Inferring Network-Level Alarm Influence Pathways for Cellular Infrastructure Reliability

Xiao Cai[a,b], Anandarup Mukherjee[b,*], Qiqi Wang[a,b], Arjun Parekh[c], Ajith Kumar Parlikad[b], Min Xie[a]

[a]*Department of Systems Engineering, City University of Hong Kong, Hong Kong, China*
[b]*Institute for Manufacturing, University of Cambridge, Cambridge, CB3 0FS, United Kingdom*
[c]*BT Group plc, Assembly Building, Cheese Lane, Bristol, BS2 0JJ, United Kingdom*

Abstract

Large-scale telecommunications infrastructure, such as Radio Access Networks (RANs), is a critical component of the modern digital industry. To ensure infrastructure intelligence and operational reliability, it is essential to understand how alarms influence one another across base stations. Existing alarm analysis approaches primarily focus on isolated temporal patterns at individual sites, overlooking cross-site alarm influence mechanisms at the infrastructure level. To address this gap, this paper proposes a Co-Occurrence and Time-Aware Graph Attention Network (COTA-GAT). It infers alarm influence pathways without relying on explicit physical or logical topology information. COTA-GAT constructs a directed influence graph from alarm co-occurrence patterns and incorporates time-decay weighting to represent both the recurrence intensity and temporal proximity of cross-site alarm interactions. Two time-aware graph attention layers are then used to learn delay-sensitive node embeddings by jointly considering node-level features and time-decay-weighted edges. Subsequently, a link-prediction decoder estimates potential alarm influence pathways. Additionally, a controlled Network Simulator-3 (NS-3) Long Term Evolution (LTE) RAN simulation with independently generated fault-cascade ground truth is used to validate the true propagation edges beyond chance. Experiments conducted on a large-scale real-world telecommunications alarm dataset demonstrate that COTA-GAT outperforms most graph-based baselines and provides an interpretable topology-agnostic influence graph. Considering a day-long propagation window, COTA-GAT achieves area under the receiver operating characteristic curve and average precision values of up to 0.9109 and 0.9235, respectively. The proposed framework provides a

novel data-driven foundation for proactive reliability management in large-scale communication networks, even in the absence of information about the underlying network topology.
Keywords: Alarm influence pathway inference, Topology-agnostic graph learning, Time-aware graph attention networks, Radio access network reliability, Telecommunications alarm analysis, Infrastructure fault management

1. Introduction

Modern telecommunications infrastructure forms the backbone of the contemporary digital industry. As large-scale and highly interconnected infrastructure systems, Radio Access Networks (RANs) must operate under high reliability and resilience requirements [1]. Although RAN architectures differ in physical deployment and functional organisation, they introduce strong interdependencies among components through shared resources, network traffic coupling, and control mechanisms. Alarm events seldom occur in isolation and may trigger a cascade, or may be a result of an already ongoing cascade of alarms. Mukherjee *et al.* use these alarm dynamics within RAN Base Stations (BSs) to analyse, interpret and forecast alarms and score the BS's resilience [2]. Alarms triggered at one BS often propagate to other sites through latent network-level interactions. This gives rise to complex alarm-influence pathways. A notable example is the U.K.'s October 2025 major network outage, which left tens of thousands of U.K. customers without internet or mobile connectivity [3]. This illustrates the potential for widespread disruption caused by large-scale RAN failures. Therefore, the understanding and modelling of these network-level propagation mechanisms is crucial for infrastructure intelligence, resilience enhancement, and proactive operational management [4, 5].

Most existing alarm analysis approaches specifically focus on individual BSs or isolated alarm sequences, and typically work with local temporal patterns. They overlook how alarms spread across sites and how influence pathways emerge among related BSs. As a result, they are limited in their ability to capture the collective behaviour of alarms in large-scale RAN deployments.

*Corresponding author
Email address: am2910@cam.ac.uk (Anandarup Mukherjee)

Early studies on alarm management in telecommunication networks mainly relied on statistical and rule-based techniques to accelerate fault detection and mitigate alarm floods [6–10]. These approaches exploited temporal patterns, predefined parent-child relationships, or domain-specific expertise to identify abnormal behaviours and support operator diagnosis. Although effective in controlled settings, such methods typically require extensive expert knowledge and exhibit limited scalability.

The increasing availability of operational data often leads to the selection of data-driven methods as the primary choice to improve automation and robustness in alarm handling [11, 12]. Deep learning models have demonstrated strong performance in industrial fault diagnosis and condition monitoring tasks, particularly for single-asset or signal-level analysis [13–15]. However, these approaches are typically applied to isolated entities. They rarely model cross-entity interactions or propagation effects, which are critical in large-scale RANs.

Graph-based learning has also emerged as a powerful paradigm for modelling relational structures in networked systems. Graph convolutional models have been widely adopted in industrial applications to capture structured dependencies, including defect localisation, component segmentation, and fault diagnosis [16–18]. By representing BSs as nodes and inter-site dependencies as edges, graph-based methods enable the modelling of alarm interactions beyond isolated temporal sequences [19, 20]. Graph neural networks (GNNs) have further enhanced this capability by learning latent dependency structures directly from data and have been applied to alarm analysis, root-cause localisation, and network-wide reasoning [21, 22]. However, most existing GNN-based approaches rely on predefined or static topologies and do not fully exploit the temporal characteristics of alarm co-occurrence in large-scale RANs.

Similarly, temporal and dynamic graph learning [23, 24] have attracted significant attention in recent years. A variety of models have been proposed to handle evolving graph structures and dynamic link prediction, enabling the learning of time-dependent dependencies in event-driven networks [25–27]. In industrial monitoring and anomaly detection, graph attention mechanisms have been introduced to model time-varying dependencies among multiple variables [28]. These methods demonstrate the importance of jointly modelling graph structure and temporal dynamics when analysing sequential relational data, but they are rarely tailored to the characteristics of

operational alarm streams in telecommunication networks.

Another related line of research concerns event diffusion and propagation in complex networks, including information spreading [29], failure cascades [30–32], and relational modelling [33]. Such studies have shown that events triggered at one node can propagate through networked systems via latent interaction pathways, giving rise to cascades and correlated behaviours. These models provide a useful abstraction for understanding how alarms may spread across BSs, although they are typically developed for social networks, traffic networks or power grids rather than RANs.

Finally, graph-based approaches have also been explored for anomaly detection and failure analysis in networked systems [34–36]. These methods aim to identify abnormal nodes or edges that deviate from typical network behaviour. Although the value of graph-based methods is increasingly recognised for fault-related tasks, most existing approaches focus on detecting anomalies after failures occur rather than on inferring latent alarm-influence pathways from historical co-occurrence data.

Existing approaches to alarm management, graph-based learning, and temporal modelling lack explicit modelling of cross-site propagation, rely on known topologies, and seldom, fully integrate temporal dynamics with relational inference. To address these gaps, this work proposes a topology-agnostic inference of alarm-influence pathways in large-scale RANs through a co-occurrence and time-aware graph attention network (COTA-GAT). The COTA-GAT models alarm propagation as a graph-based inference problem driven by historical alarm interactions. First, heterogeneous raw alarm records are transformed into unified BS-level feature representations. Specifically, categorical attributes are encoded using normalised frequency histograms, while numerical attributes are standardised to preserve scale comparability. Subsequently, a directed alarm propagation graph is constructed using a co-occurrence mechanism that identifies implicit relationships between BSs based on historical alarm interactions. To emphasise recent and more influential interactions, edge weights are assigned using a time-decay kernel. A graph attention mechanism is then applied to adaptively refine the propagation strength between nodes, enabling the model to focus on critical interdependencies. The temporal aspect of alarm propagation is incorporated through time-decay-weighted edges, which allows the graph attention mechanism to

emphasise temporally proximal and recurrent cross-site alarm interactions.

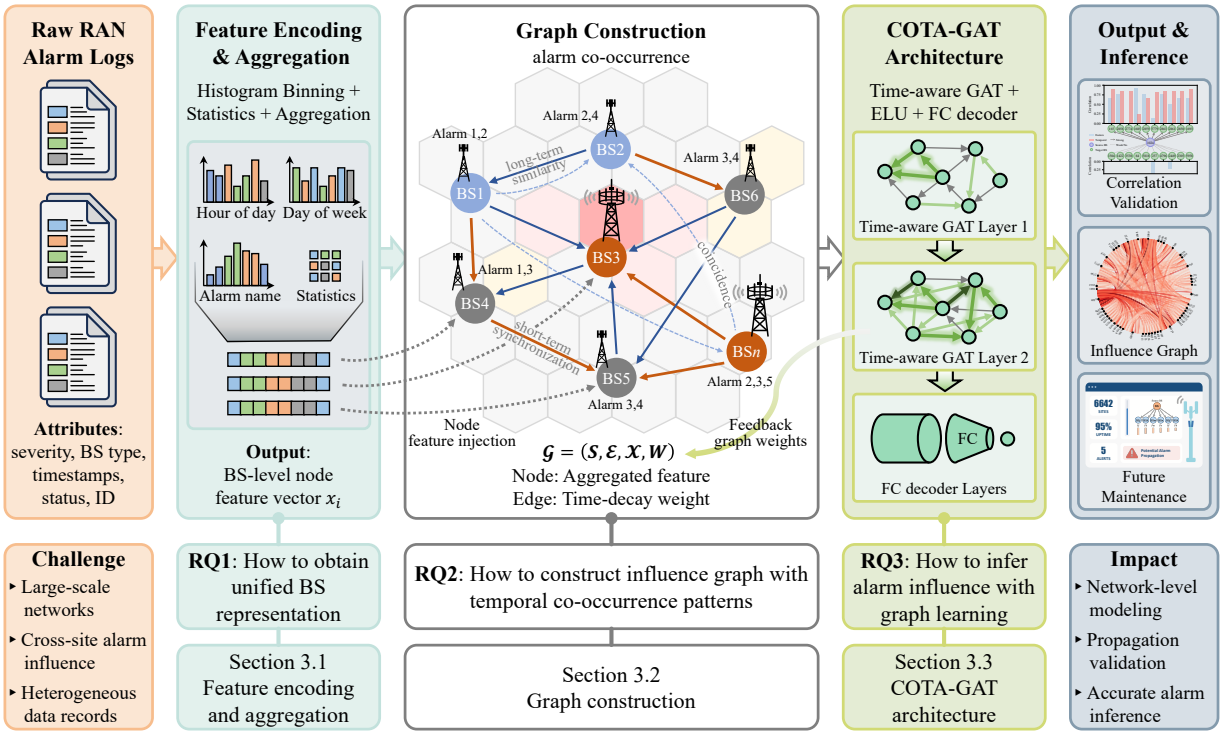


Figure 1: The graphical outline of the proposed COTA-GAT for RAN alarm influence pathway inference approach.

The proposed COTA-GAT model overcomes the limitations of traditional methods by reducing reliance on expert knowledge and improving robustness against noisy or redundant alarms. Unlike existing approaches that focus primarily on individual BS and alarm sequences [2], COTA-GAT is designed to capture cross-site (inter-BS) alarm effects, which are prevalent in practical RAN deployments. Importantly, the proposed framework does not require any prior knowledge of physical, geographical, or logical RAN topology, and instead infers cross-site relationships directly from alarm co-occurrence dynamics. For instance, a hardware malfunction at one BS may trigger congestion alarms at neighbouring BSs due to backhaul overload or load redistribution. By modelling such latent alarm influence pathways, COTA-GAT provides insight into how cross-site alarm dependencies emerge across the RANs. Based on the overarching goal of topology-agnostic alarm influence pathway inference, this work aims to answer the following research questions:

RQ1) How can heterogeneous alarm records be encoded and aggregated into informative BS-level

representations suitable for topology-agnostic graph learning?

RQ2) How can temporal co-occurrence patterns be exploited to construct a directed influence graph that reflects propagation-consistent cross-site dependencies?

RQ3) How can time-aware graph attention learning be used to infer delay-sensitive alarm influence pathways across BS in the absence of explicit network topology?

Figure 1 provides a graphical overview of the proposed framework to facilitate understanding of the overall research pipeline.

The remainder of this paper is structured as follows. Section 2 defines the alarm influence pathway inference problem and outlines the necessary preliminaries. Section 3 describes the methodology of proposed COTA-GAT framework, including the feature construction strategy, graph modelling process, and time-aware learning architecture. Section 4 describes the real-world British Telecom (BT) RAN alarm dataset, the experimental platform , implementation details, and preliminary evidence of cross-site alarm dependencies. Section 5 presents experimental evaluation of the method on the real-world BT RAN alarm dataset (which does not include the network topology information), followed by Section 6 that validates the approach using a controlled NS-3 LTE RAN simulator-based data (which provides topology information to understand true propagation of the approach). Finally, Section 7 concludes the paper and discusses future research directions.

2. Problem Formulation and Preliminaries

2.1. RAN Alarm Graph Construction

RAN alarms are key indicators of network health and performance in the telecommunication field. The data is usually collected from the alarm management system, which monitors and records various alarm events generated by BSs. As for the telecommunication network, it can naturally be considered a complex topological graph, with BSs represented as nodes, their correlations or inferred dependencies captured as edges, and alarm records as features. Therefore, the

graph can be constructed as

$$\begin{cases} \mathcal{G} = (\mathcal{S}, \mathcal{E}, \mathcal{X}, W) \\ \mathcal{S} = \left\{ s_i \middle| i = 1, 2, \ldots, n \right\} \\ \mathcal{E} = \left\{ (s_i, s_j) \middle| s_i, s_j \in \mathcal{S} \right\} \\ \mathcal{X} = \left\{ x_i \middle| i = 1, 2, \ldots, n \right\} \end{cases}, \tag{1}$$

where $\mathcal{G}$ denotes the directed graph, $\mathcal{S}$ is the collection of BSs, $\mathcal{E}$ is the set of edges, where (s_i, s_j) indicates a directed edge from BS s_i to s_j, $\mathcal{X}$ is the selected features, and $W \in \mathbb{R}^{n \times n}$ represents the weighted adjacency matrix.

2.2. Graph-based Alarm Influence Pathway Inference

Alarm influence pathway inference aims to identify how alarms influence across networked entities from the current network state and historical alarm observations. In this work, we focus on inferring potential alarm influence pathways between BSs in a RAN. To be specific, given a graph constructed from historical alarm data, the goal is to estimate whether alarms observed at one BS s_i are likely to be followed by related alarms at another BS s_j within a predefined time window. This can be formulated as a link prediction problem, where the task is to estimate the probability of a potential edge (s_i, s_j) appearing in the graph $\mathcal{G}$, i.e.,

$$\hat{y}_{ij} = \mathbb{P}\left((s_i, s_j) \in \mathcal{E} \mid \mathcal{G} \right), \tag{2}$$

In this work, an alarm influence pathway refers to a directed, temporally conditioned dependency between BSs, inferred from repeated patterns of alarm co-occurrence rather than from explicit physical connectivity. Such pathways do not necessarily imply deterministic causality, but capture statistically and operationally meaningful propagation tendencies in large-scale RANs. As manually verified propagation labels are unavailable for our real-world dataset, the edge label y_{ij} is defined as a weak operational label derived from the co-occurrence structure of the alarm logs. Specifically, $y_{ij} = 1$ indicates that BS pair (s_i, s_j) satisfies the same-alarm, temporal-precedence, and propagation-window criteria in the held-out evaluation split, while $y_{ij} = 0$ indicates that no such dependency is observed. These labels are not treated as manually verified causal ground

truth. Instead, they represent propagation-consistent supervision signals extracted from operational alarm logs.

To avoid train-test leakage, a single co-occurrence influence graph $\mathcal{G}$ is constructed from all alarm records. For evaluation, its edges are randomly partitioned into disjoint training, validation, and testing sets, so that the model is trained to score held-out edges against sampled negative pairs (which follows the standard transductive link-prediction protocol).

2.3. Graph Convolutional Networks

In this study, COTA-GAT is built upon principles established by Graph Convolutional Networks (GCN). The GCN is employed to capture spatial interactions within the BS graph, utilising spectral graph convolution. The normalised graph Laplacian matrix L of a graph is defined as:

$$L = I_n - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}, \tag{3}$$

where I_n is the identity matrix, $D \in \mathbb{R}^{n \times n}$ is the diagonal degree matrix with diagonal elements $D_{ii} = \sum_j W_{ij}$. Since L is a real symmetric matrix, it can be decomposed as

$$L = U \Lambda U^{\mathrm{T}}, \tag{4}$$

where $U \in \mathbb{R}^{n \times n}$ represents the Eigenvector matrix of L, while $\Lambda \in \mathbb{R}^{n \times n}$ is the diagonal matrix composed of the corresponding Eigenvalues.

Eventually, the spectral graph convolution of the input features $\mathcal{X}$ is defined as:

$$\Theta * \mathcal{X} = \Theta(L)\mathcal{X} = \Theta\left(U \Lambda U^{\mathrm{T}}\right)\mathcal{X} = U \Theta(\Lambda) U^{\mathrm{T}} \mathcal{X}, \tag{5}$$

where $*$ denotes the graph convolution operator, and Θ represents the kernel function, which can be viewed as a filter in the Fourier domain.

3. Methodology

The primary objective of this work is to effectively extract latent temporal dynamics and spatial relationships among BS alarms from historical data and to infer alarm influence pathways within RANs. As previously outlined, the alarm influence inference task is formulated as a link prediction

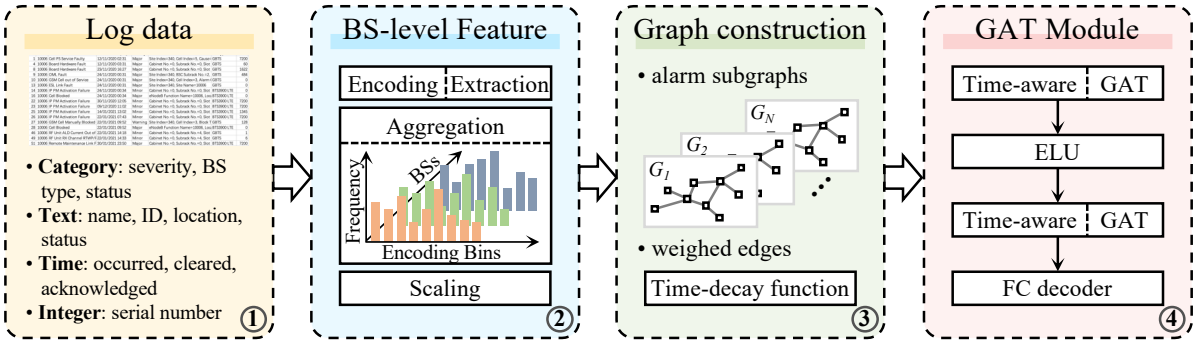


Figure 2: The methodological flow for the proposed alarm influence pathway inference.

problem on a graph. To address this, the proposed COTA-GAT framework is explained in this section. The flowchart is illustrated in Figure 2.

Firstly, the raw alarm attributes are preprocessed and encoded using dedicated encoders. Both pattern and statistical features are then extracted and aggregated to form BS-level node representations. Subsequently, a directed alarm-influence graph is constructed in which nodes represent BSs and directed edges represent propagation-consistent alarm dependencies inferred from the co-occurrence mechanism. The edge weights are assigned using a time-decay function to reflect the recurrence intensity and temporal proximity of cross-site alarm interactions. Third, two time-aware Graph Attention Network (GAT) layers are stacked with an Exponential Linear Unit (ELU) activation function to learn node embeddings by jointly considering BS-level features and delay-sensitive edge weights. These embeddings are then passed through a Fully Connected (FC) link prediction decoder to estimate the likelihood of alarm influence between BS pairs. The detailed process of each step is described below.

3.1. Feature Encoding and Aggregation

RAN alarm records typically include attributes describing alarm events, such as the alarm name, occurrence timestamp, severity level, and maintenance status. These attributes provide valuable information about alarm propagation patterns. However, they are inherently heterogeneous in data types, comprising categorical, temporal, numerical, and textual fields, and cannot be directly utilised in graph-based models. Furthermore, each BS node may generate a large number

of alarm records over time, making raw-record-level modelling impractical. As the objective is to learn influence pathways at the node level, the feature representation of each BS should not depend on individual alarms but should reflect aggregated statistical characteristics of its alarm behaviour. This makes it essential to construct informative, aggregated, and numerically stable node features before graph modelling.

In this section, a unified feature encoding and aggregation method is proposed to transform heterogeneous alarm records into expressive BS-level node features. In the experimental implementation, each BS is represented by a static operational profile aggregated over its alarm history (alarm-name, categorical, and temporal histograms and statistical summaries). These are node attributes describing long-term site behaviour, not link labels. This is which is summarised in Algorithm 1. To keep this consistent with the transductive setting, node features and normalisation parameters are computed over the full graph. As the prediction target is a randomly held-out set of edges rather than a future time period, this does not leak the held-out labels into the features.

For categorical and some textual attributes, such as alarm severity levels and alarm name, categorical encoding is first applied to convert them into numerical representations. Then, for each BS node s_i, the frequency of each attribute is calculated to form a histogram-based representation. Specifically, if $C = c_1, c_2, \ldots, c_K$ denotes the set of categorical values for a given attribute, the categorical feature vector of BS s_i is defined as:

$$h_i(cat) = \left[\frac{n_i(c_1)}{N_i}, \frac{n_i(c_2)}{N_i}, \ldots, \frac{n_i(c_K)}{N_i} \right] \tag{6}$$

where $n_i(c_k)$ represents the occurrence count of category c_k at BS s_i, and $N_i = \sum_{k=1}^{K} n_i(c_k)$ is the total number of alarms recorded at s_i. This normalisation mitigates the scale effect caused by varying alarm volumes across different BSs and ensures comparability of categorical patterns.

For temporal attributes, such as alarm occurrence timestamps, time-dependent and periodic patterns are often observed due to operational or human activity cycles. To capture such periodicity, features such as hour of the day (hod) and day of the week (dow) are extracted. The histogram is constructed based on the frequency of occurrences within each HoD and DoW:

$$h_i(hod) = \left[\frac{n_i(hod_1)}{N_i}, \frac{n_i(hod_2)}{N_i}, \ldots, \frac{n_i(hod_{24})}{N_i} \right], \; h_i(dow) = \left[\frac{n_i(dow_1)}{N_i}, \frac{n_i(dow_2)}{N_i}, \ldots, \frac{n_i(dow_7)}{N_i} \right], \tag{7}$$

Algorithm 1 Feature encoding and aggregation algorithm

Input: Alarm record set $\mathcal{A} = a_1, a_2, \ldots, a_M$, BS set $\mathcal{S} = s_1, s_2, \ldots, s_N$

Output: Node feature matrix $\mathcal{X} \in \mathbb{R}^{N \times d}$, encoders, normalization parameters Θ_{feat}

For each BS $s_i \in \mathcal{S}$ **do**

  $\mathcal{A}_i \leftarrow$ alarm records of s_i, $N_i \leftarrow |\mathcal{A}_i|$ ▷ Total number of alarms at s_i

  1. Categorical attributes:

  Encode categorical attributes into numerical values

  For each category c_k **do**

    $h_i(c_k) \leftarrow n_i(c_k)/N_i$

  End For

  $h_i(cat) \leftarrow [h_i(c_1), \ldots, h_i(c_K)]$ ▷ concatenate category histogram

  2. Temporal attributes:

  Extract hour of the day (hod) and day of the week (dow) from timestamps

  For each hod_k bin **do**

    $h_i(hod_k) \leftarrow n_i(hod_k)/N_i$

  End For

  $h_i(hod) \leftarrow [h_i(hod_1), \ldots, h_i(hod_{24})]$ ▷ concatenate hod histogram

  For each dow_k bin **do**

    $h_i(dow_k) \leftarrow n_i(dow_k)/N_i$

  End For

  $h_i(dow) \leftarrow [h_i(dow_1), \ldots, h_i(dow_7)]$ ▷ concatenate dow histogram

  3. Statistical summary:

  For each histogram $h_i(cat), h_i(hod), h_i(dow)$ **do**

    $st_i(h_i(\cdot)) \leftarrow [\mu_i, \tilde{x}_i, \sigma_i]$ ▷ mean, median, standard deviation

  End For

  $st_i \leftarrow [st_i(h_i(cat)), st_i(h_i(hod)), st_i(h_i(dow))]$

  $\mathbf{x}_i \leftarrow [h_i(cat) \parallel h_i(hod) \parallel h_i(dow) \parallel st_i]$

End For

$\mathcal{X} \leftarrow [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}N]^T$

Return: $\mathcal{X}$, encoders, Θfeat

where $n_i(hod_k)$ and $n_i(dow_k)$ denote the occurrence counts of alarms at hour k and day k respectively for BS s_i.

After that, statistical measures are extracted to summarise the distributional characteristics of $h_i(cat)$, $h_i(hod)$, and $h_i(dow)$. For each histogram $h_i(\cdot)$, we compute the mean, median, and standard deviation, and then concatenate them to form the statistical feature vector:

$$st_i = [\mu_i(cat), \tilde{x}_i(cat), \sigma_i(cat), \ldots, \sigma_i(dow)]. \tag{8}$$

Finally, the categorical, textual, temporal, and statistical feature vectors are concatenated to form the complete node representation:

$$\mathbf{x}_i = [h_i(cat); |; h_i(hod); |; h_i(dow); |; st_i]\,. \tag{9}$$

Eventually, the node feature $\mathbf{x}_i$ aggregates both the categorical and temporal distributional characteristics and the numerical statistical properties of alarms at BS s_i, providing a rich and comprehensive input for graph-based learning models.

3.2. Graph Construction

After feature encoding and extraction, a directed graph is constructed to represent the relationships among BSs based on alarm co-occurrence as shown in Figure 3. In the graph, each BS is represented as a node, and edges and weights are established based on the temporal proximity of identical alarms occurring across different BSs. The constructed graph is purely data-driven and does not correspond to any predefined network topology.

To be specific, let $A = \{a_1, a_2, \ldots, a_N\}$ denote the complete set of alarm events, where each alarm a_i is represented as a tuple:

$$a_i = (s_i, t_i, c_i), \tag{10}$$

where s_i is the BS that triggers the alarm, t_i is the timestamp of occurrence, and c_i denotes the alarm name. For each specific alarm name $c \in C$, we first identify all occurrences belonging to this alarm, i.e., $A_c = a_i \mid a_i \in A, c_i = c$. To capture potential influence pathways or co-occurrence relationships among BSs, we define a temporal window parameter $t_w > 0$. For any two alarms $a_i, a_j \in A_c$, a directed edge is established if the following condition is satisfied:

$$0 < t_j - t_i \leq t_w. \tag{11}$$

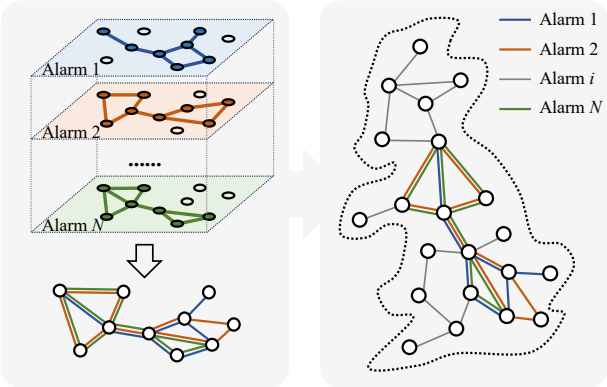


Figure 3: The construction process of alarm graphs.

In such a case, the directed edge is created from the BS that triggered the alarm earlier to the one that triggered it later:

$$e_{ij} = (s_i \rightarrow s_j), \quad e_{ij} \in \mathcal{E}. \tag{12}$$

This is based on the assumption that alarms occurring in close temporal proximity may be related, possibly due to shared network conditions or cascading failures. It should be emphasised that the directed edge defined in Eq. (12) does not by itself constitute proof of physical or causal fault propagation. Since the available alarm logs are observational data, correlated alarms may also be caused by unobserved external factors, such as weather conditions, shared power supply, common transmission infrastructure, or maintenance operations. Therefore, the constructed graph is interpreted as a data-driven alarm influence graph, in which edges represent propagation-consistent dependencies supported by temporal precedence, alarm-type consistency, temporal proximity, and repeated cross-site occurrence, rather than definitive causal links. It should also be noted that the time window t_w is crucial, as it determines the sensitivity of the co-occurrence detection. A smaller t_w captures more immediate relationships, while a larger t_w may reveal longer-term dependencies. In practice, t_w can be determined from the statistical distribution of the time intervals between identical alarms across different BSs.

To reflect the strength of temporal correlation, each edge e_{ij} is assigned a weight w_{ij} inversely proportional to the time interval between the two alarms:

$$w_{ij} = \frac{1}{1 + (t_j - t_i)/\tau}, \tag{13}$$

where τ is a time-decay hyperparameter controlling the sensitivity to temporal distance. This ensures that closely occurring alarms contribute more significantly to the influence graph.

The inverse proportional decay in Eq. (13) is adopted because it is bounded, simple, and relatively insensitive to the exact value of τ. It assigns the maximum weight when $\Delta t = t_j - t_i \to 0$ and then decreases smoothly as the temporal interval increases. Compared with an exponential kernel, this form avoids overly aggressive suppression of moderately delayed alarms, which may occur in operational RANs due to reporting latency, network management delay, or cascaded effects across subsystems. To examine whether the results depend on this kernel choice, we also compared the inverse proportional kernel with the exponential kernel $\exp(-\Delta t/\tau)$ in Section 6.

For a pair of BSs (s_i, s_j), multiple alarm co-occurrences may be observed across different alarm types and timestamps within the predefined time window. Let each occurrence be associated with a time-decay weight $w_{ij}^{(k)}$, computed by Eq. (13). Denoted by K_{ij}, the total number of such occurrences is:

$$K_{ij} = \sum_{m \in \mathcal{A}} M_{ij}^{(m)}, \tag{14}$$

where $\mathcal{A}$ is the set of alarm types and $M_{ij}^{(m)}$ is the number of occurrences of alarm type m that induce a directed interaction from s_i to s_j.

To aggregate multiple time-decay weights, a cumulative-strength formulation is adopted. It jointly reflects both temporal closeness and recurrence frequency. Specifically, the aggregated influence strength S_{ij} is defined as

$$S_{ij} = \sum_{k=1}^{K_{ij}} w_{ij}^{(k)}, \tag{15}$$

This cumulative design is crucial for enabling a topology-agnostic approach without learning spurious relationships. It emphasises edges that consistently exhibit high-frequency and short-delay temporal correlations. Consequently, a rare coincidence between two geographically or logically unrelated BSs will result in a negligible cumulative weight, effectively filtering out random environmental noise and isolating genuine operational cascades. To reduce the impact of extremely frequent alarm pairs dominating the graph, a logarithmic compression is applied to S_{ij}.

Specifically, the compressed weight is defined as

$$\tilde{w}ij = \frac{\log(1 + Sij)}{\log(1 + S_{\max})}, \tag{16}$$

where, $S_{\max} = \max_{(p,q)} S_{pq}$. The final edge weight $\tilde{w}ij \in (0, 1]$ represents the relative cumulative influence intensity among all BS pairs. Then the adjacency matrix W can be obtained as:

$$W = \begin{cases} \tilde{w}ij, & \text{if } e_{ij} \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

Finally, by aggregating all individual alarm-type graphs $\mathcal{G}^{(m)}$, a comprehensive co-occurrence-based alarm graph $\mathcal{G} = (\mathcal{S}, \mathcal{E}, \mathcal{X}, W)$ is constructed as follows:

$$\mathcal{G} = \bigcup_{m=1}^{N} \mathcal{G}^{(m)}. \tag{18}$$

Eq. (18) is applied to all alarm records to construct the complete co-occurrence influence graph used as model input. Its edges are then randomly partitioned into training, validation, and testing sets, as detailed in Section 5.1.

3.3. COTA-GAT Architecture

The COTA-GAT architecture consists of two time-aware enhanced GAT layers followed by a link prediction decoder. It operates on a static BS-level feature matrix and a time-decay-weighted co-occurrence graph. The temporal information is incorporated through the edge weights $\tilde{w}_{ij}$, which are derived from alarm occurrence intervals and recurrent co-occurrence strength. Therefore, COTA-GAT captures propagation-consistent temporal proximity at the edge level rather than applying sequence modelling to static node features.

In GCN, all neighbouring BSs contribute equally during feature aggregation, which may limit its ability to capture the varying importance of different neighbours. To address this limitation, a learnable attention mechanism is incorporated to assign adaptive weights to BSs based on their relevance. The convolution operation in GCN is then replaced with an attention-based aggregation function. Specifically, let the input feature of the layer be denoted as $h = \{h_1, h_2, \ldots, h_n\}$, $h_i \in \mathbb{R}^F$, where n represents the number of nodes, and F corresponds to the input feature dimension of each

node. The attention coefficients e_{ij} are then calculated according to Eq. (19), which quantifies the importance of node j to node i.

$$e_{ij} = \text{LeakyReLU}\left(a\left(\mathbf{\Phi}h_i, \mathbf{\Phi}h_j\right)\right), \tag{19}$$

where LeakyReLU is a non-linear activation function, $a(\cdot,\cdot) : \mathbb{R}^F \times \mathbb{R}^{F'} \rightarrow \mathbb{R}$ represents a shared attention function parameterised by a trainable weight vector $\vec{a}$, $\mathbf{\Phi} \in \mathbb{R}^{F' \times F}$ denotes the learnable weight matrix used to transform the input features.

To ensure comparability across different nodes, a softmax function is applied to the attention coefficients. Meanwhile, the computation is restricted to nodes $j \in \mathcal{N}i$, where $\mathcal{N}i$ represents the first-order neighbours of node i, so as to preserve the inherent graph structure. Consequently, the normalized attention coefficients αij are obtained as

$$\alpha ij = \frac{\exp\left(e_{ij}\right)}{\sum_{k \in \mathcal{N}i} \exp\left(eik\right)}. \tag{20}$$

Assuming the output feature is $h' = \{h'_1, h'_2, \ldots, h'_n\}$, where $h'i \in \mathbb{R}^{F'}$ and F' is the output feature dimension. The feature aggregation for node i is then computed as:

$$h'i = \sigma\left(\sum j \in N(i)\alpha ij\mathbf{\Phi}h_j\right), \tag{21}$$

where σ is the activation function.

To stabilise the attention learning process, a K-head attention mechanism is introduced, in which K independent attention operations are performed in parallel based on Eq. (21). The resulting features from each head are then concatenated to form the final representation, expressed as:

$$h'i = \Big\| k = 1^K \sigma\left(\sum_{j \in N(i)} \alpha_{ij}^{(k)} \mathbf{\Phi}^{(k)} h_j\right), \tag{22}$$

where $\alpha_{ij}^{(k)}$ and $\mathbf{\Phi}^{(k)}$ denote the attention coefficients and weight matrix of the k-th head, respectively, and $\|$ represents vector concatenation. Finally, the outputs from multiple heads are averaged to achieve feature compression and maintain a consistent dimensionality, as shown below:

$$h'_i = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j \in N(i)} \alpha_{ij}^{(k)} \mathbf{\Phi}^{(k)} h_j\right). \tag{23}$$

COTA-GAT operates on a single static graph and there is no temporal dimension and no sequence of snapshots $\mathcal{G}_t$. This mechanism allows the model to learn how strongly node s_i should attend to node s_j by jointly considering their transformed feature representations. Considering the temporal nature of alarm influence, the attention mechanism is further enhanced by incorporating time-aware edge weights $\tilde{w_{ij}}$ as defined in Eq. (16), enabling the model to focus on more relevant neighbours based on temporal proximity. The adjusted attention coefficients and normalised attention scores are computed as follows:

$$\tilde{e}ij = \tilde{w}ij \cdot e_{ij}, \tag{24}$$

$$\alpha_{ij} = \frac{\exp(\tilde{e}ij)}{\sum k \in \mathcal{N}(i) \exp(\tilde{e}_{ik})}, \tag{25}$$

This allows the model to assign higher importance to neighbouring BSs whose alarms occur shortly after the source BS and repeatedly appear within the predefined time window. In this way, temporal proximity and recurrence are incorporated directly into the graph attention process.

After obtaining the node embeddings from the two time-aware GAT layers, a link prediction decoder is employed to estimate the probability of alarm influence between BSs. Given nodes i and j and their embeddings $\mathbf{h}_i, \mathbf{h}j$, the probability of edge existence between them is computed via inner product:

$$\hat{y}ij = \sigma\left(\mathbf{h}_i^\top \mathbf{h}_j\right), \tag{26}$$

where $\sigma(\cdot)$ is the sigmoid function. The model is trained using Binary Cross-Entropy (BCE) loss:

$$\mathcal{L} = -\sum_{(i,j)\in E^+} \log \hat{y}ij - \sum (i,j) \in E^- \log(1 - \hat{y}_{ij}), \tag{27}$$

where E^+ and E^- denote positive and negative edge sets, respectively.

3.4. Topology-agnostic Design and Mitigating Spurious Correlations

A natural concern with a topology-agnostic approach is the potential for learning spurious influence pathways between geographically unrelated BSs. However, the topology-agnostic design is driven by critical practical necessities in telecommunication operations. In large-scale deployments, constructing an accurate physical or geographical connection graph is exceptionally

difficult. It requires extensive prior domain knowledge and manual preparation. Precise geographical tracking is often hindered by commercial data confidentiality and anonymisation policies. Furthermore, in modern centralised RANs, alarm influence pathways are mainly determined by logical coupling rather than strict geographic adjacency. Some geographically unrelated sites may share centralised baseband resources or virtualised network slices, leading to synchronised logical cascades. By operating without predefined explicit topologies, COTA-GAT captures these logical pathways while bypassing the operational bottlenecks of physical graph construction.

To prevent the model from learning spurious links and random coincidences between completely independent BSs caused by external environmental noise, COTA-GAT employs a dual-filtering mechanism. The first is algorithmic filtering using cumulative weights. The graph construction leverages a cumulative scoring mechanism rather than simple binary co-occurrence. As defined in Eqs. (13) and (15), the overall influence strength S_{ij} is the cumulative sum of time-decayed weights over all historical instances. This cumulative design emphasises edges that consistently exhibit high-frequency and short-delay temporal correlations. Consequently, a rare or sporadic coincidence between two geographically or logically unrelated BSs will result in a mathematically negligible cumulative weight, effectively filtering out random environmental noise.

The second is implicit filtering using graph attention. Beyond the initial edge weights, the GAT module implicitly filters out spurious correlations by evaluating interactions among node features. As defined in Eq. (19), the attention scores are computed based on the transformed features of the connected nodes. These features encapsulate the long-term statistical and operational profiles of the BSs (aggregated in Section 3.1). Geographically and logically unrelated BSs are highly unlikely to share matching long-term profiles. Therefore, even if a weak edge forms through temporal coincidence, the graph attention mechanism will assign it a low attention score due to a structural feature mismatch, effectively suppressing isolated geographic anomalies.

4. Data Description and Preprocessing

To validate the effectiveness of the proposed method, we conducted a case study using a real-world dataset of RAN alarms provided by BT. The dataset comprises approximately 13.3 million raw alarm records ($\simeq$ 0.95 million after cleaning and de-duplication), spanning predominantly

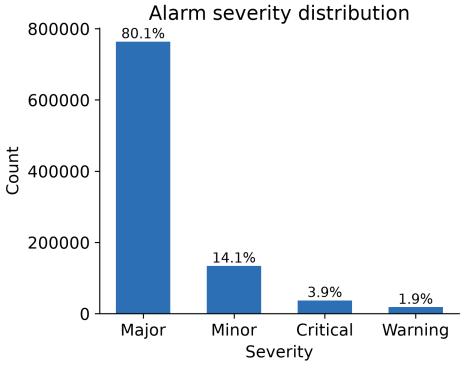
2019–2021, collected from 2G, 4G, and 5G network equipment (GSM, LTE, and NR base stations) and covering 6,773 base stations across the United Kingdom. Since it contains sensitive operational information, it underwent strict double anonymisation. All identifying fields were removed, and the anonymised data were further scrambled to prevent re-identification. This process preserves the statistical and temporal characteristics of the data while ensuring confidentiality. Each raw alarm record contains 21 attributes. Among them, 13 features that specifically relate to alarms and BSs have been extracted. The details of these selected features are summarised in Table 1.

Table 1: Summary of selected features for alarm records

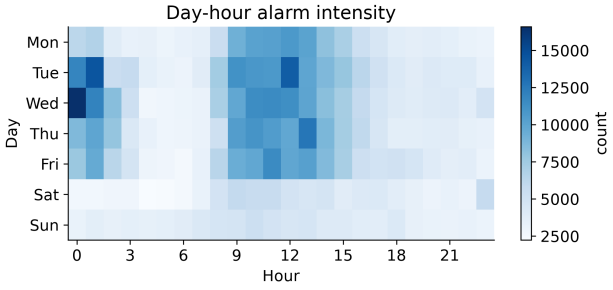
No.	Description	Type
1	Alarm severity	Category
2	Alarm name	Text
3	Source (BS) ID	Text
4	Source (BS) type	Category
5	Alarm location information (log)	Text
6	Alarm occurred timestamp	Time
7	Alarm cleared timestamp	Time
8	Alarm acknowledged timestamp	Time
9	Alarm cleared status	Category
10	Alarm acknowledged status	Category
11	Alarm log serial number	Integer
12	Equipment alarm serial number	Integer
13	BS maintenance status	Text

Figure 4 provides an overview of the dataset through several visualisations. (a) shows the distribution of alarm severity levels; (b) represents the aggregated alarm intensity over the day-hour grid, highlighting pronounced temporal regularities; (c) illustrates the distribution of alarm counts across BSs; and (d) reports the BSs with the highest alarm numbers.

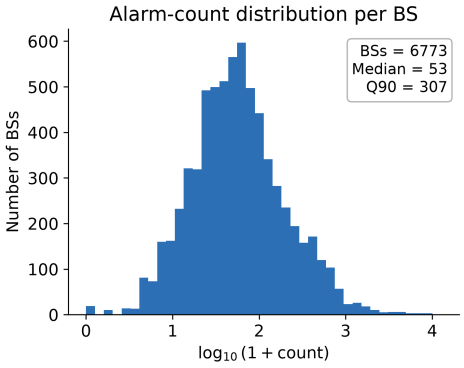
The dataset exhibits a high degree of heterogeneity, encompassing multiple data types with varying structures and levels of complexity. In particular, Feature 5 (alarm location information)



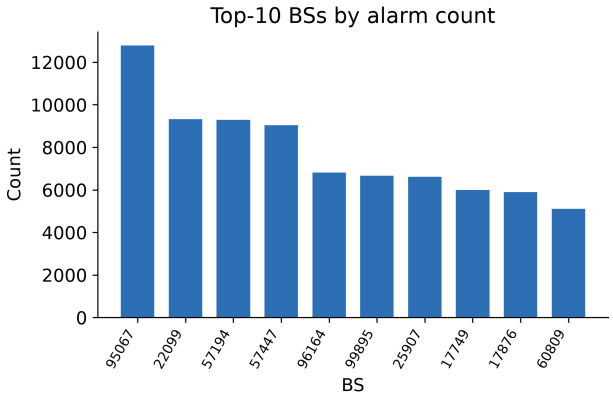
(a) Alarm severity distribution



(b) Hour-of-day (weekly) intensity of alarms



(c) Alarm counts per Base Station



(d) Top-10 BSs with highest alarm counts

Figure 4: Description of the real-world RAN alarm dataset

includes complex, detailed textual descriptions to assist engineers with fault diagnosis and troubleshooting. It is presented in various formats depending on the alarm type. In contrast, the remaining features are relatively concise and structured, consisting primarily of short string identifiers and numerical values, which are more straightforward to process and encode for subsequent analysis.

For convenience and consistency, the dataset is preprocessed following a series of standardised steps described in Section 4. Firstly, duplicate entries and records containing missing values in key fields, such as alarm severity, source ID, and timestamp, are removed to ensure data reliability. Attribute 3 (source ID) is extracted separately for BS nodes. Next, an encoding process is applied to transform the raw alarm attributes into structured numerical representations. Subsequently, the frequency of each category is calculated to generate a histogram-based representation. Finally, all features are concatenated with several extracted statistical measures to form the complete node feature vector.

4.1. Hardware Platform and Experimental Tooling

This work's experiments were conducted on a computing platform with the following hardware configuration: an Intel Core i5-12600 CPU, an NVIDIA RTX 4060 GPU, and 32 GB of RAM. Python 3 was used as the primary programming language, with support from various shell scripts. Python3 and Google Colab were used to generate the plots from the model runs' results saved as CSV and XLSX files. During the research process, the authors used the generative AI assistant Claude (Anthropic) as a software-engineering and automation aid. AI was used with human oversight to refactor and maintain the experiment codebase, automate the orchestration of multiple training and evaluation runs across different time-window configurations, and debug the code to address GPU and system errors. These tools were not used to design the study, to generate or modify experimental data or results, or to derive scientific interpretations or conclusions. All code, outputs, and analyses were reviewed, verified, and validated by the authors.

4.2. Verification of Cross-site Alarm Propagation

Understanding whether alarm events propagate across BSs is essential for justifying the graph-based formulation of the proposed model. Therefore, the following assumption is introduced (refer

to Assumption 1).

Assumption 1 (Cross-site alarm propagation). *Alarm events in RANs are not independent across BSs. Instead, the occurrence of an alarm at a source BS may increase the likelihood of subsequent alarms at other BSs within a certain temporal window.*

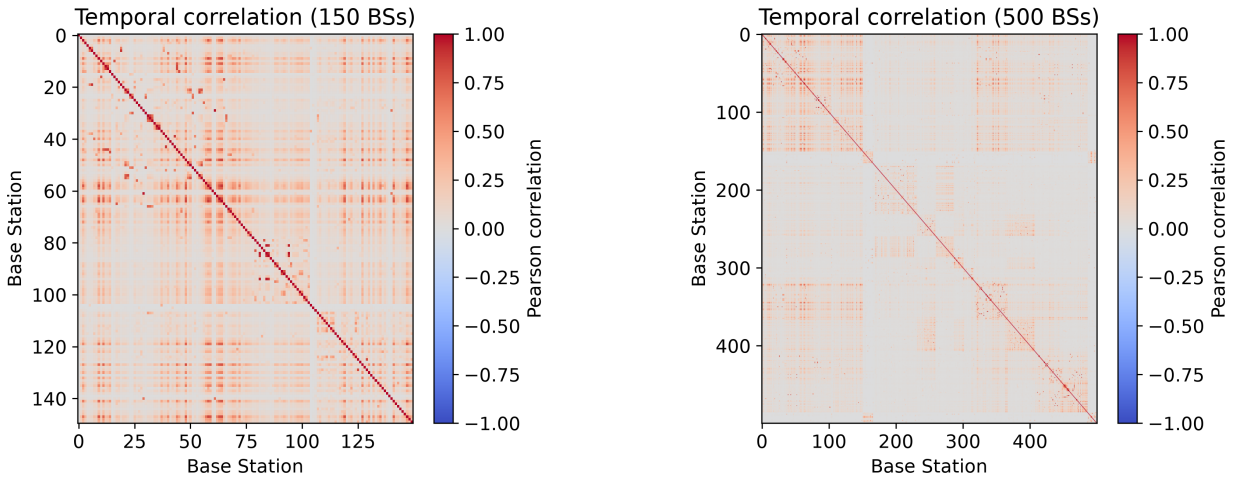
To empirically examine Assumption 1, we further investigate cross-site dependencies using correlation analysis from two complementary perspectives: (a) short-term temporal correlations and (b) long-term feature correlations.

Temporal correlation measures the degree of synchronisation between alarm occurrences across different BSs. It is computed from the time series of alarm counts within fixed time windows and captures whether alarms at two BSs tend to occur closely in time. This analysis is introduced to examine non-random temporal co-occurrence of alarms, which is a necessary condition for cross-site propagation. Feature correlation quantifies the similarity between BSs based on their long-term statistical characteristics. It is derived by correlating BS-level feature vectors. Unlike temporal correlation, feature correlation reflects the intrinsic, slowly evolving characteristics of BSs rather than short-term alarm dynamics. By jointly analysing temporal and feature correlations, we differentiate genuine alarm propagation from spurious temporal coincidences. Temporal correlation alone may arise from shared periodic patterns or global network effects and does not necessarily imply meaningful interaction. In contrast, consistent alignment between temporal and feature correlations indicates that synchronised alarms stem from inherent cross-site relationships rather than random coincidences.

For short-term temporal correlation, let D denote the number of BSs, and $\{t_{i,k}\}$ be the timestamps of alarm occurrences at BS s_i. Raw alarm records are first discretised into uniform time bins of width Δ (e.g., $\Delta = 1$ hour). Then, a time-aligned alarm count matrix $C = [c_{t,i}] \in \mathbb{R}^{T\times D}$ is obtained, where T denote the total number of time bins, and the element $c_{t,i}$ is the number of alarms observed at BS s_i in the t-th time bin. The temporal Pearson correlation coefficient for each BS pair (s_i, s_j) is computed as

$$\rho_{ij}^{\text{time}} = \frac{\sum_{t=1}^{T}(c_{t,i} - \bar{c}_i)(c_{t,j} - \bar{c}_j)}{\sqrt{\sum_{t=1}^{T}(c_{t,i} - \bar{c}_i)^2}\sqrt{\sum_{t=1}^{T}(c_{t,j} - \bar{c}_j)^2}}, \tag{28}$$

where $\bar{c}_i$ is the empirical means of alarm counts for BS s_i. Figure 5 presents the heatmap of the temporal correlation matrix. For a clearer visualisation, parts of the correlation matrix (500 BSs and 150 BSs) are zoomed in as well.



(a) 150 Base Stations

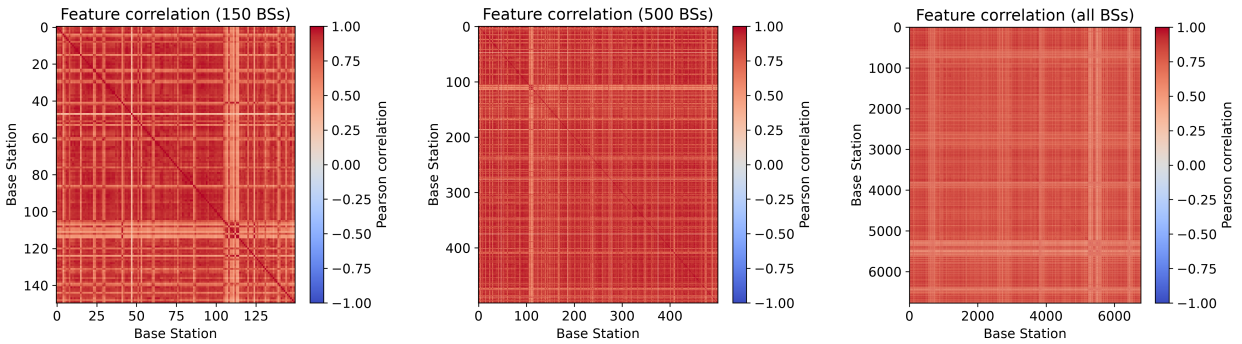
(b) 500 Base Stations

Figure 5: Heatmap of correlation matrix based on temporal alarm counts.

From a system perspective, high values $\rho_{ij}^{\text{time}} \approx 1$ suggest that the disturbances at BS s_i may trigger or coincide with subsequent alarms at BS s_j within a limited time span, whereas $\rho_{ij}^{\text{time}} \approx 0$ suggests weak or no dependence in their alarm activity. It can be observed that the inferred correlation matrix is sparse yet clearly structured. It contains several pronounced blocks and bands with high positive coefficients. These clusters correspond to groups of BSs that experience co-occurring alarm bursts, forming local communities of jointly active stations. Such a block structure is difficult to explain by purely random coincidences among independent alarm processes and instead points to underlying coupling mechanisms, which could be either direct propagation of faults or shared latent factors that synchronously affect multiple sites.

For long-term feature correlation, the BS-level feature vectors obtained in Section 4 are used to compute pairwise Pearson correlations, with the resulting matrix shown in Figure 6. The magnified views of parts of the correlation matrix (500 BSs and 150 BSs) are also provided for better visualisation.

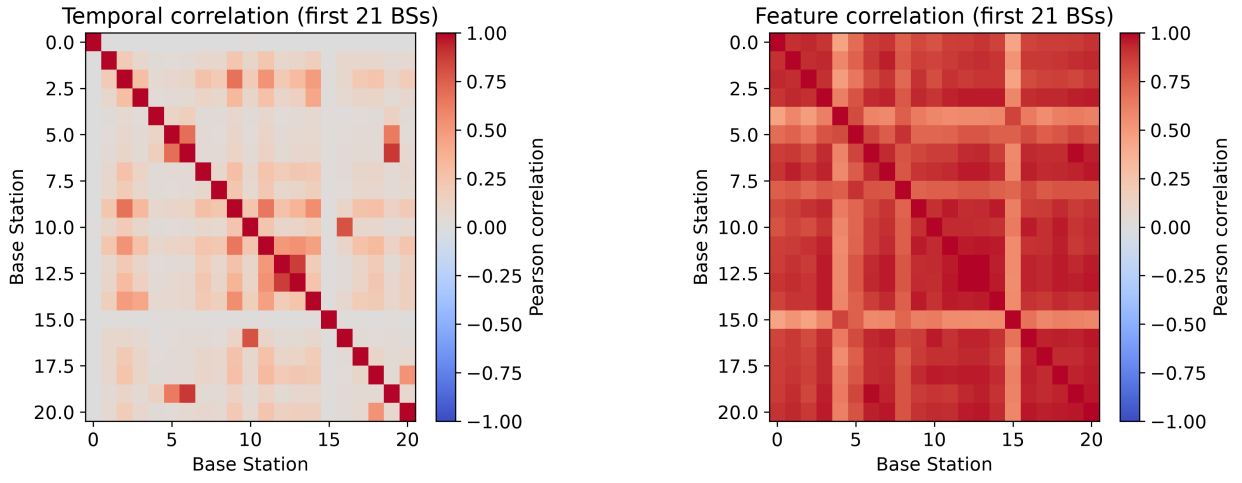
Many BS pairs show strong positive correlations in their statistical features, which indicates



(a) 150 Base Stations (b) 500 Base Stations (c) All Base Stations

Figure 6: Heatmap of correlation matrix based on BS-level statistical features.

similar long-term alarm behaviour profiles even when individual alarm events do not occur at exactly the same time. BSs within the same high correlation blocks tend to have comparable alarm frequencies, type distributions, severities, and temporal usage patterns. This pattern suggests that these BSs operate under similar conditions, configurations, or network roles, which make them susceptible to similar types of failures and alarm dynamics over extended periods. In other words, the network is not homogeneous, as it contains groups of stations that are both operationally similar and tightly coupled in terms of alarms. Further, Figure 7 strengthens these findings by directly comparing the two correlation matrices for BSs s_0 through s_{20}.



(a) Temporal correlation (b) Feature correlation

Figure 7: Heatmap of correlation matrix comparing temporal alarm counts and BS-level statistical features.

The figure shows that BS pairs with high temporal correlation also tend to exhibit high BS-level correlation. Examples include (s_4, s_5), (s_4, s_6), (s_4, s_{19}), (s_5, s_6), (s_5, s_{19}), (s_{11}, s_{12}), (s_{11}, s_{13}), and (s_{12}, s_{13}), all of which simultaneously appear as strongly correlated in both matrices. Conversely, BS pairs such as (s_0, s_4), (s_0, s_5), and (s_0, s_6) consistently show weak correlation in both dimensions. This alignment indicates that short-term dynamic synchronisation of alarms is rooted in the long-term statistical behaviours and operational characteristics. Taken together, the two correlation matrices quantify complementary aspects of dependency across the network. Strong positive relationships in both the short-term temporal domain and the long-term feature domain indicate that the cross-site dependencies observed in the alarm records are systematic rather than incidental.

These findings provide empirical support for Assumption 1 and indicate that the alarm records contain systematic cross-site dependencies that are consistent with propagation behaviour. Nevertheless, these results should not be interpreted as definitive causal proof. Some dependencies may still arise from shared latent factors, such as regional environmental conditions, common infrastructure, or coordinated maintenance activities. The purpose of this analysis is therefore to verify that the inferred graph captures non-random and propagation-consistent alarm relationships, rather than purely incidental co-occurrences.

5. Case Study: COTA-GAT on Real-world Alarm Dataset

5.1. Graph Construction

After preprocessing the raw real-world RAN data, the graph is constructed as mentioned in Section 3.2. Each BS is treated as a node, and edges are established based on alarm co-occurrence, directed from the earlier-occurring BS to the later one, with weights determined by the time-decay function. The complete co-occurrence graph is built over all alarm records, and its edges are then randomly partitioned into training, validation, and testing sets using RandomLinkSplit, with ratios of 0.8, 0.1, and 0.1, respectively. During training, only the training edges are used for message passing, while the held-out validation and testing edges serve as positive labels. Negative labels are sampled from BS pairs that do not form a qualifying co-occurrence edge under the same

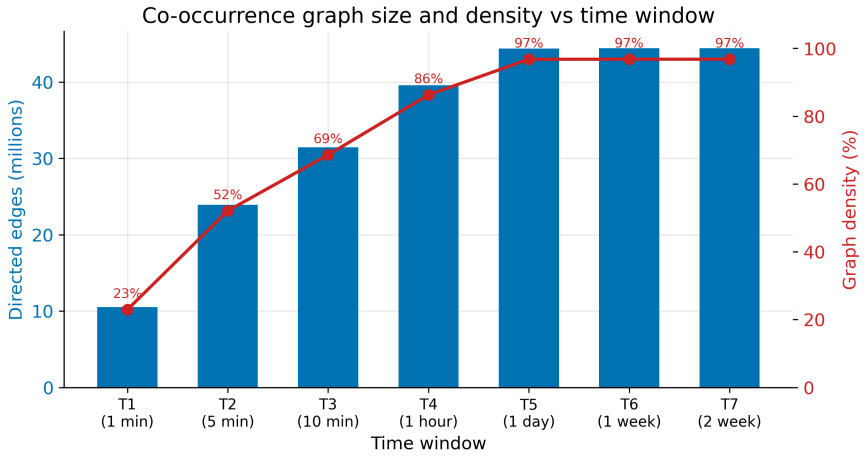


Figure 8: Co-occurrence graph size under different time windows for the real-world BT dataset.

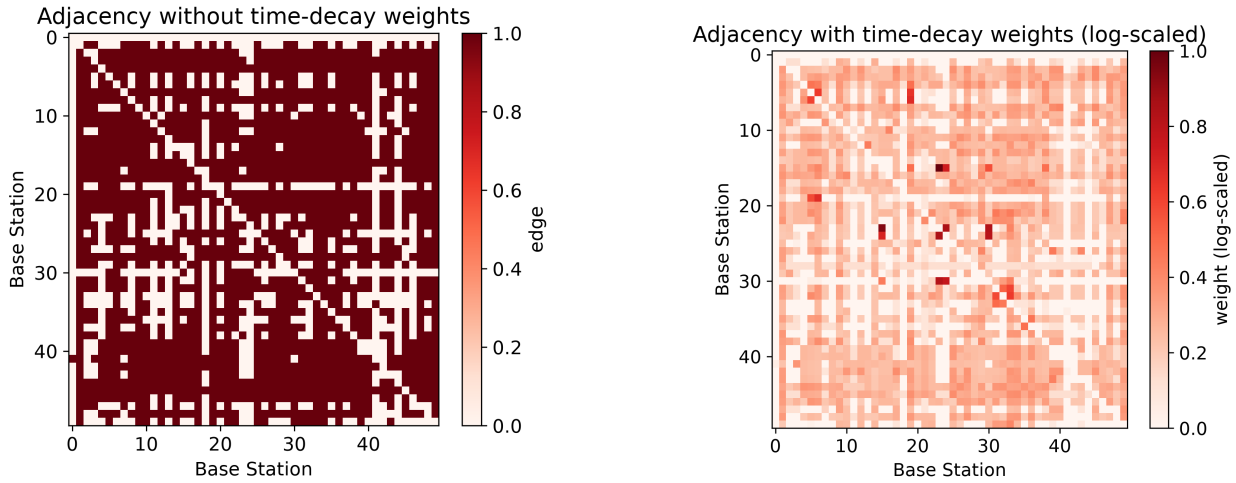
alarm type, temporal precedence, and propagation-window constraint in the corresponding split. Node features are static per-BS profiles aggregated across the dataset, so no temporal sequence information is exploited by the model.

It should be noted that the same alarm type across different BSs happening at the same time is caused by power outages in that region. Therefore, when constructing the edges between BSs, this situation is eliminated. This preprocessing step reduces one typical source of common-cause correlation. However, other exogenous factors may still remain unobserved in the alarm dataset. For this reason, the inferred links are interpreted as likely alarm-influence pathways under temporal and statistical constraints, rather than direct causal fault-transmission links.

To investigate the influence of temporal context on alarm propagation inference and evaluate the robustness of the proposed model, multiple datasets are generated using different time window sizes. Each time window defines the temporal scope within which alarm co-occurrence relationships among BSs are established. All constructed datasets contain 6773 nodes, while the number of edges varies depending on the selected time window size. By adjusting this parameter, we can analyse how different temporal granularities affect the learned dependency structures and predictive performance. The detailed configurations of each dataset are summarised in Figure 8.

Figure 9 illustrates the adjacency matrices constructed with and without time-decay weights for 50 BSs within a one-hour time window. Figure 9a shows that the adjacency is binary, which

results in saturation of the matrix due to every base-station pair co-occurring at least once during the 1-hour window. This leads to a failure in clearly distinguishing strong propagation links from incidental ones. In contrast, in Figure 9b, it is observed that time-decay weighting yields the same edge sets, thereby acquiring a heavy-tailed strength hierarchy. This results in the frequent, closely timed domination of the few co-occurrence pairs, while suppressing the rest. Essentially, Figure 9 demonstrates that the time-decay weighting restores discriminability on an otherwise saturated graph, and plays a critical role in accurately capturing the dynamics of alarm propagation.



(a) Without time-decay weights

(b) With time-decay weights

Figure 9: Directed co-occurrence adjacency among 50 base stations (1-hour window, BT dataset)

5.2. *Alarm Propagation Inference Results*

The case study experiments are conducted on the previously mentioned hardware configuration. The training hyperparameters are summarised in Table 2. To efficiently train on the large-scale graph [37], we employed a mini-batch approach using the LinkNeighborLoader. This sampler is specifically designed for link prediction methods and operates by inductively generating connected subgraphs surrounding target edges. Regarding computational scalability, COTA-GAT is trained with mini-batch neighbour sampling rather than full-batch graph attention. Specifically, LinkNeighborLoader constructs local subgraphs around target edges, so the computational and memory cost of each training step mainly depends on the batch size, sampled neighbourhood size,

and hidden feature dimension, rather than on all possible N^2 BS pairs. For a sampled subgraph with $|E|$ edges and feature dimension F, the attention computation in each layer is approximately $O(|E| \cdot F)$. In the experiments, the full BT RAN graph, comprising 6773 BS nodes and millions of constructed edges, is trained on a single GPU using this sampling strategy. The graph builds in $\simeq$ 31 s, which is measured on the 10.5M edge T1 graph. During inference, predictions can be computed either on candidate co-occurrence edges or, when a full BS-to-BS propagation matrix is required, by block-wise dot-product decoding.

Table 2: Training parameter setting for the operational real-world RAN (BT) dataset.

Parameter	Value	Parameter	Value
num neighbors	[10,5]	hidden size of GAT1	128
negative ratio	1	hidden size of GAT2	64
validation ratio	0.1	embedding size	64
testing ratio	0.1	Multi-head	2
Batch size	4096	Dropout rate	0.2
Maximum epochs	30	Learning rate	0.001
patience	8	Weight decay	0.00005

The parameter num_neighbours = [10, 5] determines the number of neighbouring nodes considered across different graph layers, thereby affecting the receptive field of the GAT model. The negative ratio is set to 1, indicating that one negative sample is paired with each positive sample during contrastive learning. The dataset is split into training, validation, and testing subsets, with validation and testing ratios both set to 0.1. The batch size is 4096, ensuring efficient computation while maintaining gradient stability. Training is performed for up to 30 epochs, with an early stopping patience of 8 epochs to prevent overfitting.

On the model side, the hidden layer dimensions for GAT1 and GAT2 are set to 128 and 64, respectively, to balance representational capacity and computational efficiency. A two-head multi-head attention mechanism is employed to enhance feature learning. The dropout rate of 0.2 helps regularise the model. Optimisation is performed using the Adam optimiser with a learning rate of 0.001 and a weight decay of 0.00005 to control overfitting and improve generalisation. In the

experiments, the input tensor is of size 6773×202×4096, where 6773 denotes the number of nodes, 202 denotes the aggregation feature length, and 4096 denotes the batch size. To evaluate alarm propagation inference performance, ranking-based metrics commonly used in link prediction [38], including the Area Under the receiver operating characteristic Curve (AUC) and Average Precision (AP), are used.

The AUC measures the probability that a randomly selected positive edge $(u, v) \in E^+$ is assigned a higher score than a randomly selected negative edge $(i, j) \in E^-$. Formally,

$$\mathrm{AUC} = \Pr\left(s(u,v) > s(i,j)\right) + \frac{1}{2}\Pr\left(s(u,v) = s(i,j)\right), \tag{29}$$

where $s(\cdot)$ denotes the link prediction score. In practice, AUC can be estimated by

$$\mathrm{AUC} = \frac{1}{|E^+||E^-|}\sum_{(u,v)\in E^+}\sum_{(i,j)\in E^-}\left[\mathbb{I}(s(u,v) > s(i,j)) + \frac{1}{2}\mathbb{I}(s(u,v) = s(i,j))\right]. \tag{30}$$

where $\mathbb{I}(\cdot)$ is the indicator function. A higher AUC indicates that the model is better at ranking existing edges ahead of non-existing ones. AP summarises the precision-recall trade-off by computing the average precision over all recall levels. It is represented by

$$\mathrm{AP} = \sum_{k=1}^{|E^+|+|E^-|} P(k)\,\Delta R(k), \tag{31}$$

where $P(k)$ is the precision at cutoff k, and $\Delta R(k)$ is the increase in recall from position $k-1$ to k. Unlike AUC, AP is more sensitive to the ranking of positive edges at higher confidence levels.

Figure 10 illustrates the validation AUC curves of COTA-GAT under the T1 time-window (T1= 1 min). It is observed that the validation AUC rises from approximately 0.56 and increases steadily over training, plateauing near 0.70 by around the 29th epoch, indicating that COTA-GAT converges on the T1 graph. The held-out test AUC closely tracks the validation curve throughout, with the early-stopping-selected checkpoint reaching a test AUC of approximately 0.703, consistent with the T1 entry in Table 3 and indicating that the model generalises without overfitting. The across-window performance is reported separately in Table 3.

To show the comparative performance of the proposed model, the COTA-GAT is also compared with some deep learning models, including Logistic Regression (LR), Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), Graph Convolutional Network (GCN),

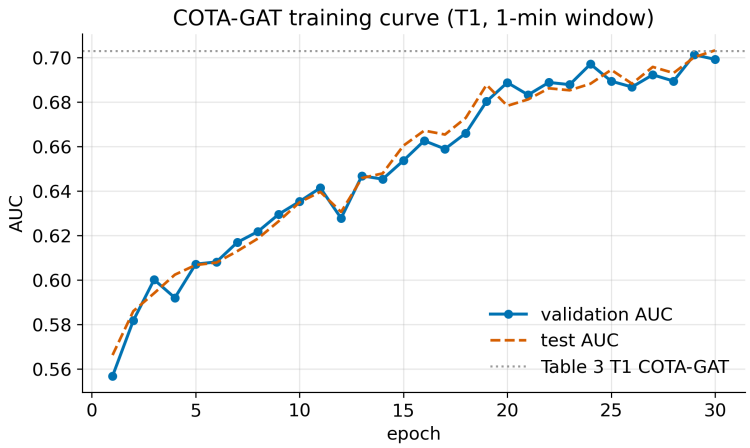


Figure 10: Test AUC curves of COTA-GAT under the T1 (1-min.) window.

GraphSAGE, and GAT. To ensure a fair comparison, all baseline models are trained and evaluated on the same training, validation, and test splits as COTA-GAT. The objective of this work is not to position COTA-GAT as a state-of-the-art temporal link-prediction model, but to demonstrate that alarm propagation can be inferred directly from co-occurrence and time-decay weighting without any physical or logical topology information. Accordingly, the baselines highlight this claim. The feature-only models (LR, SVM, XGBoost) assess whether the constructed graph encodes learnable cross-site structure beyond per-node features, while the static graph models (GCN, GraphSAGE, GAT) assess the contribution of time-aware edge weighting to the same graph. Dynamic temporal networks such as TGAT, TGN, and DySAT address a fundamentally different formulation. They assume an explicitly evolving graph given as a sequence of timestamped snapshots or interactions, and therefore do not apply to our single, topology-agnostic influence graph in which temporal information is encoded in the edge weights rather than supplied as an external structure.

To enable GPU acceleration on the large dataset, LR and SVM are implemented as single-layer linear networks, trained with cross-entropy and hinge losses, respectively, since both models correspond to linear classifiers defined by $f(x) = w^T x + b$. Optimisation is conducted using the Adam optimiser with a learning rate of 0.01. The maximum number of training epochs is set to 200, and an early stopping mechanism with a patience of 20 is employed to prevent overfitting. XGBoost is implemented using the XGBoost library with default hyperparameters, optimised via

gradient boosting on decision trees. The input edge feature for these three models is constructed by computing the Hadamard product of the features of two BSs involved in each edge. The output is the edge status, which is a binary label indicating whether an alarm propagation link exists between the two BSs. The input edge features for the aforementioned three models are computed as the Hadamard product of the aggregated feature vectors for the two BSs connected by each edge. The label is a binary vector indicating whether an alarm propagation link exists between the two BSs. GCN and GAT are implemented through two graph convolutional layers and two graph attention layers, respectively, each followed by a link prediction decoder. Since GCN and GAT are also graph-based models, their inputs are the same as those of COTA-GAT, including the constructed graph and node features. The training hyperparameters are kept consistent with those of COTA-GAT for a fair comparison.

In addition, an architectural variant of COTA-GAT named COTA-Var is designed for ablation studies. It maintains the same overall structure as COTA-GAT but removes the time-aware module, i.e., it discards the time-decay weights in the attention computation and treats all historical interactions as temporally homogeneous. It is trained with the same optimisation settings as COTA-GAT, allowing us to isolate and quantify the contribution of each component. Table 3 exhibits the comparison results of the aforementioned evaluation metrics, where T1 to T7 represent different time windows illustrated in Figure 8.

Table 3: AUC and AP performance across temporal windows. Results above 0.9 are in bold.

Method	AUC							AP						
	T1	T2	T3	T4	T5	T6	T7	T1	T2	T3	T4	T5	T6	T7
LR	0.6590	0.6666	0.6977	0.7298	0.8103	0.8141	0.8141	0.6647	0.6361	0.6634	0.6870	0.7594	0.7627	0.7627
SVM	0.6476	0.6241	0.6697	0.6892	0.7872	0.7967	0.7967	0.6527	0.6013	0.6301	0.6421	0.7345	0.7435	0.7435
XGBoost	0.8018	0.8332	0.8566	0.8982	**0.9689**	**0.9707**	**0.9707**	0.8070	0.8388	0.8674	**0.9133**	**0.9754**	**0.9766**	**0.9766**
GCN	0.6243	0.6401	0.6445	0.6363	0.6602	0.6523	0.6523	0.5870	0.5966	0.6030	0.6054	0.6085	0.6045	0.6045
GraphSAGE	0.7545	0.7820	0.7979	0.8439	0.8992	**0.9038**	**0.9038**	0.7564	0.7677	0.7848	0.8485	0.8982	**0.9031**	**0.9031**
GAT	0.6784	0.6967	0.7434	0.7760	0.8437	0.8430	0.8430	0.6636	0.6529	0.7014	0.7665	0.8217	0.8251	0.8251
COTA-GAT	0.7029	0.7366	0.7598	0.8219	**0.9109**	**0.9071**	**0.9070**	0.7074	0.7137	0.7433	0.8381	**0.9235**	**0.9205**	**0.9203**
COTA-Var	0.6980	0.7156	0.7585	0.8164	**0.9054**	**0.9065**	**0.9059**	0.7007	0.6782	0.7347	0.8303	**0.9199**	**0.9204**	**0.9197**

It can be observed that classical ML baselines (LR and SVM) show nearly identical performance and exhibit very limited sensitivity to increasing time window size. Their AUC remains between 0.65 and 0.81, and AP around 0.66 to 0.76. XGBoost shows substantial improvement over linear models, confirming that non-linear feature interactions carry meaningful information for inferring alarm occurrences. Nevertheless, because all three baselines treat each BS independently, they are unable to exploit the relational structure induced by co-occurrence windows, and therefore, cannot differentiate rapid, meaningful propagation patterns from slow or incidental alarm overlaps.

In contrast, graph-based models with adaptive neighbour weighting or sampling, i.e., GraphSAGE, GAT, and COTA-GAT, achieve markedly higher performance than the linear baselines. This confirms that the propagation graph based on the co-occurrence mechanism carries meaningful cross-site relational structure. GCN, which weights all neighbours uniformly, fails to benefit from this structure and performs comparably to the linear models, further indicating that adaptive neighbour selection or weighting is essential in this setting. XGBoost attains the highest raw scores overall. This behaviour is attributed to its strength as a feature memoriser for the reconstruction task, rather than to its ability to produce an interpretable, inductive influence graph. Amongst these graph-based approaches, COTA-GAT is slightly sub-par with GraphSAGE for shorter windows, but shows significant AUC performance improvement of 0.9109, 0.9071, and 0.9070 for longer windows T5, T6, and T7, respectively. These results demonstrate that COTA-GAT can effectively capture cross-site alarm propagation patterns among BSs and accurately infer their propagation dynamics at an early stage.

The ablation study focuses on the contribution of the time-aware attention mechanism. COTA-Var removes the time-aware module, which causes a modest but consistent drop, though it still surpasses most baselines. This indicates that the time-aware module plays a crucial role in distinguishing temporally-proximate propagation-consistent links from delayed or incidental co-occurrences. It is also noteworthy that for long-term windows (e.g., T5-T7), the performance gap between COTA-GAT and COTA-Var narrows. This phenomenon does not contradict the usefulness of the time-aware module. Instead, it reveals the boundary under which time-decay weighting is most informative.

The resulting graph is relatively sparse and structurally cleaner, and most candidate edges are more likely to represent short-lag, propagation-consistent cross-site dependencies. In this setting, the time-decay mechanism provides a strong inductive bias by assigning greater weight to rapid and recurrent alarm interactions while suppressing delayed or incidental co-occurrences. This explains the small, consistent advantage of COTA-GAT over COTA-Var under short-window settings. However, there is an interesting accuracy/complexity trade-off between these two approaches. COTA-GAT is best suited for accuracy and a time-weighted influence graph, whereas COTA-Var is better as a lighter and faster model when edge weighting becomes impractical.

When the time window is short (e.g., 1-10 min), only temporally adjacent alarms are connected, and most edges are likely to represent propagation-consistent cross-site dependencies. The resulting graph is sparse and structurally pure, enabling the attention mechanism to focus on temporally proximate alarm interactions. In this setting, the time-decay mechanism provides a strong inductive bias by assigning larger importance to rapid and recurrent alarm interactions, while suppressing delayed or incidental co-occurrences.

However, as the window enlarges (e.g., 1 day or 1 week), almost all alarms occurring within the same period become connected. The graph then becomes excessively dense, introducing numerous spurious connections due to diurnal patterns, network load fluctuations, or external disturbances. In such cases, although the time-decay function down-weights delayed co-occurrences, a fixed decay parameter τ may still assign non-negligible weights to long-delay interactions in large windows. These additional weak or common-cause correlations reduce the discriminative effect of time-aware attention, making it harder to separate propagation-consistent links from noisy statistical co-occurrences. Consequently, the learned representations at longer horizons tend to capture broader co-occurrence regularities rather than short-lag alarm influence patterns, which explains why COTA-GAT and COTA-Var show more similar performance under long time-window settings.

This observation is also consistent with the operational nature of alarm propagation in RANs. Direct alarm influence is usually a time-local phenomenon, whereas day-level or week-level windows are more likely to reflect slowly changing operational similarity or recurrent global patterns. Therefore, in practical deployment, COTA-GAT is more suitable for short-to-medium propaga-

tion windows, where the constructed graph emphasises temporally proximate and propagation-consistent alarm interactions. Long-window graphs may still be useful as auxiliary correlation graphs for analysing broad co-occurrence statistics or slow-changing dependencies, but they should not be interpreted as direct propagation graphs.

In general, non-graph models do not explicitly use relational information among BSs and are therefore less effective at modelling cross-site alarm dependencies. Standard graph-based models can benefit from neighbourhood aggregation, but their performance may degrade when the co-occurrence window is too large, and the graph contains many weakly related edges. Compared with these baselines, COTA-GAT achieves better overall performance by combining BS-level alarm features, time-decay weighted graph construction, and attention-based neighbourhood aggregation. XGBoost significantly outperforms both COTA-GAT and COTA-Var, which can be attributed to its role as a feature memoriser, as with other similar approaches. In contrast, COTA-GAT and COTA-Var show significant added benefits of being topology-agnostic and their inductive and interpretable formulation. This allows for the attention coefficients and time-decay weights to yield an inspectable, directed influence graph over base stations that feature-only baselines cannot produce.

Several observations support this interpretation. Across different time-window settings, the model exhibits stable convergence. Although the absolute AUC increases with window size, which may be attributed to denser co-occurrence graphs providing more positive structure for reconstruction, the relative advantage of the time-decay mechanism over its time-agnostic variant (COTA-Var) and the standard GAT is most pronounced at short-to-medium windows. This indicates that the learned time-decay weighting is most informative precisely where the graph is dominated by temporally proximal, propagation-consistent interactions. Further, compared with GAT and GCN baselines that use graph aggregation without explicitly encoding alarm occurrence intervals, COTA-GAT consistently achieves superior performance, highlighting the benefit of explicitly encoding temporal proximity in edge representations. Finally, the correlation analysis in Section 4.2 shows that temporal synchronisation alone does not fully account for the observed cross-site dependencies, indicating the need for adaptive edge weighting, which is facilitated by the attention mechanism.

To demonstrate the interpretability of the inferred alarm influence relationships, Figure 11 shows a case study for a randomly selected source BS s_{1024}. The figure visualises the ten highest-scoring candidate target BSs inferred by the model (grey directed arrows with high probabilities), together with ten low-scoring or inactive candidate links (grey dashed lines with probabilities approaching zero). For each target BS, we additionally illustrate the feature and temporal correlations with respect to s_{1024}, which provide complementary evidence for interpreting the inferred results.

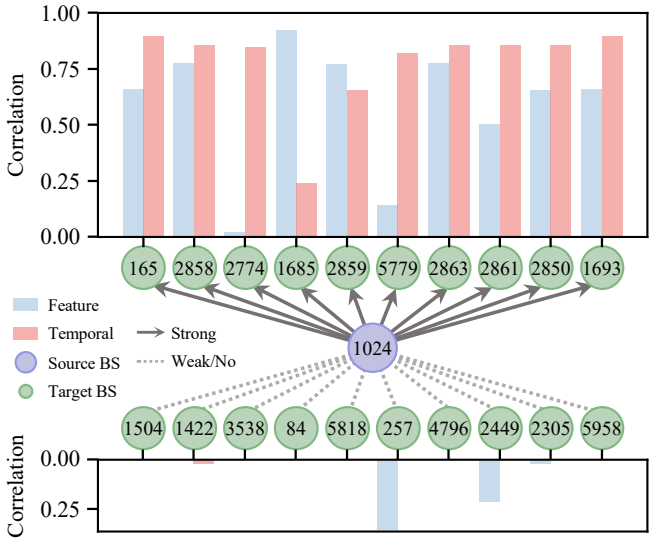


Figure 11: Visualisation of link prediction results. Strong and weak propagation targets of source BS s_{1024} inferred by COTA-GAT, along with their temporal and feature correlations.

Among the high-scoring candidate targets, most BSs exhibit both high temporal synchrony and high feature similarity with s_{1024}. These BSs display tightly coupled alarm dynamics in the short term and similar operational characteristics in the long term, which provide supporting evidence for propagation-consistent alarm influence relationships. COTA-GAT assigns higher scores to these BSs, indicating that the model can prioritise candidate links jointly supported by temporal and feature-based evidence.

Interestingly, some targets, such as s_{2774} and s_{5779}, exhibit high temporal correlation but relatively low feature similarity. COTA-GAT still assigns them relatively high alarm influence probabilities, suggesting that the inferred links are not determined solely by long-term feature similarity.

While the joint alignment of temporal and feature correlations provides stronger contextual support for cross-site alarm dependency, some high-scoring candidate links may be mainly supported by short-term temporal evidence. The proposed model captures such heterogeneous alarm influence patterns by integrating static BS-level features, time-decay-weighted edge information, and attention-based cross-site aggregation, rather than relying solely on static similarity measures.

On the weaker side, most BSs exhibit minimal correlation across both dimensions. They are naturally assigned near-zero alarm influence probability. Additionally, certain BSs such as s_{257} and s_{2449} exhibit high feature similarity but almost no temporal alignment. COTA-GAT assigns low scores to these candidates. It confirms that static feature similarity alone is insufficient to produce high alarm influence scores. This result is consistent with the design of the time-decay weighted attention mechanism, which emphasises temporally proximate cross-site alarm interactions.

Overall, the case study illustrates that COTA-GAT assigns alarm influence probabilities that closely align with the underlying temporal and feature-based relationships among BSs. The results help interpret the inferred alarm influence graph, but they should be regarded as evidence of propagation-consistent dependency patterns rather than direct proof of physical causal propagation. We further visualise the global inferred alarm influence structure by selecting the 50 most connected BSs with the highest combined in- and out-degrees, and plotting only the top 5% of alarm influence links (by probability) in a chord diagram, as shown in Figure 12. The predicted alarm influence probability is represented by the colour intensity of the ribbons.

It reveals several patterns in the model outputs. A small subset of nodes, such as s_{91338}, s_{98189}, and s_{60086}, exhibits substantially stronger and more numerous connections. By contrast, some BSs such as s_{57194} display noticeably weaker and sparser alarm influence links. Although these BSs rank among the most connected nodes in the full inferred graph, their high-probability connections within this subset remain comparatively limited. This indicates that the model consistently assigns higher probabilities to these influential sites, which may serve as highly connected nodes (by inferred degree) and potentially become operational bottlenecks or high-traffic regions within the network. The edge colour distribution is highly heterogeneous, presenting a clear contrast between dominant and weak alarm influence links. This observation suggests that the model is capable of generating well-differentiated inferences. Several local regions in the diagram display noticeably



Figure 12: Chord diagram of cross-site alarm influence patterns inferred by COTA-GAT among Top-50 BSs

denser interconnections, reflecting localised clustering effects. Meanwhile, strong links frequently extend across distant positions on the circle, indicating that high-probability alarm influence is not restricted to adjacent nodes but can widely span across the selected node set. Overall, the diagram demonstrates that COTA-GAT yields structured, sparse, and sharply discriminative alarm influence probabilities among the most influential nodes.

To further understand why certain BSs emerge as highly connected nodes in the inferred graph, we conduct a statistical analysis of their alarm characteristics. Figure 13 summarises several indicators for these BSs and compares them with the overall BS population through percentile-based analysis.

The results show that these highly connected nodes (by inferred degree) do not uniformly correspond to the most alarm-intensive sites. Instead, their dense connectivity may be associated with different alarm profile characteristics, including heterogeneous aggregation of alarm types, recur-

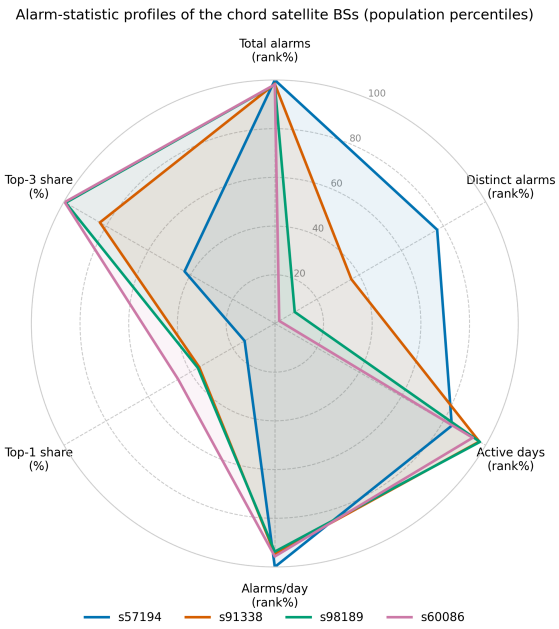


Figure 13: Radar chart comparing alarm statistics of highly connected BSs against overall population percentiles.

rent triggering patterns, and stable participation in propagation-consistent cross-site co-occurrence patterns. For instance, in Figure 12, s_{60086} exhibits extremely high alarm activity with a diverse set of alarm types, whereas s_{57194} shows moderate alarm volume but a high repetition of specific alarm types.

5.3. Sensitivity and Robustness Analysis

To further examine the robustness of the proposed framework, we conduct two additional analyses. First, we evaluate the sensitivity of COTA-GAT to different feature aggregation settings. Second, we examine its performance under increasingly imbalanced positive-to-negative edge ratios, which better reflect practical link prediction scenarios in large-scale RANs.

To evaluate feature aggregation sensitivity, i.e., whether the proposed model is sensitive to the choice of BS-level feature aggregation strategy, an additional ablation study is conducted on the real BT dataset. The co-occurrence window is fixed to t_w = 60s (T1), and the graph construction, data split, model architecture, and training settings are kept unchanged. Only the node feature representation is varied. Six feature configurations are compared — (1) the full feature set, which includes alarm-name histograms, categorical histograms, temporal histograms, and statistical sum-

maries; (2) removing statistical summaries; (3) using only the alarm-name histogram; (4) using only histogram-based features; (5) removing temporal histograms, including hour-of-day and day-of-week features; and (6) removing the alarm-name histogram. The results are shown in Figure 14. All feature configurations achieve similar performance, with AUC values between 0.645 and 0.698 and AP values between 0.798 and 0.812. The small variation indicates that COTA-GAT is robust to different choices of feature aggregation. The full feature set gives the highest value (0.698), while removing the statistics lowers it to 0.681. This suggests that the statistical summaries are useful as complementary descriptors, but they are not essential for maintaining high link-prediction accuracy.

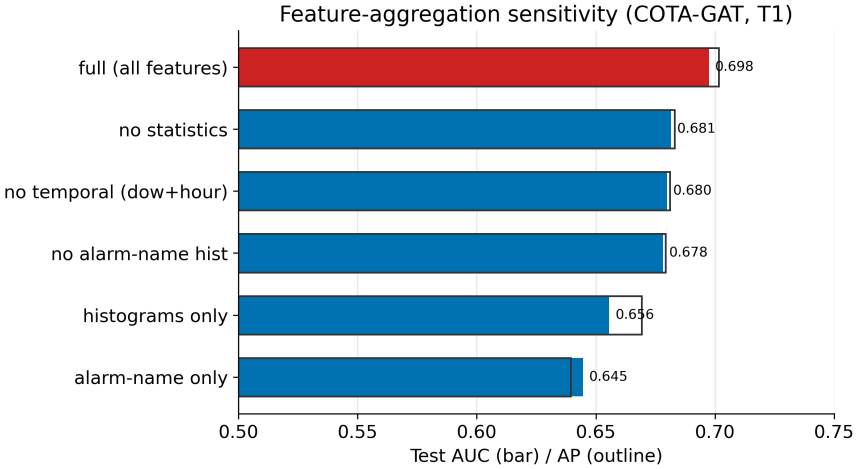


Figure 14: Feature-aggregation sensitivity analysis on the real BT dataset with $t_w = 60$ s.

Among all tested variants, removing the alarm-name histogram causes the largest performance decrease. This indicates that the distribution of alarm types is the most informative feature component for characterising BS-level alarm behaviour. Nevertheless, the performance drop remains limited, showing that COTA-GAT does not depend on a single feature block. Instead, it can exploit multiple complementary sources of information, including temporal patterns, categorical attributes, and aggregated alarm statistics.

To estimate the robustness of COTA-GAT to imbalanced negative sampling, the main experiments use a 1:1 negative sampling ratio, which is a common contrastive setting for link prediction training. However, in practical RAN operation, the number of non-propagation BS pairs is much

larger than the number of propagation-consistent links. Therefore, we further evaluate COTA-GAT under different positive-to-negative ratios, including 1:1, 1:5, 1:10, and 1:50. As shown in Figure 15, the AUC remains stable across different imbalance levels, ranging from approximately 0.69 to 0.70. This indicates that the model's ranking ability is not sensitive to the negative sampling ratio. In contrast, the absolute AP decreases as the negative ratio increases, which is expected because AP depends on the positive base rate. To better reflect performance under imbalance, we also report the AP lift over the random baseline. The AP lift increases from approximately 1.4× under the 1:1 setting to 4.4× under the 1:50 setting, indicating that COTA-GAT still ranks propagation-consistent links substantially above random expectation under heavily imbalanced conditions.

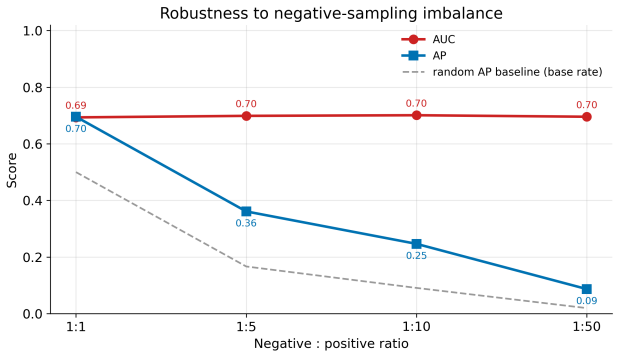


Figure 15: Evaluation of negative sample imbalance on the real BT dataset

6. COTA-GAT Validation on Controlled Simulator Data

Although the real-world operational RAN dataset provides large-scale evidence of cross-site alarm dependencies, it does not contain manually verified propagation labels or confidential physical/logical topology information (as described in Section 4). Therefore, evaluation on the operational data alone cannot fully eliminate the concern that the model is learning a self-referential co-occurrence target. To provide an external validation of the proposed graph construction, we further conducted a controlled NS-3 LTE RAN simulation in which the ground-truth propagation process is known and easily traceable from the generated logs. Table 4 shows the adjusted parameters on the COTA-GAT architecture to accommodate the significantly smaller dataset.

Table 4: Training parameter setting for the NS-3 simulator validation.

Parameter	Value	Parameter	Value
num neighbors	[15,10,5]	hidden size of GAT1	128
negative ratio	1	hidden size of GAT2	64
validation ratio	0.1	embedding size	128
testing ratio	0.1	Multi-head	2
Batch size	2048	Dropout rate	0.2
Maximum epochs	60	Learning rate	0.001
patience	25	Weight decay	0.00005
time window	30 s		

In the simulation, a small LTE RAN topology containing 21 BSs was constructed. Single-element faults, including backhaul, X2, and radio-related faults, were injected into the simulated network. For each injected fault, the resulting alarm cascade was recorded in accordance with the known fault propagation process in the simulator. This produced 42 ground-truth causal propagation edges. Importantly, these causal edges were generated independently of the proposed temporal co-occurrence heuristic and were not used to construct the co-occurrence graph.

We then applied the same co-occurrence construction to the simulated alarm logs and evaluated whether the resulting candidate edges could recover and rank the independently known causal edges. The results are shown in Figures 16, 17, and 18. First, in Figure 16, the construction achieved 100% recall of the ground-truth causal edges, indicating that all true causal pairs were covered by the co-occurrence candidate set. However, binary co-occurrence alone did not distinguish true causal edges from non-causal co-occurring pairs, yielding an AUC of 0.50, which is equivalent to chance-level ranking. This result confirms that mere co-occurrence is insufficient as evidence of propagation.

In contrast, when candidate edges were ranked using recurrence count or the proposed time-decay cumulative weight, true causal edges were ranked above non-causal co-occurrences. The time-decay weighting achieved an AUC of approximately 0.70 under both 30 s and 60 s propagation windows. A label-permutation test was further conducted by randomly shuffling the causal

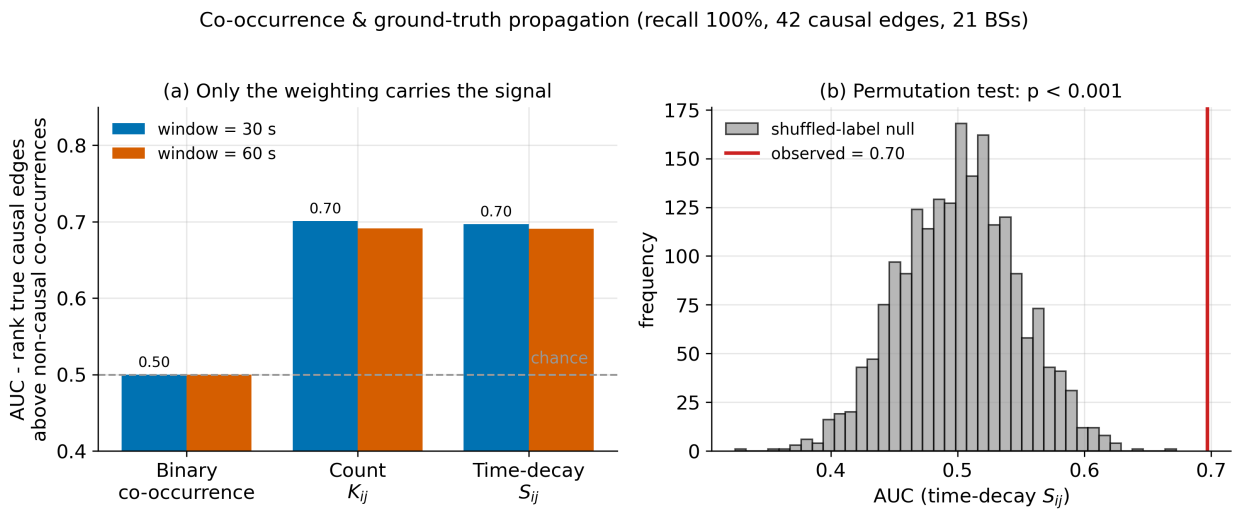
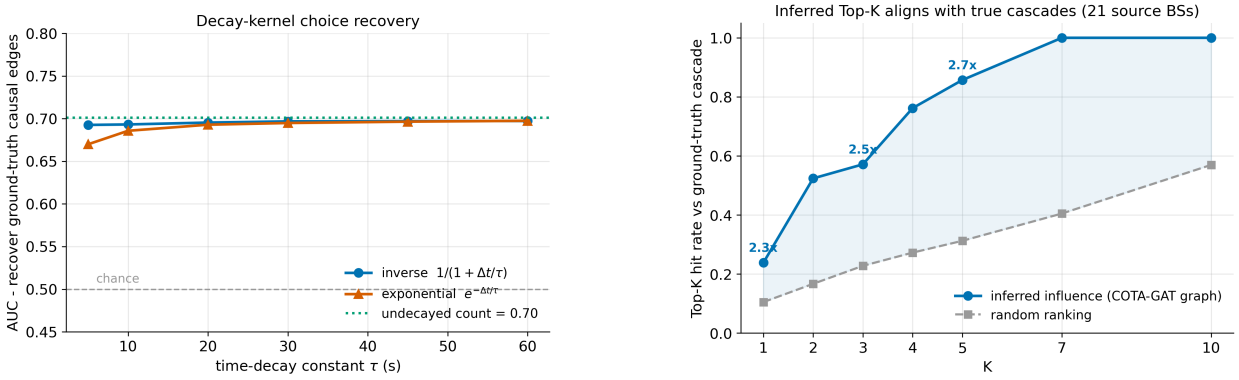


Figure 16: Ground-truth propagation validation using a NS-3-based RAN simulator

labels while preserving the constructed co-occurrence scores. The observed AUC lay far in the tail of the null distribution, with p < 0.001. This demonstrates that the ranking ability of the time-decay construction is unlikely to be caused by random label alignment.

We further tested whether the ground-truth recovery depends on the specific decay kernel. Figure 17a compares the inverse proportional kernel $1/(1 + \Delta t/\tau)$ and the exponential kernel $\exp(-\Delta t/\tau)$ over different values of τ. Both kernels produce stable AUC values around 0.69-0.70, substantially above the chance level of 0.50. This result indicates that the recovery of propagation-relevant edges is robust to the choice of decay kernel.



(a) Comparison of decay kernel choice

(b) Inferred Top-K alarms for 21 BSs

Figure 17: Validation of COTA-GAT on NS-3-based RAN simulated data

To further evaluate whether the inferred influence ranking aligns with independently known propagation behaviour, we also conducted a Top-K validation on the NS-3 ground-truth cascades. For each source BS, candidate target BSs were ranked by the inferred influence scores from the constructed COTA-GAT graph. A hit is counted if at least one true cascade target of the source BS appears within the Top-K ranked candidates.

As shown in Figure 17b, the inferred ranking aligns well with the ground-truth cascades. The Top-1, Top-3, and Top-5 hit rates reach 24%, 57%, and 86%, respectively, compared with 11%, 23%, and 31% under random ranking. This corresponds to approximately 2.3–2.7× improvement over the random baseline. When K is increased to 7 and 10, the inferred ranking covers all source BSs with true cascade targets, whereas the random baseline remains substantially lower.

We further examine whether the topology-agnostic influence links inferred by the proposed method are related to the underlying physical topology. This analysis cannot be conducted on the real BT dataset because the physical RAN topology and connection records are unavailable under confidentiality constraints. Therefore, we perform this validation on the NS-3 simulator dataset, where the physical topology is known. Figure 18 compares the inferred influence links with the physical topology.

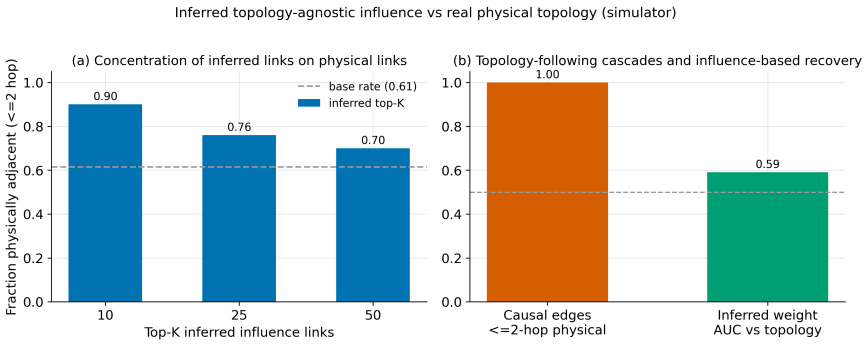


Figure 18: Comparison of the topology-agnostic inference with real topology on an NS-3-generated RAN alarm dataset

In Figure 18a, the top-ranked inferred links are strongly enriched around physically close BS pairs. Specifically, 90% of the top-10 inferred influence links connect BS pairs within two physical hops, compared with a 61% base rate. The fractions for the top-25 and top-50 inferred links are 76% and 70%, respectively, which remain above the base rate. This indicates that the strongest

topology-agnostic influence links tend to concentrate around the physical topology, even though no topology information is provided to the model.

Figure 18b further shows that all ground-truth fault-cascade edges in the simulator occur within two physical hops, confirming that the simulated cascades are topology-driven. The inferred edge weights also show a positive, although moderate, ability to distinguish physically close BS pairs from non-close pairs, with an AUC of 0.59. This result suggests that the proposed influence graph captures the propagation structure related to physical topology, but is not simply a reconstruction of the physical topology.

Overall, the NS-3 validation provides three complementary pieces of evidence. First, the AUC and permutation-test results show that the proposed co-occurrence and time-decay construction can rank ground-truth propagation edges above non-causal co-occurrences. Second, the Top-K analysis shows that the highest-ranked inferred links are strongly enriched with true cascade targets, rather than being arbitrary outputs of the co-occurrence heuristic. Third, the topology-overlap analysis indicates that the inferred topology-agnostic influence links concentrate around physically close BS pairs, even though no physical topology information is used as model input.

Taken together, these results provide external evidence that the proposed graph construction captures propagation-relevant and physical-topology-related influence structure under controlled ground-truth conditions. Nevertheless, this does not imply that all links inferred from the real operational RAN dataset are definitive physical causal links or verified physical routes. The real-data evaluation should therefore be interpreted as propagation-consistent influence prediction on observational alarm logs. Non-adjacent inferred links in real centralised or virtualised RAN deployments may correspond either to residual noisy correlations or to genuine logical coupling through shared baseband resources, backhaul, or network slices.

7. Conclusion

This paper addresses the problem of cross-site alarm influence inference from the perspective of intelligent telecommunication infrastructure management. By modelling alarm interactions as a topology-agnostic and time-aware graph learning task, the proposed COTA-GAT framework

transforms heterogeneous alarm streams into structured representations for network-level reliability analysis. With reference to the three research questions in Section 1, COTA-GAT introduces several novel components. To deal with RQ1, a unified feature encoding and aggregation method is introduced to obtain BS-level representations. Subsequently, a consolidated directed alarm influence graph, constructed by aggregating per-alarm co-occurrence graphs within predefined time windows addresses RQ2. Finally, the approach of assigning a time-decay weight to capture both the intensity and temporal proximity of cross-site alarm interactions to each edge addresses RQ3. The time-aware graph attention layers further allow the model to learn heterogeneous inter-BS dependencies by combining BS-level features with delay-sensitive edge information.

To support the modelling assumption of cross-site alarm dependency, a comprehensive correlation analysis was conducted from both short-term temporal synchronisation and long-term feature similarity perspectives. The results reveal non-random, structurally meaningful dependencies among BSs, thereby providing empirical support for the propagation-consistent graph formulation. The use of real-world BT alarm data establishes operational relevance but lacks ground truth to rule out circular co-occurrence, while the NS-3 simulator provides verifiable causal and topology ground truth but not real-world scale. Therefore, both are used complementarily, where one provides validity and the other realism.

COTA-GAT outperforms the linear baselines (LR, SVM) and most graph-based baselines at medium-to-long windows, and while a strong feature-memorising baseline (XGBoost) attains higher raw scores on this reconstruction task, COTA-GAT provides an interpretable, topology-agnostic influence graph that feature-only models cannot. The results also show that short co-occurrence windows are better suited to capturing delay-sensitive alarm influence patterns, whereas longer windows tend to introduce broader co-occurrence regularities and potential common-cause correlations. Detailed case studies further indicate that COTA-GAT effectively distinguishes dominant alarm influence pathways from weak or spurious similarities. This further illustrates the interpretability of the inferred alarm influence graph.

From an industrial perspective, the proposed approach provides a data-driven tool for prioritising candidate BSs in fault investigation and supporting proactive reliability management in large-scale communication infrastructures. A limitation of this study is that the proposed frame-

work is based on observational alarm logs and does not include explicit exogenous variables, such as weather records, maintenance schedules, power-system events, or detailed physical topology. As a result, the inferred graph cannot fully distinguish direct causal propagation from all possible common-cause correlations. Future work will integrate external topology and operational information, causal validation strategies, and dynamic graph formulations to better distinguish true fault propagation from shared environmental or operational disturbances and to support online modelling of evolving alarm influence pathways.

References

[1] J. Chen, X. Liang, J. Xue, Y. Sun, H. Zhou, X. Shen, Evolution of RAN architectures toward 6G: Motivation, development, and enabling technologies, IEEE Communications Surveys and Tutorials 26 (2024) 1950–1988. doi:10.1109/COMST.2024.3388511.

[2] A. Mukherjee, A. Chandra, M. Herrera, L. Li, H. P. Indiran, A. Parekh, A. K. Parlikad, Raptor: Resilience-aware prediction and tracking of operational risks from alarm information flows, Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability (2026) 1748006X261431895. doi:10.1177/1748006X261431895.

[3] M. Sweney, Vodafone outage: Thousands of broadband and mobile users report problems, The Guardian (2025).

[4] L. Li, M. Herrera, A. Mukherjee, G. Zheng, C. Chen, M. Dhada, H. Brice, A. Parekh, A. K. Parlikad, Predictive alarm models for improving radio access network robustness, Expert Systems with Applications 259 (2025) 125312. doi:10.1016/j.eswa.2024.125312.

[5] M. Polese, L. Bonati, S. D'Oro, S. Basagni, T. Melodia, Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges, IEEE Communications Surveys and Tutorials 25 (2023) 1376–1411. doi:10.1109/COMST.2023.3239220.

[6] Ž. Deljac, M. Randić, G. Krčelić, Early detection of network element outages based on customer trouble calls, Decision Support Systems 73 (2015) 57–73. doi:10.1016/j.dss.2015.02.014.

[7] J. Wang, C. He, Y. Liu, G. Tian, I. Peng, J. Xing, X. Ruan, H. Xie, F. L. Wang, Efficient alarm behavior analytics for telecom networks, Information Sciences 402 (2017) 1–14. doi:10.1016/j.ins.2017.03.020.

[8] P.-H. Wu, W.-C. Peng, M.-S. Chen, Mining sequential alarm patterns in a telecommunication database, in: W. Jonker (Ed.), Databases in Telecommunications II, Springer, Berlin, Heidelberg, 2001, pp. 37–51. doi:10.1007/3-540-45432-2_4.

[9] G. Liang, J. Hu, K. Yang, S. Song, T. Liu, N. Xie, Y. Yu, Data augmentation for predictive digital twin channel: Learning multi-domain correlations by convolutional TimeGAN, IEEE Journal of Selected Topics in Signal Processing 18 (2024) 18–33. doi:10.1109/JSTSP.2024.3358980.

[10] M. Boldt, S. Ickin, A. Borg, V. Kulyk, J. Gustafsson, Alarm prediction in cellular base stations using data-driven methods, IEEE Transactions on Network and Service Management 18 (2021) 1925–1933. doi:10.1109/TNSM.2021.3052093.

[11] A. J. García, M. Toril, P. Oliver, S. Luna-Ramírez, M. Ortiz, Automatic alarm prioritization by data mining for fault management in cellular networks, Expert Systems with Applications 158 (2020) 113526. doi:10.1016/j.eswa.2020.113526.

[12] M. W. Asres, M. A. Mengistu, P. Castrogiovanni, L. Bottaccioli, E. Macii, E. Patti, A. Acquaviva, Supporting telecommunication alarm management system with trouble ticket prediction, IEEE Transactions on Industrial Informatics 17 (2021) 1459–1469. doi:10.1109/TII.2020.2996942.

[13] D. Zhao, T. Wang, F. Chu, Deep convolutional neural network based planet bearing fault classification, Computers in Industry 107 (2019) 59–66. doi:10.1016/j.compind.2019.02.001.

[14] Y. Pan, C. Yi, X. Liao, Y. Lin, P. He, J. Lin, Detecting unexpected faults of high-speed train axlebox bearings using multi-node network structure, Measurement 246 (2025) 116641. doi:10.1016/j.measurement.2024.116641.

[15] W. Zhang, C. Yi, L. Yan, Q. Liu, Q. Zhou, P. He, L. Ran, Y. Lin, Dictionary learning method for cyclostationarity maximization and its application to bearing fault feature extraction, IEEE Transactions on Instrumentation and Measurement 73 (2024) 1–13. doi:10.1109/TIM.2024.3484531.

[16] B. Yang, W. Xu, F. Bi, Y. Zhang, L. Kang, L. Yi, Multi-scale neighborhood query graph convolutional network for multi-defect location in CFRP laminates, Computers in Industry 153 (2023) 104015. doi:10.1016/j.compind.2023.104015.

[17] W. Zhang, J. Joseph, Y. Yin, L. Xie, T. Furuhata, S. Yamakawa, K. Shimada, L. B. Kara, Component segmentation of engineering drawings using graph convolutional networks, Computers in Industry 147 (2023) 103885. doi:10.1016/j.compind.2023.103885.

[18] Z. Xu, Z. Wang, C. Gao, K. Zhang, J. Lv, J. Wang, L. Liu, A digital twin system for centrifugal pump fault diagnosis driven by transfer learning based on graph convolutional neural networks, Computers in Industry 163 (2024) 104155. doi:10.1016/j.compind.2024.104155.

[19] S. Mukherjee, O. Coudert, C. Beard, An open approach to autonomous ran fault management, IEEE Wireless Communications 30 (2023) 96–102. doi:10.1109/MWC.004.2200244.

[20] A. Mukherjee, M. Herrera, H. P. Indiran, L. Li, H. Brice, A. Parekh, A. K. Parlikad, Alarm webs: A framework for decoding RAN alarm dynamics, IFAC-PapersOnLine 58 (2024) 103–108. doi:10.1016/j.ifacol.2024.08.057.

[21] W. Jiang, Y. Bai, APGNN : Alarm propagation graph neural network for fault detection and alarm root cause analysis, Computer Networks 220 (2023) 109485. doi:10.1016/j.comnet.2022.109485.

[22] Y. Yang, S. Yang, C. Zhao, Z. Xu, TelOps: AI-driven operations and maintenance for telecommunication networks, IEEE Communications Magazine 62 (2024) 104–110. doi:10.1109/MCOM.003.2300055.

[23] A. Longa, V. Lachi, G. Santin, M. Bianchini, B. Lepri, P. Lio, F. Scarselli, A. Passerini, Graph neural networks for temporal graphs: State of the art, open challenges, and opportunities, 2023. doi:10.48550/arXiv.2302.01018. arXiv:2302.01018.

[24] S. M. Kazemi, R. Goel, K. Jain, I. Kobyzev, A. Sethi, P. Forsyth, P. Poupart, Representation learning for dynamic graphs: A survey, Journal of Machine Learning Research 21 (2020) 1–73. doi:10.5555/3455716.3455786.

[25] E. Rossi, B. Chamberlain, F. Frasca, D. Eynard, F. Monti, M. Bronstein, Temporal graph networks for deep learning on dynamic graphs, 2020. doi:10.48550/arXiv.2006.10637. arXiv:2006.10637.

[26] D. Huang, F. Lei, Temporal group-aware graph diffusion networks for dynamic link prediction, Information Processing & Management 60 (2023) 103292. doi:10.1016/j.ipm.2023.103292.

[27] H. Wang, Z. Cui, R. Liu, L. Fang, Y. Sha, A multi-type transferable method for missing link prediction in heterogeneous social networks, IEEE Transactions on Knowledge and Data Engineering 35 (2023) 10981–10991. doi:10.1109/TKDE.2022.3233481.

[28] Q. Li, Y. Ji, L. Sun, N. Zhang, T. Yang, Anomaly detection for industrial time series in process industry using informed machine learning with graph attention networks, Computers in Industry 176 (2026) 104445. doi:10.1016/j.compind.2026.104445.

[29] F. Meng, J. Xie, J. Sun, C. Xu, Y. Zeng, X. Wang, T. Jia, S. Huang, Y. Deng, Y. Hu, Spreading dynamics of information on online social networks, Proceedings of the National Academy of Sciences 122 (2025) e2410227122. doi:10.1073/pnas.2410227122.

[30] N. Guo, P. Guo, H. Dong, J. Zhao, Q. Han, Modeling and analysis of cascading failures in projects: A complex network approach, Computers & Industrial Engineering 127 (2019) 1–7. doi:10.1016/j.cie.2018.11.051.

[31] X. Guo, Q. Du, Y. Li, Y. Zhou, Y. Wang, Y. Huang, B. Martinez-Pastor, Cascading failure and recovery of metro–bus double-layer network considering recovery propagation, Transportation Research Part D: Transport and Environment 122 (2023) 103861. doi:10.1016/j.trd.2023.103861.

[32] S. He, Y. Zhou, Y. Zhou, J. Wu, M. Zheng, T. Liu, Fast identification of vulnerable set for cascading failure analysis in power grid, IEEE Transactions on Industrial Informatics 19 (2023) 5645–5655. doi:10.1109/TII.2022.3202917.

[33] F. Bianchi, E. Filippi-Mazzola, A. Lomi, E. C. Wit, Relational event modeling, Annual Review of Statistics and Its Application 11 (2024) 297–319. doi:10.1146/annurev-statistics-040722-060248.

[34] M. Jin, H. Y. Koh, Q. Wen, D. Zambon, C. Alippi, G. I. Webb, I. King, S. Pan, A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 46 (2024) 10466–10485. doi:10.1109/TPAMI.2024.3443141.

[35] X. Zhou, J. Wu, W. Liang, K. I.-K. Wang, Z. Yan, L. T. Yang, Q. Jin, Reconstructed graph neural network with knowledge distillation for lightweight anomaly detection, IEEE Transactions on Neural Networks and Learning Systems 35 (2024) 11817–11828. doi:10.1109/TNNLS.2024.3389714.

[36] K. Murphy, A. Lavignotte, C. Lepers, Fault prediction for heterogeneous telecommunication networks using machine learning: A survey, IEEE Transactions on Network and Service Management 21 (2024) 2515–2538. doi:10.1109/TNSM.2023.3340351.

[37] W. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, in: Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017. doi:10.5555/3294771.3294869.

[38] L. Lü, T. Zhou, Link prediction in complex networks: A survey, Physica A: Statistical

Mechanics and its Applications 390 (2011) 1150–1170. doi:10.1016/j.physa.2010.11.027.