Knowledge-Based Systems 351 (2026) 116662

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Bridging dynamics and semantics: A unified perspective on explainable Graph Neural Networks based stream reasoning

Gözde Ayşe Tataroğlu Özbulak [a,b,*], Yash Raj Shrestha [a], Jean-Paul Calbimonte [b,c]

[a] Department of Information Systems, University of Lausanne, Lausanne, 1015, Vaud, Switzerland
[b] University of Applied Sciences and Arts Western Switzerland HES-SO, Sierre, 3960, Valais, Switzerland
[c] The Sense Innovation and Research Center, Lausanne, 1007, Vaud, Switzerland

ARTICLE INFO

Keywords:
Graph Neural Networks
Explainable artificial intelligence
Stream reasoning
Knowledge graphs

ABSTRACT

This paper presents a structured review of explainable stream reasoning with Graph Neural Networks (GNNs) in settings where graph structures and background knowledge evolve over time. Although prior work has advanced GNN modeling, temporal reasoning, and Knowledge Graph (KG) integration, the literature remains fragmented regarding how explanations should be generated, semantically grounded, and evaluated in knowledge-enriched graph streams. Existing studies provide limited guidance on how explanations should align with ontologies, preserve relational coherence, and remain stable under temporal evolution. As a review article, this paper provides a taxonomy-driven synthesis of GNN explanation methods, semantic integration patterns, and stream-oriented constraints rather than a new benchmark or deployment system. It analyzes how established explanation families apply to evolving, knowledge-enriched graphs and systematizes KG integration strategies for reasoning and explanation, including embedding-based, message-passing, neuro-symbolic, and KG-assisted approaches. The review also presents an implementation-oriented architectural roadmap illustrating how semantic constraints can be incorporated into temporal message passing. Building on this synthesis, the paper proposes knowledge-aware evaluation dimensions, including semantic fidelity, relational coherence, temporal semantic stability, and human/domain-centered alignment. Where possible, these dimensions are accompanied by illustrative formal metric definitions, while their standardization remains open. A bounded empirical feasibility illustration on temporally ordered healthcare graph data demonstrates the computability of selected dimensions. Representative use cases in healthcare, security, social media, and autonomous systems further illustrate how the proposed perspective can guide future research on trustworthy and semantically grounded GNN-based stream reasoning.

1. Introduction

1.1. Motivation

Graph Neural Networks (GNNs) have become a central paradigm for learning over complex graph structures in domains such as bioinformatics, recommender systems, and applications centered on Knowledge Graphs (KGs) [1–3]. Their ability to jointly exploit node and edge features together with graph structure has led to state-of-the-art performance in node-level, link-level, and graph-level prediction tasks, including large-scale KGs [4,5]. In high-stakes settings such as healthcare and patient monitoring, recent work on patient-centric KGs highlights the need to combine expressive graph models with rich domain knowledge [6,7]. At the same time, real-world decision-making increasingly occurs in dynamic streaming environments, where both data and associated knowledge evolve continuously over time. GNNs encode relational inductive biases that align naturally with structured and semantic reasoning tasks [8–11].

Recent research addresses different aspects of this problem space, but existing efforts remain largely fragmented. Surveys on knowledge graph reasoning over static, dynamic, and multimodal graphs, together with studies on streaming KGs, emphasize the increasing importance of temporal and streaming aspects in semantic reasoning pipelines [9,12]. In parallel, explainability methods have been proposed for temporal and dynamic GNNs, including motif-based explanations for temporal GNNs, self-explainable temporal architectures based on information bottlenecks, and counterfactual explainers for dynamic graphs [13–15]. Complementary work on explainable reasoning over temporal KGs and on explainable GNN-based recommenders shows that semantic

* Corresponding author at: Department of Information Systems, University of Lausanne, Lausanne, 1015, Vaud, Switzerland.
E-mail address: gozdeayse.tatarogluozbulak@unil.ch (G.A. Tataroğlu Özbulak).

https://doi.org/10.1016/j.knosys.2026.116662
Received 30 January 2026; Received in revised form 22 June 2026; Accepted 14 July 2026
Available online 20 July 2026

and temporal constraints are critical for generating user- and domain-understandable explanations in practice [16,17]. However, these efforts typically focus either on temporal GNN architectures, on stream reasoning with KGs, or on specific application settings [11,18,19].

1.2. Problem statement

Despite substantial advances in GNNs, explainability methods, and KG-based semantic enrichment, there is still no unified framework for synthesizing and rigorously evaluating semantically grounded explanations in GNN-based reasoning over evolving, knowledge-enriched graph streams. Existing research typically addresses these dimensions in isolation:

- *Dynamic and temporal graph studies* primarily focus on representation learning under evolving structure, without explicit consideration of explanation quality or semantic grounding [9,20,21].
- *GNN explainability evaluation metrics* propose criteria for faithfulness, stability, and robustness, but largely assume static graphs and do not account for external knowledge or semantic consistency [22–24].
- *Knowledge-enhanced GNN surveys* emphasize improved reasoning performance and interpretability through semantic integration, yet offer limited guidance on how explanations should be evaluated in dynamic or streaming settings [25,26].

As a consequence, it remains unclear how to systematically assess whether explanations produced by GNN-based stream reasoning models are semantically faithful to KGs, coherent at the relational level, and stable under temporal evolution. This lack of systematic evaluation criteria hinders principled comparison of existing methods and limits the development of trustworthy systems.

1.3. Research gaps

While GNNs, explainability techniques, and semantic enrichment driven by KGs have advanced rapidly, several core issues surrounding explainable semantic stream reasoning remain underexplored and insufficiently systematized. Specifically, current literature exhibits three distinct gaps:

- *Dominance of structural over semantic explainability*. Existing explainability methods for GNNs primarily emphasize structural relevance and statistical features. However, the semantic validity of these explanations regarding external knowledge representations remains underexplored. Consequently, current approaches often fail to anchor their outputs in the underlying domain ontology, producing explanations that are mathematically faithful but semantically disconnected.
- *Lack of a unified framework linking stream reasoning, semantics, and explainability*. Research on dynamic GNNs, knowledge integration, and explainability has largely progressed in isolation. As a result, the interaction between continuously evolving graph structures and semantic knowledge has not been systematically analyzed from an explainability perspective. There is currently no consolidated framework that treats streaming dynamics, semantic enrichment, and transparency as interconnected design dimensions within a single system.
- *Inadequacy of evaluation protocols for knowledge-enriched settings*. Current evaluation practices rely heavily on generic metrics such as faithfulness, sparsity, and robustness. These measures offer little guidance on how to assess semantic fidelity, relational coherence, or consistency with background knowledge. There is a marked absence of standardized protocols designed to quantify these semantic properties, particularly when explanations must be generated and validated over evolving data streams.

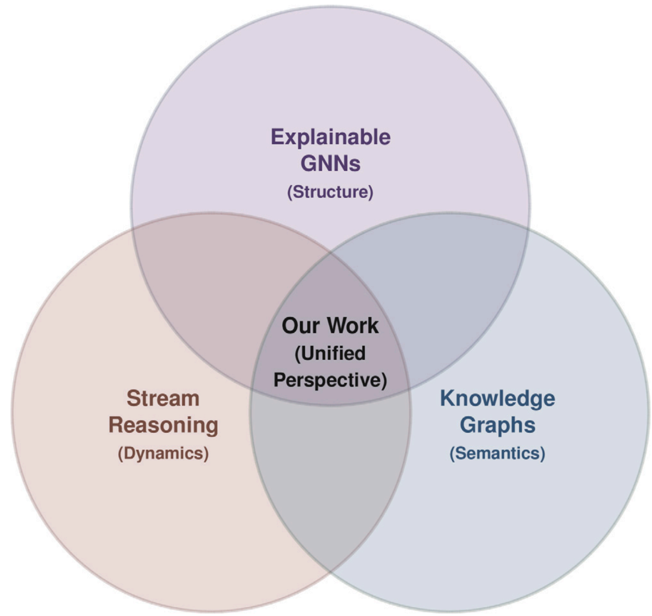

Fig. 1. Scope of the survey located at the intersection of Explainable GNNs (Structure), Stream Reasoning (Dynamics), and Knowledge Graphs (Semantics). The diagram illustrates the unified perspective proposed in this work.

1.4. Literature scope and review strategy

To ensure transparency and reproducibility, this study follows the systematic literature review guidelines established by Kitchenham et al. [27]. The relevant literature is currently fragmented across the distinct domains of explainable GNNs, KGs, and stream reasoning. Therefore, a unified semantic perspective requires systematic synthesis of these isolated research areas. Consequently, we adopt a protocol-driven literature selection strategy that mitigates selection bias while cohesively integrating studies at their intersection. The scope is explicitly defined to identify influential works located at the convergence of these three domains. As visualized in Fig. 1, this study specifically examines the unified perspective at their intersection, while considering not only methods explicitly designed for online stream-oriented reasoning, but also temporally and dynamically evolving graph methods that are analytically relevant to stream-oriented reasoning under semantic and explainability constraints. To capture this intersection in a systematic yet field-sensitive manner, we adopted the foundational Kitchenham protocol while adapting the inclusion criteria to reflect the rapid evolution of the field.

Search protocol: Search strings included "explainable GNN" AND ("stream reasoning" OR "temporal graph" OR "knowledge graph"). Time window: 2018–2025. *Inclusion criteria:* peer-reviewed journal/ /conference papers, English, full-text accessible. *Exclusion criteria:* non-GNN methods, static-only graphs. *Inter-rater reliability:* Two independent reviewers achieved 95% agreement on final inclusions (92 studies). In this regard, Fig. 2 provides a schematic overview of the protocol and filtering criteria used to curate the reviewed studies. This approach prioritizes the construction of a unified theoretical framework over a statistical aggregation of prior results.

1.5. Research questions

Building on the research gaps identified above, this study is guided by the following research questions, which aim to clarify how explainability, semantic enrichment, and streaming constraints can be jointly addressed in GNN-based stream reasoning systems.

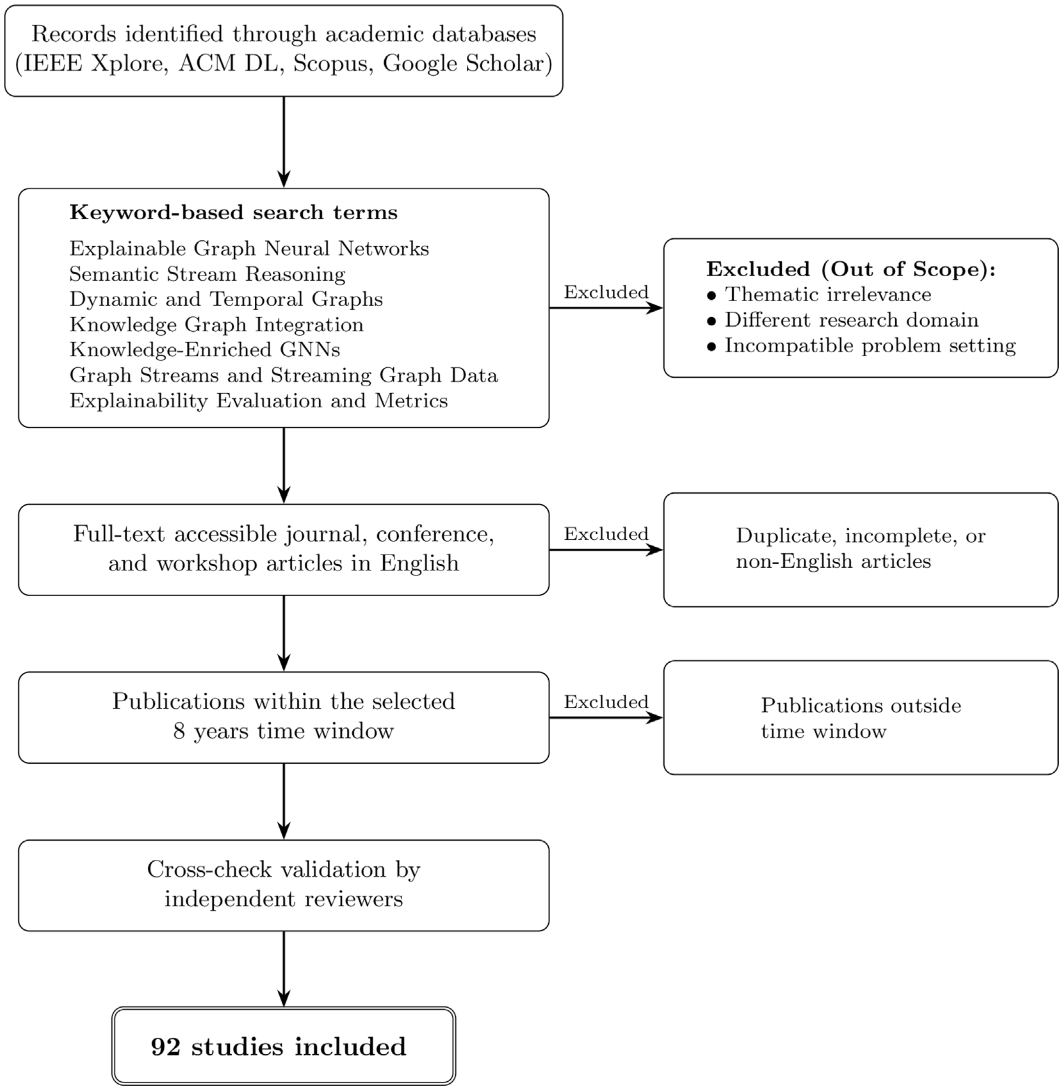

Fig. 2. Schematic view of the systematic literature search and scope definition process.

RQ1 *– Semantic grounding.* How can explanations for GNN-based stream reasoning be structured so that they are not only locally faithful to the underlying model, but also semantically consistent with domain specific ontologies?

RQ2 *– Dynamics and semantics.* How can semantic knowledge be integrated into dynamic and temporal GNNs such that explanations remain coherent and informative as both the graph structure and the associated background knowledge evolve over time?

RQ3 *– Knowledge-aware evaluation.* Which evaluation criteria and dimensions are required to assess the semantic fidelity, relational coherence, and knowledge consistency of explanations produced by GNN-based stream reasoning models?

RQ4 *– Unified perspective.* How can explainability, semantic enrichment through KGs, and stream reasoning constraints be combined into a unified conceptual framework for the analysis and design of trustworthy GNN based stream reasoning systems?

To systematically address these research questions, the remainder of this paper is structured as follows: Section 3 provides a taxonomy of explainability methods, establishing the structural foundation required for RQ1. Section 4 focuses on the integration of KGs into dynamic GNNs, addressing the semantic and temporal challenges posed in RQ2. Section 5 responds to RQ3 by defining knowledge-aware evaluation dimensions. Finally, Sections 6 and 7 synthesize these components into a unified perspective and demonstrate its application in critical domains, thereby addressing RQ4.

1.6. *Objectives and expected contributions*

To bridge the fragmentation identified in the literature, this study establishes a unified perspective on stream reasoning that is explainable and enriched with knowledge. The primary objective is to move beyond isolated analysis by synthesizing current research into a consolidated, knowledge-aware framework. This framework explicitly links the mechanics of GNNs with semantically grounded constraints from KGs and the temporal dependencies of streaming data. Accordingly, the specific contributions of this work are as follows:

- *Unified framework for explainable semantic stream reasoning.* The paper formulates a knowledge-aware conceptual framework that treats explainability, semantic enrichment, and stream reasoning constraints as coupled design dimensions. This framework serves as a structural lens to organize existing methods and to identify underexplored regions within the current literature, ensuring that transparency is built into the reasoning process rather than added as a post-hoc augmentation. The synthesis of this framework is finalized in Section 7.
- *Semantically grounded and stream-aware synthesis of explainability methods.* The study provides a structured taxonomy of explainability approaches, organizing established explanation families according to their compatibility with evolving graph structures and

patterns of semantic integration. In addition to preserving comparability with prior GNN explainability taxonomies, the review further characterizes these methods along stream-semantic axes, including semantic dependency, temporal sensitivity, and online incrementality. This two-layer organization clarifies how standard attribution mechanisms should be interpreted and adapted when reasoning over graphs that change over time, as presented in Section 3.
- *Systematization of semantic integration patterns.* The work identifies and categorizes recurring strategies for incorporating domain knowledge from KGs into dynamic GNNs. It analyzes how these integration patterns support the generation of explanations that remain coherent and interpretable even as the underlying data streams evolve, which is detailed in Section 4.
- *Knowledge-aware evaluation perspective.* The paper develops a knowledge aware evaluation perspective that extends beyond traditional metrics. These dimensions include semantic fidelity, relational coherence, temporal stability, and alignment with expert knowledge. This contribution provides conceptual guidance for designing future metrics that can quantify the semantic quality of explanations in streaming environments and is further discussed and partially illustrated in Section 5.
- *Domain-driven use case instantiation.* The proposed framework is illustrated through representative scenarios in high-stakes domains, including healthcare, security, social media, and autonomous systems. These use cases demonstrate how semantically grounded constraints and stream-specific requirements shape both model design and the interpretation of results in practice, as exemplified in Section 7.

Overall, the perspective proposed in this study emphasizes that semantic integration is not merely a modeling choice for prediction, but a fundamental mechanism for strengthening explainability. By anchoring explanations in explicit semantic relations and temporal constraints, this approach enables the detection of subtle, context-dependent patterns in continuous data streams that purely structural methods often fail to capture.

2. Background and foundations

This section establishes the core concepts underlying explainability and transparency in GNN-based stream inference. It first defines stream inference and its primary challenges, followed by the fundamental principles of GNNs. Subsequently, it examines explainability and transparency in machine learning, focusing on GNN-specific issues, and finally assesses the role of semantic context and Knowledge Graph (KG) integration.

2.1. Stream reasoning: Definitions, importance, and challenges

Stream reasoning is a research area dedicated to deriving meaningful and timely insights from continuous, variable data streams [28]. In real-world applications such as sensor networks, social media, and surveillance systems, the high velocity and continuous evolution of data render classical batch learning approaches insufficient [20]. Such dynamic environments require models capable of continuously updating internal representations to maintain robustness against rapidly shifting patterns. A critical requirement in these settings is the ability to analyze complex, large-volume data streams in real-time, capturing both temporal evolution and changing relational dependencies among entities [20].

Beyond predictive performance, high-stakes domains like healthcare, security, and finance demand models that are transparent and understandable by humans [29]. In these settings, interpretability is as crucial as accuracy, as domain experts must be able to justify or audit system outputs. Recent literature increasingly emphasizes the need for Explainable Artificial Intelligence (XAI) solutions that effectively process modern data streams [30]. Consequently, research is shifting toward approaches that balance real-time efficiency with human-centric interpretability. In this context, GNN-based approaches offer a promising solution for explainable stream inference by integrating relational reasoning with dynamic structural modeling [2]. At the same time, it is important to distinguish stream reasoning from the broader literature on temporal or dynamic graph learning. Stream reasoning places greater emphasis on requirements such as low-latency updating, limited computational budgets, window-based semantics, and robustness to continuously arriving or out-of-order data. By contrast, a substantial portion of the GNN literature reviewed in this paper focuses more broadly on evolving graph structure and temporal dependency modeling. Accordingly, the remainder of this survey also considers temporal and dynamic graph methods that are relevant to reasoning over evolving data streams, while noting that not all of them address the full requirements of online stream processing.

2.2. Core concepts and recent developments of GNNs

GNNs have emerged as a central paradigm for learning over relational data, leveraging message-passing mechanisms to propagate information across graph structures [1,2]. Through iterative neighborhood aggregation, GNNs capture local and higher-order dependencies, achieving strong performance in domains such as social networks, bioinformatics, and recommendation systems [2]. Learning effective representations often relies on self-supervised objectives, particularly when labeled data are scarce or evolving. Deep Graph Infomax [31] is a representative approach that learns robust node and graph representations without explicit labels.

Recent developments extend GNNs to dynamic and temporal environments, where topology and attributes evolve over time. Temporal Graph Networks (TGNs) and related architectures incorporate time-aware message passing and memory modules to represent evolving relational processes [20,28]. Surveys highlight the importance of dynamic representation learning for reasoning over high-frequency event streams [20]. Beyond explicit temporal architectures, context-aware graph embedding methods capture evolving interaction patterns in session-based and streaming settings [32]. To address scalability in large, rapidly changing graphs, efficient strategies such as sampling-based neighborhood approximation and edge pruning are employed to reduce computational overhead while preserving representational quality [2,33].

Despite these advances, explainability in GNNs remains an open challenge. The non-linear nature of message passing often renders them black boxes, limiting their adoption in safety-critical domains [29]. While instance-level methods such as GNNExplainer provide local explanations, and subsequent surveys organize related GNN explainability methods into broader taxonomies [34,35], much of this literature primarily addresses structural properties such as faithfulness and robustness. These methods and evaluation practices typically do not account for semantic coherence with external knowledge sources or consistency under temporal evolution [36,37]. This gap underscores the need for further research into trustworthy, human-aligned stream reasoning systems.

2.3. Explainability and transparency in machine learning

The complexity of modern machine learning systems has intensified the demand for interpretability, particularly in high-stakes domains. Explainability refers to a model's ability to provide human-understandable justifications for its predictions, clarifying the contribution of input features or relational patterns [30,38]. Transparency, conversely, concerns the accessibility of a model's internal mechanisms, including its architecture and parameters, to users [30,39]. While classical models like decision trees are inherently interpretable, deep learning architectures, including GNNs, often operate as opaque systems, posing barriers to trust and accountability [40,41].

The field of XAI aims to develop frameworks that render these models more reliable and accountable [42]. Explainability approaches are commonly categorized as *intrinsic*, where interpretability is built into the model design, or post-hoc, where explanations are generated after prediction for otherwise opaque models [30,38]. Applied XAI research increasingly integrates explainability into ensemble-based and feature-informed predictive frameworks [43]. Hybrid modeling has also been used to support interpretable decision-making in medical diagnosis, including liver disease identification [44]. Work on skin cancer detection further suggests that multi-scale feature fusion can be combined with interpretable design choices in hybrid architectures [45]. Recent work has emphasized rigorous evaluation, with frameworks like OpenXAI providing metrics for fidelity, stability, and human alignment [29,46,47]. The importance of trustworthy explanation is equally evident in high-stakes diagnostic settings, where privacy-aware approaches have been proposed for diabetes risk diagnosis [48]. Evidence from maternal health shows that explainable machine learning can reveal clinically relevant risk factors [49]. Explainable machine learning has also been applied in obstructive pulmonary disease care to support severity classification, quality-of-life prediction, and treatment impact assessment [50]. Achieving explainability in graph-based learning is uniquely challenging, requiring the elucidation of how connectivity patterns and multi-hop dependencies influence outputs [34].

An emerging direction involves enriching GNNs with semantic context from KGs. Semantic structures, such as type hierarchies and relation constraints, provide an interpretable scaffolding that enhances both reasoning capability and explanation clarity. By grounding decisions in semantically meaningful patterns, KG-driven enrichment supports robust reasoning over complex, dynamic graph streams [36,51,52].

2.4. KGs for semantic stream reasoning

Integrating semantic context is essential for modern stream reasoning systems. Semantic context, typically represented through ontologies or KGs, introduces structured domain knowledge that extends beyond raw data streams [37]. KGs capture static and evolving relational information, enriching feature representations and guiding reasoning processes [12,37,53]. KG enriched GNNs demonstrate stronger relational generalization and more interpretable decision pathways [25].

In GNN-based stream reasoning, semantic context serves two critical functions. First, it enables generalization to unseen scenarios by leveraging domain-specific relational patterns and hierarchical structures [37]. Second, KGs improve interpretability by making the internal decision process transparent and allow users to trace reasoning steps through semantically meaningful connections. This is vital in domains like medical diagnosis and financial forecasting, where output justification is mandatory [11].

Incorporating KG-derived information into GNNs improves predictive performance as well as the stability and clarity of explanations, especially in temporal environments [52]. Temporal KGs, which model evolving entities and relations, provide a structured context that enables robust reasoning over rapidly changing data [18]. KGs enhance dynamic inference and explainability through temporal grounding.

2.5. Related surveys and positioning

Early surveys established the foundations of deep learning on graphs but devoted limited attention to explainability or semantic integration in streaming settings [1,2,54]. Domain-focused surveys in bioinformatics or KG tasks emphasize predictive performance rather than transparency [3–5]. While standard KG literature prioritizes representation over explainability [36,37,53], recent studies explicitly examine how structured knowledge enhances ML interpretability [55,56]. Similarly, surveys on knowledge-enhanced GNNs discuss how external knowledge improves comprehensibility [25,57]. Main XAI surveys catalog explanation methods and evaluation criteria but often treat graphs as just another modality, by overlooking the specific challenges of relational structure and temporal evolution [29,38,39,42]. While dedicated surveys on GNN explainability have introduced taxonomies and evaluation perspectives [22,35,58,59], they generally focus on static or mini-batch settings. They do not explicitly cover semantic stream reasoning, which necessitates aligning explanations with external knowledge in continuously evolving graphs. Table 1 clarifies the distinct positioning of this survey. In particular, this review considers KG structure not only as a source of semantic enrichment for prediction, but also as a mechanism for generating, constraining, and validating explanations. This positioning also informs how the taxonomy in Section 3 should be understood. Accordingly, the role of the taxonomy in Section 3 is not to introduce wholly independent explanation categories, but to synthesize and reinterpret established GNN explanation families within the semantic and streaming perspective developed in this review.

In contrast to the above literature, this paper focuses on explainable and transparent *semantic stream reasoning* with GNNs. It provides a structured synthesis of GNN explanation techniques in semantically enriched streaming settings and a taxonomy of KG integration mechanisms. Furthermore, it introduces a knowledge-aware evaluation perspective emphasizing semantic fidelity, relational coherence, and temporal stability.

3. Explainability techniques in GNN-based stream reasoning

This section provides a systematic taxonomy of the major explainability approaches used in GNN-based models and analyzes their applicability in semantically enriched streaming contexts. The taxonomy is not intended to replace prior GNN explainability taxonomies or to define entirely new explanation families. Rather, it reorganizes established explanation approaches through the specific lens of semantic stream reasoning, with emphasis on their relevance, limitations, and interpretive role in settings involving semantic enrichment and temporal evolution. Because the literature at this intersection remains fragmented, the methods reviewed in this section should be understood as explanation families that are relevant to semantically enriched streaming contexts to different degrees, rather than as methods uniformly designed for online stream processing. In addition to these explanation families, the taxonomy can also be read through the distinction between intrinsic and post-hoc explainability introduced in Section 2.3. Intrinsic approaches provide explanatory signals as part of the model architecture or reasoning process, whereas post-hoc approaches reconstruct explanations after prediction. This distinction is particularly relevant in evolving graph streams, since intrinsic signals may be easier to update under latency constraints, while post-hoc methods often provide richer explanatory detail at higher computational cost. A structured comparison of these methods is provided in Table 2.

The method families summarized in Table 2 provide a reference layer that preserves comparability with prior GNN explainability surveys. However, the unified perspective developed in this review requires a second organizing layer that reflects the specific requirements of semantic stream reasoning. We therefore characterize explanation methods along three stream-semantic axes: *semantic dependency*, which indicates the extent to which an explanation relies on KG structure, ontological constraints, or domain rules; *temporal sensitivity*, which captures whether the explanation depends on event order, persistence, temporal windows, or evolving motifs; and *online incrementality*, which reflects whether explanations can be updated under latency and resource constraints without complete recomputation. The categories in Table 3 are not intended as mutually exclusive method classes. Rather, they provide a qualitative stream-semantic reorganization of established explanation families according to these axes.

Table 1
Positioning regarding related surveys. Legend: + = primary focus/comprehensive taxonomy, ± = discussed as sub-topic/application, − = not systematically addressed.

Comparative prior works	GNN-XAI	KG/semantics	Stream/temporal	Eval. of explanations	Unified view
Foundational GNN Surveys (architectures, learning paradigms) [1,2,54]	−	±	±	−	−
Domain-focused GNN surveys (e.g., bioinformatics, RecSys) [3–5]	−	±	±	−	−
KG fundamentals surveys (representation, reasoning) [36,37,53]	−	+	±	−	−
KGs for explainable ML (KG-grounded explanations) [55,56]	±	+	−	±	±
Knowledge-enhanced AI (external knowledge for interpretability) [25,57]	±	+	±	−	±
GNN explainability surveys (XAI taxonomies + metrics) [7,22,35,59]	+	±	−	+	−
Our work (Explainable GNN-based Stream Reasoning by Semantic Integration)	+	+	+	+	+

Table 2
Systematic comparison of explainability methods in GNN-based semantic stream reasoning.

Method	Key principle	Main strengths	Main limitations	Recommended use
Graph Attention [60–66]	Learn attention weights to highlight influential nodes and edges	Naturally interpretable attention maps; effective for local explanations in evolving graphs	Attention bias; intrinsic signals may be unstable over time; limited ability to capture global or multi-hop dependencies	Real-time anomaly detection, relational monitoring
Semantic Embedding [16,25,26,52,53,67–70]	Integrate external semantic context (KGs, ontologies) into node/edge representations	Context-aware reasoning; improved generalization across heterogeneous semantic environments	Embeddings are dense and hard to interpret directly for end users	Healthcare, recommender systems, multi-KG integration
Subgraph-based Explanation [22,34,71–75]	Identify compact or prediction-relevant subgraphs for a given prediction	Instance-specific and visually interpretable explanations; useful for local justification	High computational cost; limited scalability for large or streaming graphs; may omit relevant external structural factors	Biomedical justification, legal reasoning, critical support
Perturbation-based Explanation [22,29,47,76–79]	Modify node/edge features or topology to evaluate influence on predictions	Model agnostic; provides quantitative causal insights (e.g., Shapley values)	Slow and computationally expensive; not suitable for real-time stream settings	Regulatory auditing, offline validation, post-hoc inspection
Counterfactual Explanation [15,75,80–82]	Generate minimal perturbations to input graph that flip the prediction	Provides actionable recourse ("what-if" analysis); distinguishes causation from correlation	Optimization is difficult on discrete graph structures; sparsity issues	Actionable decision making, root cause analysis
Temporal & Evolution Explanation [13,14,83,84]	Identify critical historical events or time-steps driving the current prediction	Captures dynamic evolution patterns; explicitly addresses the time dimension	High complexity in tracking temporal dependencies; difficulty maintaining temporal fidelity, stability, and low-latency explanations	Event forecasting, dynamic trend analysis, evolving anomaly detection

3.1. *Established explanation method families*

3.1.1. *Graph attention mechanisms*

Graph Attention Networks (GAT) and related architectures employ learnable weights to assign importance scores to specific nodes and edges during the message-passing phase [60,63]. These coefficients function as inherent interpretability signals, enabling observers to verify which neighborhoods or relations contributed most significantly to a prediction [61,62,65,66]. However, attention weights should be interpreted as intrinsic explanatory signals rather than as complete explanations. In evolving graph streams, their reliability may be affected by changing neighborhoods, temporal shifts in relation importance, and instability in attention distributions across consecutive updates. Therefore, attention-based explanations are attractive for low-latency monitoring, but they may require additional semantic or post-hoc validation before being treated as sufficient evidence of model reasoning [7,58]. In streaming contexts, this means that attention scores may need to be interpreted together with window-aware normalization, periodic recalibration, or semantic consistency checks, since the set of relevant neighbors and relations may change as new events arrive. For instance, in a citation network, the model might assign higher attention coefficients to references from the same research topic, effectively filtering out irrelevant citations [60]. While this mechanism offers an efficient route to local interpretability for tasks like node classification, the resulting attention maps can sometimes be biased. Furthermore, standard attention mechanisms often struggle to capture long-range or global dependencies in complex graphs, limiting their effectiveness for deep semantic reasoning [63,64].

3.1.2. *Semantic embedding methods*

Semantic embedding strategies focus on integrating external domain knowledge, such as KGs or ontologies, directly into the vector representations of nodes and edges [53,67,68,70]. For instance, models such as TransE represent relationships as geometric translations in a latent space, while architectures like KGAT propagate these embeddings via attention mechanisms to capture high-order semantic connectivity [53,70]. By encoding hierarchical structures and relational constraints, these models improve generalization in evolving environments [25,26,52]. However, the main limitation is that high-dimensional embeddings are difficult for humans to interpret directly. Consequently, tracing the influence of specific semantic concepts on the model output is difficult without auxiliary decoding or visualization tools [16,69]. In

Table 3
Stream-semantic taxonomy of explanation methods for knowledge-enriched graph streams.

Stream-semantic category	Typical explanation families	Core semantic and temporal axes	Stream-reasoning role and limitation
Intrinsic online signals	Attention-based methods and self-explainable temporal models [14,60,63,83]	Low to medium semantic dependency; medium temporal sensitivity; high online incrementality	Suitable for low-latency monitoring because explanatory signals are produced during inference, but reliability can be limited without semantic validation
Semantic constraint-grounded explanations	Semantic embeddings, KG-assisted paths, rule-guided explanations, and semantic validation [10,53,55,56,66,70,85]	High semantic dependency; medium to high temporal sensitivity when the KG evolves; medium online incrementality	Useful for domain-grounded interpretation and semantic consistency, but requires maintaining KG constraints and validating explanations as background knowledge changes
Incremental temporal attribution	Temporal explainers, evolution-based explanations, evolving motifs, and temporally bounded subgraphs [13,14,81,83,84]	Variable semantic dependency; high temporal sensitivity; medium to high online incrementality	Captures event order, persistence, drift, and evolving patterns, but requires efficient update mechanisms to avoid recomputing explanations after every event
Deferred structural reconstruction	Subgraph-based, perturbation-based, and feature attribution methods [22,34,47,71,72,77,78]	Low to medium semantic dependency unless KG constraints are added; low to medium temporal sensitivity; low online incrementality	Provides detailed post-hoc evidence for auditing or high-risk cases, but is often costly for continuous explanation at every stream update
Selective intervention and recourse	Counterfactual explanation methods [15,80,82]	Medium to high semantic dependency when interventions must satisfy domain rules; medium temporal sensitivity; selective online incrementality	Supports actionable what-if reasoning, but is better suited to selective or high-risk cases because the feasible intervention space changes as the stream evolves

streaming settings, semantic embedding methods face additional challenges because both the input graph and the associated KG may evolve over time. Embeddings learned on earlier graph states can become stale when new entities, relations, or ontology constraints arrive, while frequent retraining may be too costly for latency-sensitive applications. Practical use in stream reasoning may therefore require incremental embedding updates, temporal regularization, or selective refreshing of only the affected semantic neighborhoods. These adaptations can improve scalability, but they also make it harder to preserve semantic fidelity and explanation stability over time.

3.1.3. Subgraph-based explanation methods

Subgraph-based approaches, exemplified by instance-level methods such as GNNExplainer, PGExplainer, and SubgraphX, identify compact subgraph structures and, in some cases, feature masks that are most relevant to a specific prediction [34,71,72]. Model-level approaches such as XGNN differ from these instance-level subgraph explainers: rather than extracting a minimal explanatory subgraph for a specific input instance, they generate graph patterns that characterize what a trained GNN has learned at the class or model level [24]. Instance-level subgraph explainers are highly valued for their fidelity and visual clarity, as they present prediction-relevant substructures to the user in a human-interpretable format [41,74]. For example, in molecular property prediction tasks such as mutagenicity testing, such an explainer may identify specific chemical substructures, such as carbon rings, as important evidence for a molecule's classification [34]. Nevertheless, the optimization process required to identify these subgraphs is computationally expensive. This cost creates significant scalability bottlenecks when applied to large-scale graphs or environments where data arrive in high-frequency streams [22,73]. Moreover, recent findings suggest that restricting explanations to local subgraphs may inadvertently exclude relevant external structural factors [75]. In streaming or dynamically evolving graphs, adapting subgraph-based explanation requires more than rerunning a static explainer after each update. A more feasible direction is incremental or temporally bounded subgraph extraction, where previously identified explanatory neighborhoods are updated only around changed nodes, edges, or time windows. Such adaptations can reduce recomputation costs, but they also introduce new challenges: the extracted subgraph must remain temporally valid, semantically coherent, and sufficiently faithful as the underlying graph evolves.

3.1.4. Perturbation-based explanation methods

Perturbation-based techniques evaluate model behavior by systematically modifying input features graph topology, or spatio-temporal constraints, and observing the variations in output confidence [47,78,79]. Being model-agnostic, they provide robust quantitative insights into feature importance, often utilizing game-theoretic concepts like Shapley values [76,78]. For instance, GraphSVX adapts Shapley value estimation to graph data, systematically masking node features to determine their marginal contribution to the classification score [78]. However, the iterative nature of perturbation requires multiple inference passes, making these methods computationally prohibitive for real-time applications. Therefore, their utility is generally confined to post-hoc analysis rather than active stream processing [22,29,77]. For evolving graph streams, perturbation-based explanation may therefore need to rely on selective or approximate perturbation strategies rather than exhaustive recomputation. Examples include perturbing only recently changed graph regions, sampling a limited set of candidate nodes or edges, or applying perturbation analysis periodically or only for high-risk predictions. These adaptations make the method more compatible with streaming constraints, although they may reduce the completeness of the causal assessment compared with full offline perturbation analysis.

3.1.5. Counterfactual explanation methods

Unlike standard attribution techniques that highlight existing important features, counterfactual approaches focus on the concept of recourse. They aim to identify the minimal changes to the input graph necessary to flip the model's prediction [80,82]. These methods are particularly valuable for decision support, as they provide actionable "what-if" scenarios that distinguish true causality from mere correlation [80]. For instance, in a hypothetical loan approval graph, such an explanation might demonstrate that establishing a 'guarantor' link with an existing high-credit node would flip a rejection decision to an acceptance. Recent advancements have extended this reasoning to dynamic environments, attempting to generate counterfactuals over evolving graph structures [15,81]. However, generating validity-preserving counterfactuals remains challenging due to the discrete nature of graph data and the complexity of optimizing for structural externalities without violating domain rules [75]. In streaming settings, counterfactual explanation becomes even more challenging because the feasible intervention space changes as new nodes, edges, and semantic relations arrive. Practical adaptation may therefore require approximate or temporally bounded counterfactuals that search only within recent graph updates, local neighborhoods, or semantically admissible intervention sets. Such counterfactuals can provide actionable explanations under time constraints, but their validity must be checked against both temporal consistency and domain-specific semantic rules. For this reason, counterfactual explanations in streaming environments are often more suitable for selective, on-demand, or high-risk cases than for continuous generation after every incoming graph update.

3.1.6. Temporal and evolution-based explanation methods

Temporal and evolution-based explanation methods address the limitations of static explainers by explicitly modeling how graph structure, node states, and relational patterns change over time. In contrast to static explanation paradigms, these methods attribute predictions not only to a fixed neighborhood or feature set, but also to prior events or interactions, such as specific time steps, temporal motifs, or evolving relational patterns that shape the current model output [13,84]. For example, in financial fraud detection, a temporal explainer may identify a burst of high-frequency transactions within a short time window as the relational pattern responsible for triggering an alert.

A useful distinction can be drawn between discrete-time snapshot representations and continuous-time event-based temporal graphs. In snapshot-based settings, the evolving graph is represented as a sequence of graph instances, and explanations may highlight important snapshots, transitions between snapshots, or persistent substructures across time. In event-based settings, timestamped interactions or updates are modeled directly, and explanations may identify influential events, messages, memory updates, evolving paths, or temporally recurring motifs. This distinction is important because the unit of explanation differs across these settings.

Temporal explanation also introduces challenges that are less visible in static GNN explanations. First, *temporal fidelity* requires that an explanation reflect not only the structural components used by the model, but also their correct temporal order and timing. Second, explanation *stability* must be interpreted carefully: explanations should not fluctuate arbitrarily across adjacent time steps, yet they should still be sensitive enough to capture genuine concept drift or meaningful changes in the data stream. Third, *incremental attribution* becomes important when explanations must be updated as new events arrive, since recomputing a complete explanation after every graph update may be computationally prohibitive. Finally, *latency-aware explanation* is critical in streaming scenarios, where the explanation process itself must not undermine the timeliness of the reasoning pipeline.

Despite their importance for stream reasoning, temporal explanation methods remain less mature than static GNN explainers. Tracking long-range temporal dependencies, distinguishing meaningful evolution from noise, and maintaining semantically coherent explanations over time all introduce additional computational and methodological complexity. Consequently, temporal and evolution-based explanation should be viewed not only as a separate explanation family, but also as a cross-cutting requirement for adapting other explanation methods to evolving, knowledge-enriched graph streams.

3.2. Comparative analysis and applicability

The choice of an explainability method in semantic stream reasoning requires balancing specific trade-offs based on the application requirements. In scenarios where low latency is the primary constraint, such as real-time anomaly detection, *graph attention mechanisms* are the standard choice. Their ability to provide immediate importance scores allows for continuous monitoring without the overhead of complex optimization [60]. On the contrary, domains relying on rich background knowledge, like healthcare, prioritize handling heterogeneous data over visual simplicity. Here, *semantic embedding methods* are preferred as they generalize across diverse ontology schemas, utilizing external knowledge to enhance reasoning [53,70]. For high-stakes applications like legal or clinical decision support, explicit justification is mandatory. In these cases, *subgraph-based explanations* are indispensable because they provide high-fidelity, visual evidence of the reasoning process, even if they are computationally expensive [34,41].

When the goal is rigorous offline validation or regulatory auditing, *perturbation based techniques* offer the deepest quantitative insights. By systematically testing feature influence, they provide robust causal metrics (e.g., Shapley values) that are ideal for post-hoc inspection [47,78]. Distinct from mere auditing, user-centric applications often require actionable advice (e.g., “what to change to get a loan”). For such tasks, *counterfactual explanations* are superior, as they generate prescriptive “what-if” scenarios that distinguish causation from correlation, offering a path to recourse [80]. Finally, for tasks explicitly driven by dynamic trends, such as event forecasting in streams, static methods are insufficient. *Temporal and evolution explanation methods* are uniquely required here to capture the historical trajectory and evolving motifs that drive the current prediction, addressing the critical time dimension missed by other approaches [13,84]. Their importance stems from the fact that temporal explanation is not limited to identifying influential structures. Also it requires preserving the order, persistence, and timing of the events that shape the prediction. While these associations highlight the most natural fit of each explanation family, they should not be understood as rigid boundaries. More generally, these applicability patterns lie along a spectrum. At one end are methods that are more readily compatible with latency-sensitive streaming settings, while at the other are methods that remain primarily offline or batch-oriented and therefore require substantial adaptation before they can be applied in evolving stream contexts. A further distinction concerns whether explanations are intrinsic or post-hoc. Intrinsic methods, such as attention-based mechanisms, are generally more feasible in latency-sensitive streaming settings because they produce explanatory signals during the forward reasoning process. However, these signals may not always be sufficient to establish reliability or trustworthiness, particularly when attention distributions change over time or when semantic constraints are not explicitly checked. By contrast, post-hoc methods such as subgraph extraction, perturbation analysis, and counterfactual explanation can provide more detailed explanatory reconstruction, but they often require additional inference, optimization, or graph manipulation. This makes them more suitable for offline validation, auditing, or selective high-risk cases than for continuous explanation at every stream update.

3.2.1. Intrinsic versus post-hoc explanations under streaming constraints

A central practical distinction in stream reasoning concerns whether explanations are produced intrinsically during the forward reasoning process or reconstructed after prediction. Intrinsic approaches, such as attention-based mechanisms or self-explainable temporal architectures, are generally more compatible with latency-sensitive settings because they generate explanatory signals together with the prediction and introduce limited additional inference overhead. This makes them suitable for continuous monitoring, early warning, and rapid inspection of incoming graph events. However, their reliability and trustworthiness may be limited when attention distributions shift over time, when neighborhoods change rapidly, when the active temporal window changes, or when semantic constraints are not explicitly checked. In such cases, intrinsic signals should be treated as fast indicators rather than complete evidence of model reasoning.

By contrast, post-hoc approaches such as subgraph extraction, perturbation analysis, and counterfactual explanation can provide richer explanatory reconstruction and stronger support for auditing, debugging, expert review, or regulatory inspection. Their main limitation is computational: they often require additional inference passes, optimization procedures, graph manipulation, or search over possible interventions. These costs become more problematic in stream reasoning because explanations may need to be updated repeatedly as new nodes, edges, relations, or delayed events arrive. For this reason, post-hoc explanations are usually more appropriate for offline validation, delayed explanation, or selectively triggered high-risk cases than for continuous generation after every stream update.

A practical stream reasoning pipeline can therefore combine both types of explanation. Lightweight intrinsic signals can support online monitoring under latency and resource constraints, while more detailed post-hoc explanations can be triggered for selected events, windows, alerts, low-confidence predictions, or semantically inconsistent outputs. This hybrid interpretation also clarifies that explainability in graph streams is not only a question of explanation quality, but also of when, how often, and under which computational budget explanations can be updated.

4. Enhancing GNN-based stream reasoning via semantic integration

4.1. Impact of semantic context on explainability and reasoning performance

Incorporating semantic context into GNN-based stream reasoning models is essential for improving both interpretability and predictive robustness, particularly in dynamic and continuously evolving graph environments. As discussed in Section 2.4, semantic structures capture entity semantics, relational dependencies, and high-level schema constraints beyond raw streaming data [36,37,41]. These semantic structures help models move beyond purely data-driven correlations by leveraging type hierarchies, relational rules, and domain-specific interaction patterns. This enriched knowledge enables GNNs to generalize observed behaviors to previously unseen scenarios, improving reasoning accuracy in tasks involving relational abstraction or hierarchical inference [86]. Importantly, integrating semantic context contributes not only to stronger model performance but also to greater transparency by grounding predictions in explicit knowledge structures that are understandable to humans [52].

In high-stakes domains such as healthcare and cybersecurity, this semantic grounding becomes critical for enabling clinicians and operators to trace and justify model decisions [68,87]. Despite this growing need, systematic analyses of how semantic context directly enhances explainability in GNN-based stream reasoning remain limited in the current literature. Recent work demonstrates that embedding KG-derived semantics into GNN architectures can significantly improve reasoning coherence, stability, and the interpretability of explanations in temporal, heterogeneous, and rapidly changing environments [11,51,52,67]. These findings highlight semantic enrichment as a promising direction for developing more transparent, reliable, and domain-aligned GNN based stream reasoning systems suitable for real world continuously evolving graph streams. Thus, this section establishes the conceptual rationale for semantic enrichment in GNN-based stream reasoning, while the following subsections examine concrete integration mechanisms, architectural implications, and an illustrative explanation-level comparison.

4.2. Knowledge graph integration for reasoning and explanation in GNNs

Integrating structured knowledge from KGs into GNN-based stream reasoning models provides a principled mechanism for encoding semantic relations, symbolic constraints, and temporal structure directly into neural computations. Across the reviewed literature, KG integration can be understood as serving two complementary functions. First, it supports representation learning and reasoning by enriching node and relation representations, guiding message passing, and improving relational inference. Second, it supports explanation by providing semantic structures through which model outputs can be traced, organized, and validated. Accordingly, this section discusses KG embedding-based integration, knowledge-aware message passing, temporal and neuro-symbolic integration, and KG-assisted explanation strategies. These strategies exhibit different strengths and limitations in terms of scalability, representational fidelity, and explainability [2,36].

KG embedding integration embeds entities and relations into continuous vector spaces using models such as TransE or ComplEx [53,88]. The resulting embeddings are then used as node features or auxiliary signals for downstream GNN architectures. This approach is computationally efficient and compatible with standard GNN pipelines. However, collapsing multi-relational KG structure into dense vectors may reduce semantic transparency and limit the preservation of symbolic constraints and higher order dependencies [36,53].

Knowledge-aware message passing incorporates KG semantics directly into the propagation mechanism of GNNs. For example, Relational GCNs (R-GCNs) introduce specific relation transformations to handle different edge types and thus maintain the multi-relational characteristics of the KG [86]. More recent architectures, such as epistemic GNNs, extend this idea by incorporating rule guided reasoning into the learning process. They align neural inference with domain-specific logical structures, thereby enhancing interpretability [89]. These approaches preserve richer semantic information compared to only embedding methods. Despite this, they require significantly more computation, which limits scalability for large or dense KGs [36].

Temporal or evolving KGs introduce additional challenges. TGNs employ time-aware message passing to update node states as relational structure changes over time [19]. In recommender systems, hierarchical GNNs coupled with multimodal KGs have shown effectiveness in modeling user preference evolution for more semantically grounded predictions [64,90]. In spite of their advantages, these approaches often face latency constraints and rely on stable temporal semantics that may not consistently hold in high-volume data streams.

Neuro-symbolic approaches represent a complementary direction, jointly training neural models with symbolic reasoning modules [10]. These methods aim to combine the expressiveness of logical rules with the flexibility of statistical learning [26,89]. Nevertheless, this hybrid integration introduces significant computational overhead, increasing training complexity and posing scalability challenges for large KGs with diverse relational patterns [25,68].

KG-assisted explanation strategies represent a distinct use of semantic knowledge for explanation [51,55,56]. Unlike KG embedding or message-passing approaches, which primarily inject knowledge into representation learning and prediction, KG-assisted explainers use the structure of the KG to make model outputs more interpretable, traceable, and semantically valid. Three directions are particularly relevant in this regard: path-based KG explanation, rule-guided explanation, and semantic validation of explanations.

Path-based KG explanation reconstructs interpretable relational paths between entities involved in a prediction [85]. In this setting, an explanation is not limited to a set of influential nodes or edges, but is expressed as a semantically meaningful chain of relations grounded in the KG. Such paths can help users understand why two entities are connected, why a recommendation is produced, or why a clinical or security-related prediction is supported by specific relational evidence. However, path-based explanations may become difficult to maintain in evolving graphs, because newly arriving entities and relations can change the available explanatory paths over time.

Rule-guided explanation uses domain rules, ontology constraints, or logical patterns to structure and filter explanations [10,89]. Rather than treating every extracted subgraph as equally meaningful, rule-guided approaches evaluate whether the explanation conforms to domain-specific reasoning patterns. This direction is particularly relevant in high-stakes settings where explanations should not only be faithful to the model, but also compatible with expert knowledge and domain protocols.

Semantic validation of explanations further extends this idea by checking whether the explanation respects KG-level constraints, such as entity-type compatibility, admissible relation types, schema dependencies, and temporal validity [9,53,66,86]. In semantic stream reasoning, this validation is especially important because explanations may become invalid when the underlying KG evolves. Therefore, KG-assisted explanation should be understood not only as a post-hoc visualization mechanism, but also as a way to assess whether explanations remain relationally meaningful and semantically consistent over time.

Overall, KG-GNN integration remains an active and rapidly evolving research area. The field is motivated by the need to combine semantic richness, computational efficiency, and interpretability within a single framework. The discussion above shows that KGs contribute to explainability in two complementary ways: internally, by shaping representation learning and message passing, and externally, by supporting path-based, rule-guided, and semantically validated explanations. Designing mechanisms that remain scalable, temporally stable, and explanation-aware in real-time stream reasoning remains an unresolved challenge.

4.3. Concrete architectural roadmap for semantically constrained temporal message passing

The integration patterns discussed above can be translated into a more explicit architectural workflow showing how semantic constraints may be incorporated into temporal GNN pipelines. As discussed in Section 3.1.6, temporal GNNs include both discrete-time snapshot representations and continuous-time or event-based formulations. The workflow in this section adopts a TGN-style event-based formulation for concreteness, because stream reasoning commonly involves timestamped events that incrementally update graph structure, node memories, and predictions. This choice is illustrative rather than definitional: Algorithm 1 is intended as one representative implementation-oriented design pattern for semantic constraint enforcement, not as a general definition of TGNNs or as an exhaustive architecture covering all temporal GNN families. In snapshot-based temporal GNNs, analogous semantic validation could be applied during snapshot construction, across transitions between consecutive snapshots, or during temporal aggregation.

In this event-based setting, semantic constraints are not treated as an external layer applied only after prediction, but as elements that can shape event representation, message propagation, memory updating, and explanation generation throughout the reasoning process. Let an incoming event at time t be represented as

$$e_t = (u, r, v, x_t, t),$$

where u and v denote source and target entities, r is the observed relation type, and x_t represents event-specific features. In a standard TGN-style pipeline, such an event contributes to message construction, temporal memory updates, and the recomputation of dynamic node embeddings as the graph evolves. In a semantically constrained variant, this pipeline is augmented with a semantic control layer that evaluates whether incoming events and propagated messages remain compatible with ontology-level or KG-level constraints. This step-by-step workflow is summarized in Algorithm 1, which shows how semantic compatibility checking and constraint enforcement can be integrated into a TGN-style temporal message passing pipeline as an event-based illustration of the broader framework.

First, raw stream events are aligned with a dynamic graph representation enriched by semantic types, relation schemas, and temporal validity conditions. This alignment ensures that observed interactions are encoded not only as structural events, but also as semantically typed relations grounded in domain knowledge. Next, auxiliary semantic context associated with the event is retrieved, including entity classes, relation constraints, hierarchical dependencies, admissible interaction patterns, and temporal rules. Candidate messages can then be constructed from both temporal signals and semantic context rather than from structural information alone.

Algorithm 1 Semantically Constrained Temporal Message Passing

Require: Event stream $S = \{e_t\}$, dynamic graph G_t, ontology / knowledge graph K, temporal GNN backbone (e.g., TGN), semantic constraint set C
Ensure: Updated node states H_t, prediction y_t, semantically grounded explanation E_t

Initialize node memories M_0 and node embeddings H_0
for each incoming event $e_t = (u, r, v, x_t, t)$ in S **do**
  $z_t \leftarrow \text{EventEncoder}(u, r, v, x_t, t)$
  $k_t \leftarrow \text{SemanticContextRetriever}(u, r, v, K)$
  $c_t \leftarrow \text{SemanticValidator}(z_t, k_t, C)$ ▷ check type, relation, and temporal consistency
  $G_t \leftarrow \text{UpdateGraphStructure}(G_{t-1}, e_t, k_t)$ ▷ incorporate the new event into the evolving graph
  $m_t \leftarrow \text{MessageFunction}(M_{t-1}[u], M_{t-1}[v], z_t)$
  $\alpha_t \leftarrow \text{SemanticCompatibilityScore}(c_t)$ ▷ higher score indicates stronger semantic compatibility
  $m'_t \leftarrow \text{ApplySemanticGate}(m_t, \alpha_t)$ ▷ block invalid messages or down-weight weakly compatible ones
  $M_t[u], M_t[v] \leftarrow \text{UpdateNodeMemory}(M_{t-1}, m'_t, t)$ ▷ update internal temporal states after semantic filtering
  $H_t \leftarrow \text{TemporalAggregator}(M_t, \text{Neighborhood}(G_t, t))$
  $y_t \leftarrow \text{Predictor}(H_t)$
  $E_t \leftarrow \text{Explainer}(H_t, G_t, K, C)$ ▷ paths, subgraphs, or motifs satisfying semantic constraints
end for

Semantic constraint enforcement is then introduced directly into message passing. Here, semantic filtering refers to checking whether an incoming event or propagated message remains consistent with ontology-level or KG-level constraints, including entity-type compatibility, semantic validity of the relation, and temporal consistency. In this sense, semantic filtering does not necessarily imply binary rejection. Rather, the architecture may either block semantically invalid interactions or reweight them according to their degree of compatibility with background knowledge. This provides a mechanism for preserving the expressiveness of temporal message passing while reducing the influence of interactions that are structurally plausible but semantically inconsistent.

Graph and memory updates then proceed as distinct but complementary operations. Graph updating incorporates the new event into the evolving relational structure, whereas memory updating revises the node-specific temporal states used for downstream inference. In addition to recency and event content, memory updates may therefore be conditioned on semantic compatibility scores reflecting the degree to which the current event conforms to the background knowledge encoded in the ontology or KG.

Finally, updated node representations are obtained by aggregating semantically admissible or semantically weighted messages over the evolving graph. Explanations may then be generated over paths, subgraphs, or motifs that satisfy the same semantic constraints used during reasoning. In this way, explanation generation remains aligned with the internal model state while also preserving compatibility with domain knowledge. The resulting workflow offers a concrete architectural interpretation of the unified perspective proposed in this paper and clarifies how semantic grounding can be introduced directly into temporal inference rather than only into post-hoc analysis.

Notably, this roadmap is intended as an implementation-oriented design pattern for semantically grounded stream reasoning rather than as a claim of a newly benchmarked architecture. Its role within this review is to make explicit how the integration patterns identified in the

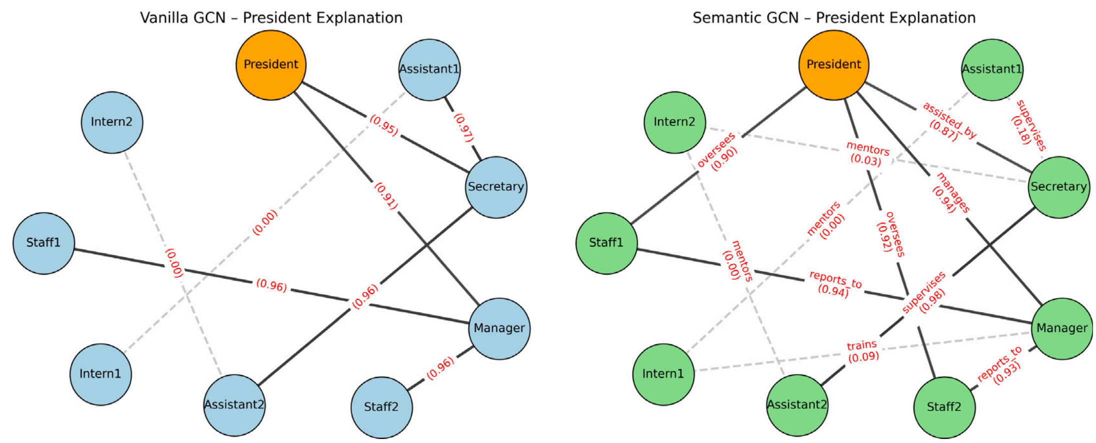

Fig. 3. Visualization of explainability gains achieved by integrating semantic context into the GNN-based reasoning process. The figure compares a Vanilla GCN (left) [92] and a semantically enriched GCN (right) [86,93] for the prediction of the 'President' node. The visualization is based on a synthetic graph designed to simulate organizational hierarchies. While the vanilla model predominantly emphasizes topologically proximate neighbors, the semantically enriched GCN highlights semantically informative links such as the "*President → Manager*" relation, which reflects a *supervisory* role, and the "*President → Secretary*" connection, which indicates *administrative support*. These additional connections are guided by external semantic knowledge injected into the model. As a result, the explanation becomes broader in scope and the decision process becomes more transparent and easier to interpret. This demonstrates the effectiveness of semantic enrichment in enhancing both the fidelity and the interpretability of GNN-based reasoning.

literature can be translated into concrete design choices for dynamic GNN systems. In this workflow, semantic constraints may be operationalized either as hard constraints, which block invalid interactions entirely, or as soft constraints, which reweight messages according to their compatibility with background knowledge. This distinction is particularly relevant in stream reasoning environments, where strict symbolic filtering may improve semantic validity but reduce robustness under noisy or incomplete observations, whereas soft semantic gating can provide a more scalable compromise for real-time inference. Thus, the event-based formulation should be understood as a concrete illustration of the framework in one common temporal GNN setting, rather than as a restriction of the framework to TGN-like architectures.

4.4. Illustrative effect of semantic enrichment on explanation structure

Whereas Section 4.1 discusses the general conceptual role of semantic context in improving reasoning, interpretability, and predictive robustness, this subsection focuses on an illustrative comparison of how semantic enrichment changes the structure of GNN explanations. Semantic enrichment not only enhances learning over graph-structured data but also shapes how model decisions are interpreted and evaluated [91]. This section examines how integrating semantic information influences the explainability and transparency of GNN-based reasoning. In particular, we analyze how semantic relations alter edge importance assignments and refine the structure of the explanatory subgraphs produced by the model. A comparative example is provided to illustrate how semantic information systematically modifies the model's internal explanation patterns.

Fig. 3 illustrates the effect of semantic enrichment on the explainability of GNN-based reasoning by comparing two trained models: a Vanilla GCN [92], which operates solely on structural edges, and a Semantic GCN [93], which incorporates additional ontological relations. In the visualized example, explanations generated by GNNExplainer reveal a distinct qualitative difference. In the Vanilla GCN (left), the explanation highlights only a small set of structurally proximate edges, without any semantic role information. In contrast, the Semantic GCN (right) yields a richer explanatory subgraph that incorporates meaningful semantic relations (e.g., manages, reports_to, mentors). These relations provide clearer functional context and reveal semantically meaningful relational paths within the hierarchy. The inclusion of semantic edges introduces additional meaningful connections while also increasing the importance assigned to several edges already present in the structural-only model. This indicates that semantic context enhances the model's ability to identify relational patterns that reflect the structure and constraints of the underlying domain. As a result, the Semantic GCN yields more coherent and semantically grounded explanatory paths. Accordingly, the example provides a concrete visualization of the mechanism discussed conceptually in Section 4.1: semantic context can reshape explanatory evidence by adding relation-level meaning to otherwise purely structural neighborhoods.

4.5. Methodological challenges and design heuristics for semantic integration

The integration of semantic context through KGs has substantially advanced the transparency and reasoning capabilities of GNN-based stream processing systems. However, achieving explainable and transparent semantic stream reasoning requires more than embedding or injecting semantic information into the model. As the scope and complexity of real-world streaming environments increase, significant methodological and computational challenges emerge. These challenges directly influence the reliability, clarity, and operational feasibility of explanations generated by GNN-based systems. Therefore, a systematic assessment of the limitations of current semantic integration techniques is essential for understanding why existing methods still fall short of delivering robust and interpretable reasoning in practice. Beyond identifying these limitations, it is also important to clarify which practical design directions may help balance semantic richness, explainability, and the computational constraints of stream reasoning over evolving graphs.

Table 4 summarizes the central challenges identified across recent studies together with practical design directions that may help mitigate them in streaming settings, including latency-sensitive cases. The listed challenges reveal not only the technical bottlenecks in scaling or modeling semantic information, but also the conceptual gaps that undermine the goal of producing consistent, semantically grounded, and user-trustworthy explanations. By complementing open questions with implementation-oriented heuristics, the table clarifies how current limitations may be approached without overstating the maturity of existing solutions. These proposed strategies should be understood as implementation-oriented design heuristics synthesized from the reviewed literature, rather than as fully benchmarked universal solutions.

Table 4
Major challenges, research gaps, and practical design directions in semantic integration for GNN-based stream reasoning.

Challenge/ Research gap	Description	Existing methods	Suggested strategy/ Heuristic	Open questions
Scalability with Large KGs	Inference becomes difficult in large or dense KGs under high-volume streaming conditions, especially when semantic multi-relational reasoning must support latency-sensitive or real-time processing	GraphSAGE (neighborhood sampling) [94], Cluster-GCN (mini-batch clustering) [1], DropEdge (structure-aware) [33]	Selective semantic processing through semantic relevance filtering, soft semantic gating, incremental updates over changed graph regions, and staged explanation scheduling	How can these strategies preserve semantic fidelity and relational completeness under computational and latency constraints, including real-time stream processing requirements?
Interpretability of Semantic Embeddings	Latent embedding dimensions lack explicit semantic meaning and are difficult to interpret	Post-hoc projection and mapping [46], embedding-level attribution (GNNExplainer) [34], relation-aware attention for semantic cues [35]	Concept-to-dimension alignment, relation-level attribution, and post-hoc semantic decoding layers that map latent factors back to ontology concepts	How can embeddings be made traceable to explicit semantic concepts in stream reasoning?
Multi-relational Reasoning	Models must handle many relation types while preserving semantic fidelity	R-GCN [86], Multi-relational GAT [62], Heterogeneous GNNs [61]	Relation-type selection, schema-aware routing, and semantically weighted message passing to prioritize the most relevant relation families during inference and explanation	How can multi-relational inference scale while preserving semantic consistency across diverse relation types?
Dynamic/Temporal KG Integration	Semantic structure evolves over time, requiring continuous updates and stable explanations	Temporal GNNs [19], Dynamic Graph Embeddings [20]	Incremental temporal updates, temporal consistency constraints, and explanation smoothing across adjacent time windows	How can explanations remain stable as KGs change over time without masking genuine concept drift?
Evaluation Metrics for Semantic Explainability	Lack of standardized evaluation metrics for semantic clarity and faithfulness of explanations	Faithfulness and robustness metrics [46], human evaluation [30], taxonomy-based assessment [35]	Hybrid evaluation protocols combining semantic fidelity, relational coherence, temporal stability, and expert validation	How can semantic explainability be evaluated in a standardized way in evolving graph environments?
Generalizability Across Domains	Models overfit to specific ontologies or KG schemas, reducing cross-domain transferability	Pre-training [95], transfer learning on KGs [68], inductive GNNs [94]	Ontology-agnostic semantic abstractions, schema alignment layers, and cross-KG pretraining objectives	How can semantic stream reasoning generalize across different ontologies while preserving explanation quality?

First, scalability in large-scale and rapidly evolving KGs remains a persistent barrier. Streaming applications such as healthcare monitoring or social media analytics often involve graphs with millions of entities and highly diverse relation types. In latency-sensitive scenarios, including real-time stream processing, the computational overhead of semantic multi-relational reasoning can quickly become prohibitive when explanations must also remain semantically grounded. Techniques including neighborhood sampling (GraphSAGE [94]), mini-batch clustering (Cluster-GCN [1]), and edge pruning (DropEdge [33]) mitigate computational load but risk discarding semantically important structure. To move beyond framing scalability only as an open challenge, a practical methodological direction is selective semantic processing. Rather than applying full multi-relational reasoning over the complete KG at every update, the system can first restrict reasoning to semantically relevant relation types, ontology classes, and temporally local neighborhoods associated with the current prediction. Building on the semantic-aware workflow introduced earlier, this may be implemented through soft semantic gating, where messages are reweighted according to semantic compatibility instead of being removed entirely. This offers a more scalable compromise than strict symbolic filtering while preserving more relational meaning than aggressive pruning. A complementary staged explanation policy can further balance explanatory richness with computational feasibility: lightweight relation-aware explanations may be used for continuous monitoring, whereas richer subgraph-based or counterfactual explanations can be triggered selectively for high-risk, low-confidence, or user-queried cases. In this way, semantic relevance filtering, incremental updates over changed graph regions, and budget-aware explanation scheduling together provide a realistic heuristic for balancing semantic fidelity with the computational overhead of semantic multi-relational reasoning under the latency constraints of streaming inference, including real-time decision settings.

Second, the interpretability of semantic embeddings remains limited. Although embedding-based approaches such as TransE, ComplEx [53,88], or post-hoc mapping methods [46] encode rich semantic patterns, the latent dimensions lack explicit meaning. As a result, explanations derived from embedding features [34] are difficult to associate with concrete semantic concepts. This challenges the core objective of producing explanations that are semantically coherent rather than purely numerical. A key research direction is to determine how embedding spaces can reflect interpretable semantic factors without compromising predictive performance.

Third, multi-relational reasoning is essential for domains like biomedicine, law, or enterprise knowledge management, where decisions depend on complex interaction types. Models such as R-GCN [86], Multi-relational GAT [62], and heterogeneous GNNs [61] support diverse relation schemas, yet their computational cost increases rapidly with relation count. This directly impacts explanation quality, as oversimplification or compression of relation types can distort the explanation's semantic grounding. A promising design direction is therefore to combine relation-type selection with schema-aware message routing, so that only the most semantically relevant relation families are emphasized during inference and explanation generation. Such a strategy does not eliminate the challenge, but it offers a practical way to reduce unnecessary multi-relational overhead while preserving the relation patterns most critical to semantic consistency.

Fourth, dynamic and temporal KGs present additional complexity. Domains such as event stream analysis and fraud detection demand models that update their internal representations continuously. Methods like Temporal GNNs [19] and dynamic graph embeddings [20] address this challenge, but most fail to ensure temporal consistency in explanations. Explanations that fluctuate unpredictably over time can reduce trust. Hence, identifying mechanisms that maintain semantic and temporal coherence under rapid concept drift remains an important challenge.

Fifth, evaluation metrics for semantic explainability are underdeveloped. Existing metrics including faithfulness [46], human evaluation [30], and taxonomy based comparison frameworks [35], provide partial insights yet lack domain-invariant structure. Without standardized metrics, the reliability of semantic explanations cannot be systematically assessed, limiting progress toward transparent and comparable reasoning systems.

Finally, generalizability across domains remains limited. Semantic stream reasoning systems frequently overfit to a specific ontology or KG schema, hindering adaptation to new environments. Although approaches such as pre-training [95], transfer learning [68], and inductive GNNs [94] offer potential solutions, they do not yet guarantee robust semantic transfer. Since explanations depend on consistent semantic grounding, domain dependence reduces the interpretability and trustworthiness of reasoning outputs in cross-domain settings.

Together, these challenges illustrate that current semantic integration strategies are still insufficient to fully support the requirements of explainable and transparent stream reasoning. At the same time, the practical design directions outlined above indicate that progress does not depend only on richer semantic modeling, but also on selective processing strategies that explicitly balance semantic fidelity, explainability, and computational efficiency under streaming and latency-sensitive conditions. Addressing these research gaps is therefore essential for developing next-generation GNN-based systems capable of producing semantically coherent, scalable, and temporally reliable explanations in real-world, continuously evolving environments.

5. Evaluation metrics and frameworks for explainability

5.1. Current status and limitations of explainability metrics

Existing work on model interpretability has provided several criteria for evaluating explanations, including usefulness for human understanding, clarity of reasoning, and alignment with decision-making requirements. Early studies emphasized these qualitative dimensions as the foundation for assessing explanation quality [29,38,96]. Later analyses argued that, without clearer standards for what an explanation should capture, evaluations tend to diverge across methods and domains [40,42].

In the context of GNNs, more recent surveys have catalogued technical metrics such as faithfulness, sparsity, stability, and local consistency, and shown that each of these metrics captures only limited aspects of explanation behavior [7,35,46,76]. Furthermore, empirical studies demonstrate that different explanation techniques often behave inconsistently under the same evaluation criteria, by highlighting significant gaps in the reliability and comparability of current metric suites [22,23]. Together, these findings suggest that while the field possesses a diverse range of evaluation tools, there is limited consensus on determining which metrics are most appropriate for specific explanation objectives.

These limitations become more pronounced when semantic knowledge is incorporated into GNN-based reasoning. As shown in the previous section, semantic enrichment modifies both the structure of the model's reasoning process and the form of the resulting explanations. However, existing metrics do not assess whether an explanation preserves the meaning of KG relations, conforms to ontology-level constraints, or remains consistent as the underlying graph evolves over time. Because these semantic and temporal properties are central to stream-based reasoning, current evaluation practices offer only a partial view of explanation quality.

Overall, the literature provides useful tools for assessing structural and behavioral aspects of GNN explanations. Yet these tools remain insufficient for settings where semantic fidelity, relational coherence, and temporal stability are essential. This gap underscores the need for evaluation frameworks that explicitly account for the knowledge-aware and dynamic characteristics of semantic stream reasoning.

To the best of our knowledge, no widely accepted benchmark jointly evaluates these semantic and temporal explanation properties in knowledge-aware stream reasoning. While several benchmark suites exist for static graph learning and GNN explainability, they typically focus on fixed datasets, single-task settings, and structure-level explanations. Consequently, they fail to capture the combined requirements of semantic grounding, temporal evolution, and knowledge consistency that characterize stream reasoning over knowledge-enriched graphs. As a result, existing benchmarks cannot be directly reused to rigorously evaluate semantic fidelity, relational coherence, or temporal semantic stability in streaming environments.

In this regard, rather than proposing a new benchmark, this survey identifies key evaluation dimensions that future benchmark design should consider, including alignment with KG semantics, stability under temporal updates, and human-centered domain validity. These dimensions aim to guide the development of rigorous and meaningful evaluation protocols for explainable GNN-based stream reasoning.

To support future research on explainability in GNN-based stream reasoning, Table 5 provides a compact survey-oriented overview of representative datasets and data sources relevant to this area. The table is not intended to define a new benchmark suite, compare dataset difficulty, or reposition this review as an empirical benchmarking study. Instead, it identifies data resources used in temporal graph learning, temporal KG reasoning, and semantically grounded explanation analysis, and summarizes the basic dataset statistics that are commonly needed for future evaluation design.

The table includes temporal interaction graphs, temporal KGs, and clinically derived healthcare graph sources. For standardized public datasets, the statistics reflect commonly reported dataset descriptors, including nodes or entities, temporal interactions or facts, timestamps, feature information, and label availability. For MIMIC-IV FHIR, the statistics correspond to the patient-level graph construction used in the bounded feasibility illustration in Section 5.4. Thus, MIMIC-IV FHIR is included as a representative real-world healthcare data source for illustrating the computability of selected knowledge-aware evaluation dimensions, rather than as a standardized real-time stream-processing benchmark.

5.2. Common metrics for evaluating explainability and transparency

The evaluation of explanations generated by GNN models necessitates metrics that capture multiple dimensions of explanatory quality. Prior research in explainable machine learning (ML) and GNN explainability has identified several widely adopted metric families, as summarized in Table 6. Each family targets a specific property of explanations. Although these categories provide a structured foundation for existing evaluation practices, they face the limitations discussed in Section 5.1, particularly when explanations incorporate semantic or KG structure and standard metrics often face additional challenges in streaming environments.

A central category of metrics focuses on *faithfulness*, which measures whether an explanation accurately reflects the model's decision process. Faithfulness is typically assessed by perturbing or removing the features or graph components highlighted by the explainer and observing the resulting change in prediction confidence. The studies in both general XAI [38] and GNN-specific attribution evaluation [76] emphasize that

Table 5
Survey-oriented overview of representative datasets and data sources for future research on explainable GNN-based stream reasoning.

Dataset	Data setting	Key statistics and features	Representative use
Wikipedia [97]	Temporal interaction graph; event sequence	9227 nodes; 157,474 temporal interactions; text features; dynamic binary labels	JODIE, TGAT, TGN, temporal GNN explainers
Reddit [97]	Temporal interaction graph; event sequence	10,984 nodes; 672,447 temporal interactions; text features; dynamic binary labels	JODIE, TGAT, TGN, temporal GNN explainers
MOOC [97]	Temporal interaction graph; event sequence	7144 nodes; 411,749 temporal interactions; interaction features; task-dependent binary labels	JODIE/TGN-style temporal interaction studies
LastFM [98]	Temporal interaction graph; event sequence	1980 nodes; 1,293,103 temporal interactions; task-dependent features; no fixed class labels	JODIE/TGN-style temporal interaction studies
ICEWS14 [99]	Temporal KG; 365 timestamps	7128 entities; 230 relations; 90,730 temporal facts; relation prediction setting; no standard node features	Temporal KG reasoning and explanation studies
ICEWS05-15 [100]	Temporal KG; 4017 timestamps	10,488 entities; 251 relations; 479,329 temporal facts; relation prediction setting; no standard node features	Temporal KG reasoning and explanation studies
MIMIC-IV FHIR Demo [101]	Healthcare temporal graph source	45 patient-level graph sequences derived in Section 5.4; at least two time steps per patient; FHIR resource types and typed clinical relations	Bounded illustration of Relational Coherence and Temporal Semantic Stability.

Table 6
Evaluation metrics for explainability in GNN-based stream reasoning and associated streaming challenges.

Metric category	Purpose/What it measures	Representative works	Semantic/KG awareness	Challenges in stream settings
Faithfulness	Measures whether the explanation reflects the model's actual decision process; typically evaluated via perturbation/removal tests.	Lipton [38], Sanchez-Lengeling et al. [76], Agarwal et al. [22,46]	Focuses on model behavior; only indirectly reflects KG semantics.	High Computational Latency: Testing fidelity requires running the model multiple times per event, which creates significant delays in real-time streams.
Sparsity	Assesses how concise the explanation is by quantifying the minimal subgraph/feature set required for a prediction.	Adadi & Berrada [29], Luo et al. [71], Yuan et al. [35]	Measures compactness; does not guarantee semantic coverage.	Throughput Trade-off: The optimization process needed to find minimal subgraphs slows down the overall data processing rate (throughput).
Stability	Evaluates consistency of explanations under small input perturbations or across similar instances.	Shan et al. [73], Huang et al. [77], Agarwal et al. [22]	Captures temporal consistency but ignores temporal validity; fails to distinguish between stable explanations and those grounded in obsolete facts.	Sensitivity to Concept Drift and Knowledge Obsolescence: As the stream evolves, older KG facts may become irrelevant. Current metrics often fail to penalize explanations that rely on this outdated information.
Human-centered/ Task-alignment	Measures alignment with expert expectations, human reasoning, domain knowledge, and task performance improvement.	Gilpin et al. [96], Rudin [40], Doshi-Velez & Kim [30], Mittelstadt et al. [41]	Aligns with expert reasoning; semantic awareness depends on how KG knowledge is incorporated.	Cognitive Bottleneck: Human evaluation speeds cannot match data stream velocity; necessitates automated, semantically-grounded proxy metrics.

faithfulness is essential for determining whether an explanation corresponds to the computations performed by the model. Recent evaluation frameworks further systematize faithfulness-based assessment for GNN explanations by combining it with other dimensions such as stability and fairness [22,46].

A second family of metrics focuses on *sparsity* or compactness. These metrics assess whether an explanation highlights only the essential nodes, edges, or features that influence the model's prediction. The importance of concise and interpretable explanations has been emphasized in the broader interpretability literature [29]. Following this principle, several GNN explainers introduce explicit sparsity constraints to ensure that explanations concentrate on the most relevant components [71]. However, sparsity alone does not guarantee explanatory quality, because excessively minimal explanations may fail to reflect the reasoning that the model actually performs. Thus, sparsity is most appropriately evaluated together with faithfulness, which are often discussed jointly in recent surveys on GNN explainability [35].

A third family of metrics examines *stability*, which assesses whether explanations remain consistent under small perturbations of the input or across similar graph instances. Stability is especially relevant in streaming scenarios, where the graph structure evolves incrementally and explanations are expected to change smoothly over time. Prior analyses of learning-based explainers show that even minimal input modifications can lead to noticeably different explanations [73]. Similar observations have been reported for local GNN explanation methods, where minor variations in node features or graph structure can alter the extracted explanatory subgraphs [77]. Recent evaluation studies further emphasize stability as a central dimension of explanation quality for GNN models, considered alongside other robustness-related criteria such as faithfulness and fairness [22].

Finally, *human-centered* or task-alignment metrics assess how well explanations correspond to human expectations, domain knowledge, or task-specific reasoning processes. Foundational discussions in interpretability research emphasize that aligning explanations with human

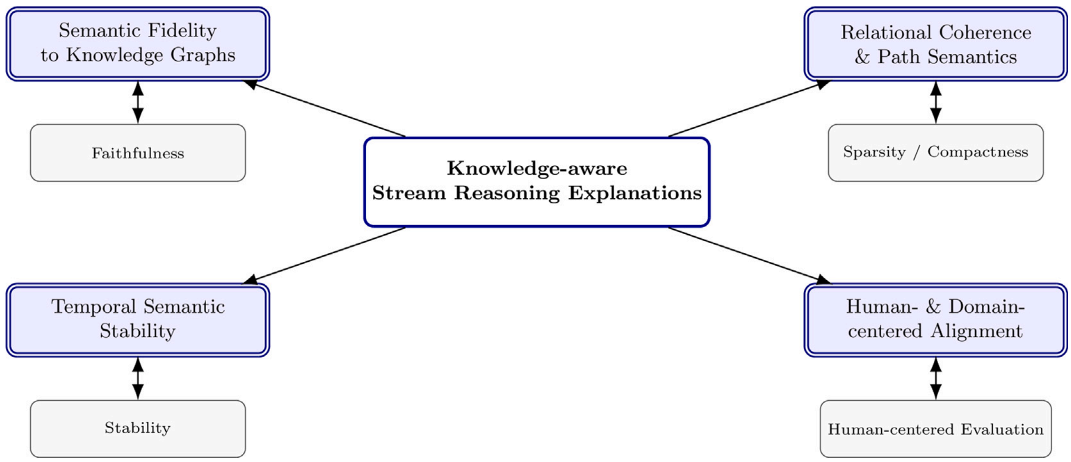

Fig. 4. Conceptual diagram illustrating how the four proposed *knowledge-aware evaluation dimensions* relate to and extend classical GNN explanation metrics. These dimensions such as semantic fidelity, relational coherence, temporal semantic stability, and human-centered alignment extend existing metric families by adding a semantic perspective. They complement traditional metrics such as faithfulness, sparsity, stability, and human-centered evaluation. Together, they provide a conceptual basis for future *knowledge-aware evaluation protocols* and benchmark design across static, temporal, and knowledge-enriched graph reasoning tasks.

judgment is essential, particularly in high-stakes settings where users should trust and validate model outputs [40,96]. Complementary perspectives argue for more principled and structured forms of human evaluation, highlighting the need for rigorous frameworks that connect explanatory quality with human understanding and task performance [30,41]. Despite their importance, human-centered evaluations remain costly and subjective, and they offer limited insight into explanation quality for semantic reasoning tasks that depend on structured knowledge resources.

In summary, the metric categories in Table 6 offer a structured basis for assessing explanations in standard GNN settings. However, they do not address whether explanations remain semantically meaningful, ontology-consistent, or relationally coherent when models operate over dynamic KG streams. This limitation is reflected in the semantic-awareness column of Table 6, which highlights that existing metrics provide only indirect insight into KG-level semantics. These observations motivate the knowledge-aware evaluation dimensions in the following section, which conceptually extend current metrics toward semantic and KG-oriented explanation quality.

5.3. Knowledge-aware evaluation dimensions for GNN explanations

While existing metrics such as faithfulness, sparsity, stability, and human-centered evaluation provide a structured basis for assessing GNN explanations, they predominantly focus on structural and behavioral properties, with only limited and indirect coverage of semantic and KG-aware aspects [1,2]. In knowledge-aware stream reasoning, decisions rely on evolving relational structure and often incorporate KGs [36,37]. Explanations in such settings are therefore expected to reflect ontology-level meaning, coherent relational patterns, and semantics that remain consistent over time [36,37]. This subsection outlines four complementary analytical evaluation dimensions that extend current metrics toward these knowledge-aware requirements. These dimensions are intended to guide future metric and benchmark design rather than to constitute finalized evaluation protocols. To provide an overview of these dimensions and their relationship to classical evaluation metrics, Fig. 4 illustrates the conceptual framework used in this section.

5.3.1. Semantic fidelity to KGs

Semantic fidelity captures the extent to which an explanation is consistent with the underlying KG, including entity types, relation constraints, and schema-level dependencies [36]. Classical faithfulness metrics assess whether explanations align with the model's decision process such as perturbation-based tests, feature evaluation or gradient-based analysis [22,38,46,76,102]. Semantic fidelity instead evaluates whether an explanation aligns with the semantics of the underlying KG, such as using appropriate relation types and respecting basic schema constraints [37,53,86]. Existing work on KG embeddings and knowledge-enhanced GNNs demonstrates how such semantic information can be encoded into models [52,67,68], but evaluation usually remains task-centric or relies on model-level faithfulness. However, rigorously assessing this fidelity presents significant challenges; if the underlying KG is highly complex or dense, verifying the semantic consistency of explanations against large-scale schemas can be computationally prohibitive [36,68]. Consequently, semantic fidelity emphasizes the need for efficient metrics that assess whether explanations follow KG semantics without incurring excessive overhead. These metrics complement existing faithfulness measures by adding a semantic perspective that current evaluations do not capture. Indicative measurement strategies for quantifying this dimension are summarized in Table 7.

5.3.2. Relational coherence and path semantics

Relational coherence focuses on whether an explanation reveals semantically meaningful relational patterns and paths, rather than arbitrary subsets of edges. Subgraph and path-based explainers such as information-theoretic, prototype-level, and counterfactual methods already extract influential subgraphs or relational motifs in GNNs, including systematic subgraph exploration approaches [24,34,47,72,103]. Their evaluation focuses on structural aspects such as faithfulness, sparsity, or visual interpretability, with little attention to whether the extracted paths reflect domain-specific reasoning in KGs or other structured sources [35,59,71]. In domains such as bioinformatics and recommendation, knowledge-enhanced GNNs use KG structure to generate more interpretable decision pathways [5,11,52,67]. However, explanation quality is often evaluated indirectly through predictive performance or case studies. A relational coherence dimension encourages metrics that compare explanation paths with KG-based reasoning patterns and evaluate how well they capture semantically meaningful, domain-aligned relational chains. The operational strategy for measuring such path similarity is outlined in Table 7.

For example, within a medical diagnosis knowledge graph, a semantically coherent explanation should follow a diagnostic reasoning chain:

$$\text{Fever} \xrightarrow{symptom\_of} \text{Infection} \xrightarrow{treated\_by} \text{Antibiotic}$$

Conversely, a model relying solely on structural connectivity might produce an irrelevant path that fails to explain the decision:

$$\text{Fever} \xrightarrow{\textit{reported\_by}} \text{Patient} \xrightarrow{\textit{works\_at}} \text{Bank}$$

To distinguish between these cases, evaluation metrics should verify whether the extracted paths match expert-validated "reasoning templates" (meta-paths), rather than merely checking for node connectivity.

5.3.3. Temporal semantic stability in streaming graphs

In stream reasoning, both the observed graph and any associated KG may evolve over time, so creating a need for explanations that remain semantically stable under temporal change [20,28]. Existing stability metrics for GNN explanations primarily evaluate robustness to small input perturbations or local modifications of the graph structure [22, 73,77]. These metrics provide valuable insight into local robustness but do not capture how explanation semantics drift as relational structure and KG content are updated. Temporal GNNs and inductive temporal graph models explicitly represent evolving relationships in dynamic graphs [28,54]. Also temporal KG reasoning approaches address time-dependent inference over KGs [18]. Nevertheless, explanation quality in these works is typically assessed at the level of prediction accuracy or static snapshots. The temporal semantic stability dimension highlights the need for metrics that relate changes in explanatory subgraphs to the evolution of underlying graphs and KGs. This includes tracking semantic drift in explanation paths over time in streaming environments, as formulated in Table 7.

5.3.4. Human and domain-centered semantic alignment

Human-centered or task-aligned evaluation remains essential, particularly in high-stakes domains where stream reasoning systems support critical decisions. Existing human-based metrics in interpretability research assess properties such as trust, clarity, usefulness, and decision support quality [29,39,96]. These studies emphasize that explanations should align with human reasoning patterns and improve task performance, especially when black-box models are deployed in sensitive settings [29,40,41]. In knowledge-aware stream reasoning, domain experts additionally rely on structured knowledge often captured in KGs or domain-specific schemas when judging whether explanations are meaningful [36,37]. The human and domain-centered semantic alignment dimension calls for evaluation protocols that incorporate expert judgment. Experts should assess not only whether explanations seem reasonable, but also whether they align with key domain concepts, relations, and constraints. Such protocols can be combined with quantitative measures from the other dimensions to provide a more complete view of explanation quality in practice. Indicative strategies for assessing this alignment against domain rules or expert-defined criteria are summarized in Table 7.

To make the computable aspects of these dimensions more explicit, we define below illustrative formalizations for cases in which an explanation can be represented as a typed subgraph or path. These definitions are not intended to constitute a finalized benchmark protocol; rather, they provide concrete metric instantiations that can support future evaluation design.

Let $E_t = (V_E^t, A_E^t)$ denote the explanation subgraph produced at time t, where each explanatory edge is represented as $(u, r, v) \in A_E^t$. Let $\tau(u)$ and $\tau(v)$ denote the entity types of nodes u and v, and let C_K denote the set of type–relation–type constraints licensed by the KG schema. A simple semantic fidelity score can then be defined as

$$\mathrm{SF}(E_t, K) = \frac{1}{|A_E^t|} \sum_{(u,r,v)\in A_E^t} \mathbf{1}\left[(\tau(u), r, \tau(v)) \in C_K\right],$$

where $\mathbf{1}[\cdot]$ is an indicator function. This score measures the proportion of explanatory relations that are schema-consistent with the KG.

For relational coherence, let $\Pi(E_t)$ be the set of explanation paths extracted from E_t, and let $\mathcal{M}$ denote a library of expert-defined or schema-derived meta-path templates. If $\rho(\pi)$ denotes the sequence of relation types in path π, a path-level coherence score can be expressed as

$$\mathrm{RC}(E_t, \mathcal{M}) = \frac{1}{|\Pi(E_t)|} \sum_{\pi\in\Pi(E_t)} \max_{m\in\mathcal{M}} \left(1 - \frac{d_{\mathrm{edit}}(\rho(\pi), m)}{\max(|\rho(\pi)|, |m|)}\right),$$

where d_{edit} is the edit distance between the extracted relation sequence and the closest valid meta-path. This formulation corresponds to the normalized path similarity strategy used to assess whether explanations follow domain-relevant relational chains.

For temporal semantic stability, let $\phi(E_t)$ be a semantic representation of the explanation at time t, constructed from relation types, node types, KG concepts, or stable domain codes. Over a sequence of T explanation subgraphs, temporal semantic stability can be computed as

$$\mathrm{TSS} = \frac{1}{T-1} \sum_{t=2}^{T} \frac{1 + \cos(\phi(E_t), \phi(E_{t-1}))}{2}.$$

The normalization maps cosine similarity to the interval $[0, 1]$, where higher values indicate greater semantic continuity between consecutive explanations.

Finally, when expert knowledge can be represented as a rule-derived or expert-validated reference explanation E_t^*, domain alignment can be measured through edge-level overlap. Let A_E^t and $A_{E^*}^t$ denote the edge sets of the model explanation and the reference explanation, respectively. Precision, recall, and their harmonic mean can be defined as

$$P_{\mathrm{DA}} = \frac{|A_E^t \cap A_{E^*}^t|}{|A_E^t|}, \qquad R_{\mathrm{DA}} = \frac{|A_E^t \cap A_{E^*}^t|}{|A_{E^*}^t|},$$

$$\mathrm{DA} = \frac{2P_{\mathrm{DA}}R_{\mathrm{DA}}}{P_{\mathrm{DA}} + R_{\mathrm{DA}}}.$$

This formalization applies when domain expectations can be encoded as reference rules, paths, or subgraphs. In settings where expert alignment is assessed through human judgment rather than graph overlap, the same dimension may require complementary qualitative or task-based evaluation.

Collectively, these dimensions extend existing evaluation practices by incorporating the requirements of knowledge-aware stream reasoning [22,35,46]. Rather than replacing traditional metrics like faithfulness or sparsity, they are designed to complement them with a semantic perspective. Table 7 translates these dimensions into indicative measurement strategies that can support future protocol and benchmark design. However, their standardization, validation, and full integration into deployment-level benchmarks remain open directions for the research community.

Maintaining these dimensions under realistic streaming constraints also requires additional design choices. In latency-sensitive settings, semantic fidelity or relational coherence cannot always be recomputed over the full evolving graph after every update. Practical implementations may therefore rely on incremental computation over changed graph regions, window-based evaluation, cached schema checks, and selective evaluation for high-risk or low-confidence predictions. Throughput constraints may further require separating lightweight continuous checks, such as type and relation validity, from more expensive assessments, such as full path coherence, counterfactual consistency, or expert-alignment evaluation. Therefore, the dimensions proposed here should be understood as analytical targets and indicative measurement directions whose deployment under online stream processing constraints remains an important topic for future work.

5.4. Bounded empirical feasibility study on a real-world dynamic healthcare dataset

This study is not intended as a real-time stream processing benchmark, an evaluation of latency or throughput, or a validation of an

Table 7
Knowledge-aware evaluation dimensions and indicative measurement strategies for assessing explanation quality in knowledge-enriched streaming environments.

Dimension	Operational definition	Measurement strategy (How to calculate)
Semantic Fidelity	Ensures that the explanation respects entity types, relation constraints, and schema-level dependencies defined in the underlying Knowledge Graph.	Calculate the percentage of relations in the explanation subgraph that validly connect two entity types according to the domain schema (e.g., checking if "Treatment" is valid for "Disease").
Relational Coherence	Assesses whether the explanation path follows semantically meaningful, domain-relevant relational chains rather than arbitrary graph connections.	Compare the extracted explanation path against a library of expert-defined reasoning patterns (meta-paths) and compute a path similarity score (e.g., normalized path edit similarity).
Temporal Semantic Stability	Measures whether the explanation's semantic meaning remains consistent over consecutive time steps, distinguishing genuine concept drift from model instability.	Track the semantic embedding of the explanation subgraph across consecutive time steps. A high stability score implies low variance in the embedding space unless the input event significantly changes.
Domain Alignment	Evaluates how well the explanation matches expert-specified domain logic, rules, or protocols.	Measure the overlap (Recall and Precision) between the edges in the model's explanation and a "Ground Truth" graph derived from expert rules or protocols.

end-to-end temporal GNN system. Rather, its purpose is to examine whether the proposed knowledge-aware evaluation dimensions can be operationalized on temporally ordered, real-world healthcare graph data within the scope of a review article. To complement the conceptual framework introduced in Section 5.3 with a quantitative proof-of-concept, we conducted a bounded empirical feasibility study on the *MIMIC-IV Clinical Database Demo on FHIR* (v2.1) [101,104,105]. We selected the healthcare domain because it is a high-stakes setting in which semantically grounded and temporally stable explanations are particularly important. Temporally evolving patient trajectories, heterogeneous clinical entities, and knowledge-rich relations make this dataset well suited for instantiating the proposed knowledge-aware evaluation dimensions, and provide a real-world clinical setting in which event sequences can be modeled as typed temporally evolving healthcare graph data. In contrast to synthetic benchmark data, the dataset preserves authentic encounter-level clinical irregularities through interoperable resources such as observations, diagnoses, procedures, and medication events.

We constructed patient-level temporally ordered graph sequences from encounter-centered FHIR resources, including *Patient*, *Encounter*, *Observation*, *Condition*, *MedicationAdministration*, and *Procedure*. For each encounter, abnormal laboratory observations were first identified, and strongest abnormal event was selected as the anchor of the local explanation context. Around this anchor, we extracted a compact proxy explanation subgraph consisting of a patient node, an encounter node, an abnormal observation node, the two most relevant condition nodes, and the single most relevant intervention node (medication or procedure). Relevance was estimated using lexical similarity across code labels and textual descriptors, together with a small temporal consistency bonus favoring concepts that recur across a patient's prior encounters. When available, interventions explicitly linked to the selected condition layer were given additional priority. Applying this procedure to the dataset produced temporally ordered explanation streams for 45 patients, each with at least two valid time steps.

The resulting explanation subgraphs are organized around the following typed relations:

$$\texttt{Patient} \xrightarrow{\texttt{has\_encounter}} \texttt{Encounter} \xrightarrow{\texttt{has\_observation}} \texttt{Observation},$$

$$\texttt{Encounter} \xrightarrow{\texttt{has\_condition}} \texttt{Condition},$$

$$\texttt{Encounter} \xrightarrow{\texttt{has\_medication/has\_procedure}} \texttt{Medication/Procedure},$$

$$\texttt{Medication/Procedure} \xrightarrow{\texttt{addresses}} \texttt{Condition}.$$

Table 8
Empirical feasibility results on the MIMIC-IV Clinical Database Demo on FHIR, reported over patient-level temporal explanation graph sequences.

Dimension	Mean	Std.
Relational Coherence	0.764	0.033
Temporal Semantic Stability	0.812	0.060

Accordingly, the temporal dimension is represented by the ordering of encounters and abnormal observations, whereas the knowledge-aware dimension is represented by typed clinical entities and semantically constrained relations derived from the FHIR resources.

The present case study instantiates the knowledge-aware evaluation dimensions summarized in Table 7. In particular, it operationalizes *Relational Coherence* and *Temporal Semantic Stability*, as these two dimensions most directly capture whether explanations preserve domain-relevant relational structure and semantically consistent meaning over time in streaming settings.

For *Relational Coherence*, explanation paths were extracted from each proxy explanation graph and compared against a library of clinically meaningful meta-path templates using normalized path edit similarity. This operationalization evaluates whether the extracted explanation follows semantically meaningful and domain-relevant relational chains rather than arbitrary graph connections. *Temporal Semantic Stability*, in turn, was quantified by converting each explanation subgraph into a compact semantic representation based on relation types, node types, and stable clinical codes, and then comparing consecutive explanations using cosine similarity across time steps. This operationalization evaluates whether the semantic content of the explanation remains stable over time despite the evolution of the underlying clinical stream. Although the present empirical use case focuses on these two dimensions, the graph construction procedure remains compatible with the broader knowledge-aware evaluation space, including schema-constrained validity checks and rule-based overlap analysis.

As reported in Table 8, the resulting patient-level streams show that the proposed operationalizations of *Relational Coherence* and *Temporal Semantic Stability* can be computed in a stable and interpretable way on a real-world, temporally evolving healthcare dataset. Across the evaluated streams, *Relational Coherence* reached a mean of 0.764, while *Temporal Semantic Stability* reached a mean of 0.812. Rather than serving as benchmark-style performance targets, these values provide

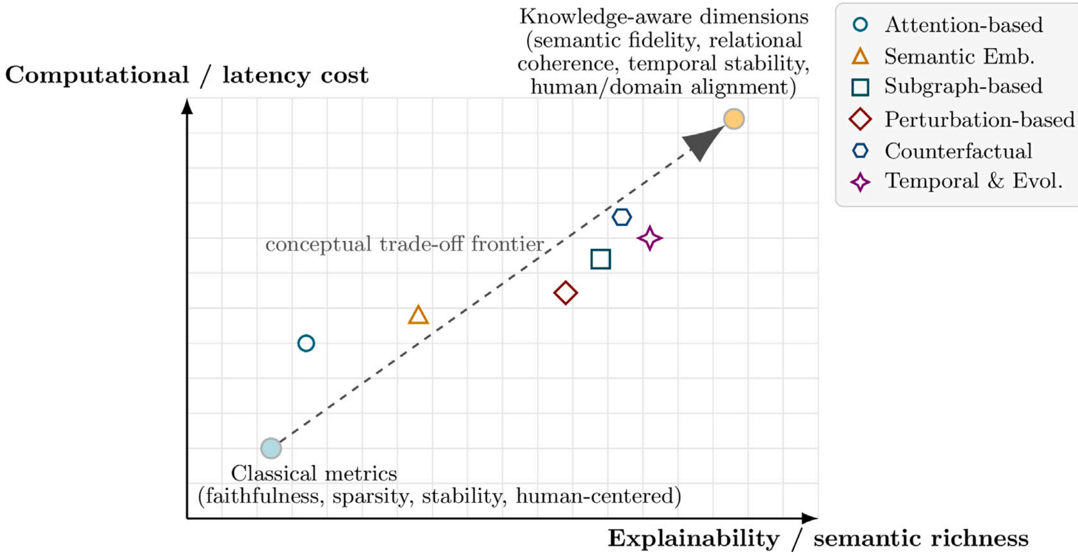

Fig. 5. Conceptual trade-off frontier between explainability and computational cost in stream reasoning. Classical evaluation criteria (faithfulness and stability, discussed in Section 5.2) and knowledge-aware dimensions (semantic fidelity, relational coherence, and temporal semantic stability, detailed in Section 5.3) anchor the two ends of the spectrum. The markers indicate the typical regions occupied in streaming settings by attention-based, semantic-embedding, subgraph-based, and perturbation-based explainers (overviewed in Section 3.1), alongside the semantically richer counterfactual and temporal evolution approaches. The dashed arrow illustrates the conceptual progression toward semantically richer yet more computationally intensive explanations.

quantitative evidence that the proposed measurement strategies can capture semantically meaningful relational patterns and their temporal persistence in practice. In this respect, the case study offers a bounded empirical illustration showing that selected knowledge-aware evaluation dimensions can be instantiated and computed on real-world temporally evolving healthcare graph data. These results should not be interpreted as validation of online stream reasoning performance, since the study does not measure latency, throughput, online updating behavior, or explanations generated by a deployed GNN model. Instead, the study provides a bounded proof-of-concept for metric computability and interpretability, while leaving online evaluation, model-generated explanations, and deployment-level streaming benchmarks as future work.

6. Explanability and performance trade-offs in stream reasoning

In stream reasoning applications, explainability requirements interact directly with constraints on predictive accuracy, latency, and resource usage. Existing GNN explainability methods were largely developed for static or batched settings, and their computational profiles can change significantly when applied to high-frequency graph streams [1,2]. When models use structured knowledge such as KGs, explanations should reflect both the evolving graph and the knowledge-awared evaluation criteria discussed in Section 5.3. [36,37]. In this section, we provide a conceptual synthesis of the associated performance and semantic trade-offs in streaming settings, rather than an empirical comparison [36,37].

6.1. Impact of explainability on accuracy and latency

Different families of GNN explainability methods impose distinct accuracy latency profiles in stream reasoning scenarios, as sketched in Fig. 5. Perturbation-based and feature-based explanation techniques often introduce a substantial computational burden because they require multiple forward passes or iterative optimization procedures [47,76,103]. While effective for static analysis, this cost is prohibitive for high-frequency streams.

Subgraph extraction methods and instance level explanation models can provide detailed and faithful explanations. However, their computational overhead make them difficult to execute at the frequency required for real-time decision making in large or rapidly evolving graphs [34,71,73]. In contrast, attention based mechanisms and intrinsic interpretability modules integrate explanation signals directly into the forward pass. They generally introduce lower additional computational cost, although they provide less precise control over explanation fidelity and robustness [59–61].

Semantic embedding methods incorporate KG information into node and edge representations. This integration can increase model complexity and explanation overhead, particularly when embeddings need to be updated in response to changes in the data stream or the underlying KG [52,53,67]. The semantically richest approaches impose the highest latency penalties due to their algorithmic complexity. Counterfactual explainers provide actionable recourse but rely on computationally intensive iterative optimization to identify decision boundaries [75,80,82]. Similarly, temporal evolution methods incur a continuous overhead as they require tracking historical dependencies over evolving time windows to capture dynamic causal patterns [13,83].

In dynamic environments, these costs are amplified because models must adapt to changes in node features, edge structure, and temporal patterns [20,28]. Temporal GNN architectures and dynamic representation learning methods must recompute or incrementally update explanations as new events are observed. This continual updating can place significant pressure on latency budgets in high-throughput streaming environments [28,65,106]. Designing explainability pipelines for such settings therefore requires explicit choices about the frequency, granularity, and type of explanations that can be supported without degrading predictive accuracy or violating real-time constraints [40,41]. In some applications, lightweight explanations that are generated at periodic intervals are preferable to dense explanations produced for every event. In high-stakes applications, higher latency may be acceptable when it enables the delivery of richer and more faithful explanatory insight [40,96]. This trade-off is summarized conceptually in Fig. 5, which positions major explanation families along the accuracy–latency and semantic richness axes.

6.2. Trade-offs in semantic and knowledge-aware evaluation

Incorporating semantic and knowledge-aware criteria defined in Section 5.3 introduces additional forms of tension beyond traditional concerns with accuracy and latency. First, enforcing *semantic fidelity* requires explanations to conform to rigid ontology constraints even when data-driven patterns suggest otherwise [36,37,53]. While necessary for consistency, these constraints can limit the model's flexibility and reduce short-term predictive performance by overriding purely statistical associations [25,52,67]. Similarly, prioritizing *relational coherence* encourages explanations that preserve meaningful multi-hop chains rather than minimal edge sets. This requirement directly conflicts with the goal of sparsity, as it often increases the size of the explanation to ensure semantic completeness [24,34,35].

On the other hand, *Temporal semantic stability* (as discussed in Section 5.3.3) adds another axis of tension. Stream reasoning demands explanations that evolve smoothly over time, yet models must also adapt quickly to concept drift [18,20]. Strong stability constraints may conflict with the objectives of temporal evolution methods, which aim to faithfully highlight rapid changes in reasoning patterns when new KG facts emerge [22,73,77]. Finally, *human and domain-centered alignment* creates a trade-off between statistical optimization and expert validation. Aligning explanations with expert heuristics often necessitates counterfactual reasoning to verify causal pathways, requiring the model to sacrifice some degree of compactness or raw predictive fit to ensure the output matches domain mental models [29,30,96]. In practice, harmonizing these conflicting objectives requires domain-specific strategies to establish the optimal trade-off between computational efficiency and semantic consistency [41,56].

7. Use cases in critical domains

The scenarios presented in this section are intended to concretely instantiate the research gaps identified in Section 1.3, the stream-reasoning requirements discussed in Section 2.1, and the methodological challenges of semantic integration analyzed in Section 4.5. In particular, they focus on the interaction between semantic knowledge, explainability requirements, and stream-specific constraints in GNN-based reasoning over evolving graphs. Inspired by the abstraction from stream to graph introduced in STKGNN [69], we consider a generic framework for GNNs enhanced with knowledge, operating on spatio-temporal KGs derived from heterogeneous sensor, video, social interaction, mobility, and log streams.

In this formulation, explainability is treated conceptually as a primary modeling requirement rather than a purely diagnostic step performed post hoc. To support this objective, the framework integrates explanation techniques at the GNN level [47,73] together with strategies for local and counterfactual interpretation [74,77,80]. These are combined with mechanisms for rule-based and path-centric reasoning from the literature on KGs [10,51,85] and informed by established evaluation principles for models enhanced with knowledge [22,25].

To illustrate how these principles apply across diverse domains, Fig. 6 presents a conceptual architectural framework. As a representative example, this framework abstracts the workflow into three phases: an *Input* phase that structures heterogeneous streams into graph abstractions ($\mathcal{G}$); a *Reasoning* phase where semantic context explicitly guides the GNN explanations; and an *Inference* phase that validates predictions against domain constraints. This conceptual model serves as the common architectural logic for the specific use cases discussed below. At a more operational level, Section 4.3 and Algorithm 1 detail how semantic constraint enforcement can be incorporated into the *Reasoning* phase of this conceptual framework.

These scenarios are not intended to report fully implemented domain-specific systems or benchmark results. Rather, they make the proposed framework more concrete by specifying representative GNN backbones, KG structures, explanation forms, evaluation dimensions, and stream-specific operational constraints for each domain. To distinguish these scenarios from standard temporal graph learning examples, each use case is interpreted not merely as a setting in which a graph evolves over time, but as a stream-reasoning setting in which explanations must be generated under window-based semantics, possible late or out-of-order event arrivals, limited computational budgets, resource-aware explanation scheduling, and latency-sensitive decision requirements. These constraints determine when explanations are updated, whether lightweight or detailed explanations are produced, and how semantic consistency is maintained as new events enter the stream.

7.1. Stream-aware healthcare and security instantiations

7.1.1. Healthcare monitoring

A representative healthcare instantiation could use a spatio-temporal GCN or a TGN-style temporal GNN over sensor, pose, encounter, and clinical event streams. The associated KG may include patients, body joints, sensor events, clinical conditions, and risk factors, with relations such as *has_posture*, *observed_during*, *near_ground*, and *has_risk_factor*. Spatio-temporal KGs constructed from continuous monitoring streams can support activity recognition grounded in clinically meaningful relations [6,11,93]. For example, fall detection may be modeled as a concise temporal condition over relations between entities representing body posture, proximity to the ground, and relevant clinical risk factors. An explanation for a fall-risk alert would therefore be supported by a temporally persistent path linking abnormal posture, proximity to the ground, and patient-specific risk information. The most relevant evaluation dimensions are Temporal Semantic Stability and Relational Coherence.

Stream-reasoning constraints. This scenario requires explanations to be computed over clinically meaningful time windows, since sensor observations, pose estimates, laboratory values, and encounter events may arrive at different rates. Late, missing, or corrected clinical events may require retrospective updates to the explanation, while resource constraints favor lightweight continuous semantic checks and more detailed explanations only for persistent, high-risk, or clinically ambiguous alerts. This distinguishes the scenario from a standard temporal graph benchmark: the central challenge is not only to identify which historical observations influenced a prediction, but also to maintain semantically valid explanations as heterogeneous clinical streams arrive asynchronously.

7.1.2. Security and surveillance

A representative security instantiation could use a temporal heterogeneous GNN or an R-GCN with temporal updates over video-derived object tracks, trajectories, and scene events. The KG may represent persons, objects, locations, cameras, and trajectories, with relations such as *associated_with*, *left_behind*, *near*, and *located_in*. In security and surveillance settings, relations between owners and objects, trajectories, and scene context can be encoded as an evolving KG [10]. Under this representation, an abandoned-object alert may be explained by a relational path showing that an object remains stationary after its associated person leaves the scene. Rule-guided and path-centric explanation mechanisms are especially relevant here because the explanation depends on persistent and semantically valid relations between persons, objects, and locations rather than transient visual cues [85,89]. The most relevant evaluation dimensions are Relational Coherence and Semantic Fidelity.

Stream-reasoning constraints. Window semantics are essential in this setting because object tracks, ownership relations, and scene context must be interpreted over short temporal intervals. Delayed camera detections, missing frames, or out-of-order trajectory updates may change the explanatory path. Therefore, latency-sensitive alerts require fast intrinsic or rule-based explanations, whereas more expensive post-hoc explanations can be reserved for escalated or forensic analysis. This illustrates how stream-aware explanation design must balance immediate alerting with later, richer semantic reconstruction.

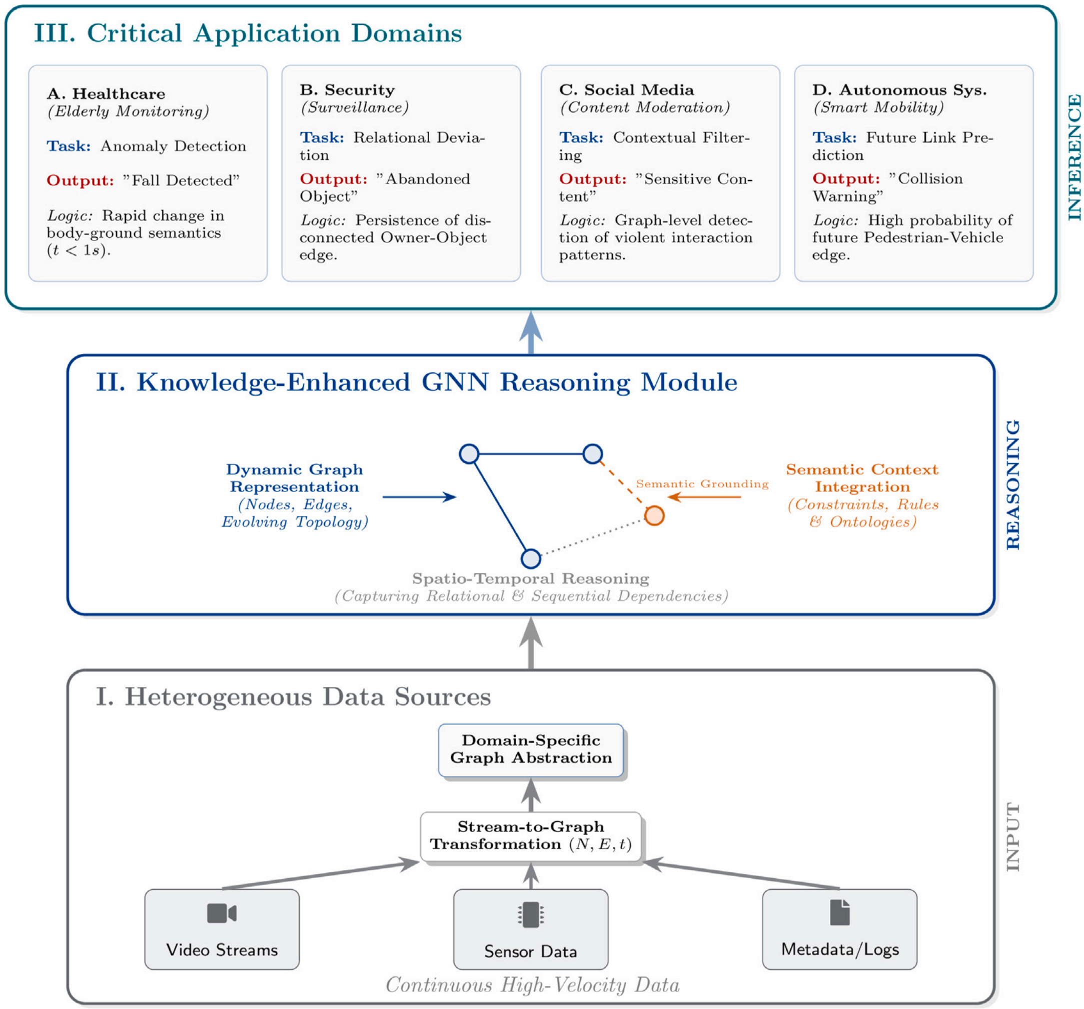

Fig. 6. Conceptual instantiation of the unified framework for explainable semantic stream reasoning (Section 1.6). The three-phase abstraction integrates (I) **Stream-to-Graph transformation** $\mathcal{G} = (N, E, t)$ capturing semantically grounded entities (N), rule-governed relations (E), and temporal validity (t); (II) **Knowledge-enhanced reasoning** with Semantic-Aware XAI; and (III) **Domain-specific inference**. This synthesis addresses the research gaps (Section 1.3) and illustrates applicability across high-stakes domains.

7.2. Stream-aware social media and autonomous-system instantiations

7.2.1. Social media moderation

A representative social media instantiation could use a heterogeneous GAT, R-GCN, or temporal interaction GNN over posts, replies, mentions, hashtags, and user interaction streams. The KG may include users, posts, topics, groups, and moderation categories, with relations such as *replies_to*, *mentions*, *targets*, and *coordinated_with*. For content moderation, user interactions and conversation context can be modeled as multi-relational graphs enriched with domain ontologies [55]. This allows harmful-content predictions to be explained through semantically coherent interaction patterns rather than isolated posts [56,58,64]. For example, an explanation may connect repeated targeting, coordinated replies, and abusive-topic relations to support a moderation decision. The most relevant evaluation dimensions are Semantic Fidelity and Human/Domain-Centered Alignment.

Stream-reasoning constraints. In social media streams, high throughput and delayed interactions make resource-aware explanation generation particularly important. Explanations may need to be maintained over sliding windows of posts, replies, mentions, and hashtags, while out-of-order replies or delayed moderation signals can alter the interpretation of a conversation thread. Deeper semantic or counterfactual explanations are therefore better triggered selectively for high-impact, repeated, low-confidence, or policy-sensitive decisions. This stream-aware interpretation goes beyond standard temporal interaction modeling by emphasizing when explanations should be updated and when more expensive explanation procedures should be scheduled.

7.2.2. Autonomous systems

A representative autonomous-systems instantiation could use a spatio-temporal GNN or a TGN-style event-based GNN over vehicle, pedestrian, road-sign, lane, and sensor streams. The KG may represent vehicles, pedestrians, lanes, traffic lights, road segments, and traffic rules, with relations such as *in_lane*, *approaches*, *has_right_of_way*, and *must_yield_to*. In autonomous and smart mobility settings, trajectories and dynamic traffic states can be represented as temporal KGs [13,14]. A braking decision or collision warning may be explained by a temporally valid path linking pedestrian crossing, lane position, traffic signal state, and right-of-way constraints, following the design principles for stream-to-graph conversion in STKGNN [69]. Such decisions can also be analyzed using causality-inspired explainers for dynamic GNNs [15,81,83]. The most relevant evaluation dimensions are Temporal Semantic Stability, Relational Coherence, and Human/Domain-Centered Alignment.

Stream-reasoning constraints. In autonomous systems, explanation generation is constrained by strict latency budgets and heterogeneous sensor streams. Windowed reasoning, delayed sensor updates, missing observations, and out-of-order events must be handled without interrupting safety-critical decisions. Lightweight online explanations are therefore more suitable during operation, while richer semantic, counterfactual, or causal explanations may be generated for post-event audit and incident analysis. This exemplifies the trade-off between explanation richness and latency, since semantically grounded relational explanations must remain compatible with real-time operational constraints.

Across these examples, the proposed framework differs from standard temporal GNN benchmark settings by treating explanation generation as a stream-aware process. The relevant design question is not only which historical graph elements influenced a prediction, but also how explanations can be updated, validated, and scheduled under semantic constraints, temporal windows, late events, and limited latency budgets.

8. Discussion, implications, and research directions

8.1. Current limitations and open questions

The findings of this study reveal both the emerging potential and the persistent gaps of explainable, knowledge-aware stream reasoning with GNNs. As detailed in the challenge analysis in Section 4.5 and summarized in Table 4, existing work at the intersection of GNNs, KGs, and stream reasoning still exhibits distinct structural limitations. Most explainability approaches primarily emphasize structural relevance and statistical attribution, while providing limited guarantees regarding semantic consistency with external knowledge representations. This limitation becomes particularly pronounced in dynamic settings, where evolving graph structures may cause explanations to drift away from domain semantics, reducing their interpretability in streaming environments.

Moreover, the research on temporal GNNs and on semantic enrichment by KGs has largely progressed in parallel rather than as part of an integrated framework. Although temporal GNNs capture evolving relations and KGs provide rich semantic grounding, their joint role in shaping explainability has rarely been examined systematically. Consequently, it remains unclear how to integrate semantics to ensure temporal coherence and capture rare yet significant patterns.

Evaluation practice constitutes another critical limitation. As discussed in Section 5.1, existing assessments predominantly focus on conventional criteria such as faithfulness, sparsity, and robustness, while paying relatively little attention to semantic fidelity or relational coherence. Furthermore, scalability, the interpretability of high-dimensional semantic embeddings, and the transferability of semantic reasoning across heterogeneous ontologies remain unresolved challenges that call for deeper methodological advances.

8.2. Implications for theory, methods, and practice

The semantically grounded perspective proposed in this study carries several implications for theory, methodology, and practice. At the theoretical level, the knowledge-aware evaluation dimensions introduced in Section 5.3—semantic fidelity, relational coherence, temporal semantic stability, and domain alignment—extend conventional understandings of explanation quality. They suggest that semantic validity should be treated as a primary criterion rather than a secondary consideration. This motivates more formal analysis of how GNN message passing interacts with logical constraints and temporal semantics in KGs.

At the methodological level, the trade-offs discussed in Section 6 indicate that explanation design is closely linked to model architecture. As conceptualized in Fig. 5 compares explainability methods defined in Section 3.1, impose different costs and benefits in knowledge-enriched streams. Their effectiveness depends on where semantic context is introduced within the GNN pipeline. Future approaches should therefore make semantic entities, relation types, and temporal structure explicitly visible in explanations instead of relying solely on uninterpretable latent representations. In this respect, grounding explanation mechanisms in KG semantics has the potential to support the interpretation of subtle, context-dependent patterns in continuous data streams.

The real-world scenarios in Section 7 (Healthcare, Security, Social Media, and Autonomous Systems) demonstrate that explainable semantic stream reasoning is relevant to real-world deployment rather than being a purely theoretical goal. In these domains, explanations must remain consistent with domain semantics and knowledge constraints while preserving interpretability over time. Consequently, evaluation strategies should explicitly consider the semantic consistency and temporal stability metrics proposed in this survey alongside traditional performance and latency measures. As illustrated in the architectural framework in Fig. 6, these domains benefit significantly from explanations that remain grounded in domain knowledge even as data streams and background semantics evolve.

8.3. Future directions for knowledge-aware stream reasoning

The analysis in this survey points to several promising directions for advancing explainable and knowledge-aware stream reasoning with GNNs. A first direction involves the development of semantically constrained learning objectives in which ontological rules, type constraints, and temporal relations are incorporated directly into training. Such approaches may help align predictions and explanations with domain semantics, although they also raise questions regarding efficiency, scalability, and the appropriate balance between constraint enforcement and model flexibility.

A second direction concerns evaluation methodology. Streaming environments require explanation quality to be treated as a temporally evolving property. Future research should therefore develop metrics and protocols that measure how explanations change under evolving structure, features, and background knowledge. Notions such as explanation drift and semantic path stability appear particularly relevant and should be translated into practical, testable evaluation procedures.

A third direction relates to benchmarking. Dedicated benchmark suites that combine temporal graph structure, explicit KG semantics, and explanation objectives would enable joint assessment of predictive capability and semantic explanation quality. Studies centered on human and domain expertise are also needed to verify whether knowledge-aware metrics align with expert judgment and genuinely support decision-making. Although promising, these directions introduce challenges concerning dataset design, annotation effort, and cross-domain generalization.

In summary, advancing explainable semantic stream reasoning necessitates the synergistic evolution of training, evaluation, and benchmarking, underpinned by a pervasive focus on semantic grounding.

9. Conclusions

This paper has presented a structured, conceptually driven survey that develops a semantically grounded perspective on explainable GNN-based stream reasoning. The analysis reveals that, despite notable progress in GNN architectures and temporal modeling, current research lacks a unified view that treats explainability, semantic knowledge, and streaming dynamics as interconnected elements. This fragmentation limits the ability of existing methods to provide explanations that withstand the evolving nature of graph environments, offering only partial support for domain experts who require actionable insights from continuous data streams.

By synthesizing developments across explainability techniques and integration strategies, the study provides a coherent conceptual lens for bridging these gaps. The proposed knowledge-aware evaluation perspective establishes that semantic fidelity, relational coherence, and temporal stability must be treated as core properties of explanation quality rather than secondary considerations. This shift in perspective is critical for ensuring that machine-generated explanations align with the logical and causal structures inherent in expert knowledge.

The application scenarios discussed in this work confirm that these requirements are not merely theoretical but operationally essential in domains where decisions carry significant safety implications. Ultimately, future work will depend on the deeper integration of semantic constraints with temporal learning dynamics. Achieving this synergy is a prerequisite for developing systems that are not only predictive but also genuinely transparent and trustworthy in real-world streaming environments.

CRediT authorship contribution statement

Gözde Ayşe Tataroğlu Özbulak: Writing – review & editing, Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Yash Raj Shrestha:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Jean-Paul Calbimonte:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Swiss National Science Foundation (SNSF) through the StreamKG project (Grant No. 213369).

Data availability

No data was used for the research described in the article.

References

[1] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, S.Y. Philip, A comprehensive survey on graph neural networks, IEEE Trans. Neural Netw. Learn. Syst. 32 (1) (2020) 4–24.
[2] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, M. Sun, Graph neural networks: A review of methods and applications, AI Open 1 (2020) 57–81, http://dx.doi.org/10.1016/j.aiopen.2021.01.001.
[3] S. Wu, F. Sun, W. Zhang, X. Xie, B. Cui, Graph neural networks in recommender systems: a survey, ACM Comput. Surv. 55 (5) (2022) 1–37.
[4] Z. Ye, Y.J. Kumar, G.O. Sing, F. Song, J. Wang, A comprehensive survey of graph neural networks for knowledge graphs, IEEE Access 10 (2022) 75729–75741.
[5] X.M. Zhang, L. Liang, L. Liu, M.J. Tang, Graph neural networks and their current applications in bioinformatics, Front. Genet. 12 (2021) 690049.
[6] H.S. Al Khatib, S. Neupane, H. Kumar Manchukonda, N.A. Golilarz, S. Mittal, A. Amirlatifi, S. Rahimi, Patient-centric knowledge graphs: a survey of current methods, challenges, and applications, Front. Artif. Intell. 7 (2024) 1388479.
[7] H. Zhang, B. Wu, X. Yuan, S. Pan, H. Tong, J. Pei, Trustworthy graph neural networks: Aspects, methods, and trends, Proc. IEEE 112 (2) (2024) 97–139.
[8] L. Waikhom, R. Patgiri, A survey of graph neural networks in various learning paradigms: methods, applications, and challenges, Artif. Intell. Rev. 56 (7) (2023) 6295–6364.
[9] K. Liang, L. Meng, M. Liu, Y. Liu, W. Tu, S. Wang, S. Zhou, X. Liu, F. Sun, K. He, A survey of knowledge graph reasoning on graph types: Static, dynamic, and multi-modal, IEEE Trans. Pattern Anal. Mach. Intell. 46 (12) (2024) 9456–9478.
[10] Z. Wang, S. Ma, K. Wang, Z. Zhuang, Rule-guided graph neural networks for explainable knowledge graph reasoning, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 39, (12) 2025, pp. 12784–12791.
[11] R.K. Tripathy, Z. Frohock, H. Wang, G.A. Cary, S. Keegan, G.W. Carter, Y. Li, Effective integration of multi-omics with prior knowledge to identify biomarkers via explainable graph neural networks, Npj Syst. Biol. Appl. 11 (1) (2025) 43.
[12] G.A. Tataroğlu Özbulak, G. Manzo, Y.R. Shrestha, J.-P. Calbimonte, A comprehensive survey of stream reasoning and its integration with knowledge graphs: GA Tataroğlu Özbulak et al., Knowl. Inf. Syst. (2025) 1–33.
[13] J. Chen, R. Ying, Tempme: Towards the explainability of temporal graph neural networks via motif discovery, Adv. Neural Inf. Process. Syst. 36 (2023) 29005–29028.
[14] S. Seo, S. Kim, J. Jung, Y. Lee, C. Park, Self-explainable temporal graph networks based on graph information bottleneck, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, pp. 2572–2583.
[15] Z. Qu, D. Gomm, M. Färber, Cody: Counterfactual explainers for dynamic graphs, in: Forty-Second International Conference on Machine Learning.
[16] Q. Li, G. Wu, Explainable reasoning over temporal knowledge graphs by pre-trained language model, Inf. Process. Manage. 62 (1) (2025) 103903.
[17] A.R. Mohammadi, Explainable graph neural network recommenders; challenges and opportunities, in: Proceedings of the 17th ACM Conference on Recommender Systems, 2023, pp. 1318–1324.
[18] Z. Li, X. Jin, W. Li, S. Guan, J. Guo, H. Shen, Y. Wang, X. Cheng, Temporal knowledge graph reasoning based on evolutional representation learning, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 408–417.
[19] S. Gao, Y. Li, Y. Shen, Y. Shao, L. Chen, Etc: Efficient training of temporal graph neural networks over large-scale dynamic graphs, Proc. VLDB Endow. 17 (5) (2024) 1060–1072.
[20] S.M. Kazemi, R. Goel, Representation learning for dynamic graphs: A survey, J. Mach. Learn. Res. 21 (70) (2020) 1–73, URL http://jmlr.org/papers/v21/19447.html.
[21] S. Liao, S. Liang, Z. Meng, Q. Zhang, Learning dynamic embeddings for temporal knowledge graphs, in: Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021, pp. 535–543.
[22] C. Agarwal, O. Queen, H. Lakkaraju, M. Zitnik, Evaluating explainability for graph neural networks, Sci. Data 10 (1) (2023) 144.
[23] T. Funke, M. Khosla, M. Rathee, A. Anand, Zorro: Valid, sparse, and stable explanations in graph neural networks, IEEE Trans. Knowl. Data Eng. 35 (8) (2022) 8687–8698.
[24] H. Yuan, J. Tang, X. Hu, S. Ji, XGNN: Towards model-level explanations of graph neural networks, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 430–438.
[25] X. Wang, Y. Liu, X. Chen, et al., Knowledge-enhanced graph neural networks: A survey, IEEE Trans. Knowl. Data Eng. (2023) http://dx.doi.org/10.1109/TKDE.2023.3234567.
[26] T. Ma, L. Huang, Q. Lu, S. Hu, Kr-gcn: Knowledge-aware reasoning with graph convolution network for explainable recommendation, ACM Trans. Inf. Syst. 41 (1) (2023) 1–27.
[27] B. Kitchenham, et al., Procedures for performing systematic reviews, Keele UK Keele Univ. 33 (2004) (2004) 1–26.
[28] D. Xu, C. Ruan, E. Korpeoglu, S. Kumar, K. Achan, Inductive representation learning on temporal graphs, in: International Conference on Learning Representations (ICLR), 2020, URL https://openreview.net/forum?id=rJeW1yHYwH.
[29] A. Adadi, M. Berrada, Peeking inside the black-box: a survey on explainable artificial intelligence (XAI), IEEE Access 6 (2018) 52138–52160, http://dx.doi.org/10.1109/ACCESS.2018.2870052.
[30] W. Samek, K.R. Müller, Towards explainable artificial intelligence, in: Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Springer, Cham, 2019, pp. 5–22.
[31] P. Velickovic, W. Fedus, W.L. Hamilton, P. Liò, Y. Bengio, R.D. Hjelm, Deep graph infomax, 2, (3) 2019, p. 4s, ICLR (poster).
[32] H.S. Sheu, S. Li, Context-aware graph embedding for session-based news recommendation, in: Proceedings of the 14th ACM Conference on Recommender Systems, 2020, pp. 657–662.
[33] J. Han, W. Huang, Y. Rong, T. Xu, F. Sun, J. Huang, Structure-aware dropedge toward deep graph convolutional networks, IEEE Trans. Neural Netw. Learn. Syst. (2023).
[34] Z. Ying, D. Bourgeois, J. You, M. Zitnik, J. Leskovec, Gnnexplainer: Generating explanations for graph neural networks, Adv. Neural Inf. Process. Syst. 32 (2019).
[35] H. Yuan, et al., Explainability in graph neural networks: A taxonomic survey, IEEE Trans. Pattern Anal. Mach. Intell. 45 (2) (2022) 163–185.
[36] S. Ji, S. Pan, E. Cambria, P. Marttinen, P.S. Yu, A survey on knowledge graphs: Representation, acquisition, and applications, IEEE Trans. Neural Netw. Learn. Syst. 33 (2) (2022) 494–514, http://dx.doi.org/10.1109/TNNLS.2021.3070843.
[37] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G.D. Melo, C. Gutierrez, A. Zimmermann, Knowledge graphs, ACM Comput. Surv. 54 (4) (2021) 1–37.
[38] Z.C. Lipton, The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery, Queue 16 (3) (2018) 31–57.
[39] W.J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, B. Yu, Definitions, methods, and applications in interpretable machine learning, Proc. Natl. Acad. Sci. 116 (44) (2019) 22071–22080.

[40] C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, Nat. Mach. Intell. 1 (5) (2019) 206–215, http://dx.doi.org/10.1038/s42256-019-0048-x.
[41] B. Mittelstadt, C. Russell, S. Wachter, Explaining explanations in AI, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, 2019, pp. 279–288.
[42] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM Comput. Surv. 51 (5) (2018) 1–42.
[43] M.I. Hussain, A. Munir, M. Mamun, S.H. Chowdhury, N. Uddin, M.M. Hossain, A transparent house price prediction framework using ensemble learning, genetic algorithm-based tuning, and ANOVA-based feature analysis, FinTech 4 (3) (2025) 33.
[44] S.H. Chowdhury, M. Mamun, T.A. Shaikat, M.I. Hussain, S. Iqbal, M.M. Hossain, An ensemble approach for artificial neural network-based liver disease identification from optimal features through hybrid modeling integrated with advanced explainable AI, Medinformatics 2 (2) (2025) 107–119.
[45] S.H. Chowdhury, Hrnet-xgb: A comprehensive hybrid framework for automated and interpretable skin cancer detection with multi-scale feature fusion, 2025, SSRN.
[46] C. Agarwal, S. Krishna, E. Saxena, M. Pawelczyk, N. Johnson, I. Puri, H. Lakkaraju, Openxai: Towards a transparent evaluation of model explanations, Adv. Neural Inf. Process. Syst. 35 (2022) 15784–15799.
[47] P.E. Pope, S. Kolouri, M. Rostami, C.E. Martin, H. Hoffmann, Explainability methods for graph convolutional neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 10772–10781.
[48] M. Mamun, M.S. Ali, M.S.A. Chowdhury, S.H. Chowdhury, M.I. Hussain, M.M. Hossain, A differential privacy and TOPSIS enhanced explainable machine learning framework for diabetes risk diagnosis, in: 2025 International Conference on Electrical, Computer and Communication Engineering, ECCE, IEEE, 2025, pp. 1–6.
[49] M. Mamun, S.H. Chowdhury, M.O. Faruq, M.M. Azad, B.R. Biswas, M.I. Hussain, M.M. Hossain, Identification of maternal health risk from optimal features using explainable machine learning, Eng. Rep. 7 (11) (2025) e70491.
[50] M.T.A. Shaikat, S.H. Chowdhury, M. Shovon, M.M. Hossain, M.I. Hussain, M. Mamun, Explainability elevated obstructive pulmonary disease care: severity classification, quality of life prediction, and treatment impact assessment, in: 2025 International Conference on Electrical, Computer and Communication Engineering, ECCE, IEEE, 2025, pp. 1–6.
[51] D. Tena Cucala, B. Cuenca Grau, E.V. Kostylev, B. Motik, Explainable GNN-based models over knowledge graphs, 2022.
[52] Z. Lyu, Y. Wu, J. Lai, M. Yang, C. Li, W. Zhou, Knowledge enhanced graph neural networks for explainable recommendation, IEEE Trans. Knowl. Data Eng. 35 (5) (2022) 4954–4968.
[53] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey of approaches and applications, IEEE Trans. Knowl. Data Eng. 29 (12) (2017) 2724–2743.
[54] Z. Zhang, P. Cui, W. Zhu, Deep learning on graphs: A survey, IEEE Trans. Knowl. Data Eng. 34 (1) (2020) 249–270.
[55] I. Tiddi, S. Schlobach, Knowledge graphs as tools for explainable machine learning: A survey, Artificial Intelligence 302 (2022) 103627.
[56] E. Rajabi, K. Etminani, Knowledge-graph-based explainable AI: A systematic review, J. Inf. Sci. 50 (4) (2024) 1019–1029.
[57] S. Schramm, C. Wehner, U. Schmid, Comprehensible artificial intelligence on knowledge graphs: A survey, J. Web Semant. 79 (2023) 100806.
[58] H. Yuan, H. Yu, S. Gui, S. Ji, Explainability in graph neural networks: A taxonomic survey, IEEE Trans. Pattern Anal. Mach. Intell. 45 (5) (2022) 5782–5799.
[59] N. Liu, Q. Feng, X. Hu, Interpretability in graph neural networks, in: Graph Neural Networks: Foundations, Frontiers, and Applications, Springer, 2022, pp. 121–147.
[60] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, et al., Graph attention networks, Stat 1050 (20) 10–48550.
[61] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, P.S. Yu, Heterogeneous graph attention network, in: Proceedings of the World Wide Web Conference, 2019, pp. 2022–2032.
[62] Z. Li, Y. Zhao, Y. Zhang, Z. Zhang, Multi-relational graph attention networks for knowledge graph completion, Knowl.-Based Syst. 251 (2022) 109262.
[63] S. Brody, U. Alon, E. Yahav, How attentive are graph attention networks? in: 10th International Conference on Learning Representations, ICLR 2022, 2022.
[64] X. Han, X. Dou, User recommendation method integrating hierarchical graph attention network with multimodal knowledge graph, Front. Neurorobotics 19 (2025) 1587973.
[65] X. Gao, W. Zhang, J. Yu, Y. Shao, Q.V.H. Nguyen, B. Cui, H. Yin, Accelerating scalable graph neural network inference with node-adaptive propagation, in: 2024 IEEE 40th International Conference on Data Engineering, ICDE, IEEE, 2024, pp. 3042–3055.
[66] E. Panagiotaki, D. De Martini, L. Kunze, Semantic interpretation and validation of graph attention-based explanations for gnn models, in: 2023 21st International Conference on Advanced Robotics, ICAR, IEEE, 2023, pp. 375–380.
[67] L. Werner, N. Layaïda, P. Genevès, S. Chlyah, Knowledge enhanced graph neural networks, in: 2023 IEEE 10th International Conference on Data Science and Advanced Analytics, DSAA, IEEE, 2023, pp. 1–10.
[68] Z. Ye, et al., A comprehensive survey of graph neural networks for knowledge graphs, IEEE Access 10 (2022) 75729–75741.
[69] G.A. Tataroğlu özbulak, Y.R. Shrestha, J.-P. Calbimonte, STKGNN: Scalable spatio-temporal knowledge graph reasoning for activity recognition, in: Proceedings of the 34th ACM International Conference on Information and Knowledge Management, 2025, pp. 2853–2862.
[70] X. Wang, X. He, Y. Cao, M. Liu, T.-S. Chua, Kgat: Knowledge graph attention network for recommendation, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 950–958.
[71] D. Luo, W. Cheng, D. Xu, W. Yu, B. Zong, H. Chen, X. Zhang, Parameterized explainer for graph neural network, Adv. Neural Inf. Process. Syst. 33 (2020) 19620–19631.
[72] H. Yuan, H. Yu, J. Wang, K. Li, S. Ji, On explainability of graph neural networks via subgraph explorations, in: International Conference on Machine Learning, PMLR, 2021, pp. 12241–12252.
[73] C. Shan, Y. Shen, Y. Zhang, X. Li, D. Li, Reinforcement learning enhanced explainer for graph neural networks, Adv. Neural Inf. Process. Syst. 34 (2021) 22523–22533.
[74] W. Sloneker, S. Patel, H.-J. Wang, L. Crawford, R. Singh, BetaExplainer: A probabilistic method to explain graph neural networks, J. Stat. Theory Appl. (2025) 1–20.
[75] L. Wu, D. Hao, Z. Fan, Explainable graph neural networks via structural externalities, in: Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, 2025, pp. 583–591.
[76] B. Sanchez-Lengeling, J. Wei, B. Lee, E. Reif, P. Wang, W. Qian, A. Wiltschko, Evaluating attribution for graph neural networks, Adv. Neural Inf. Process. Syst. 33 (2020) 5898–5910.
[77] Q. Huang, M. Yamada, Y. Tian, D. Singh, Y. Chang, Graphlime: Local interpretable model explanations for graph neural networks, IEEE Trans. Knowl. Data Eng. 35 (7) (2022) 6968–6972.
[78] A. Duval, F.D. Malliaros, GraphSVX: Shapley value explanations for graph neural networks, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2021, pp. 302–318.
[79] R. Yu, Y. Li, T. Shi, W. Xie, Explaining spatio-temporal graph convolutional networks with spatio-temporal constraints perturbation for action recognition, Inf. Fusion (2025) 103387.
[80] A. Lucic, M.A. Ter Hoeve, G. Tolomei, M. De Rijke, F. Silvestri, Cf-gnnexplainer: Counterfactual explanations for graph neural networks, in: International Conference on Artificial Intelligence and Statistics, PMLR, 2022, pp. 4499–4511.
[81] K. Zhao, L. Zhang, Causality-inspired spatial-temporal explanations for dynamic graph neural networks, in: The Twelfth International Conference on Learning Representations, 2024.
[82] W. Lin, H. Lan, B. Li, Generative causal explanations for graph neural networks, in: International Conference on Machine Learning, PMLR, 2021, pp. 6666–6679.
[83] T. Wang, D. Luo, W. Cheng, H. Chen, X. Zhang, Dyexplainer: Self-explainable dynamic graph neural network with sparse attentions, ACM Trans. Knowl. Discov. Data 19 (4) (2025) 1–21.
[84] W. Xia, M. Lai, C. Shan, Y. Zhang, X. Dai, X. Li, D. Li, Explaining temporal graph models through an explorer-navigator framework, in: The Eleventh International Conference on Learning Representations, 2022.
[85] H. Chang, J. Ye, A. Lopez-Avila, J. Du, J. Li, Path-based explanation for knowledge graph completion, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, pp. 231–242.
[86] M. Schlichtkrull, T.N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: The Semantic Web: 15th International Conference, ESWC 2018, Proceedings, in: Lecture Notes in Computer Science, vol. 10843, Springer, 2018, pp. 593–607.
[87] E. Choi, M.T. Bahadori, E. Searles, C. Coffey, M. Thompson, J. Bost, J. Sun, Multi-layer representation learning for medical concepts, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1495–1504.
[88] X. Han, S. Cao, X. Lv, Y. Lin, Z. Liu, M. Sun, J. Li, OpenKE: An open toolkit for knowledge embedding, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2018, pp. 139–144.
[89] I. Khalid, S. Schockaert, Systematic relational reasoning with epistemic graph neural networks, in: The Thirteenth International Conference on Learning Representations.
[90] Z. Lyu, Y. Wu, J. Lai, M. Yang, C. Li, W. Zhou, Knowledge enhanced graph neural networks for explainable recommendation, IEEE Trans. Knowl. Data Eng. 35 (5) (2022) 4954–4968.

[91] H. Xuanyuan, P. Barbiero, D. Georgiev, L.C. Magister, P. Liò, Global concept-based interpretability for graph neural networks via neuron analysis, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, (9) 2023, pp. 10675–10683.
[92] B. Jiang, Z. Zhang, D. Lin, J. Tang, B. Luo, Semi-supervised learning with graph learning-convolutional networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 11313–11320.
[93] L. Zhao, X. Peng, Y. Tian, M. Kapadia, D.N. Metaxas, Semantic graph convolutional networks for 3d human pose regression, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3425–3435.
[94] W. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, Adv. Neural Inf. Process. Syst. 30 (2017).
[95] W. Hu, B. Liu, J. Gomes, M. Zitnik, P. Liang, V. Pande, J. Leskovec, Strategies for pre-training graph neural networks, in: International Conference on Learning Representations, ICLR, 2020.
[96] L.H. Gilpin, D. Bau, B.Z. Yuan, A. Bajwa, M. Specter, L. Kagal, Explaining explanations: An overview of interpretability of machine learning, in: 2018 IEEE 5th International Conference on Data Science and Advanced Analytics, DSAA, IEEE, 2018, pp. 80–89.
[97] S. Kumar, X. Zhang, J. Leskovec, Predicting dynamic embedding trajectory in temporal interaction networks, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 1269–1278.
[98] I. Cantador, P. Brusilovsky, T. Kuflik, Second workshop on information heterogeneity and fusion in recommender systems (HetRec2011), in: Proceedings of the Fifth ACM Conference on Recommender Systems, 2011, pp. 387–388.
[99] A. García-Durán, S. Dumančić, M. Niepert, Learning sequence encoders for temporal knowledge graph completion, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 4816–4821.
[100] S.S. Dasgupta, S.N. Ray, P. Talukdar, Hyte: Hyperplane-based temporally aware knowledge graph embedding, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 2001–2011.
[101] A. Bennett, H. Ulrich, J. Wiedekopf, P. Szul, J. Grimes, A. Johnson, MIMIC-IV Clinical Database Demo on FHIR, PhysioNet (2025) Version 2.1.0.
[102] J. He, A. Rafiey, G. Mishne, Y. Wang, Explaining gnn explanations with edge gradients, in: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2, 2025, pp. 884–895.
[103] J. Chen, L. Song, M. Wainwright, M. Jordan, Learning to explain: An information-theoretic perspective on model interpretation, in: Proceedings of the International Conference on Machine Learning, ICML, PMLR, 2018, pp. 883–892.
[104] A.M. Bennett, H. Ulrich, P. Van Damme, J. Wiedekopf, A.E. Johnson, MIMIC-IV on FHIR: converting a decade of in-patient data into an exchangeable, interoperable format, J. Am. Med. Inform. Assoc. 30 (4) (2023) 718–725.
[105] A.L. Goldberger, L.A.N. Amaral, L. Glass, J.M. Hausdorff, P.C. Ivanov, R.G. Mark, J.E. Mietus, G.B. Moody, C.-K. Peng, H.E. Stanley, PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals, Circulation 101 (23) (2000) e215–e220, RRID:SCR_007345.
[106] W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, J. Leskovec, Open graph benchmark: Datasets for machine learning on graphs, Adv. Neural Inf. Process. Syst. 33 (2020) 22118–22133.