Visually Detecting Anomalies in Temporal Event Sequences: The Ghost Employee Case Study

Matthew Ray
Masters of Applied Data Science
Data Science and Society Department
University of North Carolina at Chapel Hill
Email: raymatt@unc.edu, raymm10@hotmail.com, matthew.m.ray4.mil@army.mil

***Abstract*—Detecting anomalies in temporal event sequences is notoriously difficult, especially when sequences are complex, sparse, and context-dependent. Conventional statistical and machine learning methods often lack the interpretability needed for exploratory and audit-focused analysis. In this paper, we introduce a visualization methodology centered around a meaningful temporal anchor, allowing users to intuitively identify anomalies through spatial reasoning and visual pattern recognition. Inspired by real-world issues in Army Human Resources and demonstrated using a synthetic financial fraud case, our method emphasizes interactive control, flexible event ordering, and cognitive alignment. We contrast our approach with prominent systems like EventFlow, DecisionFlow, and EVA, outlining how it overcomes their limitations in interpretability and pattern clarity. Ultimately, our design better supports human-in-the-loop analysis for detecting nuanced or adversarial anomalies.**

I. Introduction

The analysis of temporal event sequences is an essential task across a wide range of domains, including healthcare, fraud detection, human resource systems, and cybersecurity. Yet despite the proliferation of high-dimensional data and increasingly complex behaviors, tools that support intuitive, accurate anomaly detection remain lacking. In particular, anomaly detection across these temporal domains often suffers from three intersecting challenges: sparsity, contextual dependency, and interpretability. Most statistical or machine learning models built to tackle anomaly detection optimize for accuracy, but offer little to no support for exploration, transparency, or human judgment. These models often require strong assumptions about event frequency or known labels, and even when effective, they generally fail to articulate ”why” an event is anomalous.

Our work began with a practical challenge in Army Human Resources—making sense of dense and scattered personnel data across an employee's career timeline. We noticed that when trying to evaluate events chronologically (e.g., promotions, training, leaves), many anomalies were not clear in the data itself but only surfaced when viewed contextually. In these datasets, sparsity is not an edge case—it is the norm. Events often appear just once, such as a pay grade reassignment, a document submission, or a deployment order, and their relevance is heavily tied to temporal positioning and human process expectations.

This problem led us to conceptualize a new approach: visualizing temporal sequences with a human-centric model, structured not around raw time but around contextually anchored timeframes. Instead of treating time as a linear continuum, we treat it as a flexible axis with anchors that give meaning—like a hire date, a suspicious login, or a contract approval. From this anchor, we can examine every other event as a relative distance. This simple shift dramatically increases human interpret-ability and reveals patterns that statistical analysis methods miss or aggregate away.

To illustrate the utility of our method while protecting privacy and reducing complexity, we constructed a synthetic financial fraud case around a ghost employee. This fictitious character, John Doe [E999], is fabricated by a bad actor within an organization. The bad actor carefully backdates documents, submits process logs, and forges digital activity to blend in with real employee data. We make the assumption this individual is in a position of authority to oversee hiring and has some level of access to insert records into the HR System. But in doing so, they create sequences of events that don't match the patterns of real employees. Our visualization system, built around temporal anchoring and interactive features like reordering and filtering, makes these discrepancies immediately obvious.

This paper introduces that visualization methodology, details the principles behind its architecture, compares it against existing methods in anomaly detection and visual analytics, and demonstrates its value through the lens of synthetic—but realistic—fraud scenarios.

II. Related Work

Anomaly detection in temporal data is a longstanding and active area of research. However, most traditional approaches are either statistical in nature or based on black-box machine learning models, and they often fall short in terms of explainability and interactive analysis. Our approach is informed both by gaps in these existing methods and by emerging work in visual analytics for anomaly detection.

EventFlow, developed by Plaisant et al. [1], was one of the first widely-used systems for visualizing temporal patterns in patient record data. It introduced key mechanisms for sequence alignment and cohort comparison, but the tool was heavily oriented around known outcomes and hypothesis-driven queries. It lacked support for exploratory detection of

edge-case anomalies and required significant manual effort to configure effective views.

DecisionFlow, proposed by Gotz and Stavropoulos [2], advanced the field by enabling more flexible visualization of high-dimensional sequences and introducing milestone-based comparisons. This system introduced useful abstractions for temporal correlation but still leaned heavily on outcome analysis and statistical filtering. It did not fully address the challenge of identifying subtle anomalies embedded in sparse, real-world event flows.

Recent developments in deep learning, such as those explored by Guo et al. in EventThread3 and its extensions [3], [4], apply variational autoencoders (VAEs) and sequence reconstruction to identify outlier event sequences. These systems use latent vector comparisons and reconstruction probability maps to flag anomalies. While effective in terms of detection performance, they often struggle with interpretability—analysts must interpret model outputs through multiple coordinated views, and the detection logic remains opaque to non-specialists.

In financial fraud contexts, Leite et al. introduced EVA [5], a visual scoring system for detecting suspicious transactions. The EVA interface combined scatterplots, temporal sequences, and risk metrics, yet placed heavy cognitive demand on the user. EVA also relied on rule-based scoring and did not effectively support contextual, user-driven exploration of temporal anomalies.

More broadly, Leite et al. [6] and Shi et al. [7] emphasized the need for visualization systems to support domain experts—those without ML backgrounds—in understanding and exploring anomalies. Their surveys point to interaction mechanisms like sequence comparison, opacity control, and filtering as essential tools for interpretation. Similarly, Liu et al. [8] outlined principles for integrating visual tools with machine-driven anomaly detection pipelines, supporting annotation, zooming, and subgroup analysis. These ideas align directly with our system design, standing out as tested methods for tackling complex visualizations - reducing clutter and improving human cognition.

While the body of related work has made impressive strides in performance and dimensionality, we argue that few systems fully align with the way humans intuitively perceive temporal anomalies—especially when events are sparse, dependent on organizational policy, or inserted with malicious intent. Our work seeks to fill that gap by combining visual pattern recognition, intuitive anchoring, and flexible interaction into a tool tailored for exploratory, human-centric anomaly analysis.

III. Design Methodology

Our visualization system is built on a foundation of human-centered design principles, informed by practical analysis needs in HR, finance, and auditing contexts. Unlike tools that rely solely on statistical modeling, our system empowers users to identify anomalies through interactive visual exploration, cognitive pattern recognition, and contextual anchoring. The architecture was iteratively designed to reflect how humans process irregular sequences, not by measuring z scores or vector distances, but by identifying what 'does not look right'. Humans have powerful pattern recognition when given a singular focus and long exposure. We rely on the assumption that the auditor or analyst is very familiar with their data.

At the core of our design is a temporal anchoring mechanism. Rather than plotting events along an absolute timeline, we allow the user to define a central reference point—a temporal "anchor"—from which all other events are measured as relative distances. This anchor can be the hire date, first large commission, or any significant event of analytical interest. This concept forms the vertical axis and enables a truncation or expansion relative to the anchor. This enables the dynamic measure to be introduced as days or years. Meanwhile, the event categories are distributed along the X-axis. The result is a clean, cognitive-friendly view of event spread across time, as seen in Fig. 1.

A. Central Temporal Anchoring

The baseline view (Fig. 1) shows all employees' event trajectories aligned to a common anchor—typically the hire date. This makes the dataset immediately interpretable: events stack and align across individuals, and patterns become visually distinguishable through their relative timing. For example, early onboarding events cluster near the top of the plot, while milestone or benefit-related events fan out further below.

When a user changes the anchor to another event, such as a large commission (see Fig. 2), the entire dataset repositions. This reorientation allows users to analyze whether key events are being properly preceded by validation steps. In the ghost employee case, we can observe that financial transactions preceded necessary HR clearances—an inversion of typical workflow.

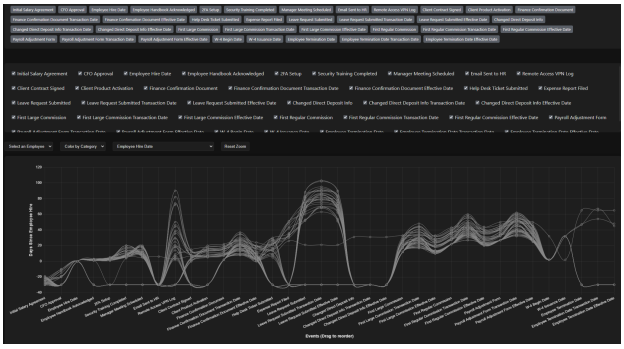

Fig. 1. Default state with central anchoring on hire date. Events are plotted as vertical splines across event categories.

B. Event Reordering and X-Axis Control

The X-axis represents the categories of events (e.g., documents, transactions, approvals). These are draggable and reorder-able in real time. By reordering the X-axis (Fig. 3), users can align the view to reflect organizational process flow, such as placing approvals before transactions. This function

Fig. 2. Anchor shifted to “First Large Commission.” The employee timeline realigns to highlight events occurring before/after the new anchor.

allows out-of-order behavior—like a CFO approval following payroll submission—to visually stand out. In our ghost scenario, the bad actor knew an initial salary agreement was required, but was not aware of the internal HR process marked ”CFO Approval” in the database.

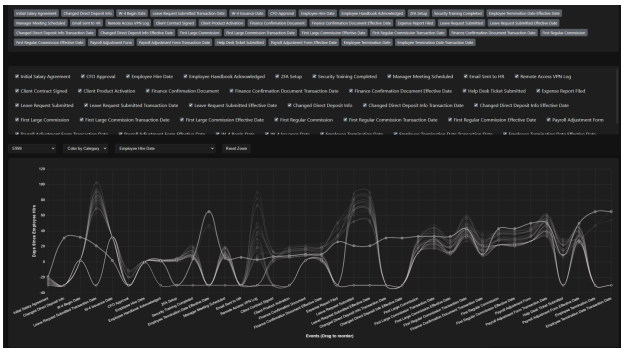

Fig. 3. Event categories reordered to reflect expected HR process. Out-of-order behavior becomes visually clear.

C. Selection and Highlighting of Anomalies

Users can select any individual (or filtered group) to highlight their sequence. In Fig. 4, the ghost employee’s spline is selected. Their event trajectory sharply deviates from population norms—showing compressed intervals, back-to-back events, and incorrectly ordered documentation as well as ”perfect” HR processing time - appearing as horizontal patterns.

D. Subgroup Coloring and Filtering

Role-based or attribute-based subgroups can be dynamically colored to support pattern discovery across cohorts. In Fig. 5, each spline is assigned a color corresponding to job function. This allows visual clustering—e.g., all support staff exhibit similar event spacing—while deviations, like a manager’s unusually short onboarding, become more obvious, or if the employee perfectly completed all online training within one day - which would appear as a straight line, implying zero difference along the central measure.

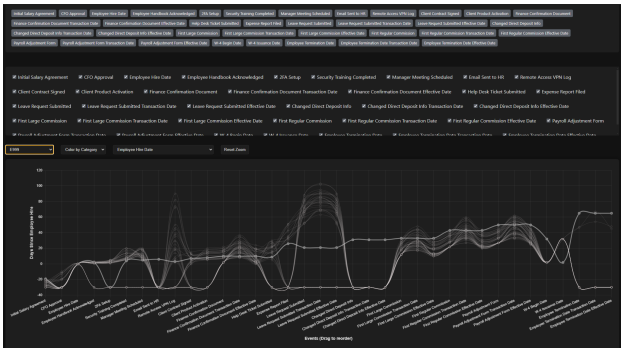

Fig. 4. The ghost employee (E999) selected. Note unusual spacing and order of document-related events.

Users can further filter the dataset to isolate a subgroup (Fig. 6), reducing clutter and enhancing comparison fidelity. Zooming in (Fig. 7) reveals precise deviations in timing or document progression unique to the filtered cohort.

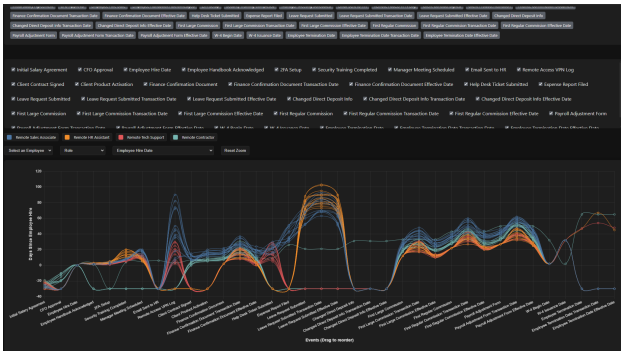

Fig. 5. Subgroup coloring by employee role. Visual clustering highlights population-based trajectory norms.

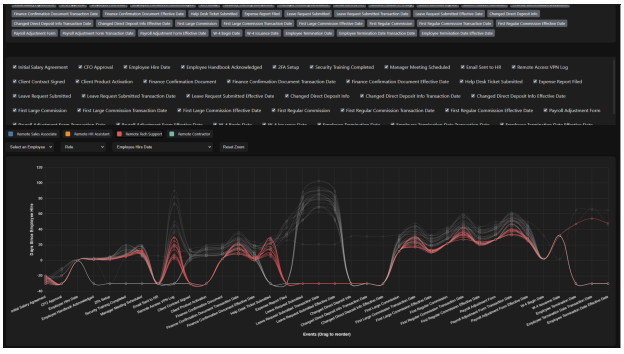

Fig. 6. Filtered view of a single subgroup (e.g., remote technical support) to analyze intra-group consistency.

E. Spline Linkage and Visual Cognition

Each employee’s event path is drawn using spline curves, not straight lines, to exploit human perception of continuity and rhythm. Smooth transitions make it easier to spot inconsistencies—such as abrupt timing gaps or compressed intervals. When a spline sharply bends or forms a V-shape between

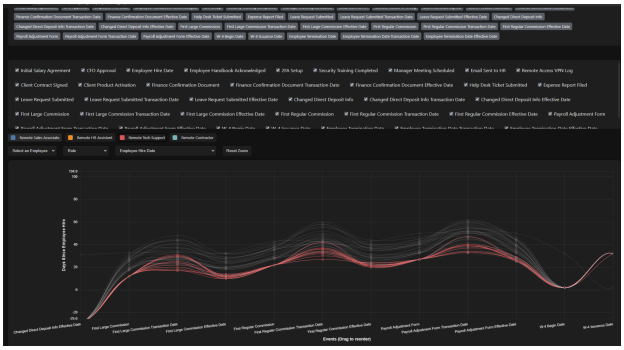

Fig. 7. Zoomed-in view of a filtered subgroup. Fine-grained analysis reveals precise inconsistencies.

two normally spaced categories, it signals a deviation worth investigating. These curves serve as a visual fingerprint of behavior, much like a waveform. We handle Null value anchors by providing a default -29.9 value denoting the sharp decline in missing information. This assists in moving the data out of the chart view, and yet still drawing attention to missing information.

The design of our system, from anchoring and reordering to highlighting and visual smoothing, aims to harness the analyst's ability to spot patterns at a glance. It avoids overloading the interface with scores, outlier flags, or statistical thresholds, and instead trusts the human-in-the-loop to explore, interpret, and act.

F. Synthetic Data Generation and Insider Threat Simulation

To avoid the ethical and operational challenges of using real personnel records, our system was developed and validated using a synthetically generated dataset. However, this synthetic data was not randomly constructed—it was carefully engineered to simulate a realistic insider threat scenario. Specifically, we created records for a fictitious employee whose data mimics what a well-informed malicious actor might insert: events that occur in the correct sequence, use the correct terminology, and appear within statistically acceptable timeframes.

This "insider-informed" data was key to validating our approach. The ghost employee's timeline was intentionally constructed to bypass traditional audit filters and trigger no flags based on basic statistical models. Only when viewed visually, in context with population norms and expected procedural ordering, did their sequence stand out as anomalous.

This design assumption, that adversaries often act just inside the limits of normality, underpins our choice to prioritize interpret-ability and human pattern recognition over automated thresholding. A human viewer, presented with anchored, ordered, and colored sequence views, can spot these edge-case anomalies that rules-based engines are not designed to catch.

IV. Case Study: Financial Fraud via Ghost Employee

To validate our visualization design, we created a synthetic—but structurally realistic—fraud scenario based on ghost employee schemes. In this example, a finance department supervisor named Alex inserts a fictitious employee, John Doe, into the organization's HR and payroll systems. This narrative simulates the subtle complexities that make adversarial fraud hard to detect with traditional tools.

The fraud is well-orchestrated. Alex assigns John Doe a unique employee ID and backdates the hire date to coincide with a known hiring surge, ensuring the fraud blends into HR trends. He then populates documents like salary agreements, on-boarding forms, and security clearances with forged timestamps. These events are entered into systems that normally do not cross-check submission latency or inter-departmental sequencing. On the surface, John Doe appears legitimate.

But within our visualization, several inconsistencies immediately surface.

A. Unrealistic Document Processing Time

One red flag is the simultaneous submission and processing of HR forms. Real employees typically experience delays of 10–20 days between form submission and HR system acknowledgment. In Fig. 4, John Doe's W-4, direct deposit setup, and training verification all occur within the same 24-hour window, and directly on the hire date anchor. While this may appear efficient in isolation, when plotted relative to population norms, the timing appears unnaturally compressed.

B. Exploratory Reordering and Rapid Hypothesis Testing

The ability to dynamically reorder event categories along the X-axis transforms the visualization from a static report into a true exploratory environment. Rather than needing to predefine a theory, write custom queries, or engineer logic tests in SQL or Python, analysts can simply rearrange the order of event types and watch how timelines shift in response.

This reordering helps surface inconsistencies or irregular event logic visually. For example, placing authentication events before provisioning can quickly reveal whether anyone accessed the system prior to having credentials. Or, reordering payment-related steps next to onboarding forms may reveal sequences that appear valid at a glance but collapse under scrutiny.

What makes this particularly powerful is the low friction involved—users don't need to formalize a hypothesis before exploring it. If something feels off, they can act on that instinct visually. The interface invites investigation by letting users go down "rabbit holes" without fear of breaking logic chains or writing incorrect filters. An analyst can follow a hunch, adjust the event timeline structure, and immediately test whether an individual's behavior diverges from organizational norms.

In systems where processes evolve or documentation is inconsistent, this kind of visual reordering reveals more than rigid rulesets can. It empowers human curiosity—enabling spontaneous pattern detection, outlier chasing, and deeper contextual analysis.

C. Missing Dependencies

Even more telling is the absence of key documents. Most contractors have a CFO-signed 1099 form and associated tax

review logged within 5–10 days of their salary agreement. John Doe lacks any CFO approval or signed financial review. By toggling event visibility and opacity (as in Figs. 6 and 7), analysts can zero in on which records are consistently missing across multiple dimensions.

D. Fabricated Sequences with Misaligned Norms

Fraudsters, even sophisticated ones, tend to make two mistakes: they move too quickly, and they fail to replicate the irregularities found in authentic human processes. Most real employees have noise in their timelines—unexpected lags, redundant submissions, rescheduled training. In contrast, John Doe's events follow a clean, idealized path with no delay, no overlap, and perfectly ordered steps. This uniformity becomes a giveaway when compared against real data.

E. Leverage for Audit and Risk Review

The anomaly visualization becomes a communication tool, not just a detection engine. A security team, finance reviewer, or compliance analyst can highlight John Doe, anchor on his first commission payment (Fig. 2), and walk stakeholders through every violation of expected workflow logic. This is particularly valuable for situations where the burden of proof rests on clear explanation—auditors and investigators can visually explain what is wrong without needing to write complex logic rules or machine learning justifications.

Our system doesn't just help detect fraud, it helps explain it.

V. Comparison with Machine Learning and Broader Applications

Machine learning has made substantial contributions to anomaly detection in temporal data. Unsupervised models such as autoencoders, recurrent neural networks (RNNs), and more recently, variational autoencoders (VAEs), have been used to identify outliers in complex sequences [3], [4]. These models are excellent at generalization and scale well for large datasets. However, they suffer from two major drawbacks in practical settings: opacity and misalignment with human intuition.

Most ML-based systems do not provide users with a transparent reason for their detection outputs. A flagged anomaly might be an outlier in latent space but offer no insight into which specific events caused it, what sequence violated protocol, or how the pattern deviated from normalcy. The burden then falls on the analyst to reverse-engineer the rationale, often without access to the model's internal logic or feature embeddings. In contrast, our system makes the deviation visually obvious. The user sees when and where something went wrong—and more importantly, how it differs from the expected pattern.

Another challenge in ML-based systems is handling sparsity and contextual noise. These models rely heavily on high-quality, labeled training data. But in real-world applications like HR, medical records, or financial workflows, many events happen only once or follow no fixed pattern. These edge cases are either missed entirely by the model or incorrectly flagged due to lack of historical precedent. Our method, by design, thrives in these "rare event" zones. Because it leverages human cognition, it doesn't require a statistically significant number of similar cases to detect irregularities—it just needs the outlier to visually stand out.

A. Human-in-the-Loop and Semantic Understanding

Rather than competing with machine learning, our method complements it. The visualization can serve as an exploratory tool to generate features that can later be embedded into ML pipelines. Conversely, ML outputs—such as probability scores or risk labels—can be overlaid on the visualization to assist in human decision-making. This hybrid approach aligns with the vision of human-in-the-loop AI, where interpretation, judgment, and domain expertise play an essential role.

The tool is particularly strong at identifying what we call "semantic anomalies." These are not violations of statistical expectation but of business process logic—like a direct deposit being filed before a contract is signed. A machine may overlook this as "normal" if the sequence appears frequently enough. But a human knows it defies real-world constraints. Our tool is built to surface and spotlight these scenarios quickly.

VI. Broader Applications and Future Work

While our system was demonstrated using a synthetic HR fraud scenario, the methodology generalizes to a wide range of domains where event sequences are irregular, sparse, and dependent on contextual order. In healthcare, temporal anomalies can occur when treatments, lab results, or medications are administered out of order—potentially with severe consequences. These are often missed by statistical thresholds but stand out visually when the event sequences are anchored and aligned.

In cybersecurity, seemingly normal login or access events may form suspicious sequences when analyzed temporally—for instance, a pattern of escalating privileges or lateral movement across systems. Similarly, in personnel or clearance systems, renewal cycles, duty assignments, or benefits administration may follow inconsistent paths that suggest negligence or insider abuse.

Our approach also fits naturally into audit, compliance, and regulatory environments. In these contexts, the challenge is not only detecting anomalies but explaining them clearly to stakeholders. Because our system surfaces discrepancies in timing, order, and documentation with visual cues rather than algorithmic abstractions, it supports defensible, human-readable narratives. Analysts can walk auditors or decision-makers through the sequence of events visually, showing not just that something is wrong—but how and why.

Looking ahead, we envision enhancements that blend visual exploration with machine intelligence. A recommendation engine could suggest potential anchors based on variance, or flag sequences with rare spline shapes. Shape-based clustering could group users with similar behavioral timelines, allowing for cohort-based comparisons. Natural language tooltips could describe why a sequence appears anomalous ("W-4 filed

before hire date; approval missing"), helping less technical users understand and act on findings.

We also anticipate integrating with supervised learning workflows. For example, analysts could export spline-based features, such as the time between the anchor and the first transaction, as input variables to a predictive model. This creates a feedback loop between intuitive, visual sensing and robust, data-driven decision support. Ultimately, our system is not just a visualization layer, but a launchpad for deeper analytics and human-centered insight.

VII. Conclusion

Temporal anomalies are rarely loud. They do not always appear in red text or trigger system alarms. More often, they hide in plain sight—blending into what "looks normal" to a machine, but not to a person. Our work sets out to build a system that trusts the analyst's eye and judgment, providing tools that extend those abilities rather than replace them.

By anchoring temporal events, dynamically ordering categories, and filtering views semantically, we help analysts see what systems miss: subtle misorderings, improbable timing, and too-perfect sequences. These traits often define insider threats, frauds, or systemic failures that fall through the cracks of rule-based checks and automated scans.

We believe visual cognition is still one of the most underutilized superpowers in anomaly detection. And we aim to prove that with the right design, one person and one visual interface can outperform complex models in the places where it matters most.

References

[1] C. Plaisant, "Visualization of temporal patterns in patient record data," *Fundamental & Clinical Pharmacology*, vol. 32, no. 1, pp. 85–87, 2018. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5771833/

[2] D. Gotz and H. Stavropoulos, "Decisionflow: Visual analytics for high-dimensional temporal event sequence data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 1783–1792, 2014.

[3] S. Guo, Z. Jin, Q. Chen, D. Gotz, H. Zha, and N. Cao, "Visual anomaly detection in event sequence data," in *2019 IEEE International Conference on Big Data (Big Data)*, 2019, pp. 2161–2170.

[4] ——, "Interpretable anomaly detection in event sequences via sequence matching and visual comparison," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 12, pp. 4531–4544, 2022.

[5] R. A. Leite, T. Gschwandtner, S. Miksch, S. Kriglstein, M. Pohl, E. Gstrein, and J. Kuntner, "Eva: Visual analytics to identify fraudulent events," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 330–339, 2018.

[6] R. A. Leite, T. Gschwandtner, S. Miksch, E. Gstrein, and J. Kuntner, "Visual analytics for event detection: Focusing on fraud," *Visual Informatics*, vol. 2, no. 4, pp. 198–212, 2018.

[7] Y. Shi, Y. Liu, H. Tong, J. He, G. Yan, and N. Cao, "Visual analytics of anomalous user behaviors: A survey," *arXiv preprint arXiv:1905.06720*, 2019. [Online]. Available: https://arxiv.org/abs/1905.06720

[8] D. Liu, S. Alnegheimish, A. Zytek, and K. Veeramachaneni, "Mtv: Visual analytics for detecting, investigating, and annotating anomalies in multivariate time series," *arXiv preprint arXiv:2112.05734*, 2021. [Online]. Available: https://ar5iv.labs.arxiv.org/html/2112.05734