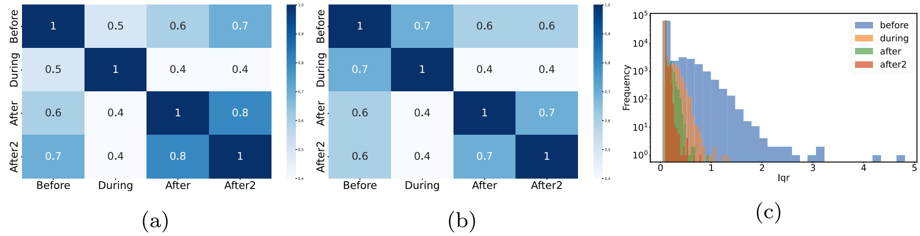
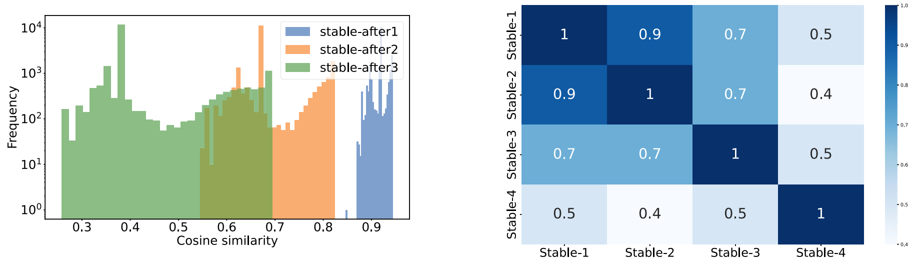


Fig. 5 **a-b** Heatmap of the normalized mutual information between user clustering assignments before, during, and after one and two months from the beginning of the discussion about the fork event. **a** TGNN text (1-hop) representations. **b** TGNN struct (1-hop) representations. **c** Distributions of IQR on node representations before, during, and after one and two months from the hard fork event. On the x-axis is the IQR, and on the y-axis its frequency. Node representations refer to TGNN text (1-hop)

that *the primary contribution to this shift, and therefore the main impact of the event, is associated with the textual content generated in the platform*.

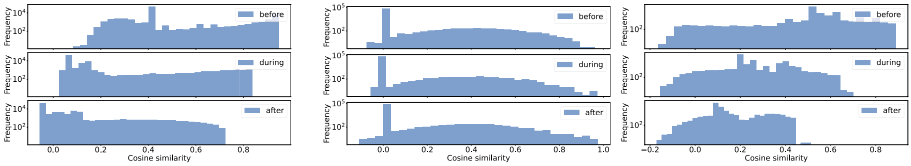
Finally, we focus on verifying if the representation of a user tends to polarize over time on certain dimensions after a shocking event. To assess if there are some dominant dimensions in the node representation of each user, we measure the Interquantile Range (IQR) of the node states. Fig. 5c shows the distributions of IQR on 1-hop node representation based on text over the different periods. In this instance, we exclude the analysis of the interquartile range on structural node states, as our primary focus lies in evaluating the degree of polarization by considering both social relationships and textual content. From the figure, we can observe that the IQR distribution abruptly decreases from *before* to *during* indicating that *there is a polarization in the node representation caused by the shocking event, but such effect rapidly vanishes over time* and the dimensions of the node representation tend to be more equally distributed. In conclusion, the response of users to the impactful event resonated within their representations; however, over time, the influence of the shock has gradually diminished.

Validation of the approach on a "stable" period. To validate our results, we conduct the same analysis over a "stable" period, i.e. a period without a network-wide shocking event, which refers to monthly snapshots from August to November 2016. By doing so, we can assess whether the observed findings regarding user migration are not merely outcomes of the network's inherent evolution. We report the cosine similarity distributions and the NMIs regarding the learned node representations over the stable period in Fig. 6. We observe both cosine similarities and NMIs decrease over time, but their diminution is smoother compared to the period of the shocking event. For instance, the cosine similarity distribution after one month in the "stable" period is concentrated around 0.9, while the same quantity for the shocking event is spread on the interval [0.3, 0.9]. Moreover, during the "stable" period, we do not observe negative similarity values between node representations in M1 and in the next three months, while in Fig. 4a cosine similarity distributions span even negative values. Finally, the NMI between the user clustering of the first period and the next two months reaches 0.9 and 0.7, respectively. Both values are higher than their migration/hard fork event counterpart, suggesting higher stability of groups of similar users over time. In conclusion, *the natural evolutionary process of the network induced a marginal shift in node representation. However, the occurrence of the user migration within the platform has consistently amplified this shift*.



(a) (b)

Fig. 6 Validation of the approach. Results on a stable period using the TGNN text (1-hop) representation



(a) (b) (c)

Fig. 7 Baselines comparison. Distributions of the cosine similarity between the node state over the user migration period, using TGNN text (1-hop) (**a**), baseline text (**b**), and baseline struct (**c**)

Baselines comparison. To highlight why leveraging temporal GNN node embeddings is profitable for detecting shifts, we compare our approach against two baselines. The first baseline uses only textual information processed by the pre-trained SBERT model as node representation, while the second leverages only topological properties summarized by the computation of statistics described in Section 3. We report the comparison of the distribution of the cosine similarities over time in Fig. 7. Results show that *raw SBERT-based embeddings do not capture relevant information from the shocking event*: the three distributions are very similar and span between zero and one. Concerning network statistics, we observe a moderate representation shift over time, especially for the highest values of similarity. However, the shift is more evident considering TGNN embeddings. To quantify the detected shift, we compute the Earth Mover's Distance between the similarities “before”-“during” and "before"-“after”. A high distance value implies a great observed shift in the distribution of embedding similarities before and during the event. The distances for TGNN-based, text-only, and graph-only embeddings are 0.30, 0.02, and 0.26, respectively. Overall, it seems *TGNN-based embeddings can capture information related to user migration that goes beyond both simple text embeddings and structural features*.

Ablation study. Given that temporal node embeddings are a more profitable choice to witness user migration than simple network statistics and textual content, we are interested in identifying which factors play the greatest role in achieving this result. To this end, we conduct an ablation study on the architecture of the temporal GNN. Specifically, we compare the embeddings obtained using the complete architecture and those without a) the UPDATE layer, b) the GNN layer, c) the current features X_t (replaced by **1**'s vectors), and d) the current topology E_t; i.e. no message-passing in the last snapshot (see Algorithm 1). We report the Earth Mover's Distances between "before"-“during” and "before"-“after” in Table 3.

Table 3 Ablation study of the temporal GNN architecture leveraged to compute node embeddings

Model	EMD
TGNN	0.30
w/o UPDATE	0.01
w/o GNN	0.09
w/o current-features	0.27
w/o current-topology	0.27

Earth Mover's Distance (EMD) is computed between the similarity distributions of node embeddings "before"-“during” and "before"-“after”

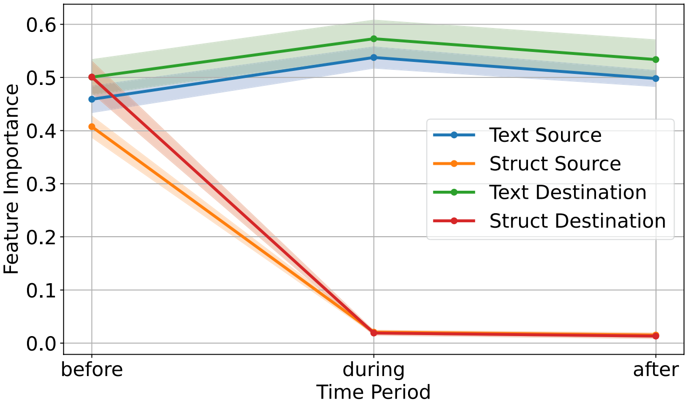


Fig. 8 Feature importance of structural and textual features of source and destination nodes before, during, and after the migration event as returned by the surrogate model for the TGN text model

Results show that the UPDATE and GNN layers play the greatest role in detecting the event since their removal exhibits the highest distance drop. In contrast, the content and relationships created during the event only have a marginal role in the distribution shift. Given that the GNN layer performs 1-hop message-passing over the “follow” relationships, *the key factor that helps a TGNN to detect the event is learning how the textual content read by the users evolves over the two months before the migration*.

Evolution of feature importance. To validate and deepen the insight that textual content read by users is the key factor to detect the event, we analyze how the importance of input features to predict new links, i.e., the evolution of the network, changes over time. Given a link prediction task, input features can be grouped into four different types: textual and structural features of source and destination nodes. We report in Fig. 8 the feature importance given by a LASSO model acting as a surrogate model for our temporal GNN architecture, grouped by the four different feature types. In the “before” period, all four feature types contribute in a relatively balanced way to the link prediction task, with comparable levels of importance. This indicates that, under stable conditions, both textual and structural signals, originating from both source and destination nodes, play a role in shaping user interactions. However, a significant shift emerges during and after the event. The importance of structural features (both source and destination; orange and red lines) sharply decreases, suggesting that traditional topological signals become less reliable for predicting user behavior when the network undergoes an abrupt change. At the same time, textual features, and in particular those of the destination node (green line), gain prominence, becoming the dominant signal in the model's predictive process. *Such a shift reinforces the role of textual content as a primary driver of behavioral adaptation in response to major platform-level events, and confirms that during such circumstances, users are particularly influenced by the content associated with the nodes they choose to interact with.*

6 Conclusions and future works

In this work, we rely on temporal node embeddings to analyze how user behavior, habits, platform discussions, and interactions are influenced by a user migration in an online social network. As a case study, we analyzed a user migration due to a hard fork event in Steemit, a novel blockchain-based online social network that allows the retrieval of validated high-resolution temporal information. To do so, we model "follow" links between users and textual content produced by users as a discrete-time dynamic attributed graph. Then, we use future link prediction as a self-supervised task for a temporal graph neural network model to obtain node representations from the dynamic graph. The embeddings represent node states before, during, and after the network-wide shocking event. Finally, we analyze how the node representations change over time.

Our results show that TGNNs can be useful for analyzing and detecting change points and node representation shifts in networked temporal data. In short, our analysis reveals that the node representation reflects changes in the behavior of individual users, the diversification of behavior among previously similar users during the event, and the event's impact on users' features. Moreover, TGNN-based embeddings can capture information related to user migration that goes beyond both simple text embeddings and structural features, thanks to the learning of graph structure and embedding update functions.

In future works, since in real applications detected shifts or anomalies potentially may result in punitive measures, the focus will be on how to produce accurate explanations for the detected change points by using explainability techniques tailored for temporal GNNs (Dileo et al., 2025). The latter point represents a big challenge in the temporal graph learning field since GNN explainability on dynamic graphs remains an open problem.

Author contributions All authors contributed to the study conception and design. Material preparation, data collection, and analysis were performed by Manuel Dileo and Matteo Zignani. The first draft of the manuscript was written by Manuel Dileo and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding This work has been partially funded by the Italian Ministry of University and Research (MUR) and the European Union – NextGenerationEU in the framework of the PRIN 2022 project "AWESOME: Analysis framework for WEb3 SOcial MEdia" – CUP: I53D23003680006.

Data availability All information about the data are available here: https://github.com/manuel-dileo/nodeshift. To guarantee the reproducibility of the results and the pseudo-anonymization, we release the input of the architecture in a tensor-based representation.

Code availability Code of our solution and the evaluation procedure is available here: https://github.com/manuel-dileo/nodeshift

Declarations

Conflicts of interest No conflicts of interest.

Ethical approval The data collected are publicly available and accessible on the Steem blockchain and on the websites reported in Section 5. We followed well-established ethical procedures for social data and obtained a waiver from the ethics committee at the University of Milan. All data is pseudo-anonymized before usage, it is stored within a secure silo, while the text is not stored in a readable format but in a vector representation.

Consent to participate Not applicable.

Consent for publication Not applicable.

References

Abdelwahab, A., & Landwehr, N. (2022). Deep distributional sequence embeddings based on a wasserstein loss. *Neural Processing Letters, 54*(5), 3749–3769. https://doi.org/10.1007/s11063-022-10784-y

Akoglu, L., Tong, H., & Koutra, D. (2015). Graph-based anomaly detection and description: A survey. *Data Mining And Knowledge Discovery, 29*(3), 626–688.

Aminikhanghahi, S., & Cook, D. J. (2017). A survey of methods for time series change point detection. *Knowledge and Information Systems, 51*(2), 339–367. https://doi.org/10.1007/s10115-016-0987-z

Ba, C. T., Dileo, M., Galdeman, A., Zignani, M., & Gaito, S. (2024). Analyzing user migration in blockchain online social networks through network structure and discussion topics of communities on multilayer networks. *Distrib. Ledger Technol.* https://doi.org/10.1145/3640020

Ba, C.T., Galdeman, A., Dileo, M., Zignani, M., & Gaito, S. (2023). User migration prediction in blockchain socioeconomic networks using graph neural networks. In: 2023 ACM Conference on Information Technology for Social Good, p. 333–341. https://doi.org/10.1145/3582515.3609552

Ba, C.T., Michienzi, A., Guidi, B., Zignani, M., Ricci, L., & Gaito, S. (2022). Fork-based user migration in blockchain online social media. In: 14th ACM Web Science Conference 2022, pp. 174–184

Dileo, M., & Zignani, M. (2024). Discrete-time graph neural networks for transaction prediction in web3 social platforms. *Machine Learning*. https://doi.org/10.1007/s10994-024-06579-y

Dileo, M., & Zignani, M. (2025). Network-wide shocking events through the lens of node representation shift. In: F. Naretto, R. Pellungrini (eds.) DS Late Breaking Contributions 2024, no. 3928 in CEUR Workshop Proceedings, pp. 09–12 . https://ceur-ws.org/Vol-3928/paper_54.pdf

Dileo, M., Ba, C. T., Zignani, M., & Gaito, S. (2022). Link prediction with text in online social networks: The role of textual content on high-resolution temporal data. In P. Pascal & D. Ienco (Eds.), *Discovery Science.* Springer Nature Switzerland.

Dileo, M., Zignani, M., & Gaito, S. (2023). Temporal graph learning for dynamic link prediction with text in online social networks. *Machine Learning*. https://doi.org/10.1007/s10994-023-06475-x

Dileo, M., Zignani, M., & Gaito, S.T. (2025). Evaluating explainability techniques on discrete-time graph neural networks. Transactions on Machine Learning Research. https://openreview.net/forum?id=JzmXo0rfry

Ekle, O. A., & Eberle, W. (2024). Anomaly detection in dynamic graphs: A comprehensive survey. *Acm Transactions On Knowledge Discovery From Data*. https://doi.org/10.1145/3669906

Fabbri, R., Fabbri, R., Antunes, D. C., Pisani, M. M., & de Oliveira, O. N. (2017). Temporal stability in human interaction networks. *Physica A: Statistical Mechanics and Its Applications, 486*, 92–105.

Galdeman, A., Zignani, M., & Gaito, S. (2023). User migration across web3 online social networks: Behaviors and influence of hubs. In: ICC 2023 - IEEE International Conference on Communications, pp. 5595–5601. 10.1109/ICC45041.2023.10278763

Goglia, D., & Vega, D. (2024). Structure and dynamics of growing networks of reddit threads. *Applied Network Science, 9*(1), 48. https://doi.org/10.1007/s41109-024-00654-y

Guidi, B. (2021). An overview of blockchain online social media from the technical point of view. *Applied Sciences, 11*(21), 9880.

Huang, S., Hitti, Y., Rabusseau, G., & Rabbany, R. (2020). Laplacian change point detection for dynamic graphs. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '20. ACM. 10.1145/3394486.3403077

Kazemi, S. M., Goel, R., Jain, K., Kobyzev, I., Sethi, A., Forsyth, P., & Poupart, P. (2020). Representation learning for dynamic graphs: A survey. *Journal Of Machine Learning Research, 21*, Article 70:1-70:73.

Khanam, K. Z., Srivastava, G., & Mago, V. (2022). The homophily principle in social network analysis: A survey. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-021-11857-1

Khazane, A., Rider, J., Serpe, M., Gogoglou, A., Hines, K., Bruss, C.B., & Serpe, R. (2019). Deeptrax: Embedding graphs of financial transactions. In: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), pp. 126–133. https://doi.org/10.1109/ICMLA.2019.00028

Kulkarni, V., Al-Rfou, R., Perozzi, B., & Skiena, S. (2015). Statistically significant detection of linguistic change. In: Proceedings of the 24th International Conference on World Wide Web, WWW '15, p. 625–635. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE. https://doi.org/10.1145/2736277.2741627

Kutuzov, A., Øvrelid, L., Szymanski, T., & Velldal, E. (2018). Diachronic word embeddings and semantic shifts: a survey. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 1384–1397. Association for Computational Linguistics, Santa Fe, New Mexico, USA. https://aclanthology.org/C18-1117

Liu, Q., Tang, S., Zhang, X., Zhao, X., Zhao, B.Y., & Zheng, H. (2016). Network growth and link prediction through an empirical lens. Proceedings of the 2016 Internet Measurement Conference

Liu, Y., Jin, M., Pan, S., Zhou, C., Zheng, Y., Xia, F., & Yu, P. S. (2023). Graph self-supervised learning: A survey. *IEEE Transactions on Knowledge and Data Engineering, 35*(6), 5879–5900.

Longa, A., Lachi, V., Santin, G., Bianchini, M., Lepri, B., Lio, P., & franco scarselli, Passerini, A. (2023). Graph neural networks for temporal graphs: State of the art, open challenges, and opportunities. Transactions on Machine Learning Research. https://openreview.net/forum?id=pHCdMat0gI

Montariol, S., Martinc, M., & Pivovarova, L. (2021). Scalable and interpretable semantic change detection. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4642–4652. Association for Computational Linguistics, Online. https://doi.org/10.18653/v1/2021.naacl-main.369

Newell, E., Jurgens, D., Saleem, H.M., Vala, H., Sassine, J., Armstrong, C., & Ruths, D. (2016). User migration in online social networks: A case study on reddit during a period of community unrest. In: ICWSM

Pareja, A., Domeniconi, G., Chen, J., Ma, T., Suzumura, T., Kanezashi, H., Kaler, T., Schardl, T.B., & Leiserson, C.E. (2019). Evolvegcn: Evolving graph convolutional networks for dynamic graphs. https://doi.org/10.48550/ARXIV.1902.10191

Peel, L., & Clauset, A. (2015). Detecting change points in the large-scale structure of evolving networks. AAAI

Peyré, G., & Cuturi, M. (2019). Computational optimal transport: With applications to data science. *Foundations and Trends®in Machine Learning, 11*(5–6), 355–607. https://doi.org/10.1561/2200000073

Reimers, N., & Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 4512–4525. ACL, Online. https://doi.org/10.18653/v1/2020.emnlp-main.365

Ribeiro, M.T., Singh, S., & Guestrin, C. (2016). why should I trust you?: Explaining the predictions of any classifier. In: KDD, pp. 1135–1144. ACM

Rossi, E., Chamberlain, B., Frasca, F., Eynard, D., Monti, F., & Bronstein, M. (2020). Temporal graph networks for deep learning on dynamic graphs. In: ICML 2020 Workshop on Graph Representation Learning

Rubner, Y., Tomasi, C., & Guibas, L. J. (2000). The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision, 40*(2), 99–121. https://doi.org/10.1023/A:1026543900054

Ruffinelli, D., Broscheit, S., & Gemulla, R. (2020). You can teach an old dog new tricks! on training knowledge graph embeddings. In: International Conference on Learning Representations. https://openreview.net/forum?id=BkxSmlBFvr

Senaweera, M., Dissanayake, R., Chamindi, N., Shyamalal, A., Elvitigala, C., Horawalavithana, S., Wijesekara, P., Gunawardana, K., Wickramasinghe, M.I.E., & Keppitiyagama, C. (2018). A weighted network analysis of user migrations in a social network. 2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer) pp. 357–362

Seo, Y., Defferrard, M., Vandergheynst, P., & Bresson, X. (2018). Structured sequence modeling with graph convolutional recurrent networks. In: Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, 2018, Proceedings, Part I 25, pp. 362–373. Springer

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B, Statistical Methodology, 58*(1), 267–288.

Tsakalidis, A., Bazzi, M., Cucuringu, M., Basile, P., & McGillivray, B. (2019). Mining the UK web archive for semantic change detection. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), pp. 1212–1221. INCOMA Ltd., Varna, Bulgaria. https://doi.org/10.26615/978-954-452-056-4_139

Wang, Y., Hooi, B., Liu, Y., Zhao, T., Guo, Z., & Shah, N. (2022). Flashlight: Scalable link prediction with effective decoders. In: B. Rieck, R. Pascanu (eds.) Learning on Graphs Conference, LoG 2022, 9-12 December 2022, Virtual Event, *Proceedings of Machine Learning Research*, vol. 198, p. 14. PMLR . https://proceedings.mlr.press/v198/wang22a.html

Xu, H., Luo, D., Zha, H., & Duke, L.C. (2019). Gromov-wasserstein learning for graph matching and node embedding. In: International conference on machine learning, pp. 6932–6941. PMLR

Yang, Y., Lichtenwalter, R. N., & Chawla, N. V. (2014). Evaluating link prediction methods. *Knowledge and Information Systems, 45*(3), 751–782. https://doi.org/10.1007/s10115-014-0789-0

You, J., Du, T., & Leskovec, J. (2022). Roland: Graph learning framework for dynamic graphs. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '22. ACM, New York, NY, USA. https://doi.org/10.1145/3534678.3539300

You, J., Wang, Y., Pal, A., Eksombatchai, P., Rosenberg, C., & Leskovec, J. (2019). Hierarchical temporal convolutional networks for dynamic recommender systems. In: L. Liu, R.W. White, A. Mantrach, F. Silvestri, J.J. McAuley, R. Baeza-Yates, L. Zia (eds.) The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, pp. 2236–2246. ACM. https://doi.org/10.1145/3308558.3313747.

Zhang, T., Ramakrishnan, R., & Livny, M. (1996). Birch: An efficient data clustering method for very large databases. *SIGMOD Record, 25*(2), 103–114. https://doi.org/10.1145/235968.233324

Authors and Affiliations

Manuel Dileo[1] · **Matteo Zignani**[1]

✉ Manuel Dileo
manuel.dileo@unimi.it

Matteo Zignani
matteo.zignani@unimi.it

[1] Department of Computer Science, University of Milan, Milan, Italy