Machine Learning (2024) 113:6395–6412
https://doi.org/10.1007/s10994-024-06579-y

Discrete-time graph neural networks for transaction prediction in Web3 social platforms

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

Received: 4 March 2024 / Revised: 24 April 2024 / Accepted: 31 May 2024 /
Published online: 25 June 2024

© The Author(s) 2024

Abstract

In Web3 social platforms, i.e. social web applications that rely on blockchain technology to support their functionalities, interactions among users are usually multimodal, from common social interactions such as following, liking, or posting, to specific relations given by crypto-token transfers facilitated by the blockchain. In this dynamic and intertwined networked context, modeled as a financial network, our main goals are (i) to predict whether a pair of users will be involved in a financial transaction, i.e. the *transaction prediction task*, even using textual information produced by users, and (ii) to verify whether performances may be enhanced by textual content. To address the above issues, we compared current snapshot-based temporal graph learning methods and developed T3GNN, a solution based on state-of-the-art temporal graph neural networks' design, which integrates fine-tuned sentence embeddings and a simple yet effective graph-augmentation strategy for representing content, and historical negative sampling. We evaluated models in a Web3 context by leveraging a novel high-resolution temporal dataset, collected from one of the most used Web3 social platforms, which spans more than one year of financial interactions as well as published textual content. The experimental evaluation has shown that T3GNN consistently achieved the best performance over time and for most of the snapshots. Furthermore, through an extensive analysis of the performance of our model, we show that, despite the graph structure being crucial for making predictions, textual content contains useful information for forecasting transactions, highlighting an interplay between users' interests and economic relationships in Web3 platforms. Finally, the evaluation has also highlighted the importance of adopting sampling methods alternative to random negative sampling when dealing with prediction tasks on temporal networks.

Keywords Graph neural networks · Temporal networks · Network analysis · Blockchain-data

Editors: Albert Bifet, Rita P. Ribeiro, Ana Carolina Lorena.

✉ Manuel Dileo
manuel.dileo@unimi.it

Matteo Zignani
matteo.zignani@unimi.it

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

1 Introduction

Financial and economic platforms built on blockchain technology are significantly transforming the existing financial systems, primarily owing to their decentralized nature, which represents a distinct paradigm compared to traditional economics and currencies. These platforms, along with the financial data they generate, can be effectively represented as financial networks, where different relationships - expressed by asset transfers and financial operations—may occur among various actors, including accounts, wallets, developers, validators, and others. By modeling blockchain-based platforms as financial networks, we can harness the potential of machine learning techniques and cope with tasks such as node classification, link classification and prediction, and graph classification (Wang et al., 2021). This enables us to better understand diverse and emerging phenomena observed within these platforms, such as fraud (Xu et al., 2021), scams (Bartoletti et al., 2018), pump-and-dump schemes (Xu et al., 2021), money laundering (Weber et al., 2019), and financial crashes (Shamsi et al., 2022). In the context of machine learning applied to financial networks, blockchain-based systems present advantages and application issues stemming from their inherent design principles:

(i) *data* are *temporal annotated* by design since operations generating relationships and features must be validated in timestamped blocks; notably, the time granularity of the data is on the order of seconds, and data may span extended periods marked by potential instability and volatility, properties that should be taken into account when designing and evaluating different solutions; and

(ii) *data* and interactions stored within blockchains tend to be *multifaceted* and *heterogeneous*. This characteristic extends beyond financial operations, encompassing social relationships and comprehensive information regarding the system's components. The data heterogeneity opens the possibility of integrating different kinds of information to tackle network-oriented tasks through diverse multimodal approaches.

These characteristics, especially data heterogeneity, find their utmost expression in Web3 social platforms (Guidi, 2021), i.e. platforms where social traits and financial aspects are strictly intertwined forming a socio-economic complex system. Web3 social platforms are social web applications whose core functions are supported by an underlying blockchain that from one side ensures the persistence and validity of operations, and on the other side introduces economic operations, such as crypto-token transfers and wealth redistribution. In these platforms, every financial operation, such as cryptocurrency exchanges, as well as social interaction, such as "follows", postings, likes, and comments, is recorded in an accessible blockchain with a high-resolution timestamp (seconds)., producing a very large set of timestamped interactions among wallets associated with the platform accounts—formally a list of triples (u, v, t), eventually accompanied by a weight w, whose meaning is application-dependent. From the viewpoint of machine learning for financial networks, Web3 social data can be used

(i) for benchmarking machine learning methods for temporal financial networks, or

(ii) for addressing node, link, or graph classification tasks in a multimodal and temporal setting, where the multimodality is related to textual content and social interactions, which can be integrated to solve financial-related tasks

In particular, among the issues addressable by machine learning methods, the *transaction prediction* task, i.e. predicting whether there will be asset transfers between pairs of accounts, is central and peculiar for these platforms since it is at the basis of forecasting their wealth, tailoring services to facilitate token transfers, and identifying active users in the economic layer.

To this aim, our study centers on the task of transaction prediction—an instance of link prediction task—using data derived from Web3 social platforms, such as textual content (posts and comments). This context poses a few application challenges especially when designing models that ask for a representation combining temporal financial data and textual content generated by social interactions. To cope with these requirements our attention is directed towards the framework of temporal graph learning (TGL) due to its aptitude for learning from temporal networks, as well as its ability to seamlessly incorporate data from text or social interactions into the models. Within this context, our primary objectives encompass three key aspects:

(i) *Benchmark and evaluation*: we evaluate existing temporal graph learning methods in solving the transaction prediction task in the novel context of Web3, wherein the textual content generated by users is incorporated as sentence embeddings;
(ii) *Methodology*: we introduce an additional model denoted as T3GNN, which integrates textual content representation into the ROLAND framework (You et al., 2022); this integration aims to handle the dynamic structure of financial transactions effectively through the live-update setting; and
(iii) *Multimodality:* a comprehensive evaluation of the impact of textual content on augmenting prediction performance; in other words, we seek to determine the usefulness of user-generated content in predicting future financial transactions

We cope with the above goals by leveraging a novel large-scale dataset with high-resolution temporal information on transactions and textual content gathered from Steemit, one of the most popular Web3 social platforms. The dataset covers the financial transactions and the textual posts of about 15K accounts over one year, posing a few technical challenges when handling the computation of the representation of high volumes of textual content. Moreover, in the evaluation protocol, we introduce more fair and robust evaluation settings based on live-update and historical negative edges as negative sampling strategy. In particular, through historical sampling we aim to make the evaluation more robust, providing models more challenging and realistic instances to classify. Indeed, the findings obtained from the experimental evaluation conducted on the Web3 dataset have yielded a few significant observations:

(i) in the evaluation setting based on live-update and historical sampling the T3GNN model outperformed other snapshot-based temporal graph learning methods previously applied to financial networks by combining into a unique representation of temporal and textual features, and, at the same time, adapting the learned representation and model in a continual learning fashion;
(ii) adopting a more fair evaluation protocol based on historical negative edges, the performances of the best-performing model have worsened as expected: this observation underscores the necessity of utilizing sampling methods other than random negative sampling to attain more realistic evaluations; and
(iii) textual content contains useful information for predicting transactions as the generated embeddings allow to obtain better performance than a pure edge-memorization

baseline. However, the integration of textual embeddings only offered marginal enhancements in prediction performance and the graph structure is crucial for forecasting links.

We can summarize our main contributions to the transaction prediction task on Web3 financial networks:

(i) we extend the set of benchmarks available for temporal graph learning by introducing a novel and large-scale high-resolution temporal and multimodal dataset gathered from an emerging blockchain-based social platform;
(ii) we train and evaluate state-of-the-art temporal graph neural networks over the transaction prediction task using fair and recently introduced protocols, such as the live-update setting and the historical negative sampling;
(iii) we propose a methodology to leverage the heterogeneity of Web3 social data by learning a representation capturing both economic and social information to predict future financial transactions, introducing a novel model architecture inspired by the ROLAND framework; and
(iv) we verify if textual content contains useful information to predict links and evaluate whether performances may be enhanced by document embeddings

The paper is organized as follows. Section 2 provides a brief introduction to the nature of blockchain-based online social networks and a review of works related to temporal graph learning for financial applications. Section 3 describes the construction of the temporal financial network, how textual features are extracted, the temporal graph neural network models for predicting future transactions, and the training and evaluation protocol. In Sect. 4 we provide a description of the dataset, the experimental setup, and all the experimental results. Finally, Sect. 5 reports the main findings of transaction prediction on Web3 social platforms and discusses potential future works.

2 Related work

Dealing with the task of transaction prediction in Web3 social networks within the framework of temporal graph learning involves methods from temporal graph neural networks for financial networks and link prediction with text, and works on the application context, i.e. blockchain-based online social networks. In the following, we describe works related to the platforms our social data comes from. Then, we review the main methods of temporal graph learning, with a special focus on integrating textual information and application on financial networks.

2.1 Blockchain-based online social networks

In the landscape of Web3 platforms, blockchain-based Online Social Networks (BOSNs) are web applications whose core functions are supported by an underlying blockchain that ensures the persistence and validity of the operations. Each "social operation" (e.g. following, voting, commenting) and "financial operation" (e.g. transfer of crypto money) is stored with a high-resolution timestamp. Since every action is recorded on a blockchain, these platforms offer an extensive data source of interactions on networks, including not only the social sphere but also the economic side. These vast collections of temporal and heterogeneous data have recently emerged as beneficial for a wide range of research fields. Most of the research studies on BOSNs have been focused on Steemit since it is one of the most popular Web3 social platforms. The most relevant advancements and issues are illustrated in a few recent works (Guidi, 2021; Ba et al., 2022a, b). With limited exceptions, the majority of previous studies have concentrated on examining the structure of financial or social relationships within BOSNs. Some of these studies have considered link dynamics (Ba et al., 2022b) and small subgraphs (Galdeman et al., 2022). However, to our knowledge, none of the prior works have addressed the task of transaction prediction, particularly in the context of incorporating both temporal links and textual content.

2.2 Temporal graph neural networks

Temporal graph neural networks (TGNNs) are deep learning models for extracting, learning, and predicting from evolving networks such as recommender systems (You et al., 2019), traffic networks (Zhao et al., 2020), or online social networks (Dileo et al., 2023). They generalize model architectures for graph neural networks (GNNs) by extending the message-passing framework (Gilmer et al., 2017) to temporal networks. Based on the way temporal networks are modeled and on the different strategies to handle temporal information on nodes and edges, several GNNs for temporal networks have been proposed in the literature. Longa et al. provides a comprehensive survey on GNNs for temporal networks (Longa et al., 2023). In the context of financial networks, several architectures based on TGNNs have been proposed to solve various tasks such as stock movement prediction, loan default risk prediction, or fraud detection (Wang et al., 2021). Financial temporal networked data extracted from blockchains have been introduced in a few works such as Kumar et al. (2018) or Weber et al. (2019). These datasets are extracted from the Bitcoin blockchain and used to evaluate models on different tasks such as fraudulent user prediction (Kumar et al., 2018), anti-money laundering (Weber et al., 2019), or transaction prediction (Pareja et al., 2020; You et al., 2022). However, the current models have been only evaluated on homogeneous financial networks based on blockchain data. In this work, we focus on more intertwined blockchain-based systems where temporal financial data are coupled with social and behavioral data.

2.3 Link prediction with text

Only a few studies have evaluated the role of textual node-related data in enhancing performances in link prediction tasks. Among these works, Xu et al. (2021) used unstructured text content from heterogeneous datasets to obtain topic-aware node embedding representations with GNNs, while in Dileo et al. (2023, 2022) we used respectively a temporal GNN with text-based features and a topic model (Blei et al., 2003) to perform “follow” link prediction in online social networks. In financial applications, textual information has been leveraged mainly for stock movement prediction (Zou et al., 2022), typically treated as a node classification task (Sawhney et al., 2020). Overall, using text to make predictions seems to improve performance and give insights into the network being studied.

To the best of our knowledge, within the aforementioned contexts, this work represents the first attempt at evaluating methods for temporal graph learning on the transaction prediction task, utilizing multimodal data from Web3 social platforms, with an emphasis on textual information.

3 Methodology

In this section, we present a comprehensive description of how financial and text data extracted from Web3 social platforms can be effectively modeled to address the transaction prediction task. Specifically, we outline the construction of a sequence of graph snapshots and elucidate our chosen approach for embedding the posts and comments published by users. Leveraging this representation, we undertake the task of transaction prediction by evaluating a selection of existing architectures for temporal graph learning, alongside a novel architecture inspired by the ROLAND model design. Furthermore, we provide a concise overview of the live-update setting employed for training and evaluating all the models. The overarching methodology devised in this study aims to assess whether user-generated content can enhance the predictive accuracy of future financial transactions, as well as to identify the most suitable model for achieving this objective.

3.1 Data modeling and problem statement

Transaction links and text information stored in blockchain-based systems can be modeled as an attributed temporal directed graph $\mathcal{G} = (V, E, T, X)$, where V is the set of users, links $(u, v, t) \in E$ denote a directed transaction link—financial interaction—from user u to user v at time t, T is the set of all possible timestamps, and X is a $|V| \times f$ matrix of node attributes related to textual content produced by users, with f the dimension of attribute vectors (Liu et al., 2023). Given a time interval $[t_0, t_1]$, the graph snapshot $\mathcal{G}_{[t_0,t_1]}$ represents a directed graph, where for each link $e = (u, v, t) \in E$, we have that $t \in [t_0, t_1]$. For simplicity, since all the edges in a certain time interval are treated as they share the same timestamp, we use the notation $\mathcal{G}_t$ to denote a graph snapshot, where t is a time interval. We chose to model data as a snapshot-based, also known as discrete-time, temporal network in agreement with recent previous works on forecasting financial networks (Gandhi et al., 2021; Shumovskaia et al., 2021; You et al., 2022).

We encoded the textual content X, by a pre-trained BERT-based language model to get a vector-based representation. In particular, we selected Sentence-BERT (SBERT) (Reimers and Gurevych, 2019) as it can derive semantically meaningful sentence embeddings and is more suitable for articulated texts, as in our case. As a result, semantically similar sentences will be close to one another in the vector space. For each time interval t, we denote as $D_{(u,t)}$ the set of documents (posts and comments) published by user u during time interval t. The initial node features $X_{(u,t)}$ of u at time t corresponds to the average of its document embeddings, i.e.

$$X_{(u,t)} = \frac{1}{|D_{(u,t)}|} \sum_{d \in D_{(u,t)}} \text{SBERT(d)} \tag{1}$$

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. Their embedding representation is then updated through standard message-passing (Gilmer et al., 2017) during the training process, together with all the other nodes.

Given a snapshot graph G_t, the goal is to predict which transactions are more likely to appear at a successive snapshot graph $\mathcal{G}_{t+1}$. The problem is known as future or dynamic link prediction and, in the context of financial networks, it can be defined as *future or dynamic transaction prediction*. We use information up to time t to predict potential edges at time $t+1$. The problem can be cast into a binary classification task, where label 1 is for existing links in the following time interval, 0 otherwise (Liben-Nowell and Kleinberg, 2003). Given a sequence of graph snapshots $[\mathcal{G}_{t_0}, ..., \mathcal{G}_{t_n}]$, we rely on the experimental setting for transductive temporal link prediction presented in Liu et al. (2016):

- $\mathcal{G}_{t_0}$ is used to retrieve the list of edges and their relative nodes, and
- $\mathcal{G}_{t_i}, i > 0$, is an induced sub-graph constrained around the nodes of G_{t_0}. This limitation makes it possible to understand how a graph and its connections effectively evolve.

In our application scenario, $\mathcal{G}_{t_0}$ represents a huge volume of historical information already happened and stored in a database, while $\mathcal{G}_{t_i}, i > 0$ are new updated information over time. Finally, when evaluating dynamic link prediction, negative edges are often randomly sampled from any node pair. Considering the sparsity of real-world graphs, the majority of node pairs are unlikely to form an edge; therefore, random negative edges are easy negative edges. For this reason, we construct the negative set using the notion of historical negative edges introduced by Poursafaei et al. (2022), i.e. edges that occurred in the previous step but are not present in the current step. This process generates a set of negative instances more challenging for the model but more robust and realistic for a fair comparison and selection.

3.2 Temporal graph neural networks

Temporal graph neural networks (TGNNs) are deep learning models for extracting, learning, and predicting from evolving networks. They generalize model architectures for graph neural networks (GNNs) by extending the message-passing framework (Gilmer et al., 2017) to temporal networks. Based on the way temporal networks are modeled and on the different strategies to handle temporal information on nodes and edges, several GNNs for temporal networks have been proposed in the literature. When comparing TGNNs models, it is important to highlight that snapshot-based and event-based models (Longa et al., 2023) are designed to handle two different kinds of temporal networks, and their training and evaluation strategies lead to two different settings (Huang et al., 2023). Hence, it is unfair to compare snapshot-based and event-based models against each other. Consequently, following the taxonomy presented in Longa et al. (2023), since we model our dataset as a discrete-time temporal network, we test four different snapshot-based TGNNs, further distinguishing two model evolution and two embedding evolution methods. Specifically, the TGNNs evaluated in this work are the following:

- **EvolveGCN-O, EvolveGCN-H** (Pareja et al., 2020). They utilize an RNN to dynamically update the weights of the internal GCNs (Kipf and Welling, 2017), which allows the GNN model to change during the test time. In EvolveGCN-H, the GCN parameters are hidden states of a recurrent architecture that takes node embeddings as input,

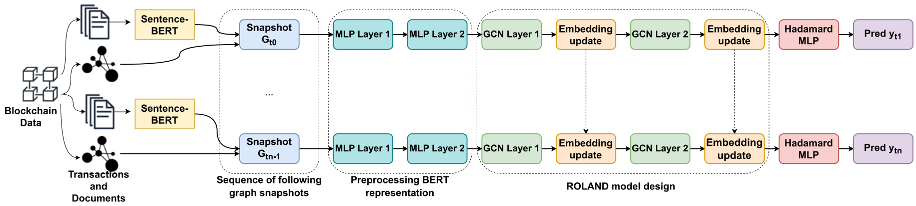


Fig. 1 Pipeline of T3GNN to perform future transaction prediction with text from a Web3 social platform. For each time interval, the method builds a snapshot of the financial network G_{t_i} and for each user, it computes the sentence embeddings of posts/comments published in the time interval. The dynamics of the network are handled by a two-layer GCN based on the ROLAND model design, preceded by two MLP layers to preprocess node features

while in EvolveGCN-O, the GCN parameters are input/outputs of a recurrent architecture. They are called model evolution methods because they only evolve the learnable parameter of a static GNN over time.

- **GCRN-GRU** (Seo et al., 2018). It is a generalization of the T-GCN model (Zhao et al., 2020), which internalizes a GNN into the GRU cell by replacing linear transformations in GRU with graph convolution operators. GCRN uses ChebNet (Defferrard et al., 2016) for spatial information and separate GNNs to compute different gates of RNNs.
- **T3GNN** (Temporal Graph Neural Network for Web3), our proposed model, based on the ROLAND model design (You et al., 2022). Figure 1 shows the pipeline of T3GNN. The architecture of our model includes: *i)* two MLP layers to preprocess the node features (high-dimensional BERT representations) and fine-tune the pre-trained text embeddings, *ii)* a simple yet effective graph augmentation technique, which consists of adding self-loops to the graph structure during the message-passing to avoid a drastically diminishing importance of the initial node features compared to neighboring features, *iii)* a dynamic two-layer GCN based on the ROLAND model design; and *iv)* an HadamardMLP (Wang et al., 2022) as decoder, typically more effective than other decoders for link prediction. ROLAND introduces two primary innovations: firstly, the view of node embeddings across various GNN layers as hierarchical node states, and secondly, the recurrent updating of these states over time using customizable embedding modules. In our model, node embeddings are updated using a one-layer Concat-MLP (You et al., 2022).

To train and evaluate all the models, we adopt the live-update setting (You et al., 2022), where models learned in previous snapshots are fine-tuned with newly observed data, by utilizing historical information and predicting future links, so capturing the evolving nature of data and models. Given the current snapshot, it is partitioned into a training and a validation set. Subsequently, the model undergoes training to minimize the binary cross-entropy loss. This training process continues until there is no further improvement in the prediction performance on the validation set, thereby satisfying the criteria for early stopping. Following this, the model's predictive performance is evaluated on the next snapshot. This procedure is systematically repeated, starting from the initial snapshot until the second last, fine-tuning the learnable parameters of the model. At the end of this process, the prediction performance of the model over time is obtained by averaging the performance over each snapshot.

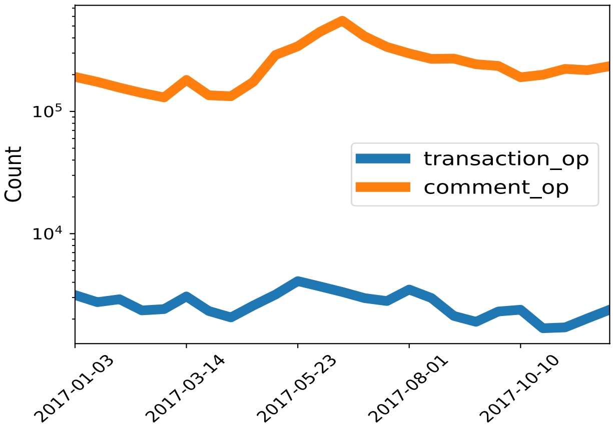


Fig. 2 Number of new comments/posts (orange line) and financial transaction (blue line) operations over the two-week snapshots. The *y*-axis is in log scale (Color figure online)

4 Experimental evaluation

In this section, we describe the experimental evaluation of temporal graph network models for the transaction prediction task on one of the most popular Web3 social platforms. Code, data, and supplementary information about the experiments can be found in our GitHub repository.[1]

4.1 Data

We perform transaction prediction using Web3 social data gathered from Steemit, one of the most popular Web3 social platforms, based on the Steem blockchain. Users on Steemit can perform many different actions, called operations. These operations, retrievable by a specific API, track users' activities with a temporal precision of 3 s, so providing high-resolution temporal data. We collected data in its early stage: from the first block on the Steem blockchain, produced on 24th March 2016, up to the end of 2017; and considered both economic and social operations, gathering two kinds of information: *(i)* transactions between users, available in the `transaction` operations; and *(ii)* post and comments written by users, available in the `comment` operations. Data related to 2016 were used to construct the initial training set, processing 274, 872 transaction operations and 241, 677 comment operations, and obtaining a snapshot graph—$\mathcal{G}_{t_0}$—with 14, 814 nodes and 39, 937 edges. Data from 2017 were collected and processed sequentially in two-week snapshots to obtain a good balance between newly executed transactions, user-generated content, and fine-grain time granularity. Figure 2 shows the number of new transactions and comment/post operations over the two-week snapshots. The number of posts and comments written by users is reasonably higher than the number of transactions but the two quantities have a quite similar trend over time.

[1] https://github.com/manuel-dileo/t3gnn

Table 1 Average AUPRC over time of the temporal graph network models for the transaction prediction task on the Steemit dataset

Model	AUPRC
EdgeBank	0,528 ± 0,0000
GCRN-GRU	0,634 ± 0,0070
EvolveGCN-O	0,706 ± 0,0005
EvolveGCN-H	0,706 ± 0,0010
T3GNN + self-loops	**0,747 ± 0,0100**

For each model, we ran experiments with 3 different random seeds, reporting the average result and standard deviation for each method

4.2 Experimental setup

We evaluated the temporal graph network models over the transaction prediction task. We used the area under the precision-recall curve (AUPRC) to evaluate models, as suggested in Yang et al. (2014) and adopted in prior works as well (Rossi et al., 2020; Pareja et al., 2020; Poursafaei et al., 2022). To highlight the learning capability of the models, we also examined the performance of EdgeBank (Poursafaei et al., 2022), a simple memorization-based baseline, which stores previously observed edges in memory, and then predicts existing edges in memory as positive at test time, that has surprisingly achieved great performance on current benchmark datasets (Huang et al., 2023). We considered historical negative edges and we adopted the live-update evaluation. We randomly chose 25% of edges in each snapshot to construct the validation set and determine the early-stopping condition. We run experiments with 3 different random seeds as in You et al. (2022), reporting the average and standard deviation of the performance measure for each model.

4.3 Results

As for the evaluation of the different models, in Table 1 we report the average performance for the transaction prediction task over the time intervals, for the temporal graph network models detailed in Sect. 3. First, the EdgeBank is outperformed by all the other TGNN models. This result suggests that in our dataset there are complex mechanisms (Ba et al., 2023) that lead to the formation of new links compared to just the repetition of past transactions with known users. Moreover, it shows that all TGNN models are learning useful information able to go beyond the pure memorization of past interactions. T3GNN achieves the best performance with an AUPRC score increase of more than 4% compared to the other baselines. Moreover, the variability of the performances in all the cases is very limited.

Thanks to the live-update setting, we can also evaluate the prediction performance of the models snapshot by snapshot. We report the performance trend of the models over the considered year in Fig. 3. The performance of EdgeBank is strictly dominated by T3GNN and EvolveGCN models. Overall, we observe a consistent level of variation from snapshot to snapshot: we move from AUPRC scores close to 0.9 to scores next to 0.4. This

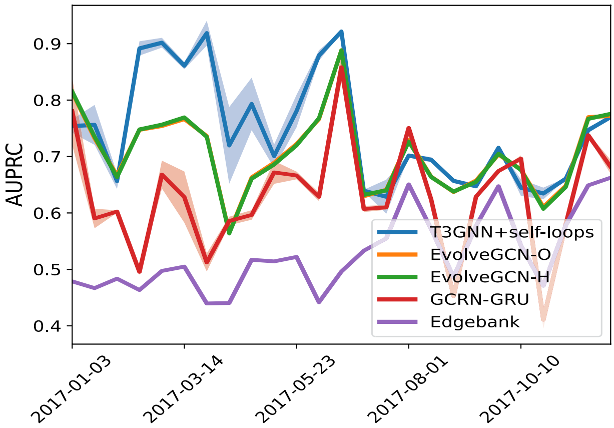
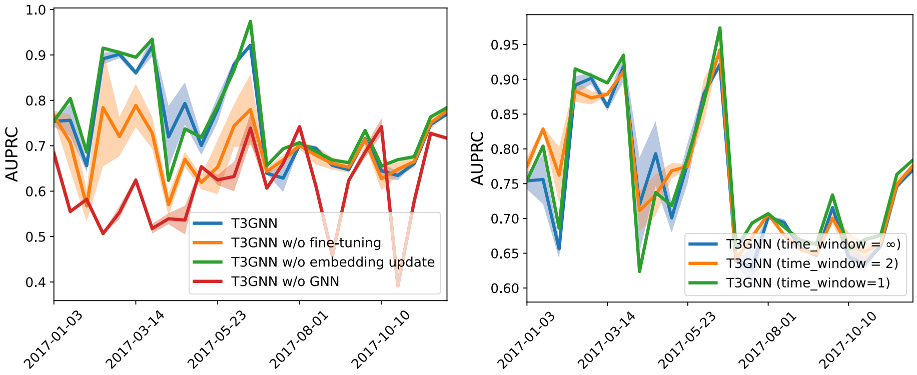


Fig. 3 Performances of temporal graph network models for the transaction prediction task over the two-week snapshots in terms of AUPRC, using live-update and historical negative links. T3GNN (blue line) performs best on most of the snapshots and does not exhibit sharp falls as in the case of GCRN-GRU (red line). The gap in the performance is more evident in the first half of the evaluation period, while in the second half, performances are comparable with EvolveGCN-X models (EvolveGCN-O and EvolveGCN-H obtained the same performances, so their lines overlap) (Color figure online)

phenomenon is much more evident in GCRN-GRU where its performance trend heavily oscillates throughout the year, showing that the model is very sensitive to changes in network evolution due to user behaviors or external financial trends (Ba et al., 2022b). In contrast, the performance trends of EvolveGCN models are less prone to sharp falls but they do not reach the performance of an evolutionary embedding model such as T3GNN. In fact, T3GNN reaches the best performances on most of the snapshots even if there is no strict dominance on the performance of the other models: better average performances in Table 1 are mainly due to the marked gap in the first half of the evaluation period, while in the second half, performances are aligned with EvolveGCN models. Overall, T3GNN seems the best candidate among the baseline models for transaction prediction on Web3 social platforms.

Ablation study. We conducted an ablation study of T3GNN by removing the following components: *(i)* the fine-tuning of the learnable parameters over time, *(i)* the GNN layers, and *(iii)* the node embedding update. Removing the node embedding update modules (e.g. a GRU cell or ConcatMLP) means that node embeddings are updated using the fine-tuning process only. We report their performance trends over time in Fig. 4a. Results show that GNN layers and fine-tuning are crucial for predicting future transactions as their removal leads to significantly worse performance for almost all the snapshots. In contrast, it seems there is no substantial gain in using node embedding update modules since their removal leads to quite a similar T3GNN performance trend. To further investigate this aspect, we run our model limiting the time window, i.e. the temporal information taken into account to obtain node embedding in a current snapshot. T3GNN, following the ROLAND model design (You et al., 2022), produces the node embeddings using the current snapshot and aggregation of the node embeddings of all the past snapshots. We symbolize this choice denoting a time window equal to infinity. However, the number of snapshots contributing to the current node embeddings could be limited to a certain time window size of *tw*. Specifically, by setting $tw = 1$ we are considering the current snapshot only, and leads to T3GNN w/o embedding update. We report the performance trends using a time window



(a) (b)

Fig. 4 **a** Ablation study of T3GNN (blue line) removing fine-tuning (orange line), the node embedding update module (green line), and the GNN layers (red line). GNN layers and fine-tuning over time are crucial for predicting future transactions. **b** Performance trends of T3GNN for different considered time windows for node embedding updates. The time window could be tuned for better performance on a specific dataset and/or period (Color figure online)

equal to one, two, and infinity in Fig. 4b. Results show that the choice of the time window leads to slightly different performances over time. Hence, the time window size could be tuned for better performance on specific transaction datasets and/or periods. Specifically, in our dataset, utilizing a time window equal to one yielded better results on the majority of the snapshots, whereas incorporating all the snapshots significantly improved performance in the third month.

The importance of sampling strategy. The generation of negative instances, i.e. random negative sampling, is often a standard procedure in training and evaluating graph neural networks. However, random negative sampling represents an over-optimistic setting, especially in very sparse networks. To highlight the necessity of a more fair negative sampling strategy, we compare the performance of T3GNN—the best-performing model—using two different sampling strategies: random negative or historical negative edges. The results in terms of AUPRC scores are depicted in Fig. 5, snapshot by snapshot. We can observe that the performance of T3GNN increases significantly using random negative edges, showing they are easy examples to classify, while the negative instances generated by historical sampling are more challenging to classify, resulting in lower scores over the entire period. This result stresses the need for a better sampling strategy for the evaluation of dynamic link prediction tasks.

Effectiveness of self-loops. Stacking several GNN layers and embedding update modules allow TGNNs to fully leverage graph and temporal information. However, the continuous update of the node embeddings through message passing and information from the past may lead the impact of the actual initial node features—the text embeddings of the current snapshot—to drastically diminish and contribute insignificantly to the final node embeddings. In the literature, some solutions based on graph augmentation, graph rewiring, and model regularization have been proposed to mitigate this problem (Rusch et al., 2023; You et al., 2020). Among the available solutions, we tested three simple solutions to enhance the contribution of the text embedding in our TGNN model:

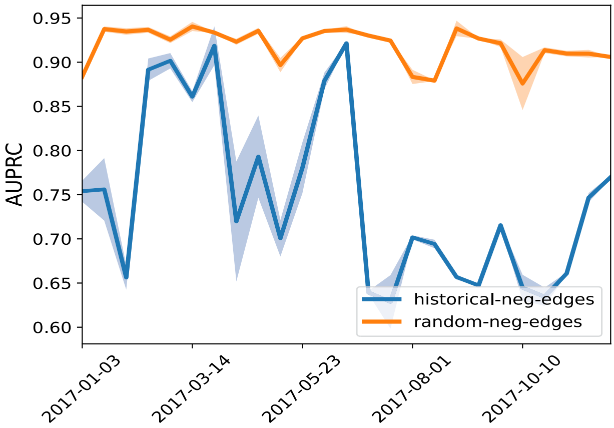


Fig. 5 Evaluation of T3GNN model for the transaction prediction task over the two-week snapshots in terms of AUPRC, using random negative edges (orange line) or historical negative edges (blue line). In the latter setting, performances are lower than in the random negative sampling setting for each two-week snapshot. It confirms that performances for random negative edges are over-optimistic based on easy examples to classify; while the historical negative sampling is a more challenging and realistic setting (Color figure online)

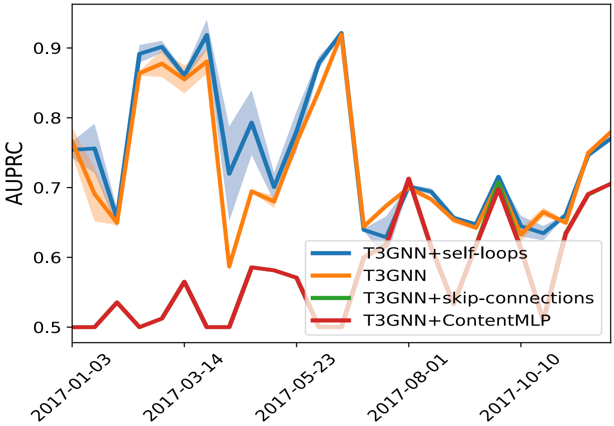


Fig. 6 Performances of temporal graph network models for the transaction prediction task over the two-week snapshots in terms of AUPRC, using live-update and historical negative links. T3GNN + self-loops (blue line) reaches the best performance on most of the snapshots, showing its effectiveness against skip-connection and ContentMLP solutions (green and red line) in enhancing the performance of a T3GNN model (orange line). Skip-connections and ContentMLP obtain the same performances, so their lines overlap (Color figure online)

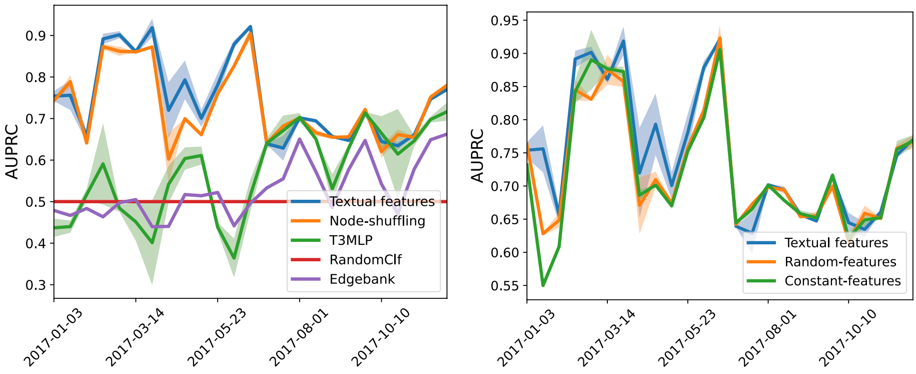
(i) The addition of self-loops, which enforce initial node features by inherently aggregating them with obtained node embedding in each GNN layer;
(ii) Skip or residual connections (You et al., 2020), which aggregate initial node features with the obtained node embedding after each embedding update module; and
(iii) ContentMLP, a solution that processes text embedding through an MLP and aggregates its output with the final obtained node embeddings

We report the performance trend of the considered solutions in Fig. 6. The results show that the addition of self-loops reaches the best performance on most of the snapshots effectively boosting the performance of a "vanilla" T3GNN model. In contrast, the solutions based on skip-connections and ContentMLP reach lower performance giving too much importance to the textual content of the current snapshot.

The role of textual content. Finally, we investigate the role of textual content on transaction prediction by comparing the results obtained (i) using SBERT textual features for node representation; (ii) shuffling the node feature matrix along the first dimension, i.e. assigning a random textual identity to each node; and (iii) replacing GNN layers in T3GNN with MLP layers—T3MLP, i.e. removing the effects of the network. We report the performance trends in Fig. 7a. In the plot, we also report the results for EdgeBank and a random classifier to better understand the role of textual content. The performance trends highlight three important phenomena. First, the graph structure is crucial for predicting future transactions between users as shuffling the node feature and leaving the network unchanged—setting (ii)—leads to performance very similar to the original ones. Then, the textual content contains useful information for predicting transactions. In fact, the performance of T3MLP, which leverages the node feature matrix only, is higher than the ones of the random classifier and the EdgeBank models, showing that using text embeddings allows discovering future transactions better than just predicting them at random or based on pure memorization. Overall, the integration of textual features increases the prediction performance of T3GNN in the first half of the period, while in the remaining snapshots, the structure is enough as the T3GNN with shuffled features reaches merely the same performances. This behavior may be related to the price trend of the Steem USD, the cryptocurrency associated with Steemit. In fact, in the first half of the period, the price was quite low, but it started to increase significantly between May and June, reaching its peak by the end of the year (Ba et al., 2022b). However, it is noteworthy to notice that T3MLP benefits from text embeddings even in the second half of the year, as it does not have access to financial information.

Besides the role of textual content in predicting future transactions, we further investigate the role of text embeddings in enhancing the performance of T3GNN. To this aim, we compare the results obtained (i) using SBERT textual features for node representation; (ii) replacing the text-based representation with random features[2]; and (iii) applying a constant encoder for node representation, i.e. no features. Recently, Sato et al. (2021) has shown that random features strengthen the power of Graph Neural Networks. Hence, an increase in performance when node features are employed could be just related to the usage of d-dimensional random feature vectors in place of constant encoders, and not related to the actual node features' content. We report the performance trends in Fig. 7b. The AUPRC scores of T3GNN models, averaged over all the snapshots, with constant, random, and textual features, are respectively 0.713, 0.713, and 0.747. Overall, leveraging textual content increases prediction performance by 3.4%. Moreover, the general trend related to the interplay between textual content and structural information described in Fig. 7a holds. Therefore, the content of the fine-tuned text embeddings plays a role in enhancing the performance of T3GNN.

[2] It is worth noting that placing random features on nodes is not equivalent to shuffling text-embeddings. Indeed, the latter setting maintains a representation coherent with the textual identity of a user while the former is not in the least connected with textual content produced by a user.



(a) (b)

Fig. 7 The role of textual content. In **a**, we report the performances for the transaction prediction task over the two-week snapshots in terms of AUPRC of T3GNN with textual features (blue line), node feature matrix shuffling (orange line), T3MLP (green line), a naive random classifier (red line), and EdgeBank (violet line). The structural information is crucial for predicting future transactions but the textual content contains useful information for this task. In **b** we report the performances of T3GNN models using SBERT textual features (blue line), random features (orange line), or constant encoding (green line). The integration of textual features increases the prediction performance in the first half of the period, while in the remaining snapshots, the random features and textual features have obtained similar AUPRC scores (Color figure online)

5 Conclusions

In conclusion, this research paper focused on the application of machine learning techniques to temporal graphs in the context of financial networks and Web3 social platforms. The study aimed to explore the transaction prediction task, incorporating both financial and textual data from Web3 platforms. Key contributions included the evaluation of temporal graph learning methods, the introduction of the T3GNN model integrating textual content, and the assessment of the impact of user-generated content on prediction performance. The study utilized a novel dataset from the Steemit platform and employed fair evaluation protocols. Results demonstrated the superior performance of T3GNN and highlighted the importance of alternative sampling methods for realistic evaluations. An extensive analysis of the performance of T3GNN shows that, despite the graph structure being crucial for making predictions, textual content contains useful information for forecasting transactions, highlighting an interplay between users' interests and economic relationships in Web3 platforms. Indeed, using text embeddings allows discovering future transactions better than just predicting them at random or based on pure memorization, and enhances the performance of T3GNN. Overall, this study extended the benchmark dataset for transaction and link prediction tasks, introduced a novel model architecture based on the ROLAND framework, and provided insights into leveraging the heterogeneity of Web3 social data for predicting future financial transactions. In future works, the heterogeneity and multimodality of Web3 data can be even more exploited by integrating the different types of social relationships as well as the roles of the different accounts in the management of the platforms.

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 Open access funding provided by Università degli Studi di Milano within the CRUI-CARE Agreement. 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.

Availability of data and material All information about the data are available here: https://github.com/manuel-dileo/t3gnn. 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 the T3GNN framework and the evaluation procedure is available here: https://github.com/manuel-dileo/t3gnn

Declarations

Conflict of interest No Conflict of interest.

Ethics approval The data collected are publicly available and accessible on the Steem blockchain and on the websites reported in Sect. 4. 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.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

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

Ba, C. T., Zignani, M., & Gaito, S. (2022b). The role of cryptocurrency in the dynamics of blockchain-based social networks: The case of steemit. *PloS one, 17*(6), e0267612.

Ba, C. T., Zignani, M., & Gaito, S. (2023). Characterizing growth in decentralized socio-economic networks through triadic closure-related network motifs. *Online Social Networks and Media, 37–38*, 100266. https://doi.org/10.1016/j.osnem.2023.100266.https://www.sciencedirect.com/science/article/pii/S2468696423000253

Bartoletti, M., Pes, B., & Serusi, S. (2018). Data mining for detecting bitcoin ponzi schemes. In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, pp. 75–84. https://doi.org/10.1109/CVCBT.2018.00014

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*. **3**, 993–1022.

Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS’16, p. 3844-3852. Curran Associates Inc.

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* (pp. 212–226). Springer Nature.

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

Galdeman, A., Zignani, M., & Gaito, S. (2022). Disentangling the growth of blockchain-based networks by graph evolution rule mining. In *2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10. IEEE.

Gandhi, A., Aakanksha, Kaveri, S., & Chaoji, V. (2021). Spatio-temporal multi-graph networks for demand forecasting in online marketplaces In ECML/PKDD (4). *Lecture Notes in Computer Science,* vol. 12978, pp. 187–203. Springer.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., & Dahl, G. E. (2017). Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning* Vol. 70, ICML'17, p. 1263-1272. JMLR.org.

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

Huang, S., Poursafaei, F., Danovitch, J., Fey, M., Hu, W., Rossi, E., Leskovec, J., Bronstein, M.M., Rabusseau, G., & Rabbany, R. (2023). Temporal graph benchmark for machine learning on temporal graphs. CoRR **abs/2307.01026**.

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SJU4ayYgl

Kumar, S., Hooi, B., Makhija, D., Kumar, M., Faloutsos, C., Subrahmanian, V. (2018) Rev2: Fraudulent user prediction in rating platforms. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 333–341. ACM.

Liben-Nowell, D., & Kleinberg, J. (2003). The link prediction problem for social networks. In: Proceedings of the Twelfth International Conference on Information and Knowledge Management, CIKM '03, pp. 556–559. Association for Computing Machinery. https://doi.org/10.1145/956863.956972

Liu, P., Guarrasi, V., & Saryuce, A. (2023). Temporal network motifs: Models, limitations, evaluation. *IEEE Transactions on Knowledge & Data Engineering, 35*(01), 945–957. https://doi.org/10.1109/TKDE.2021.3077495

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*.

Longa, A., Lachi, V., Santin, G., Bianchini, M., Lepri, B., Lio, P., Scarselli, F., & Passerini, A. (2023). Graph neural networks for temporal graphs: State of the art, open challenges, and opportunities. https://doi.org/10.48550/ARXIV.2302.01018.

Pareja, A., Domeniconi, G., Chen, J., Ma, T., Suzumura, T., Kanezashi, H., Kaler, T., Schardl, T. B., & Leiserson, C. E. (2020). EvolveGCN: Evolving graph convolutional networks for dynamic graphs. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Poursafaei, F., Huang, A., Pelrine, K., & Rabbany, R. (2022). Towards better evaluation for dynamic link prediction. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*. https://openreview.net/forum?id=1GVpwr2Tfdg.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992. Association for Computational Linguistics, Hong Kong, China. https://doi.org/10.18653/v1/D19-1410.

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*.

Rusch, T. K., Bronstein, M. M., & Mishra, S. (2023). A survey on oversmoothing in graph neural networks. CoRR **abs/2303.10993**.

Sato, R., Yamada, M., & Kashima, H. (2021). Random Features Strengthen Graph Neural Networks, pp. 333–341. https://doi.org/10.1137/1.9781611976700.38

Sawhney, R., Agarwal, S., Wadhwa, A., & Shah, R. R. (2020) Deep attentive learning for stock movement prediction from social media text and company correlations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 8415–8426. Association for Computational Linguistics, Online*. https://doi.org/10.18653/v1/2020.emnlp-main.676.

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, Proceedings, Part I 25, pp. 362–373. Springer.

Shamsi, K., Gel, Y. R., Kantarcioglu, M., & Akcora, C. G. (2022). Chartalist: Labeled graph datasets for utxo and account-based blockchains. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*, pp. 1–14. https://openreview.net/pdf?id=10iA3OowAV3.

Shumovskaia, V., Fedyanin, K., Sukharev, I., Berestnev, D., & Panov, M. (2021). Linking bank clients using graph neural networks powered by rich transactional data. *International Journal of Data Science and Analytics, 12*(2), 135–145.

Wang, J., Zhang, S., Xiao, Y., & Song, R. (2021). A review on graph neural network methods in financial applications. CoRR **abs/2111.15367**. https://arxiv.org/abs/2111.15367

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.) *Proceedings of the First Learning on Graphs Conference*, *Proceedings of Machine Learning Research*, vol. 198, pp. 14:1–14:17. PMLR. https://proceedings.mlr.press/v198/wang22a.html

Weber, M., Domeniconi, G., Chen, J., Weidele, D. K. I., Bellei, C., Robinson, T., & Leiserson, C. E. (2019). Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. CoRR **abs/1908.02591**. http://arxiv.org/abs/1908.02591

Xu, S., Yang, C., Shi, C., Fang, Y., Guo, Y., Yang, T., Zhang, L., & Hu, M. (2021). Topic-aware heterogeneous graph neural network for link prediction. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management, CIKM '21*, pp. 2261–2270. Association for Computing Machinery. https://doi.org/10.1145/3459637.3482485.

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*, pp. 2358–2366. Association for Computing Machinery.

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*, pp. 2236–2246. ACM. https://doi.org/10.1145/3308558.3313747

You, J., Ying, Z., & Leskovec, J. (2020). Design space for graph neural networks. In *NeurIPS*.

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., & Li, H. (2020). T-GCN: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems, 21*(9), 3848–3858. https://doi.org/10.1109/tits.2019.2935152

Zou, J., Zhao, Q., Jiao, Y., Cao, H., Liu, Y., Yan, Q., Abbasnejad, E., Liu, L., & Shi, J. Q. (2022). Stock market prediction via deep learning techniques: A survey.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.