

Relational Anomaly Detection in Knowledge Graphs: A Review

Dev P Chaudhary, Suhana Begum, Prabhaker Tiwari

Assistant Professor, Associate Professor

Department of Business Intelligence and Knowledge Mining

Kaveri Regional College, Nagpur, India

Email: *Devp057chaudary@yahoo.com, suhanabegurm_88@gmail.com,
prabhakert1iwari@rediffmail.com*

Abstract

Knowledge graphs (KGs) represent entities and relations in structured form, enabling intelligent reasoning and semantic search. However, large-scale KGs often contain anomalous or inconsistent relational patterns caused by noise, incompleteness, or malicious insertion. Relational anomaly detection aims to identify such irregular structures considering both entities and relationships. This paper reviews the foundations, methods, datasets, and challenges of relational anomaly detection in knowledge graphs. We categorize approaches into statistical, embedding-based, graph neural network, rule-based, and hybrid techniques. Comparative analysis highlights strengths and limitations across scalability, interpretability, and detection accuracy. Applications in fraud detection, cybersecurity, biomedical discovery, and data quality management are also discussed. Finally, open challenges such as evolving graphs, explainability, and privacy-aware anomaly detection are outlined. The review provides a consolidated understanding for researchers and practitioners working in graph mining and knowledge engineering.

Keywords: *Knowledge graph, relational anomaly detection, graph mining, link prediction, graph neural networks, outlier detection, semantic data quality*

INTRODUCTION

Knowledge graphs (KGs) have become a core representation for structured knowledge in domains such as semantic web, recommender systems, and biomedical informatics. A knowledge graph typically consists of triples (head entity, relation, tail entity), forming a multi-relational graph. Examples include large public knowledge bases like Wikidata and domain-specific graphs such as protein interaction networks.

Despite their usefulness, knowledge graphs are often noisy and incomplete. Errors may arise from automated extraction, conflicting sources, or adversarial manipulation. These errors appear as anomalous relations or abnormal substructures. For instance, a triple stating that a person is born in two distant centuries or a medical entity having incompatible relations with drugs represents relational anomalies.

Traditional anomaly detection methods focus on feature vectors or simple graphs. However, relational anomaly detection in knowledge graphs is more complex because anomalies depend not only on node properties but also on multi-hop relational patterns and semantics. Detecting anomalies in such structured data requires specialized models capable of capturing relational dependencies.

This paper provides a comprehensive review of relational anomaly detection techniques in knowledge graphs. It discusses modeling principles, detection frameworks, evaluation metrics, and research trends. The aim is to synthesize current knowledge and identify future research directions.

BACKGROUND AND PROBLEM DEFINITION

Relational anomaly detection in knowledge graphs builds on the representation of structured knowledge and the statistical notion of abnormal relational patterns. Before discussing detection methods, it is necessary to understand how knowledge graphs are structured and how anomalies manifest within relational data.

1. Knowledge Graph Structure

A knowledge graph (KG) is a structured representation of facts where entities are connected through typed relationships. Formally, a KG can be modeled as a directed labeled graph:

$$G = (E, R, T)$$

where:

- **E** represents the set of entities (nodes)
- **R** represents the set of relation types (edge labels)
- **T** $\subseteq E \times R \times E$ represents the set of triples

Each triple (h, r, t) connects a head entity **h** to a tail entity **t** via relation **r**.

This triple-based representation provides semantic meaning to graph edges. Unlike simple graphs, edges are not uniform; each relation has domain and range constraints and semantic interpretation.

Example of Knowledge Graph Triples

- (PersonA, bornIn, CityX)
- (CityX, locatedIn, CountryY)
- (DrugY, treats, DiseaseZ)
- (ProteinP, interactsWith, ProteinQ)

These triples collectively form a multi-relational network.

Heterogeneous Nature

Knowledge graphs are heterogeneous because:

- Nodes belong to different types (person, city, drug, gene, etc.)
- Edges belong to different relation categories
- Semantic constraints exist between types

For instance:

bornIn: Person $\rightarrow$ Location

Treats: Drug $\rightarrow$ Disease

This typed structure distinguishes KGs from homogeneous networks like social graphs.

Multi-Hop Relational Semantics

Knowledge graphs also encode implicit relations via paths.

Example:

(PersonA, bornIn, CityX)

(CityX, locatedIn, CountryY)

From these, we infer:

(PersonA, citizenOf, CountryY)

Such relational paths define semantic consistency. An anomaly may appear when a triple contradicts these inferred relations.

Schema and Ontology Constraints

Many KGs include ontology schemas specifying:

- Entity classes
- Relation domains and ranges
- Hierarchical relations

Example constraint: locatedIn: City $\rightarrow$ Country

If a triple violates schema, it may be anomalous.

TYPES OF RELATIONAL ANOMALIES

Relational anomalies arise when triples or subgraphs violate expected relational patterns, semantic rules, or structural regularities. Unlike attribute anomalies (which depend on numeric feature deviation), relational anomalies depend on graph structure and semantics.

Below are major categories of relational anomalies in knowledge graphs.

1. Relation Inconsistency

Relation inconsistency occurs when an entity participates in mutually incompatible relations or violates domain/range constraints.

Examples:

- (PersonA, bornIn, 2050) $\rightarrow$ temporal inconsistency
- (DrugX, treats, MountainY) $\rightarrow$ type mismatch
- (CityX, locatedIn, PlanetMars) $\rightarrow$ impossible relation

This type often arises from extraction errors or ontology violations.

2. Structural Outliers

Structural outliers are entities whose connectivity patterns differ significantly from typical nodes of the same type.

Examples:

- A person node connected to thousands of unrelated domains
- A gene linked to unrelated disease clusters
- A city connected to many continents

These nodes appear abnormal when comparing graph topology metrics such as:

- Degree distribution
- Relation diversity
- Neighborhood similarity

Structural anomalies often indicate noise or malicious insertion.

3. Link Anomalies

Link anomalies are individual edges that are unexpected given graph context.

Example:

If most universities are located in cities, then:

(UniversityA, locatedIn, OceanB)

is a link anomaly.

Link anomalies are often detected through link prediction probability. If predicted likelihood is very low, edge is anomalous.

Types of link anomalies:

- Cross-type mismatch
- Rare relation between entities
- Contradictory relation

4. Subgraph Anomalies

Subgraph anomalies involve abnormal relational motifs or patterns across multiple nodes.

Examples:

- Circular hierarchy:

A subclassOf B

B subclassOf C

C subclassOf A

- Impossible biological pathway
- Fraud ring structure in financial graph

Subgraph anomalies capture higher-order relational inconsistencies that cannot be detected from single triples.

5. Temporal Anomalies

Temporal anomalies occur when relations change in inconsistent or impossible ways over time.

Examples:

- PersonA bornIn CityX at 1980
- PersonA bornIn CityY at 1985
- CEO of company changing daily
- Historical entity linked to modern technology

Temporal anomalies require time-aware knowledge graphs where triples include timestamps.

COMPARISON: ATTRIBUTE VS RELATIONAL ANOMALIES

Table 1.

Aspect	Attribute Anomaly	Relational Anomaly
Data type	Numeric/feature	Graph relations
Context	Independent	Structural
Example	Age = 300	bornIn future
Detection	Statistical	Graph-based
Dependency	Local	Multi-hop

Relational anomalies are more complex because context spans multiple entities and relations.

3. Formal Definition of Relational Anomaly Detection

Let a knowledge graph be $G = (E, R, T)$ with triple $t = (h, r, t)$.

Relational anomaly detection aims to compute an anomaly score:

$$A(t): T \rightarrow \mathbb{R}$$

Such that:

- High score $\rightarrow$ unlikely or abnormal triple
- Low score $\rightarrow$ plausible triple

Probabilistic Interpretation

Most methods assume the knowledge graph follows an underlying relational distribution $P(h, r, t)$.

Anomaly score can be defined as:

$$A(t) = -\log P(h, r, t)$$

Thus:

- High probability $\rightarrow$ normal
- Low probability $\rightarrow$ anomaly

This probabilistic view connects anomaly detection with link prediction.

Graph Contextual Definition

A triple may be anomalous relative to its neighborhood:

$$A(t) = f(\text{local structure, relation semantics, global distribution})$$

Thus anomaly depends on:

- Entity types
- Neighbor relations
- Path patterns
- Ontology constraints

Subgraph-Level Definition

For subgraph $S \subseteq G$:

$$A(S) = \text{deviation}(S, \text{expected relational motif distribution})$$

Example:

If most hierarchies are trees, cycles are anomalous.

Goal of Relational Anomaly Detection

The objective is to identify:

- Abnormal triples
- Abnormal entities
- Abnormal subgraphs

Formally:

Find subset $T^* \subseteq T$ such that:

$\forall t \in T^*, P(t \mid G)$ is minimal

In practice:

- Rank triples by anomaly score
- Detect top-k anomalies
- Flag inconsistent relations

CHALLENGES IN RELATIONAL ANOMALY DETECTION

Relational anomaly detection in knowledge graphs presents unique challenges:

1. Multi-Relational Complexity

Entities may participate in many relation types; anomalies depend on relation semantics.

2. Sparsity and Incompleteness

Knowledge graphs often miss facts, making abnormality difficult to distinguish from absence.

3. Scalability

Large KGs contain millions or billions of triples.

4. Heterogeneity

Different entity types and relation types coexist.

5. Contextual Dependence

An edge may be normal in one context but anomalous in another.

6. Lack of Ground Truth

Few labeled anomalies available for training.

METHODOLOGICAL APPROACHES

Relational anomaly detection methods can be categorized based on representation and learning

paradigm.

1. Statistical Relational Methods

These methods model relational distributions using probabilistic assumptions.

Principle

Estimate probability of relation patterns; low-probability triples considered anomalies.

Techniques

- Bayesian relational models
- Markov logic networks
- Probabilistic soft logic

Example

If relation bornIn typically connects Person $\rightarrow$ City, then triple

(Person, bornIn, Country)

has low probability $\rightarrow$ anomaly.

Advantages

- Interpretable
- Incorporates semantics
- Handles uncertainty

Limitations

- Computationally expensive
- Difficult for large graphs

2. Embedding-Based Methods

Embedding models map entities and relations into continuous vector space. Anomaly score derived from embedding plausibility.

Common Models

- TransE
- DistMult
- ComplEx
- RotatE

Detection Strategy

Score triple using embedding function:

$$f(h,r,t) = \| h + r - t \|$$

Large distance $\rightarrow$ anomaly.

Advantages

- Scalable
- Effective for link prediction
- Captures global patterns

Limitations

- Limited interpretability
- May miss local structural anomalies

3. Graph Neural Network (GNN) Methods

GNNs learn node representations using neighborhood aggregation. Relational GNNs extend to multi-relation graphs.

Variants

- Relational Graph Convolutional Networks (R-GCN)
- Graph Attention Networks
- Heterogeneous GNN

Detection Mechanism

- Learn embeddings via GNN
- Reconstruct edges or predict links
- High reconstruction error $\rightarrow$ anomaly

Strengths

- Captures structural context
- Handles heterogeneous relations
- High accuracy

Weaknesses

- Training complexity
- Requires large memory

- Hard to interpret

4. Rule-Based and Logical Methods

Logical rules describe valid relational patterns.

Example rule:

$\text{Person}(x) \rightarrow \exists y \text{ City}(y) \wedge \text{bornIn}(x,y)$

Triples violating rule $\rightarrow$ anomalies.

Sources of Rules

- Ontology constraints
- Association rule mining
- Path patterns

Advantages

- Highly interpretable
- Domain-aware
- Precise anomaly identification

Limitations

- Requires rule discovery
- Incomplete coverage
- Sensitive to noise

5. Subgraph and Motif-Based Methods

Anomalies detected by comparing subgraph patterns.

Idea

Nodes participating in rare relational motifs are anomalous.

Example motifs:

- Star pattern
- Triangle pattern
- Hierarchical pattern

Approach

- Extract local subgraphs

- Compare motif frequency
- Rare motifs → anomalies

Benefits

- Captures structural irregularities
- Works without labels

Drawbacks

- Computationally heavy
- Limited semantic awareness

6. Hybrid Methods

Hybrid methods combine embeddings, rules, and statistics.

Example combinations:

- Embedding + rule constraints
- GNN + probabilistic scoring
- Ontology + neural networks

Hybrid models often outperform single methods.

DETECTION LEVELS

Relational anomaly detection can operate at multiple granularity levels.

Level	Description	Example
Triple	Single relation abnormal	(DrugA treats CityB)
Entity	Node abnormal connectivity	Person linked to many unrelated domains
Subgraph	Pattern abnormal	Circular hierarchy
Graph	Global structure abnormal	Dense unexpected cluster

BENCHMARK DATASETS

Common datasets used in KG anomaly detection research:

Dataset	Domain	Characteristics
Wikidata	General knowledge	Large, noisy
FB15K-237	General KG	Filtered relations
YAGO3	Semantic KG	High quality
WN18RR	Lexical KG	Hierarchical
NELL995	Extracted KG	Noisy relations

These datasets allow evaluation of relational anomaly detection models.

EVALUATION METRICS

Relational anomaly detection evaluation differs from classification.

1. AUC-ROC

Measures ranking quality of anomaly scores.

2. Precision@K

Fraction of true anomalies in top-K predictions.

3. Mean Rank

Average rank of anomalous triples.

4. Reconstruction Error

Used in embedding or GNN methods.

5. Log-Likelihood

Probability under relational model.

COMPARATIVE ANALYSIS OF METHODS

Method Type	Scalability	Interpretability	Structural Awareness	Accuracy
Statistical	Low	High	Medium	Medium
Embedding	High	Low	Low	Medium-High
GNN	Medium	Low	High	High

Method Type	Scalability	Interpretability	Structural Awareness	Accuracy
Rule-Based	Medium	High	Medium	Medium
Subgraph	Low	Medium	High	Medium
Hybrid	Medium	Medium	High	High

Observations:

- GNN and hybrid methods show best accuracy.
- Rule-based approaches best for explainability.
- Embedding methods best for scalability.

APPLICATIONS

1. Fraud Detection

Financial knowledge graphs connect accounts, transactions, and entities. Abnormal transaction relations reveal fraud rings.

Example

anomaly:

AccountA transfers funds to many unrelated shell companies.

2. Cybersecurity

Network knowledge graphs represent hosts, IPs, and activities. Relational anomalies indicate attacks.

Example:

Host communicates with rare malicious domain cluster.

3. Biomedical Discovery

Biomedical KGs link genes, diseases, drugs. Anomalous relations may reveal data errors or novel hypotheses.

Example:

Drug linked to unrelated biological pathway.

4. Knowledge Graph Cleaning

KG construction pipelines introduce errors. Relational anomaly detection helps remove inconsistent triples.

5. Recommendation Systems

User-item knowledge graphs contain preferences. Abnormal relations detect spam or fake users.

CASE STUDY EXAMPLE

Consider a biomedical knowledge graph containing:

(DrugA, treats, DiseaseX)

(DrugA, targets, ProteinY)

(ProteinY, causes, DiseaseZ)

If DiseaseZ unrelated to DiseaseX, then inferred chain:

DrugA $\rightarrow$ ProteinY $\rightarrow$ DiseaseZ

This may contradict medical ontology. The relation path becomes anomalous.

Detection methods:

- Rule: drug treats disease via valid pathway
- Embedding: path low plausibility
- GNN: unusual subgraph embedding

TEMPORAL RELATIONAL ANOMALY DETECTION

Knowledge graphs often evolve over time. Temporal anomalies arise when relations change unexpectedly.

Example:

Company CEO changes daily $\rightarrow$ anomaly.

Methods:

- Dynamic embeddings
- Temporal GNN
- Sequence modeling

Challenges:

- Concept drift
- Historical context
- Streaming data

EXPLAINABILITY IN KG ANOMALY DETECTION

Explainability is important for trust.

Explainable signals:

- Violated rules
- Abnormal paths
- Rare relation patterns
- Low embedding score components

Example explanation:

Triple anomalous because entity type mismatch and rare relation frequency.

Rule-based and hybrid methods provide best explanations.

EMERGING TRENDS

1. Self-Supervised Graph Learning

Learning relational representations without labels improves anomaly detection in sparse graphs.

2. Contrastive Learning

Normal vs corrupted triples contrasted to learn anomaly boundaries.

3. Generative Models

Graph autoencoders and diffusion models reconstruct normal relational patterns.

4. Foundation Models for Graphs

Large graph models integrate semantics and structure for anomaly detection.

5. Privacy-Preserving KG Analysis

Sensitive KGs require anomaly detection without exposing data.

OPEN CHALLENGES

1. Scalability to Web-Scale Graphs

Future KGs may contain trillions of triples.

2. Explainability vs Accuracy Trade-off

Deep models accurate but opaque.

3. Evolving Graphs

Temporal dynamics complicate detection.

4. Noisy Training Data

Unlabeled anomalies degrade models.

5. Cross-Domain Knowledge Graphs

Different semantics across domains.

6. Adversarial Attacks

Malicious insertion of realistic anomalies.

FUTURE RESEARCH DIRECTIONS

- Neuro-symbolic anomaly detection
- Online KG anomaly detection
- Cross-KG anomaly transfer learning
- Human-in-the-loop detection
- Privacy-aware relational modeling
- Multimodal knowledge graph anomalies

CONCLUSION

Relational anomaly detection in knowledge graphs is a crucial research area for ensuring quality, security, and reliability of semantic data. Unlike traditional anomaly detection, it must account for complex relational structures and semantics. This review categorized methods into statistical, embedding-based, GNN, rule-based, subgraph, and hybrid approaches. GNN and hybrid techniques currently achieve best detection accuracy, while rule-based methods remain strongest for interpretability. Applications across fraud detection, cybersecurity, biomedical analysis, and knowledge graph cleaning demonstrate practical importance. However, major challenges remain in scalability, explainability, evolving graphs, and lack of labeled anomalies. Future research integrating symbolic reasoning with deep graph learning and temporal modeling is expected to advance relational anomaly detection in large-scale knowledge graphs.

REFERENCES

1. Akoglu, L., Tong, H., Koutra, D. (2015). Graph based anomaly detection and description: a survey. *Data Mining and Knowledge Discovery*.

2. Nickel, M., Murphy, K., Tresp, V., Gabrilovich, E. (2016). A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*.
3. Schlichtkrull, M., Kipf, T., Bloem, P., et al. (2018). Modeling relational data with graph convolutional networks. *ESWC*.
4. Bordes, A., Usunier, N., Garcia-Duran, A., et al. (2013). Translating embeddings for modeling multi-relational data. *NeurIPS*.
5. Dettmers, T., Pasquale, M., Pontus, S., Riedel, S. (2018). Convolutional 2D knowledge graph embeddings. *AAAI*.
6. Ji, S., Pan, S., Cambria, E., et al. (2022). A survey on knowledge graphs: representation, acquisition and applications. *IEEE TNNLS*.
7. Paulheim, H. (2017). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web Journal*.
8. Zhao, Y., Akoglu, L. (2021). Outlier detection in graph data: a survey. *ACM SIGKDD Explorations*.
9. Rossi, R., Ahmed, N. (2020). Deep graph representation learning: A survey. *IEEE Data Engineering Bulletin*.
10. Chen, Z., Song, L., Li, X. (2023). Anomaly detection in attributed graphs: A deep learning perspective. *TKDE*.