Vilella *et al. Applied Network Science* (2025) 10:14
https://doi.org/10.1007/s41109-025-00702-1

Applied Network Science

RESEARCH Open Access

Weirdnodes: centrality based anomaly detection on temporal networks for the anti-financial crime domain

Salvatore Vilella[1], Arthur Capozzi[2], Marco Fornasiero[3], Dario Moncalvo[3], Valeria Ricci[3], Silvia Ronchiadin[3] and Giancarlo Ruffo[1*]

*Correspondence:
Giancarlo Ruffo
giancarlo.ruffo@uniupo.it
[1]DISIT, University of Eastern Piedmont, Viale Teresa Michel 11, 15121 Alessandria, Italy
[2]Computational Social Science, ETH Zürich, Zürich, Switzerland
[3]Anti-Financial Crime Digital Hub, Corso Inghilterra 3, 10138 Turin, Italy

Abstract
Analyzing the financial domain presents significant challenges, particularly due to the lack of publicly available data and the limited opportunities for the scientific community to test methods and algorithms on real datasets. This paper explores the application of network analysis to the Anti-Financial Crime (AFC) domain, leveraging a large dataset of over 80 million cross-border wire transfers. Our goal is to detect outliers potentially involved in malicious activities, in alignment with financial regulations. We address this problem with WeirdNodes, a centrality-based methodology for ranked anomaly detection in temporal networks. Unlike many existing approaches that rely on rule-based algorithms or general machine learning models, WeirdNodes harnesses the evolving structure and relationships within financial transaction networks. By focusing on minimal disruptions in otherwise stable ecosystems-such as those built upon large set of international financial transactions-our approach tracks the temporal evolution of centrality-based rankings. This enables the detection of abrupt role shifts, signaling anomalies that warrant further investigation by domain experts. Beyond anomaly detection, this analysis represents a step toward automating AFC and Anti-Money Laundering (AML) processes, equipping AFC officers with a comprehensive, top-down perspective to enhance their efforts. By providing a bird's eye view of financial data, our approach mitigates the risk of overlooking complex behaviors that single-node or narrowly focused transactional analyses may fail to detect.

Keywords Anomaly detection, Network analysis, Financial graphs, Node rankings

Introduction

Modern complex systems, encompassing fields as diverse as biology, cybersecurity, finance, healthcare, and industrial processes, present an environment where deviations from the norm can hold critical significance. These deviations constitute a focal point for anomaly detection: they can signify a range of events depending on the domain, from subtle inefficiencies to outright malicious activities. Anomaly detection serves as a critical line of defense against system failures, security breaches or other potential threats.

Early detection can mitigate the cascading effects that anomalies might trigger. By identifying deviations from the norm, anomaly detection allows to respond proactively, maintaining system integrity, data security, and operational efficiency; an ex-post identification of anomalies instead provides the analysts with useful insights on the nature of the system and its components, possibly informing them about potential future threats.

This is particularly true in the financial domain, where anomaly detection is a key tool for combating financial crime and for identifying suspicious or illegal activities. Depending on the specific tasks and needs of the controllers, both real-time approaches or post-event detection of anomalies can be exploited to unveil malicious activities. Currently, typical Anti Financial Crime (AFC) practices are rule-based.[1] An important part in acquiring the necessary knowledge is played by surveys such as the Wolfsberg Group's CBDDQ and FCCQ, i.e., a long set of questions that are used by banks or other financial institutions to provide high-level information about their Financial Crime Compliance Program; this and other AFC strategies, that are peculiar to each financial institution due to its own risk appetite and risk tolerance, are built upon complex layers of national and international regulations. The traditional approach of Anti Money Laundering toward data analysis is largely based on individual subjects as unit of analysis, in order to spot standalone customer behaviors that may suggest the presence of criminal activities leveraging a Financial Institution's accounts and means of payment. Nevertheless, there is an important conceptual overlap between the traditional approaches to AFC and the analysis of complex systems: network analysis is an embedded activity in the investigation process leading to SAR (Suspicious Activity Report) already performed by human based reasoning, with the general support of individual productivity desktop tools, or of specifically developed automated systems. This activity, starting from previously selected starting points (suspicious activities already spotted), partially balance or complement the narrow focus coming from the customer-centric transaction monitoring output.

For these reasons, improving the state of the art on network analysis applied to the detection of anomalies in the financial domain could help regulators and financial institutions in their fight against financial crimes, by switching from an *atomic* to a *high level* and *context-aware* view of the system. Nonetheless, detecting anomalies within complex systems poses a unique set of challenges. Traditional rule-based methods often fall short in capturing the subtle dynamics that characterize anomalies; on the other hand, machine learning and, more specifically, deep-learning based methods, while being very effective in capturing the embedded complexity of the system, can lack the *interpretability* that regulators expect and law enforcement agencies need in order to take action. Furthermore, the domain experts are considered liable for not having reported a suspicious transaction to an extent that is proportional to the severity of the implausibility of the unreported case, thus calling for an anomaly detection system.

We position our work in the AFC context, proposing a methodology that exploits well-consolidated network centrality measures and node rankings to unveil sudden and unexpected changes in the role of nodes in the system evolving through time. Given the peculiarities of the financial domain, we aim to allow the AFC analyst to further investigate suggested potential anomalies taking into account the activity of the actors and of their neighbors; because of the understanding that we have of the centrality measures

[1] According to the Correspondent Banking Due Diligence Questionnaire (CBDDQ) Guidance (Wolfsberg Group), https://t.ly/PIZms, last accessed: 17/08/2023.

in a complex network, and how to interpret them, the actions taken by the AFC analyst will be fully explainable. We will treat this as a problem of ranked information retrieval: we want our final output to be a list of observations, sorted by their measured deviation from the norm, so that outliers are prioritized in such a way that the probability of finding real anomalies towards the top of this list is maximized. As a by product of our work, we also introduce a set of network perturbation algorithms that allows for the creation and manipulation of synthetic data simulating a temporal network of financial transactions. Our methodology is tested on these artificially generated networks, making the results on real data more reliable.

Problem statement and formalization

As noted by Savage et al. in (2014), when detecting anomalies in social networks, the focus is on identifying unexpected patterns of interactions between individuals within the network. Generally speaking, network anomalies can be succinctly defined as "patterns of interaction that significantly differ from the norm," aligning with the definitions provided by Chandola et al. (2009), and Hodge and Austin (2004). Broadening further the perspective, rather than limiting our consideration to patterns in their typical network analysis connotation - such as motifs and paths - we can explore and identify more general forms of regularities.

Our problem can be intuitively stated as it follows: *in a network system that exhibits minimal changes across different time intervals, we aim at identifying those nodes that show a clear distinctive dynamics w.r.t. the rest of the network.* We discuss that monitoring the evolution of centrality-based nodes' ranks is more effective to identify such modifications, than just monitoring the centrality dynamics alone (values are instable), or some specific motifs (their detection in large networks is computationally expensive).

Hence, we present a general formalization for distinguishing regularities from anomalies in temporal networks. In this context, we posit that our graph exhibits a certain regular property at time t_i, and any observation deviating from this property at time t_j with $j > i$ is deemed an outlier and, potentially, as a system anomaly. Having a well-defined characterization of this property and empirical validation that it holds for our graph, our objective is to identify non-conforming outliers. An additional observation is necessary here: the overall problem conceptualization pertains to measures and features at a *node-wise analysis*; we can readily extend our definitions to also encompass link-wise analysis. This extension is particularly relevant in certain domains where the focus may be on the interactions themselves, their overall stability, and the minority of links that deviates from the norms. Although link-wise analysis is crucial in many domains, the rest of the paper will not mention this aspect for the sake of clarity, and will focus on identifying nodes whose network based features may change across different time intervals.

In our scenario, we are not focused on properties that are strictly invariant (i.e., the probability of observing an anomaly is extremely unlikely). Instead, we are searching for properties where nodes "sometimes" deviate from the general behavior. Simultaneously, we are not interested in extremely heterogeneous properties, as defining a "normal" behavior in such cases could be challenging. It is worth noting that some seemingly natural attributes may not necessarily serve as good candidates for defining the properties we seek. For instance, node strength and degree are known to be highly heterogeneous

in many real-world networks, and the probability of a node having strength and degree much higher than the mean is significantly higher than with normal distributions.

The above considerations are particularly important in a time-varying network: degree, strength and other more complex node's metrics like betweenness, closeness, or page rank, may exhibit values that change constantly and considerably over time. However, nodes' rankings based on such measures may remain quite stable over time, and they can be excellent candidates for the properties we want to observe *across* different temporal layers. Therefore, if we are interested on how a metric changes over time, we need to define a temporal property:

Definition 1 *(Temporal Property)* Given a graph $G = (N, L)$, two distinct time intervals T_x and T_y, and a measure or feature *f* associated with a generic node *i*, a temporal property $P(G, T_x, T_y, f, i)$ is a condition that evaluates to true if *f*(*i*) remains approximately unchanged when computed for G_{T_x} and G_{T_y}. We say that *Pholds* for *i* if the condition is true.

Example 1 If $f(i) = r(d^-, i)$ is the feature we are considering, that is the position of *i* in a ranking based on the indegree d^-, then we can build on such *f* a temporal property. The indegree of *i* and of all the other nodes in *G* may change, but if the position of *i* in the ranking remains approximately unchanged between T_x and T_y, i.e., $r_x(d^-, i) \approx r_y(d^-, i)$, then $P(G, T_x, T_y, f, i)$ holds for node *i*.
Finally, we need to formalize the idea that a temporal property should be tested on every node of the graph, and considered valid if it significantly holds for them:

Definition 2 *(Graph Temporal Property)* Given a graph $G = (N, L)$, two distinct time intervals T_x and T_y, a measure or feature *f* associated with a generic node *i*, and a temporal property *P*, a graph temporal property $\mathcal{P}(G, P)$ is *valid* if $P(G, T_x, T_y, f, i)$ significantly holds for nodes $i \in N$.

Example 2 Given a graph *G* with $N = \{1, \ldots, 5\}$, with $r_x(d^-)$ and $r_y(d^-)$ being indegree based node rankings in two different time intervals T_x and T_y, we can build graph temporal property $\mathcal{P}(G, P)$ in terms of temporal property $P(G, T_x, T_y, d^-, i) = (r_x(d^-, i) \approx r_y(d^-, i))$. We consider $\mathcal{P}$ valid if Spearman's or Kendall's tau correlation coefficients returns at least moderate concordance signals (i.e., values greater than 0.4 or 0.3 respectively). Of course, stronger or weaker threshold can be set depending on the domain.
The problem formalization introduced so far supports us in focusing our analysis towards temporal properties and node metrics that exhibit stability (e.g., ranks with moderate signals of concordance or even better) over time. Stability, in this context, implies that the majority of nodes do not (significantly) alter their values. Hence, a valid graph temporal property can be defined in terms of stable node's metrics. This approach would eventually enable us to promptly identify the minority of nodes that deviate from the norm.

Definition 3 *(Temporal Outlier)* Given a graph $G = (N, L)$, two distinct time intervals T_x and T_y, a measure or feature *f*, a temporal property *P*, and a valid graph temporal property $\mathcal{P}(G, P)$, a node *i* is an *outlier* when $P(G, T_x, T_y, f, i)$ is false.

Example 3 Given a graph G with $N = \{1, \ldots, 5\}$, with $r_x(d^-) = (1, 2, 3, 4, 5)$ and $r_y(d^-) = (5, 2, 3, 4, 1)$ being indegree based node rankings in two different time intervals T_x and T_y, then nodes 1 and 5 are definitely outliers since $r_x(d^-, 1) \not\approx r_y(d^-, 1)$ and $r_x(d^-, 1) \not\approx r_y(d^-, 1)$.
Let us observe that, given the set of outliers $\mathcal{O} = \{i \in N \mid \neg P(G, T_x, T_y, f, i)\}$, if we are able to *quantify* how much $P(G, T_x, T_y, f, i)$ is far from being true for each outlier $i \in \mathcal{O}$, we can also sort set $\mathcal{O}$ in function of this value, e.g., $\delta = (r_x(f, i) - r_y(f, i))$. We follow this idea, exploiting centrality measures based nodes' rankings, to define procedure WeirdNodes described in Sect. 4.2.

AFC's domain and constraints

The entire design of our methodology stems from the application of AFC, that impose a set of assumptions and requirements that are needed for its applicability in day-to-day operations. These constraints involve aspects of interpretability of the results, given the legal implications of AFC investigations; moreover, the ever-increasing complexity of financial fraud operations calls for new tools to help domain experts in detecting them in a timely manner, unveiling complex structures yet maintaining a human-in-the-loop approach. We list some of the most relevant remarks below:

Remark 1 (Interpretability) In many fields that deal with potential criminal records or with life-critical applications, such as health or safety, full interpretability of the evidence underlying any decision is required by the regulators. This partially prevents to fully deploy of the firepower of modern AI algorithms in the routine operations of AFC divisions of financial institutions, that should be able to provide to the relevant Authorities full details of the underlying dynamics leading to the identification of potentially suspicious cases. For this reason, the utilization of AI black boxes should be minimized as much as possible, unless their output are fully explainable.

Remark 2 (Unlabeled data) We are dealing with a context where no explicit notion of anomaly is known and data, when available, is usually unlabeled. This is particularly true for many domains where consolidated anomaly detection practices are already enforced, and a paradigm shift is needed in order to expand and improve the current domain knowledge. This led us to adopt an *unsupervised approach*, without introducing any prior bias or without looking for specific known money laundering patterns. It is also worthy to note that within the AFC domain the evaluation of an anomaly is non trivial also "a posteriori". Indeed, such verification is not entirely in the hands of the AFC experts of the financial institution; a fully detailed report on the individual case will be filed to the competent authority, which in turn hardly provides timely feedback to the institution. For this reason it is extremely difficult to find a ground truth to perform experiments and validate algorithms in such a specific domain.

Remark 3 (Ranked results) Algorithms should provide experts with a concise list of potentially relevant anomalies, ensuring the timely execution of their analysis and subsequent reporting to the competent authority. Algorithms returning an unordered set of potentially anomalous cases can be unsatisfactory for many reasons: the set can be too big to be timely investigated by the experts, or also too small and likely to neglect some

relevant cases. The classic "feast or famine" problem (see Manning et al. 2008) is traditionally solved assigning a score to each observation, and rank the set consequently. As a consequence, our process returns a ranked list of nodes potentially anomalous.

In light of the above constraints and characteristics of the AFC domain, we will treat the task of anomaly detection as a problem of *ranked information retrieval*, namely, by designing the process described in Sect. 4.2 whose ultimate output is an ordered set of potential anomalies, that is, a list of observations sorted by their quantified deviations from a given norm. As argued, this should be done by always keeping the *interpretability* of the results as a top priority, by providing as an output a set of nodes that can be further inspected by a human expert. In addition, in Sect. 4.4, we provide an empirical validation of the methodology using artificially generated data with *labeled perturbations* simulating a evolving temporal graphs representing financial transactions between nodes. Finally, we were able to label as "relevant" and "not relevant" the results returned by our methodology when applied to the real data (see Sect. 5), so that a validation of the method in our case study is provided as well.

Methods

Basic notation

Networks model interactions between agents, with financial transactions being a prime example. In such networks, nodes represent entities (e.g., individuals, bank accounts), and links represent transactions. Using timestamps, a *directed weighted temporal network* is a fitting representation of the system.

The graph is defined as $G = (N, L)$, where N is the set of nodes and L is the set of links. Each link (i, j, w, t) denotes a transaction from i to j at time t with weight w, such as the amount of money transferred.

Timestamps are within $\dot{T} = [t_0, t_\omega]$, the period from the first to the last recorded transaction. Various intervals T can be defined within $\dot{T}$, such as $T = \text{ Dec 2023}$. A *temporal layer* G_T of G represents the network snapshot in interval T: $G = (N, L_T)$, where L_T includes links active in T.

Entities have f_N features (e.g., demographics), enabling the creation of aggregated graphs $G_T^{\mathcal{F}}$ at different resolutions. For instance, $G^{\text{Country}}_{\text{Dec 2023}}$ aggregates transactions by country for December 2023, with an edge $(\text{ Italy }, \text{ France})$ representing total money sent from Italy to France during that period.

WeirdNodes: an anomaly detection network-based procedure

Ranking nodes by their properties and assessing the stability of such rankings over time is the cornerstone around which we build WeirdNodes, an analytical process that returns a top-k list of nodes ranked by their measured deviation from a norm. Depending on the application, and on the requirements of the experiment, any node centrality metric could be used: by monitoring the evolution of centrality-based node rankings over time, our approach effectively identifies nodes whose changes are significantly relevant.

A step-by-step description of the WeirdNodes procedure follows here:

1. *Building the graphs* Represent the data as a graph $G = (N, L)$ where nodes N are entities and links L are recorded interactions. Each link (i, j) has a timestamp $t_{ij} \in \dot{T}$ and a weight w_{ij}. Nodes are grouped by feature $\mathcal{F}$, and temporal layers $G^{\mathcal{F}}_{T_0}, G^{\mathcal{F}}_{T_1}, \ldots$

are defined based on time intervals (e.g., months). For the sake of simplicity, we omit the aggregation feature $\mathcal{F}$ from the following notations.

2. *Computing centrality metric m* For each graph, and for every node $i \in N_{T_x}$, compute the given metric $m_x(i)$.
3. *Centrality-Based Node Rankings* Produce node rankings for each graph G_{T_x} based on the computed centrality metric. Denote the ranking for metric m as $r(m)$, and for a specific time interval T_x as $r_x(m)$. The position of node i in $r_x(m)$ is $r_x(m, i)$.
4. *Stability Check* Let $P(G, T_x, T_y, m, i) = (r_x(m, i) \approx r_y(m, i))$, we verify if the graph temporal property $\mathcal{P}(G, P)$ is valid, i.e., the rank correlation coefficients for $r_x(m)$ and $r_y(m)$ over time intervals T_x and T_y returns an acceptable concordance signal. This step ensures the overall stability of node positions in centrality-based rankings, and that the number of outliers (nodes that significantly changed their ranks) are limited in number.

If the ranks based on measure *m* passes the stability test, we can exploit the temporal fluctuations of nodes' positions in centrality based rankings as a criterion for sorting nodes, and continue the procedure as it follows.

5. *Time-Varying Comparison* Compare node rankings across different time intervals. We define the residual $\delta(T_x, T_y, r(m), i) = r_x(m, i) - r_y(m, i)$ for each node i, as indicator of changes in ranks between T_x and T_y. Nodes can be sorted by $|\delta|$ (magnitude of change) or by positive/negative residuals (gaining or losing ranks). Note that if the residual is zero, it indicates that the node's rank did not change from T_x to T_y; if it is negative, it means that the node has a lower rank in T_y than in T_x; if it is positive, it means that the node has a higher rank in T_y than in T_x. Consequently, we sort our nodes by residual for metric m used to create our rankings. We can sort by $|\delta|$ if we are indifferent to whether nodes gained or lost positions in their rankings and want to sort by the magnitude of the change. Alternatively, we can sort in ascending or descending order based on whether we want to prioritize nodes that lost or gained positions in their ranks from T_x to T_y.

Select another centrality measure *m*, and repeat from step (2). When all the desired centralities have been calculated, for those rankings *r*(*m*) that passed the stability check in step (5), we generate a list of nodes sorted by residuals as described in step (5). These different sorted lists of nodes are then passed to the last step of the procedure:

6. *Returning top-k Anomalies* A final list of top-k anomalies is returned to the human analyst. We can just select one of the sorted list returned in the previous step, and select the top-k, or adopt a hybrid strategy, that *merges* all the ordered lists obtained so far, and *sorts* them with the application of some heuristic. It is worth recalling that such k should be kept as smaller as possible, after Remark 3 (see Sect. 3), to allow the human analyst to investigate in a timely manner the returned cases before submitting a report to the Authorities under their responsibility.

```
procedure WeirdNodes(N, L, T_x, T_y, K)
                    ▷ N: nodes, L: links, s.t. each link is a quadruple (i, j, w, t)
                                                    ▷ T_x, T_y: time intervals
                                  ▷ K: the number of top outliers to be returned
    G_{T_x} ← (N, L_{T_x}), G_{T_y} ← (N, L_{T_y})
    ValidCentralityMeasures ← []
    for each m ∈ CentralityMeasures do
                ▷ CentralityMeasures is a list of functions, e.g., as in Table 1
        m_x ← ComputeCentralityMeasure(G_{T_x}, m)
        m_y ← ComputeCentralityMeasure(G_{T_y}, m)
        r_x ← CentralityBasedNodeRanking(m_x)
        r_y ← CentralityBasedNodeRanking(m_y)
        if StabilityCheck(r_x, r_y) == true then
                                ▷ Compute rank correlation between r_x and r_y, and
                                ▷ check if concordance signal is moderate or greater
            ValidCentralityMeasures.append(m)
            for each i ∈ N do        ▷ Time varying comparison by residuals next
                δ_xy[i] ← r_x[i] − r_y[i]
            end for
            SortedNodesList[m] ← SortNodesByResidual(N,δ_xy)
        end if
        MergedList ← []
        for each m ∈ ValidCentralityMeasures do
            MergedList.mergeAndSort(SortedNodesList[m])
        end for
    end for
    TopKList ← top K nodes in MergedList
    return TopKList
end procedure
```

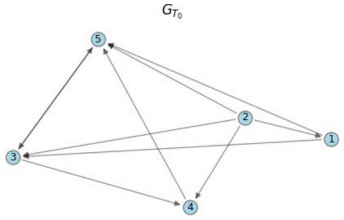
Algorithm 1 WeirdNodes

The pseudo-code of WeirdNodes is shown in Algorithm 1, terminating with the adoption of the generic "merge and sort" strategy of step (6) - that must be independently implemented and tailored to the specific data. Lets observe that in some cases, a single network metric may perform better, while in others a hybrid strategy that exploits information from all the ranks might be needed to detect as many relevant anomalies as possible. We will see examples of both kinds of strategies in Sect. 4.4, and in Sect. 5.

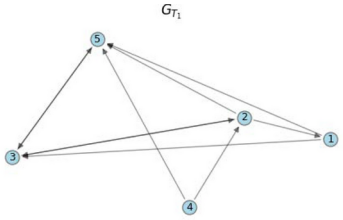
An illustrative example

Let's suppose that our data can be represented by a simple graph *G* with 5 nodes and two temporal layers G_{T_0} (Fig. 1a) and G_{T_1} (Fig. 1b), accomplishing step (1) of our procedure. We can calculate our centrality metrics for both G_{T_0} and G_{T_1} at step (2), and then we can rank nodes accordingly at step (3). For the sake of simplicity, we focus here only on one centrality, indegree d^-: the table in Fig. 1c shows, for every node *i*, the indegree centralities $d_0^-(i)$ and $d_1^-(i)$, and the indegree based node ranking $r_0(d^-, i)$ and $r_1(d^-, i)$.

We apply step (4) and (5) by assuming, as temporal property, that indegree-based node ranking is stable over time, i.e., $P(G, T_0, T_1, d^-, i) = (r_0(d^-, i) \approx r_1(d^-, i))$. Setting a (very conservative) approximation threshold to ± 1, the given temporal property *P* holds for nodes $\{1, 3, 5\}$. Since our rank correlation coefficients return moderate signals of concordance, e.g., Spearman's $\rho = 0.6$, and Kendall's $\tau = 0.4$, we can consider the stability check as quite successful, meaning that the graph temporal property $\mathcal{P}(G, T_0, T_1, d^-, i)$ is valid.



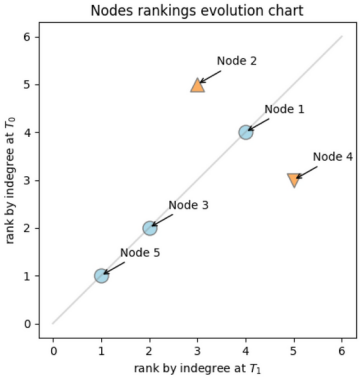
(a) Graph's temporal layer G_{T_0}



(b) Graph's temporal layer G_{T_1}

Node	G_{T_0}		G_{T_1}		
	indegree	rank	indegree	rank	δ
i	d_0^-	r_0	d_1^-	r_1	$r_0 - r_1$
1	.2	4	.2	4	0 =
2	.0	5	.4	3	+2 △
3	.6	2	.6	2	0 =
4	.4	3	.0	5	-2 ▽
5	.8	1	.8	1	0 =

(c) Evolution of nodes' indegree and rank



(d) Node rankings evolution chart; anomalies colored in light orange, regular nodes in light blue.

Fig. 1 The temporal network based anomaly detection analysis applied to a toy example

We can then continue the final part of the process that will lead us to the list of the top-k anomalies. Looking at Fig. 1c, at step (5) we execute the time-varying comparison between ranks at T_0 and T_1: it shows clearly that $|\delta| > 1$ for nodes 2 and 3. In particular, node 2's δ is positive ($= +2$), that means that it *gains* a higher rank from T_0 to T_1. On the contrary, node 4 *loses* two positions in the rank, so δ is negative. These two anomalies are returned to the domain expert. It should be noted that, in this example, step (6) trivially returns the same ordered (and only) list as in step (5).

An intuitive way to visualize and interpret what happens in step (5) is to use a *node rankings evolution chart* (in the rest of the paper referenced as *REC*) shown in Fig. 1d, a simple scatter-plot where values in the *x* axis are the nodes' ranks at one time interval (e.g., T_1), and values in the *y* axis are the nodes' ranks at the other time interval (e.g., T_0). If we draw the identity line $y = x$, we expect that the temporal property holds for nodes that will be plotted along (or very close by) the identity line. Conversely, outliers are progressively more and more distant from the identity line.

Finally, WeirdNodes would return a Top-2 list of (node id, δ) pairs equal to $((2, +2), (4, -2))$, where "regular" node have been filtered out.

Evaluating WeirdNodes with synthetic data

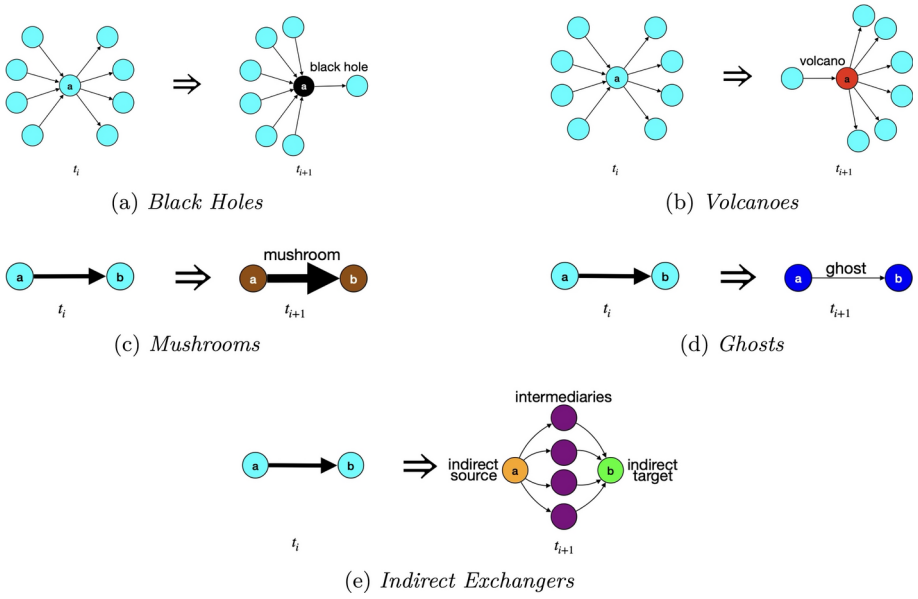
To test the WeirdNodes in a controlled environment, we implemented a set of python functions that take a graph G_0 as input and generate a perturbed version G_1 of the graph itself. This allows the simulation of a network's evolution from a time interval T_0 to T_1, so that we can compare the two snapshots looking for the anomalies that are

injected (and consequently labeled) on purpose. The full package can be accessed at the repo https://github.com/giaruffo/weirdnodes, so that experiments can be accessed, and replicated, as well as other settings can be tested. A more detailed description of the experiments can be found in the Supplementary Material.

The original network is perturbed by injecting one or more of the following anomalies (shown also in Fig. 2 - fractions and factors can be changed for different experimental settings):

- *Black holes* Nodes that, from T_0 to T_1, experience a significant increase in incoming links while showing a substantial reduction in outgoing links. In our experiments, the number of added/removed links is a random factor ≥ 0.5 of the existing edges, as in Fig. 2a.
- *Volcanoes* Nodes that, exhibit a sharp decline in incoming links while significantly expanding their outgoing connections. The number of added/removed links follows a random factor ≥ 0.5 of the existing edges, as in Fig. 2b.
- *Mushrooms* Endpoints of an edge where the edge weight increases by a random factor (between 10 and 100) from T_0 to T_1, as in Fig. 2c.
- *Ghosts* Endpoints of an edge where the edge weight decreases by a random fraction (between 0.5 and 1) from T_0 to T_1, as in Fig. 2d.
- *Indirect exchangers* Endpoints of an edge that is removed from T_0 to T_1 but subsequently form new connections through intermediary nodes, as in Fig. 2e.

It might be observed that these simple perturbations can be easily found by a set of rule-based anomaly detection procedures, that is actually what AFC analysts do when they look for unexpected irregularities that can manifest with one of more of the patterns listed above. However, it is important to stress that more articulated irregular schemes can emerge from one time interval to another, and that a combination of malicious patterns can easily bypass the thresholds and conditions embedded in a rules-based system. Moreover, a behavioral change that radically mutates the role of a node within a complex network can have an impact on the importance - according to some centrality measures



(a) *Black Holes* (b) *Volcanoes*

(c) *Mushrooms* (d) *Ghosts*

(e) *Indirect Exchangers*

Fig. 2 The anomalies that can be injected in a graph at time T_0 to get a perturbed graph at time T_1

- of the node itself. This is precisely the type of irregularity we aim to detect. To achieve this, and we designed the procedure described in Algorithm 1 to capture these temporal node-wise irregularities, as discussed in Sect. 2. Hence, the main objective of the experiments described in this section is how WeirdNodes detects random perturbations in an original graph when applied to a controlled set of predefined anomalies.

We ran four different kinds of experiments. In all of them, the first graph has been randomly generated using an Erdős-Rényi model, with the same number of nodes and edges of the $G^{\text{Country}}_{\text{Feb 2022}}$ graph described in Sect. 5.2, to assess a comparable baseline. Also, the graph is directed and weighted, and weights are generated by a Gaussian model with the same mean and standard deviation we observed in $G^{\text{Country}}_{\text{Feb 2022}}$. The four experimental settings are as follows:

- *Black holes and volcanoes* The original graph G_0 is perturbed in G_1 by picking up 10 random nodes, and - with a probability of 0.5 for each selected node - mutating them into "black holes" or "volcanoes", and labeled accordingly. For each perturbation, when the selected target or source node is detached, the stub edge is rewired to another randomly assigned node.
- *Mushrooms and ghosts* The original graph G_0 is perturbed in G_1 by picking up 5 random edges, and - with a probability of 0.5 for each selected edge - significantly increasing or decreasing their weight. Endpoints are marked as "mushroom" or "ghost" respectively.
- *Indirect exchangers and intermediaries* The original graph G_0 is perturbed in G_1 by removing up to 5 random edges, randomly selecting other 5 intermediary nodes for each removed edge, and wiring source and target nodes by means of these intermediaries. Each involved node is marked as "indirect_source", "indirect_target", or "intermediary'.
- *Mixed anomalies* The original graph G_0 is perturbed in G_1 by picking up 4 random nodes (that are modified as "black_hole" or "ghost"), 2 random edges (each used to change its endpoints to "mushroom" or "ghost"), and 2 other random edges (each used to change its endpoints to "indirect_source" and "indirect_target", and to make other 5 nodes "intermediary").

For each experimental settings, we ran the WeirdNodes procedure five times, initializing the list `CentralityMeasures` in Algorithm 1 to the centralities shown in Table 1.

The centrality-based strategies are considered "valid" after passing the stability check, which occurs when both Kendall's tau and the Spearman coefficient indicate strong or very strong correlations when comparing the results between G_0 and G_1. In some "Mushrooms and Ghosts" and "Mixed Anomalies" experiments, we observed that the HITS Hub and Authority scores are too sensible to perturbations, for which the stability checks returned weak or moderate signals of concordance. In these cases, the ranks calculated after Hub and Authority scores were skipped. Furthermore, lets observe that the structure of the graph is left unchanged under the "Mushrooms and Ghosts" experiment, since no link is removed, added or rewired. Hence, the stability check returns a perfect concordance between the ranks calculated after degree, indegree, outdegree, betweenness, closeness, and eigenvector. Within this experimental setting, only weight-sensitive measures as strength, instrength, outstrength, and pagerank were confirmed valid.

Table 1 A list of centrality measures used to generate node rankings

Measure	Intuitive interpretation	Description
Degree $d(i)$	Node *i*'s connectivity	The overall number of *i*'s connections
Indegree $d^{-}(i)$	Node *i*'s popularity	The number of *i*'s incoming connections
Outdegree $d^{+}(i)$	Node *i*'s branching factor	The number of *i*'s outgoing connections
Strength $s(i)$	Node *i*'s weighted connectivity	The overall weight of *i*'s connections
Instrength $s^{-}(i)$	Node *i*'s weighted popularity	The total weight of *i*'s incoming connections
Outstrength $s^{+}(i)$	Node *i*'s weighted branching factor	The total weight of *i*'s outgoing connections
Betweenness $b(i)$	Node *i*'s importance as a bridge	The higher the number of shortest paths passing through *i*, the greater its value
Closeness $c(i)$	Node *i*'s network topological centrality	The smaller the distances to other nodes, the greater its value
Eigenvector $\mathrm{ev}\ (i)$	Node *i*'s network relevance	The probability that a random walker will eventually reach *i* in an undirectedgraph
PageRank $\mathrm{pr}\ (i)$	Node *i*'s network relevance	The probability that a random walker will eventually reach *i* in a directed graph
Hub score $\mathrm{h}\ (i)$	Node *i*'s network relevance as a sender	The importance of *i* as a sender, based on the importance of its successors
Authority score $\mathrm{a}\ (i)$	Node *i*'s network relevance as a receiver	The importance of *i* as a receiver, based on the importance of its predecessors

Table 2 "Black Holes and Volcanoes" experiment: performances of the different top-30 lists returned by each centrality based strategy and the summation strategy

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	r@30
Degree	1.00	1.00	0.40	0.71	0.74	0.70	0.61	0.80
Indegree	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
Outdegree	1.00	1.00	1.00	1.00	1.00	0.83	0.69	1.00
Strength	1.00	1.00	0.80	0.96	0.94	0.75	0.65	0.90
Instrength	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
Outstrength	1.00	1.00	1.00	1.00	1.00	0.83	0.69	1.00
Betweenness	1.00	1.00	1.00	1.00	0.89	0.64	0.51	0.60
Closeness	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
Eigenvector	1.00	1.00	1.00	1.00	0.97	0.75	0.61	0.80
Hits_hubs	1.00	1.00	1.00	1.00	0.94	0.75	0.61	0.80
Hits_authorities	1.00	1.00	1.00	1.00	0.94	0.75	0.61	0.80
Pagerank	1.00	1.00	1.00	1.00	0.82	0.58	0.45	0.50
Summation_strategy	1.00	1.00	1.00	1.00	0.99	0.80	0.69	1.00

"Black Holes and Volcanoes" experiment Step (5) of the WeirdNodes procedure generates a top-k list (k=30k=30) for each comparison between rankings based on the centrality measures mentioned above. Nodes are ranked by the absolute value of their residuals, and performance is evaluated using precision-at-k (p@k) with $k = \{1, 2, 5\}$, average precision-at-k (avg p@k) with $k = \{10, 20, 30\}$, and recall-at-30 (r@30), as in Table 2. The evolution of node rankings for each centrality-based strategy is illustrated in Fig. 3.

In Fig. 3, we can see that, after the "Black Holes and Volcanoes" perturbation, there is a clear trend of stable nodes very close to the identity line for each of the centrality-based strategies, as predicted by the strong and very strong signals returned by the stability check. Furthermore, we can spot that among the outliers showing higher ranks residuals, every centrality-based strategy is able to identify many nodes labeled as black holes or volcanoes. It is remarkable that the first 5 positions (see p@5 and avg p@5) in almost every top-k list are correctly labeled as anomalies. Also, we have that all the strategies'

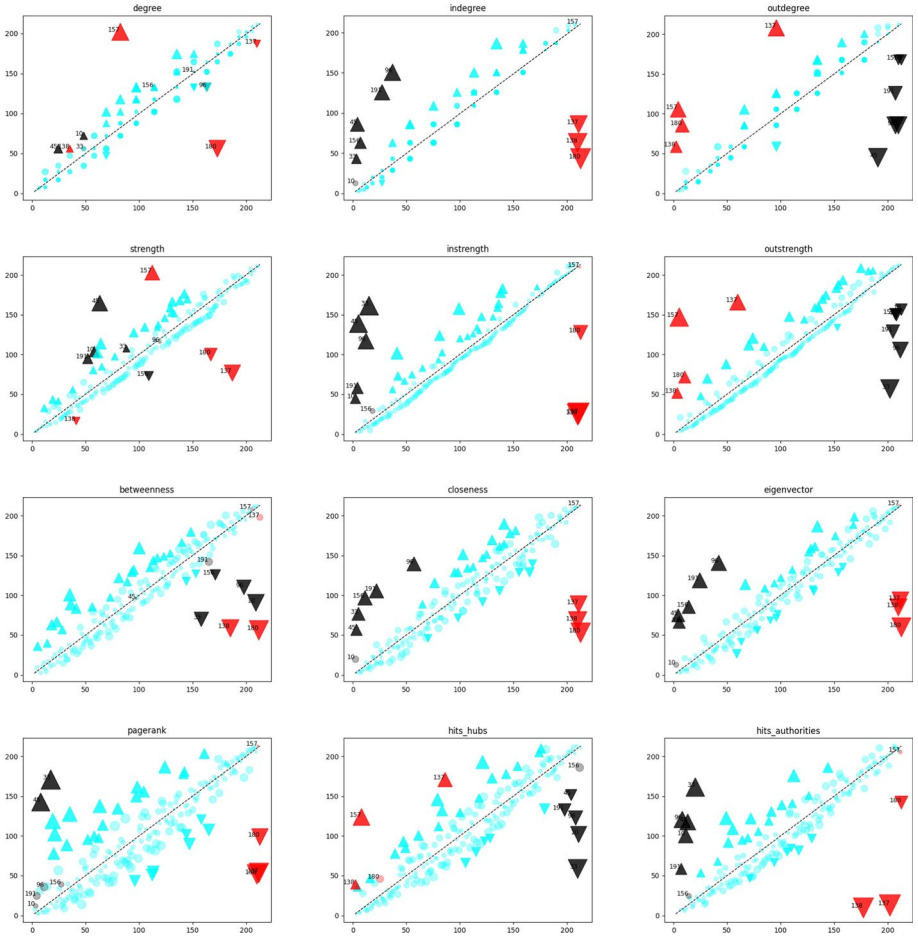


Fig. 3 "Black Holes and Volcanoes" experiment: ranking evolution charts between graph G_0 and its perturbation G_1. Plots' horizontal axes describes ranks by the given measure (at top of each plot) at time T_1, while vertical axes describes ranks at time T_0. Top-30 nodes that are gaining positions (triangles), and top-30 nodes that are losing positions (inverted triangles) in the rankings are colored according their type: normal (cyan), black holes (black), volcanoes (red)

performances are excellent (p@10 is always ≥ 0.74, r@30 is often close to 1.0), and that strategies based on outdegree and outstrength out perform the others (p@10 = 1.0).

Of course, in a more realistic scenario the analyst does not know a-priori which of the centrality-based strategies are better than the others, so we usually need to merge the information from all the top-k list in just one. Hence, as in step (6) of Algorithm 1, we tested a very simple hybrid strategy, that is calculated as it follows:

Definition 4 *(Summation Strategy)* Given all the top-k lists of nodes, for each node we calculate the sum of all the normalized ranks that it gets in each top-k list. We return a summarizing top-k list of nodes ordered by the normalized sum.

The performance of the application of the "Summation strategy" to the last step of our WeirdNodes procedure are shown in the last row of Table 2, and in Fig. 4.

Other experiments and observations We had executed five different runs for each of the experimental settings that we introduced above. Generally speaking, the results are very good, proving also that centrality-based strategies behave differently in function of the irregularities that are injected in the original graph: if degree-based strategies out perform the others when looking for black holes and volcanoes, strength-based

Fig. 4 "Black Holes and Volcanoes" experiment: evaluation of the summation strategy in terms of *precision-at-k*, *precision-at-k*, and *recall-at-k*

strategies are better with mushrooms, ghosts, indirect exchangers, and intermediaries. Mixed anomalies, instead, may need the effort of many different approaches, since no single centrality based strategy clearly stands out. For this reason, in our AFC case study, we evaluated many approaches, and tested some hybrid strategies that exploit other information on nodes that were available in the dataset.

The details of these experiments are not reported in the main paper for the sake of brevity, but they can be found in the Supplementary Material. Finally, we recall that the full experimental settings can be replicated and iterated because the code is fully available at https://github.com/giaruffo/weirdnodes.

Case study: anti-financial crime detection in cross-country money transfer temporal networks

Our approach has been applied on a large record of cross-country financial transactions. The ultimate objective is to identify anomalous nodes that, over time, experience unexpected changes in their position within the network, as stated in Sect. 2. Such nodes warrant further investigation by AFC analysts. Additionally, we can compare different top-k outliers selection strategies to maximize performances.

Dataset description

Data supporting the present case study is provided by Intesa Sanpaolo (ISP) and AFC Digital Hub, and was made available to the research team in a fully anonymized form respecting the strictest privacy and security requirements. The data supporting the findings of this study is available from ISP upon request to AFC Digital Hub. Please note that restrictions for data availability apply. Researchers interested in having access to data for academic purposes will be asked to sign a non-disclosure agreement (see the Declarations at the end of the paper for more information about the AFC Digital Hub consortium, and how to access data).

The dataset collects all the cross-border transactions that involve ISP's customers or in which ISP's business is to support the payment performing of partners financial institutions while such operations do not involve its own customers (pass-through). It encompasses 80 million SEPA SCT and SWIFT enabled wire transfers in a period of fifteen months, from September 2021 to November 2022. Every transaction has a timestamp indicating day, month and year. We also have the information about both the Country and the BIC code of the sender and the receiver; moreover, for the months of January, February and March 2022, we were also provided with the anonymized IBAN codes

involved in the transactions, allowing us to work at multiple levels of temporal and spatial aggregation. The information contained in the dataset are detailed in Table 3.

"BIC" denotes the International ISO standard ISO 9362, delineating the structure and components of a universal identifier code. This code serves as a requirement for both financial and non-financial institutions, enabling the streamlined automated processing of information on a global scale. BICs fall into two distinct categories: Connected BICs, which hold access privileges to the Swift network, and non-connected BICs utilized solely for referencing purposes without network access.[2]. Our dataset includes 8008 unique BICs and 218 countries. The volume of transactions per month is split almost in a 90–10 proportion between SEPA and SWIFT respectively.

Given the information contained in our dataset, we are able to model the transactions at different levels of aggregation, studying money transfers between either countries, BIC codes or IBAN codes. In the next sections we will focus on showcasing the application of our anomaly detection procedure on the countries and BIC aggregations, discussing some interesting insights. In the Supplementary Material we provide a comprehensive data exploration of our dataset, focusing also on the finer-grained IBAN codes resolution, discussing the pros and cons of applying our procedure at that level.

Application to the transactions dataset: identifying anomalous countries

First, we aggregate money transfers by countries, in order to identify those nations that, according to our methodology, stand out as potentially anomalous actors in a selected time period.

The application of the first three steps of WeirdNodes consists in building the graphs and computing the centrality metrics and node rankings. By considering a monthly temporal resolution, we obtain 15 temporal layers $G^{\text{Country}}_{\text{Sep 2021}}, G^{\text{Country}}_{\text{Oct 2021}}, \ldots, G^{\text{Country}}_{\text{Nov 2022}}$. Nodes represent countries in our dataset. Every temporal layer has its own edge list, such that each link represents that a given amount of money is transferred from one country to the other during that timeframe; the edge will be weighted according to the total amount of money moved, even across multiple transactions, from the source node to the destination. We have a total of 218 countries; if a country does not have transactions in a

Table 3 Dataset fields description

Field	Description
Data_ref	Transaction date
Transaction_id	Internal unique identification code of the transaction
BIC	Anonymised Business Identifier Code (8 digits) of the bank of the payer/beneficiary
IBAN code	Anonymised version of the International Bank Account Number of the payer/beneficiary (only available for transactions between January, February, and March 2022)
Countryresidence	ISO code of the country of residence of the payer/beneficiary
Countrybank	ISO code of the country of the bank of the payer/beneficiary (according to BIC)
Amount	Expressed in euros. In SEPA transactions and SWIFT transactions where the original currency is euro, the value is actual. In SWIFT transactions where the original currency is not euro, the amount expressed here has been calculated using the exchange rate at the transaction date, however it may be slightly different from the actual value applied to the client
Currency	Original currency applied in the transaction
Source	Data stream of the transaction (SEPA or SWIFT)

[2] A BIC code is formatted according to the following structure: https://www.swift.com/standards/data-standards/bic-business-identifier-code, last accessed: 06/02/2024.

Table 4 Network statistics for the graphs G^{Country} across the 15 months

	Edges	Density	Avg degree	Avg strength	Avg clustering coeff	Diameter	Avg path length
Mean	4,540.33	0.10	42.88	1.45e+09	0.68	3.47	1.90
Std	144.21	0.00	1.45	2.90e+08	0.01	0.52	0.01
Min	4,309.00	0.10	40.65	1.03e+09	0.66	3.00	1.89
Max	4,750.00	0.11	45.02	1.85e+09	0.71	4.00	1.91

Each graph contains 218 countries; more networks' statistics can be found in the Supplementary Material

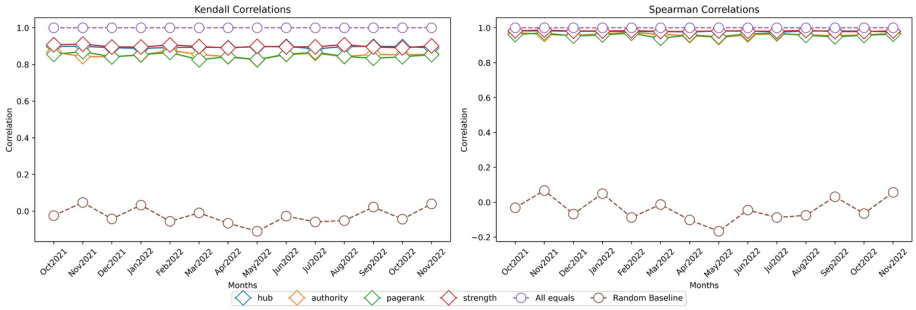


Fig. 5 Stability over time of the node rankings based on centrality metrics, at the level of countries. The stability in each month is compared to the previous and tested against two baselines: *all equals*, where no change occurs from one ranking to the other, and a *random baselines* where every ranking is randomly reshuffled. The very high levels of correlation, according to both Kendall's and Spearman's coefficients, are consistent over time, suggesting the presence of a condition of stability where nodes tend to preserve their ranking positions

certain month, it will appear as a singleton in the relative graph. A descriptive overview of the graphs can be found in Table 4.

On each graph, we compute a set of centrality metrics as in Table 1. Specifically, we evaluate our nodes based on (in/out)strength, PageRank, hub and authority scores. We are not using (in/out)degree because graphs are weighted and strength can be more informative in this case study. Moreover, we are overlooking other classical centrality measures like closeness and betweenness because these graphs show very short average distances (with very small standard deviation): the nodes would have a very similar closeness with each other, and considering that the average clustering coefficient is very high, also bridges (i.e., nodes with high betweenness) loses their informational value. It is also worth noticing that the application of WeirdNodes to synthetic data, as shown in Sect. 4.4 and further explored in the Supplementary Material, returned a signal that degree-based strategies are contained and in some cases outperformed (as in the "Mushrooms and Ghosts" experiment) by strength-based strategies, and that betweenness and closeness based strategies do not excel in novelty, meaning that they usually detect outliers that are spotted also by the other measures.

We observe in Fig. 5 that the centrality-based node rankings are extremely highly correlated according to both Kendall's τ and Spearman's ρ, and that this is consistently true throughout the entire dataset; the evaluation of the rank correlations in Fig. 5 tells us to which extent this temporal property is satisfied. The rankings based on the centrality metrics are consistently similar over time, and this is checked against two baselines: *All Equals*, where the ranking never changes, resulting in a perfect correlation, and a *random baseline*, where the rankings are randomly shuffled, resulting in a correlation that oscillates around 0.

Having passed the necessary check of Step (4), we can proceed with the identification of the top-k anomalies. In step (5) we execute the time-varying comparison between two

timeframes, namely February and March 2022.[3] In Fig. 6 we see the resulting ranking evolution charts, that display how the vast majority of nodes lie very close to the identity line, with only a minority of countries that deviate from the condition of stability identified by the identity line. The output in each metric is sorted by residual (δ).

We are now able to evaluate the outliers that emerge from step (5) as countries showing potentially anomalous financial behavior. To do so, we refer to the ISP classification of Countries according to their financial risk.[4] This classification, frequently updated, follows a low-medium-high risk scheme. We focus on the top-10 nodes gaining positions and the top-10 losing positions in the rankings, forming our top-20 outlier list. As shown, many of these nodes are high or medium-risk countries. While we cannot disclose specific country names or classifications, we consider high and medium-risk countries as true positives, allowing us to measure the algorithm's performance using average precision-at-k for each centrality metric. Table 5 details the average precision-at-k for various k values.

Ranking strategies based on centrality metrics are effective in identifying countries with medium and high financial risk, indicating that the methodology can highlight nodes potentially involved in malicious activities. However, not all reported anomalies

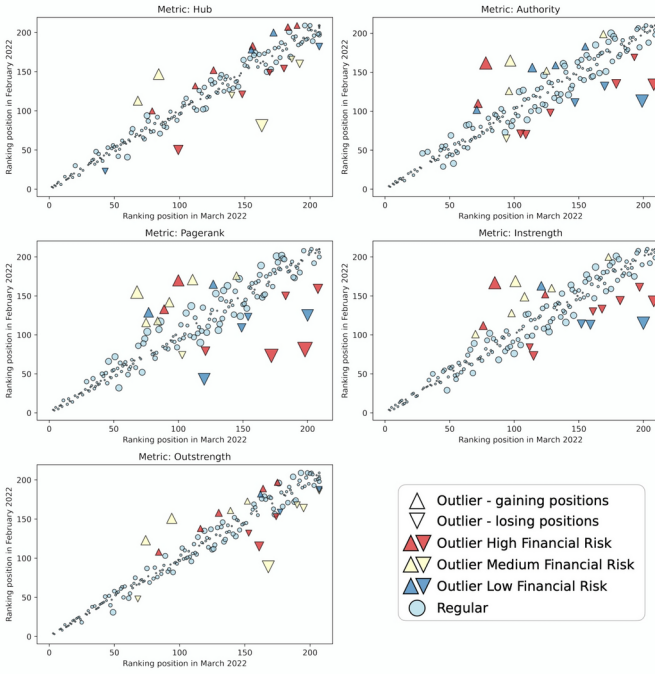


Fig. 6 Ranking evolution charts of the time-varying comparison between countries in February/March 2022. The top-10 nodes by residual that are gaining positions (triangles) plus the top-10 that are losing positions (inverted triangles) in the rankings are colored by their financial risk according to ISP internal classification. The rest of the nodes are labeled as "regular". Size is proportional to the residual. While there is a clear trend of stable nodes very close to the identity line (as suggested by Fig. 5), there is a selection of outliers with high residual that undergo significant changes in their ranking positions

[3] Although the method can be applied to any two timeframes, these time frames have been selected because of the Russo-Ukrainian War that began in February 2022. This period has been particularly observed by AFC experts, and some more straightforward interpretation and validation of the outcomes were expected.

[4] This list is the byproduct of international regulations, company policies and internal information; however, many similar lists are available, such as https://t.ly/Lvwcd (Commission 2024) or https://t.ly/qn2iK (Society 2024) (last accessed: 28/01/2025).

Table 5 Performances of the algorithm in terms of the average precision-at-k and recall of the lists of outliers generated by each metric

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	r at 20
Hub	1.0	1.0	1.0	1.00	0.95	0.88	0.80
Authority	0.0	0.5	0.8	0.68	0.68	0.66	0.60
Pagerank	1.0	1.0	0.6	1.00	0.82	0.77	0.70
Instrength	0.0	0.5	0.6	0.64	0.63	0.70	0.80
Outstrength	1.0	1.0	1.0	1.00	1.00	0.99	0.85

At this stage of analysis, since we have the full ground-truth of the financial risk, we are able to compute the recall score

Table 6 Evaluation of step (6): performance obtained by merging the output of step (5) in a final, single list of outliers sorted by Δ_{HRA}

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	r at 30
Mixed sorting strategy by Δ_{HRA}	1.0	1.0	1.0	1.0	1.0	0.93	0.86	0.73

are high or medium-risk countries. These can be considered false positives in terms of financial risk but their unexpected changes in network roles can be valuable. Such anomalies, appearing alongside usual suspects, help AFC analysts identify potential threats not on the watch-list.

Since a node can be flagged by multiple metrics, some repetitions occur. The union of all top-10 "ascending", and top-10 "descending" anomalies across five centrality metrics includes 54 unique countries: 22 (out of 82) high-risk, 18 (out of 62) medium-risk, and 14 (out of 68) low-risk. With the limited number of nodes, we execute step (6) for the final output: analysts can evaluate outliers by individual centrality metrics or merge lists. We sort the 54 countries based on the change in the volume of money exchanged with high-risk countries, highlighting significant nodes. This approach achieves good performance in precision-at-k and recall, as shown in Table 6, and we refer to it as the mixed sorting strategy throughout the rest of the paper.

Definition 5 *(Mixed Sorting strategy)* Given all the top-k lists of nodes representing countries, and that countries can be classified in terms of financial risks, we return a top-k list of nodes sorted by Δ_{HRA}, with Δ_{HRA} defined as the difference in the volume of money exchanged with high-risk countries.

It should be noted that hub score based strategy alone out performs the mixed sorting strategy at avg p@5 and recall. However, in a a realistic scenario, the analyst is unaware which of the centralities based strategy is better than the others, and returning up to 100 nodes to the analyst is unfeasible. On the contrary, mixed sorting strategy returns only 30 outliers to be analyzed further, and its performances are comparable to the best out of five centrality based strategies (with a higher p@10, too).

The output of this procedure provides the user with useful information: as argued, it is able to identify among the vast amount of transactions a number of well-known malicious agents, together - and, maybe, most importantly - with other potentially anomalous nodes that are unexpected to be detected at this resolution. Anyway, such a coarse-grained analysis, even though yielding very encouraging results, is still preliminary. We can fully exploit the potential of this methodology by examining transactions at a finer-grained level, identifying outliers that can be thoroughly inspected by AFC experts in order to provide more precise information at a smaller scale.

Table 7 Network statistics for the graphs G^{BIC} across the 15 months

	Edges	Density	Avg degree	Avg strength	Avg clustering coeff	Diameter	Avg path length
Mean	58,130.73	1.810^{-3}	20.48	54,024,165.98	0.28	6.20	2.78
Std	1,734.11	1.010^{-4}	0.66	10,903,596.64	0.00	0.41	0.02
Min	55,509.00	1.710^{-3}	19.47	38,442,552.78	0.27	6.00	2.74
Max	61,142.00	1.910^{-3}	21.69	68,983,766.61	0.29	7.00	2.82

Each graph contains 8008 BICs

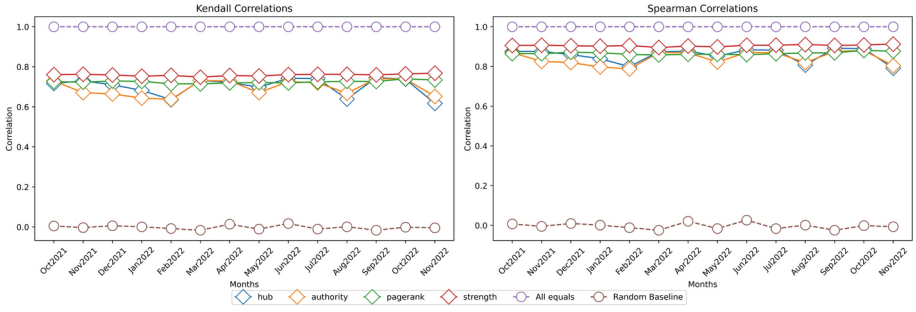


Fig. 7 Stability over time of the node rankings based on centrality metrics. The stability in each month is compared to the previous and tested against two baselines: *all equals*, where no change occurs from one ranking to the other, and a *random baseline* where every ranking is randomly reshuffled. The high levels of correlation are consistent over time, suggesting the presence of a condition of rankings stability

Application to the transactions dataset: identifying anomalous BIC codes

We now aggregate the transactions at the finer-grained resolution of BICs and we apply our outlier detection procedure in order to identify potentially anomalous BIC codes in our data. By considering the same monthly temporal resolution, we obtain once again 15 temporal layers $G^{\text{BIC}}_{\text{Sep 2021}}, G^{\text{BIC}}_{\text{Oct 2021}}, \ldots, G^{\text{BIC}}_{\text{Nov 2022}}$. Every graph has 8008 nodes, and each node represents a different (anonymized) BIC code, namely, the branch of a bank in a certain country. In each layer, an edge between two nodes is established if, at some point within the considered time window, money is transferred from one BIC to the other; the edge will be weighted according to the total amount of money moved, even across multiple transactions, from the source node to the destination. BICs that, during that month, do not have exchanged money with other BICs, will be singletons. A descriptive overview of the graphs can be found in Table 7.

As done for the country graphs, we focus here on the same time frames of February and March 2022 ($G^{\text{BIC}}_{\text{Feb 2022}}$, and $G^{\text{BIC}}_{\text{Mar 2022}}$), and build node rankings based on the same network metrics (i.e., (in/out)strength, PageRank, hub and authority scores).Then, we apply step (4) of the procedure to assess the validity of centralities-based strategies over time. Figure 7 suggests that there is a consistent stability of the node rankings over time for all the centrality measures we selected for all the graph's snapshots, including the pair we are analyzing here.

Hence, we proceed with the identification of the top-k anomalies according to every network centrality metric executing step (5) of WeirdNodes. For each sorted list, we extract the top-30 nodes by residual that are gaining positions plus the top-30 that are losing positions between Feb 2022 and Mar 2022, for a total of 60 nodes (ranking evolution charts can be found in theSupplementary Material). The cutoff of 60 nodes per metric was chosen based on experimental needs; as argued in Remark 3 (Sect. 3), the analyst needs a *reasonable* number of anomalies to inspect manually due to time and resources

limitations. The appropriate number therefore strictly depends on the specifics of the annotation process and, in this specific case, was suggested by the domain experts.

Annotation process and evaluation of the results

Given the much more specific and finer-grained quality of the results at this level of aggregation, the domain experts were able to provide a substantial feedback by undertaking a process of manual validation of the results. The goals of this process are:

- To assess the proportion of real anomalies (*true positives* within the outliers identified by WeirdNodes);
- To assess the goodness of the individual network metrics, by evaluating the position of the true positives in every centrality-based ranking;
- To assess the goodness of the outliers' sorting strategies by evaluating if the true positives occupy the highest positions of the corresponding final ranked list.

Step (5) of WeirdNodes produces five lists of top-60 outliers sorted by their residual δ. This produces a total set of 300 nodes, with repetitions; in fact, the union's size is 162. The domain expert annotated each of them finding 21 "relevant" nodes, marking the others as "not relevant". Therefore, we can quantify the performance of each of these rankings through precision@k and average precision@k shown in Table 8.

The comparative analysis shows that indegree and pagerank based strategies outperform the others in terms of average precision@k and $\overline{\text{recall}}$@60[5]. However, as already discussed, there is no guarantee that these measures will alwaysbe the best, and the analyst cannot have an a priori knowledge allowing them to select one strategy or the other. Hence, a hybrid approach taking the most out of the five top-60 lists is necessary.

One trivial "merging and sorting" strategy is to create the union set of all the 5 top-60 lists, producing in general a final list with size up to 300 nodes (reduced to "only" 162 in our case study). This is what we did for our annotation process, but this is unfeasible to tackle for the human expert on a periodic routine. However, this approach would return a $\overline{\text{recall}}$ equal to 100%, and a precision equal to $21/162 = 0.13$, that is lower than all the avg p@60 in Table 8.

One option is to employ the same *mixed sorting strategy* used in the preceding section: we compile the union of all top-60 lists as before, and then we rank them according to the Δ_{HRA} metric. Then we select the 60 nodes with highest Δ_{HRA} value.

In addition, we introduce another strategy, defined as follows:

Table 8 Evaluation of step (5): nodes in the 5 top-k lists BICs are labeled as relevant (true positive) or not relevant (false positive), and this is used to calculate performances of the valid centralities-based strategies

	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	avg p@60	$\overline{r}$@30	$\overline{r}$@60
Hub	0.0	0.0	0.20	0.20	0.22	0.23	0.22	0.19	0.19	0.29
Authority	1.0	1.0	0.40	**1.00**	0.83	**0.83**	0.66	**0.55**	0.19	0.24
PageRank	0.0	0.0	0.00	0.00	0.19	0.19	0.19	0.17	0.19	**0.33**
Instrength	1.0	1.0	0.40	**1.00**	**1.00**	0.76	**0.76**	0.39	0.14	**0.33**
Outstrength	0.0	0.0	0.20	0.20	0.26	0.29	0.29	0.26	0.19	0.24
Mean	0.4	0.4	0.24	0.48	0.50	0.46	0.42	0.31	0.18	0.29

Best performances for avg p@k and $\overline{r}$@60 are emphasized in bold

[5] We are not able to calculate recall here, because not all the BICs were annotated by the domain experts. Hence, we overline recall to stress that we are calculating this measure just *assuming* that only 21 BICS are "relevant" within the observed time frames.

Definition 6 *(Stratified Sorting strategy)* Given a set of nodes' sorted lists, we return a top-k list that executes the following steps: (1) we group all nodes that share identical positions across all the top-k lists; (3) we sort each group internally by Δ_{HRA}; (3) as one node can be in more top-k lists, we remove the duplicated entries by keeping the one with the highest position in the rank.

Intuitively, the stratified strategy (whose algorithm's pseudo-code plus other information that led us to its formulation can be found in the Supplementary Material) preserves the information of the original ranking positions of the outliers, exploiting also the information coming from Δ_{HRA}.

The stratified strategy, as shown in Table 9, shows better performances than mixed sorting strategies in terms of p@k and avg p@k. In particular, the average p@10 is 0.78, indicating very good performance in the top-10 ranking positions, surpassing all other basic ranking strategies based on a single centrality measure. The weaknesses of the stratified strategy are: (1) a lower $\overline{\text{recall}}$ than the mixed sorting strategy, and (2) lower avg p@k (with $k = [10, 20, 30, 60]$ than the instrength/authority based strategies. However, we point out that the stratified strategy is a necessary, and still commendable, trade off to give the human analyst a list of cases to inspect, and eventually report to the competent authority, as short as possible. Of course, with additional time, they can go through all the 5 top-k lists, to validate any suspicious anomalies.

Discussion

The anomalies resulting from the application of the subsequent steps of the procedure on the country graphs constitute, qualitatively, interesting results, since many of the top-k anomalous nodes are nations that are already known to be involved in potentially malicious financial activities. While we are not allowed to disclose any details about real cases linked with the operational activity of AFC Digital Hub on which our models have been applied, it is possible to make reference to publicly known cases that totally fit with our approach and show its empowering value in transaction monitoring against financial crime. In fact, at the time being, several media sources[6] have reported the sharp decreaseof German exports to Russia in the first quarter of 2023 due to EU financial sanctions against this country. Such discontinuity is almost contemporary to the unprecedented increase of the same export flow to Kyrgyzstan arising evident but heavily delayed suspicion about circumvention of sanctions practices. Specifically, the value of German exports to Russia itself slumped by more than 47% in January-March compared with the same period a year earlier, reflecting tough restrictions on trade imposed by the European Union and other Western powers. However, exports from Germany to Kyrgyzstan rose some 949%, to 170 million euros (187.14 million dollars), a Reuters analysis based on data from the German statistics office shows. Due to the fragmentation of such

Table 9 Evaluation of Step (6): we compare the performances of top-60 lists returned by the mixed and the stratified sorting strategies, with the latter out performing the first in terms of precision

Hybrid strategies	p@1	p@2	p@5	avg p@5	avg p@10	avg p@20	avg p@30	avg p@60	$\overline{r}$@30	$\overline{r}$ @60
Mixed sorting	0.0	0.0	0.2	0.33	0.29	0.25	0.25	0.23	0.29	**0.43**
Stratified sorting	1.0	1.0	0.4	**1.00**	0.78	0.59	0.53	0.43	0.29	0.38

Best performances for avg p@5 and $\overline{r}$@60 are emphasized in bold

[6] https://www.reuters.com/world/german-exports-russias-neighbours-fuel-sanctions-evasion-fears-2023-05-16/.

exporting flows among several chains of players as well as the not direct bordering of the two countries, the inferred circumventing triangulations between Germany and Russia by Kyrgyzstan has been gone undetected for months, although it is referable to one of the basic illustrative perturbations that we described in Sect. 4.4, namely the "indirect exchangers and intermediaries" case, that would be easily detected by the application WeirdNodes. Indeed, the adoption of single transactions or groups of transactions linked to specific accounts as unit of analysis, may jeopardize the ability of a transaction monitoring system to spot, inside the relevant level of noise physiologically affecting the financial flows, the signal of this kind of disrupting shift. On the contrary, the same phenomenon is apparent adopting a top-down, macro-level, network-based methodology such as the one currently described.

The validation carried out on the outcomes of the application at the level of BICs instead allowed us to validate quantitatively the different steps of the procedure. As we already mentioned, of the unique nodes handed to the experts following all the steps of the procedure, 21 are confirmed as true positives worth of further investigations by the analysts and the competent authorities. The final output performs well with respect to the Remark 1 defined in Sect. 3, with good levels of precision-at-k for the top positions of the rankings and an average precision-at-10 of 0.78.

Intuitively, the effectiveness of the application of WeirdNodes to this specific use case can be assessed also through a qualitative inspection of the resulting anomaly. As an example, let's consider the case of the *instrength* analysis. As seen in Fig. 6, there is a number of countries that are positive or negative outliers according to the residuals on the instrength centrality' rankings. Then, the AFC expert can choose, as a starting point for their analysis, to investigate the top 2 countries that lose and gain position, for a total of 4 countries. By inspecting their instrength dynamics, we find that the volume of money received by $\mathrm{Country}_1$ and $\mathrm{Country}_2$ (real references cannot be disclosed) from February to March drops by more than 99% in both cases, losing 3 orders of magnitude; similarly, $\mathrm{Country}_3$ and $\mathrm{Country}_4$ gain a staggering $> 4000\%$, increasing their volume of money by one order of magnitude. Such a drop/gain between countries could be considered extreme, and it recalls the behavior of the so called "black holes" and "volcanoes" anomalies (see Sect. 4.4) and a deeper analysis can be carried out.

Indeed, by inspecting the BICs belonging to these four countries, we are able to identify two BICs that were labeled as *relevant* by the AFC experts. BIC_a and BIC_b nodes belong respectively to $\mathrm{Country}_1$ and $\mathrm{Country}_4$. Interestingly, $\mathrm{Country}_1$ is a country at low financial risk, that would hardly be noticed as an immediate anomaly by AFC officers, that are instead more focused on the observation of high risk entities on their watchlists. Moreover, $\mathrm{Country}_1$ is interpretable also as a "ghost"-like behavior, i.e., it is the endpoint of one edge that is particularly rich at $\mathrm{Feb\ 2022}$ and then drops at $\mathrm{Mar\ 2022}$. At a finer-grained resolution, BIC_a is involved in this transaction, therefore displaying a"ghost"-like behavior as well.

On the other hand, $\mathrm{Country}_4$ is a high-financial risk country, that is probably already under the scrutiny of AFC officers, and it is behaving like a "mushroom", suddenly appearing in $\mathrm{Mar\ 2022}$ with a heavy-weighted connection. However, if we inspect deeper similarly to the previous case, we see that BIC_b is flagged as an anomaly under multiple network based strategies, specifically instrength, pagerank and authority, suggesting that a more complex pattern is underlying this anomaly and it is worth

inspecting by the experts. The stratified sorting strategy, in such a case, helps in uprising to the top interesting information. As a final note, it is worth noting that, in the BIC networks, there are 8 BICs belonging to $\mathrm{Country}_4$, but only two are flagged as anomalies by our algorithm and, of these two, BIC_b is identified as relevant by AFC experts. By applying WeirdNodes, the experts are likely able to effectively filter out unwanted information, navigating through the vast quantity of transactions involving high and low risk countries.

Related work

Anomaly detection on graphs

Broadly speaking, there are many studies that focus on the problem of anomaly detection on networks. Among them, some have leveraged graph metrics and centrality measures to develop algorithms for detecting anomalies and outliers at both the node (Pereira et al. 2019; Mihiri Shashikala et al. 2015; Hassanzadeh et al. 2012; Kaur et al. 2016; Abeer and Mahmoud 2023) and edge levels (Mitchell et al. 2019). To detect anomalous nodes and edges effectively, centrality-based algorithms rely on rankings that remain stable despite graph perturbations and temporal changes. As a result, several studies have explored the robustness and continuity of centrality measures (Pereira et al. 2019; Segarra and Ribeiro 2015; Kardos et al. 2020; Costenbader and Valente 2003). Savage et al. (2014) review a number of relevant literature, observing that anomalies in online social networks (OSNs) can signify irregular, and often illegal behaviour. Detection of such anomalies has been used to identify malicious individuals, including spammers, sexual predators, and online fraudsters. In their survey, they list existing computational techniques for detecting anomalies in online social networks. Also, they characterize anomalies as being either static or dynamic, and as being labelled or unlabelled, and they survey methods for detecting these different types of anomalies. They also suggest that the detection of anomalies in online social networks is composed of two sub-processes; the selection and calculation of network features, and the classification of observations from this feature space. In addition, this survey provides an overview of the types of problems that anomaly detection can address and identifies key areas for future research. Ranshous et al. (2015) also provides an overview of methods and techniques for detecting anomalies in dynamic networks. It covers various approaches and challenges in identifying unusual patterns or behaviors in networks that change over time, such as social networks or communication networks. Another survey by Kaur and Singh (2016) also observes that anomalous activities in OSNs can represent unusual and illegal activities exhibiting different behaviors than others present in the same structure. The authors describe different types of anomalies and their novel categorization based on various characteristics. A review of a number of techniques for preventing and detecting anomalies along with underlying assumptions and reasons for the presence of such anomalies is covered as well, altogether with data mining approaches used to detect anomalies. A special reference is made to the analysis of social network centered anomaly detection techniques which are broadly classified as behavior based, structure based and spectral based. Each one of these classifications further incorporates a wide number of techniques. In general, it is extremely common to use network analysis to extract network metrics that will inform different learning algorithms (Van Vlasselaer et al. 2015a, b; Van Vlasselaer et al. 2017) also in the field of anomaly detection in transaction networks (Van

Vlasselaer et al. 2015). Expert systems based on social network analysis were developed also for the detection of insurance frauds (Šubelj et al. 2011). Also, anomaly detection on temporal networks can be performed by identifying statistically significant graph evolution rules (Galdeman et al. 2023), thus detecting potential irregularities that may indicate unexpected deviations from the established norms.

Finally, Wang et al. (2017) address the problem of time-sensitive anomaly detection on road networks, in order to reveal unexpected traffic patterns whose identification could be helpful in road planning and management. They examine diverse anomalous traffic patterns, specifically identifying edges that exhibit complete disappearance or emergence between consecutive timeframes. They also consider patterns with a disparity in probabilities between the two time intervals exceeding a predefined threshold μ. Interestingly, and similarly to our approach, they will rank the entries of their final output, evaluating the effectiveness of the anomaly detection in terms of precision-at-k.

Anomaly detection in financial networks

Many efforts have been put in the scientific literature to address the many challenges posed by anomaly detection on financial data. Examining the AFC problem through the lenses of computer science and complex systems allowed the development of many interesting approaches. Such approaches usually exploit machine learning, deep learning and complex networks - or a mixture of them - to tackle the problem of detecting anomalies in financial transactions.

Indeed, complex networks are a powerful tool to model financial transactions: since financial data usually encodes an exchange of money between two or more entities, this interaction can easily be modeled as *edges* between two or more *nodes*. Network analysis applied to AFC, Anti-Money Laundering (AML), and more generally to the identification of anomalies in transaction networks is indeed growing in interest in recent years. García and Mateos (2021) describe the research lines and developments based on network analysis carried out from 2015 to 2020 by the Spanish Tax Agency for tax control. They present case studies demonstrating how network analysis has been effectively applied in a real world scenario. On the one hand, pattern detection algorithms on graphs can facilitate the identification of frauds; on the other, community identification and community detection techniques help to provide a more precise picture of the economic reality.

Colladon and Remondi (2017) emphasize the role of social network metrics in the detection of money laundering practices related to companies, analyzing different kinds of relational graphs to identify clusters of companies belonging to owners that were involved in court trials.

Garcia-Bedoya et al. (2020) highlight the limitations of traditional AML approaches, which rely on static analysis conducted days or months after transactions occur. They emphasize that in money laundering, the involved nodes often have complex, pre-determined connections like paths, cycles, or *smurf* transactions to obscure their activities. Liu et al. (2020) suggest that time-ordered transaction cycles may indicate money laundering in online networks. Jiang et al. (2022) conduct a comprehensive study on detecting motifs in financial graphs, exploring motif-based embeddings for various graph analysis tasks. Starnini et al. (2021) focus on smurfing, proposing an efficient method for identifying suspicious smurf-like subgraphs.

Finally, Akoglu et al. (2010) define OddBall, an anomaly detection algorithm based on complex networks that shares our same vision of the issue of anomaly detection. The authors seek for features, such as very basic centrality metrics, that present regular patterns that would allow for the identification of outliers. They compute the above metrics node-wise, considering the *egonetwork* of each node. However, contrarily to our approach, they work on static graphs, without taking into account the temporal dimension; the anomaly is computed on the values of the chosen metric at a certain time, without considering its evolution in time.

Many studies described above are based on the identification of specific patterns and, in some cases, on the study of their temporal evolution. These approaches have the limitation of having to know a priori the patterns to look for within the transaction network. Alternative techniques are based on machine learning (Chen et al. 2018) or deep learning via graph neural networks (extensively reviewed in this survey (Motie and Raahemi 2023)), that generate compact vector representations for each node. Several approaches have been proposed and applied to fraud detection (Kute et al. 2021; Dou et al. 2020; Shi et al. 2022; Liu et al. 2021). In Zhang et al. (2022), Zhang et al. develop a competitive graph neural networks (CGNN)-based fraud detection system, that uses some notion of normal behavior as weak supervision information for the model to build a profile of fraudulent users. In 2018, Weber et al. (2018) presented AMLSim,[7] a project intended to provide a multi-agent based simulator that generates synthetic banking transaction data together with a set of known money laundering patterns. The authors propose preliminary results showing that graph learning for AML is possible even in large sparse networks (1 M nodes and 9 M edges).

Given the lack of hand-labeled data to be used as a training set, many machine learning models are based on unsupervised anomaly detection (Chen et al. 2018); techniques to generate realistic synthetic datasets have also been developed (Altman et al. 2023); money laundering detection has been explored through zero-shot and meta-learning (Pan 2022).

Our contribution to the existing literature

Many of these approaches demonstrate both promise and effectiveness; however, they often do so at the cost of computational efficiency. In the AFC domain, this trade-off frequently results in an inability to fully meet at least one of the key criteria outlined in Sect. 3. A more detailed comparison and qualitative evaluation of some state-of-the-art algorithms discussed in this section are summarized in Table 10.

A notable advantage of network analysis is that, while many centrality algorithms can efficiently compute results for large graphs, their outcomes remain easily interpretable based on the chosen centrality measure. Dumitrescu et al. (2022) leverage basic node metrics and ego-networks as features for an anomaly detection algorithm, albeit on graphs smaller in size compared to those used in our work, and notably, they are labeled. Nevertheless, their results are highly encouraging for the application of simple network analysis-based methods in anomaly detection for financial transactions, similarly to the OddBall algorithm developed by Akoglu et al. (2010).

[7] https://github.com/IBM/AMLSim.

Table 10 A comparison between WeirdNodes and other anomaly detection methodologies applied to financial graphs is presented

	Interpretability	Unsupervised	Ranked	Node-wise	Temporal
WeirdNodes	✓	✓	✓	✓	✓
OddBall Akoglu et al. (2010)	✓	✓		✓	
Smurf-based Starnini et al. (2021)	✓				
ANOMALOUS Peng et al. (2018)	✓		✓	✓	
MAHINDER Zhang et al. (2022)				✓	
CARE-GNN Deguchi et al. (2014)		✓		✓	
eFraudCom Weber et al. (2018)			✓		
Pereira et al.'s method Pereira et al. (2019)	✓	✓		✓	✓

This includes both deep learning-based and traditional approaches. The various algorithms are evaluated based on the key characteristics of our approach to assess their suitability for our specific use case

Pereira et al. (2019) proposed a method for detecting changes in node behavior through the temporal analysis of centrality metrics. Their approach employs a change point scoring function, which quantifies deviations between the observed and expected values. The expected values are estimated using a weighted moving window average, and deviations exceeding a predefined threshold indicate behavioral change. This technique defines behavioral change at the individual node level, rather than relying on ranked information retrieval or assessing the overall stability of node metrics over time. In contrast, our method applies unsupervised ranking stability analysis to ensure network stability before identifying anomalies, accounting for potential perturbations that may affect a large number of nodes. Notice that Pereira et al.'s method is applied to the processing of evolving network streams, whereas WeirdNodes is designed to be applied to the comparison of two temporal snapshots.

Other aspects that differentiates our approach from the majority of previous systems is its applicability at different scales, and the stability check to estimate if models fluctuations are too large to efficiently detect anomalies. nodes can be aggregated into groups (in our financial networks, transactions between countries, banks, or accounts can be aggregated). Observing behavioral differences at various resolutions allows domain experts to zoom in and out of their data when irregularities are observed, providing an additional layer of interpretability to their analysis. However, models will be applied after an assessment on rankings stability, and they can reliable at a resolution, but unstable into another.

Conclusions, limitations, and future work

In this work, we proposed WeirdNodes, an anomaly detection procedure that applies complex network analysis to the Anti Financial Crime domain. This approach is designed to simplify the task of domain experts, as detailed in Sect. 3, by minimizing the time spent navigating large datasets for anomalies, and to get rid of the limitations of rule-based approaches that are widely used in this domain. Although there are many machine learning based anomaly detection systems that have amazing performance on general purpose problems, while some other systems is based on network analysis and the observation of centrality measures' dynamics, the unavailability to the public of financial transaction datasets, makes tailored solutions adopted internally in financial institutions - when implemented - almost invisible to the scientific community. We believe that the main contribution of this paper is the dissemination of a real-world case study, including the description of a large financial dataset, and the effort to adapt a procedure that can

be executed by the AFC expert, attacking the problem stated in 2, and satisfying the analyst's information needs. The simplicity of the procedure, and the understanding that we have of network metrics, which returns a list of the top-k suspicious nodes (countries or BICs), and that takes k as a tunable parameter, makes the result sets quite easy to interpret, allowing the procedure itself to be repeated on other time frames, and ultimately to zoom in/out the data resolution to look for further details.

We are aware of some relevant limitations of our proposal. First, the dataset we used for our experimental analysis cannot be attached to this paper due to privacy and legal constraints: researchers interested in analyzing this or similar data need to contact the Anti Financial Crime Digital Hub consortium (information in the Declarations section below). Second, while we recognize that this procedure is adaptable to problems from other fields, such as biology, genomics, and social networks, we have not been able to find a set of suitable datasets that would prove the more general applicability of WeirdNodes. Third, the procedure cannot be applied with sufficient confidence to explore the networks of individual transactions because of the failure of the stability checks at the IBAN level. Finally, we presented how to compare two different time frames (each with a given fixed length, e.g., a week, a month, a semester), but we had no opportunity to run the procedure on a sequence of different time frames during the research project. We suspect that a time-series analysis of the residuals dynamics over time could uncover more subtle changes in the network that take longer to emerge. For instance, a regular node might not abruptly sever all its outgoing links from one period to the next but instead gradually reduce the number and value of its transactions. Various models, such as AutoRegressive (AR), ARIMA, SARIMA, Exponential Smoothing, and LSTM, could be employed to predict trends and dynamics. This is a clear future direction of this work, and we think that WeirdNodes is a very promising tool to extract the variables to be analyzed with such tools.

Other future directions include the application of this approach to other domains, given the availability of datasets: the difficulty of finding publicly available financial datasets, and the promising results we obtained by inoculating our synthetic networks with controlled perturbations to simulate different evolving systems, may spur renewed efforts to create generative models that simulate as closely as possible the network properties of a real financial ecosystem like the one we studied. Another promising direction that we envision is flow analysis, which can also be performed at finer-grained levels after the procedure described in this paper has returned some BICs to be further explored: this type of analysis, which can be computationally expensive on very large graphs, can be effective if we can focus on subgraphs that have been found to be anomalous at a coarse-grained level: "follow the money" is probably still the best way to understand the sources and targets of illicit transactions, and automatic or semi-automatic new algorithms can effectively improve the reliability and productivity of the analyst.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1007/s41109-025-00702-1.

Supplementary material 1 (pdf 25541 KB)

Acknowledgements

The authors would like to acknowledge Federica Cena, Rossano Gaeta and Laura Li Puma for the feedbacks and insightful suggestions we received in the preliminary phase of this project.

Author contributions
SV, ATECL, GR, DM, MF, VR, SR designed the research. DM, MF, VR, SR provided the anonymized data. SV, ATECL, GR defined, implemented and tested the methodology. SV, ATECL, GR analyzed the data. SV, ATECL, GR, DM validated the results. SV, GR wrote and revised the manuscript. GR managed the project.

Funding
SV, ATECL, and GR acknowledge funding from the Anti Financial Crime Digital Hub consortium, whose members are Intesa Sanpaolo Innovation Center, University of Turin, Polytechnic University of Turin, and CENTAI. SV, ATECL, and GR started this collaboration when they were all affiliated at the University of Turin.

Data availibility
Real data of cross-country financial transactions is provided by Intesa Sanpaolo (ISP) and AFC Digital Hub. For more information, write to adh@pec.afcdigitalhub.com.

Declarations

Conflict of interest
The author(s) declare(s) that they have no conflict of interest.

Received: 16 July 2024 / Accepted: 23 March 2025
Published online: 28 April 2025

References

Akoglu L, McGlohon M, Faloutsos C (2010) Oddball: spotting anomalies in weighted graphs. In: Proceedings of PAKDD 2010, Hyderabad, India, June 21-24, 2010. Part II 14, pp 410–421. Springer
Altman E, Egressy B, Blanuša J, Atasu K (2023) Realistic synthetic financial transactions for anti-money laundering models. arXiv preprint arXiv:2306.16424
Chandola V, Banerjee A, Kumar V (2009) Anomaly detection: a survey. ACM Comput Surv 41(3):1–58. https://doi.org/10.1145/1541880.1541882
Chen Z, Le DV-K, Teoh E, Nazir A, Karuppiah E, Lam K (2018) Machine learning techniques for anti-money laundering (aml) solutions in suspicious transaction detection: a review. Knowl Inf Syst 57:245–285
Commission TE (2024) High risk third countries and the international context content of anti-money laundering and countering. https://t.ly/Lvwcd
Costenbader E, Valente TW (2003) The stability of centrality measures when networks are sampled. Soc Netw 25(4):283–307
Deguchi T, Takahashi K, Takayasu H, Takayasu M (2014) Hubs and authorities in the world trade network using a weighted hits algorithm. PLoS ONE 9(7):1–16
Dou Y, Liu Z, Sun L, Deng Y, Peng H, Yu PS (2020) Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp 315–324
Dumitrescu B, Bălțoiu A, Budulan Ş (2022) Anomaly detection in graphs of bank transactions for anti money laundering applications. IEEE Access 10:47699–47714
Fronzetti Colladon A, Remondi E (2017) Using social network analysis to prevent money laundering. Expert Syst Appl 67:49–58. https://doi.org/10.1016/j.eswa.2016.09.029
Galdeman A, Zignani M, Gaito S (2023) Unfolding temporal networks through statistically significant graph evolution rules. In: 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA), pp 1–10. IEEE
García IG, Mateos A (2021) Use of social network analysis for tax control in Spain. Hacienda Pública Española / Rev Public Econ 239(4):159–197
Garcia-Bedoya O, Granados O, Burgos J (2020) Ai against money laundering networks: the colombian case. J Money Laundering Control ahead-of-print. https://doi.org/10.1108/JMLC-04-2020-0033
Hassanzadeh R, Nayak R, Stebila D (2012) Analyzing the effectiveness of graph metrics for anomaly detection in online social networks. In: Wang XS, Cruz I, Delis A, Huang G (eds) Web information systems engineering - WISE 2012. Springer, Berlin, Heidelberg, pp 624–630
Hodge V, Austin J (2004) A survey of outlier detection methodologies. Artif Intell Rev 22:85–126. https://doi.org/10.1023/B:AIRE.0000045502.10941.a9
Jiang J, Hu Y, Li X, Ouyang W, Wang Z, Fu F, Cui B (2022) Analyzing online transaction networks with network motifs. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp 3098–3106
Kardos O, London A, Vinkó T (2020) Stability of network centrality measures: a numerical study. Soc Netw Anal Min 10:1–17
Kaur R, Singh S (2016) A survey of data mining and social network analysis based anomaly detection techniques. Egyptian Inf J 17(2):199–216. https://doi.org/10.1016/j.eij.2015.11.004
Kaur R, Kaur M, Singh S (2016) A novel graph centrality based approach to analyze anomalous nodes with negative behavior. Procedia Comput Sci 78:556–562. https://doi.org/10.1016/j.procs.2016.02.102
Kute DV, Pradhan B, Shukla N, Alamri A (2021) Deep learning and explainable artificial intelligence techniques applied for detecting money laundering-a critical review. IEEE Access 9:82300–82317. https://doi.org/10.1109/ACCESS.2021.3086230
Liu Z, Zhou D, Zhu Y, Gu J, He J (2020) Towards fine-grained temporal network representation via time-reinforced random walk. Proc AAAI Conf Artif Intell 34(04):4973–4980. https://doi.org/10.1609/aaai.v34i04.5936
Liu Y, Ao X, Qin Z, Chi J, Feng J, Yang H, He Q (2021) Pick and choose: a gnn-based imbalanced learning approach for fraud detection. In: Proceedings of the Web Conference 2021, pp 3168–3177
Manning CD, Raghavan P, Schütze H (2008) Introduction to Information Retrieval. Cambridge University Press, Cambridge, UK
Mihiri Shashikala HB, George R, Shujaee KA (2015) Outlier detection in network data using the betweenness centrality. In: SoutheastCon 2015, pp 1–5. https://doi.org/10.1109/SECON.2015.7133008

Mitchell C, Agrawal R, Parker J (2019) The effectiveness of edge centrality measures for anomaly detection. In: 2019 IEEE International Conference on Big Data (Big Data), pp 5022–5027. https://doi.org/10.1109/BigData47090.2019.9006468
Motie S, Raahemi B (2023) Financial fraud detection using graph neural networks: a systematic review. Expert Syst Appl 240:122156
Pan J (2022) Deep set classifier for financial forensics: an application to detect money laundering. https://doi.org/10.48550/ARXIV.2207.07863
Peng Z, Luo M, Li J, Liu H, Zheng Q et al. (2018) Anomalous: a joint modeling approach for anomaly detection on attributed networks. In: IJCAI, vol 18, pp 3513–3519
Pereira F, Tabassum S, Gama J, de Amo S, Oliveira G (2019) Processing evolving social networks for change detection based on centrality measures. In: Sayed-Mouchaweh M (ed) Learning from data streams in evolving environments. Studies in big data, vol 41. Springer, Cham. https://doi.org/10.1007/978-3-319-89803-2_7
Ranshous S, Shen S, Koutra D, Harenberg S, Faloutsos C, Samatova NF (2015) Anomaly detection in dynamic networks: a survey. WIREs Comput Stat 7(3):223–247. https://doi.org/10.1002/wics.1347
Savage D, Zhang X, Yu X, Chou P, Wang Q (2014) Anomaly detection in online social networks. Soc Netw 39:62–70. https://doi.org/10.1016/j.socnet.2014.05.002
Segarra S, Ribeiro A (2015) Stability and continuity of centrality measures in weighted graphs. IEEE Trans Signal Process 64(3):543–555
Shi F, Cao Y, Shang Y, Zhou Y, Zhou C, Wu J (2022) H2-fdetector: a gnn-based fraud detector with homophilic and heterophilic connections. In: Proceedings of the ACM Web Conference 2022, pp 1486–1494
Society TL (2024) High-risk third countries for AML purposes https://t.ly/qn2iK
Starnini M, Tsourakakis CE, Zamanipour M, Panisson A, Allasia W, Fornasiero M, Puma LL, Ricci V, Ronchiadin S, et al.: Smurf-based anti-money laundering in time-evolving transaction networks. In: Proceedings of ECML PKDD 2021, Part IV 21. Bilbao, Spain, Sept. 13–17, 2021, pp 171–186 (2021). Springer
Šubelj L, Furlan Š, Bajec M (2011) An expert system for detecting automobile insurance fraud using social network analysis. Expert Syst Appl 38(1):1039–1052
Van Vlasselaer V, Bravo C, Caelen O, Eliassi-Rad T, Akoglu L, Snoeck M, Baesens B (2015) Apate: a novel approach for automated credit card transaction fraud detection using network-based extensions. Decis Support Syst 75:38–48
Van Vlasselaer V, Eliassi-Rad T, Akoglu L, Snoeck M, Baesens B (2017) Gotcha! network-based fraud detection for social security fraud. Manage Sci 63(9):3090–3110
Van Vlasselaer V, Akoglu L, Eliassi-Rad T, Snoeck M, Baesens B (2015) Guilt-by-constellation: Fraud detection by suspicious clique memberships. In: 2015 48th Hawaii International Conference on System Sciences, pp 918–927. IEEE
Van Vlasselaer V, Eliassi-Rad T, Akoglu L, Snoeck M, Baesens B (2015) Afraid: fraud detection via active inference in time-evolving social networks. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, pp 659–666
Wang S, Zhang X, Cao J, He L, Stenneth L, Yu PS, Li Z, Huang Z (2017) Computing urban traffic congestions by incorporating sparse GPS probe data and social media data. ACM Trans Inf Syst (TOIS) 35(4):1–30
Weber M, Chen J, Suzumura T, Pareja A, Ma T, Kanezashi H, Kaler T, Leiserson CE, Schardl TB (2018) Scalable graph learning for anti-money laundering: a first look. CoRR arXiv:abs/1812.00076
Zaki AA, Saleh NA, Mahmoud MA (2023) Performance comparison of some centrality measures used in detecting anomalies in directed social networks. Commun Stat-Simul Comput 52(7):3122–3136. https://doi.org/10.1080/03610918.2021.1928192
Zhang G, Li Z, Huang J, Wu J, Zhou C, Yang J, Gao J (2022) efraudcom: An e-commerce fraud detection system via competitive graph neural networks. ACM Transactions on Information Systems (TOIS) 40(3):1–29

Publisher's Note