iJRASET
International Journal For Research in
Applied Science and Engineering Technology

Volume: 13 Issue: VIII Month of publication: August 2025

DOI: https://doi.org/10.22214/ijraset.2025.73727

Enhancing Alert Prioritization in Enterprise Security Operations via Factor Graph Modeling

Mallipudi Beulah
PG Scholar, Dept. of Computer Science and Engineering University College of Engineering Kakinada, JNTUK Kakinada, A.P., India - 533003

Abstract: Advanced Persistent Threats (APTs) remain one of the most challenging problems in enterprise cybersecurity due to their stealthy, multi-stage nature and ability to evade traditional detection methods. While existing systems like RANK utilize factor graph-based scoring to correlate alerts into actionable incident graphs, they rely on static, manually defined tactic transition weights, limiting adaptability to emerging attack pat- terns. In this paper, we propose an enhancement to the RANK architecture by introducing an adaptive scoring mechanism based on pairwise factor graphs. Our approach dynamically models contextual and temporal relationships between MITRE ATT&CK tactics, enabling more accurate and flexible incident evaluation. Experiments conducted on two enterprise-scale datasets show that our method improves detection precision and F1 scores while maintaining compatibility with real-world alert streams. These results demonstrate the potential of adaptive, context- aware scoring in reducing false positives and enhancing threat detection capabilities in modern Security Operations Centers.

Index Terms: Advanced Persistent Threats, Cybersecurity, Inc ident Graph, MITRE ATT&CK, Alert Correlation, Adaptive Scoring, Enterprise Networks, Threat Detection, Security Oper- ations Center (SOC)

I. INTRODUCTION

Advanced Persistent Threats (APTs) have emerged as a critical concern in enterprise cybersecurity due to their stealthy, multi-stage nature and long dwell times [1]. These threats typically involve a sequence of coordinated steps—such as reconnaissance, initial access, lateral movement, data exfiltra- tion, and impact—executed over extended periods to evade detection [2], [3]. Traditional intrusion detection systems (IDS) and rule-based mechanisms, while valuable, often generate an overwhelming number of alerts, leading to alert fatigue and missed critical threats [4].

To address these challenges, recent research has explored the integration of artificial intelligence (AI) with cybersecu- rity operations. One such notable system is RANK—an AI- assisted, end-to-end architecture for detecting APTs using alert templating, graph construction, partitioning, and probabilistic incident scoring through factor graphs [5]. RANK effectively reduces tens of thousands of raw alerts into a smaller number of high-confidence security incidents that are more actionable for human analysts.

Despite its effectiveness, RANK’s current implementation uses a fixed, manually defined MITRE tactic transition matrix and static factor weights in its scoring process. This static design may limit the system’s adaptability to new or evolving threat behaviors, especially in environments where attacker tactics deviate from known patterns or where simultaneous multi-vector attacks occur [6].

In this work, we propose an extension to the RANK architecture by introducing an adaptive pairwise factor graph- based scoring mechanism.

Our approach dynamically captures inter-tactic relationships and contextual dependencies within the alert graph, allowing the system to better represent nuanced attack patterns. This modification enables improved scoring precision while maintaining RANK’s ability to summarize complex alert sets into concise, meaningful incident graphs.

The remainder of this paper is organized as follows: Section II presents related work. Section III outlines the proposed methodology. Section IV discusses implementation and dataset specifics. Section V reports experimental results and analysis. Section VI explores future research directions, and Section VII concludes the paper.

II. RELATED WORK

The detection of Advanced Persistent Threats (APTs) has attracted substantial research interest due to the covert, long- term nature of such attacks. These threats typically unfold over multiple stages, often blending with legitimate activity, making them difficult to detect using traditional security systems.

A. Limitations of Conventional Intrusion Detection

Signature-based intrusion detection systems (IDS) like Snort [7] and Suricata [8] are commonly deployed in enterprise environments. These systems match observed activity against known threat patterns, offering low false positives for known attacks [4]. However, they are less effective against novel attack variants or stealthy multi-stage threats. Moreover, their reliance on static rules makes them brittle in dynamic or heterogeneous environments.

B. Alert Correlation and Attack Graphs

To address the overwhelming volume of alerts produced by IDS, researchers have proposed alert correlation techniques.

These methods aim to combine low-level alerts into meaning- ful attack chains [9]. Julisch [10] introduced alert clustering to reduce redundancy, while others employed rule-based engines or temporal correlations.

Attack graphs model causal relationships between security events, enabling visualization of multi-step attacks. However, most early graph-based systems relied on predefined rules or static mappings, which limited adaptability to new attack paths. Additionally, scalability issues and high noise sensitivity remain challenges when applied to real-world enterprise data.

C. Tactic-Based Modeling with MITRE ATT&CK

The MITRE ATT&CK framework [3] has become widely adopted as a structured taxonomy for describing adversarial behavior. By organizing attacker actions into tactics and tech- niques, it enables semantic interpretation of alerts. Several recent systems leverage ATT&CK to label and categorize events during correlation and threat modeling [2].

D. AI-Assisted Scoring with Factor Graphs

One of the more recent advancements in APT detection is the RANK architecture [5], which incorporates AI-assisted incident scoring. RANK aggregates alerts into graphs, parti- tions them into incident subgraphs, and uses a factor graph model to assign likelihood scores based on observed MITRE tactics. This probabilistic scoring helps reduce the number of irrelevant alerts presented to analysts, enhancing decision- making.

However, a key limitation of RANK lies in its use of a static tactic transition matrix and fixed edge weights. These parameters are manually tuned and do not adapt to changes in attack behavior over time. In dynamic enterprise environments with polymorphic attack techniques, such rigidity may reduce detection accuracy and delay threat recognition [6].

E. Positioning Our Work

Our work builds upon the foundations of RANK but en- hances the incident scoring stage by introducing a pairwise factor graph model with context-sensitive edge weighting. Rather than relying on static tactic transitions, our system dy- namically computes edge weights based on co-occurrence and temporal patterns of tactics observed in real-time alert streams. This adaptive scoring strategy offers improved flexibility and responsiveness to evolving attack strategies, while preserving the interpretability benefits of probabilistic graphical models.

TABLE I
COMPARISON OF RELATED APT DETECTION APPROACHES

Method	Uses Graph	Scoring Mechanism	Adaptive Capability
Julisch (2003) [10]	✗	Clustering	✗
Dantu et al. (2004) [9]	✓	Rule-based	✗
RANK (2023) [5]	✓	Static Factor Graph	✗
Ours (this paper)	✓	Adaptive Factor Graph	✓

III. PROPOSED METHOD

This section presents our extension to the RANK archi- tecture by introducing a dynamic and context-aware scoring mechanism using a pairwise factor graph. While the original RANK system utilizes a fixed transition matrix to score incidents based on tactic co-occurrence, our method introduces adaptiveness by learning inter-tactic relationships from the incident context and recent historical data.

A. System Architecture Overview

We preserve the four-stage pipeline introduced in RANK:

1) Alert Templating and Merging
2) Alert Graph Construction
3) Graph Partitioning
4) Incident Scoring

Our enhancement is entirely focused on the fourth stage—incident scoring—where a novel, pairwise factor graph is constructed for each incident. Unlike the static structure used in RANK, our approach dynamically builds the factor graph based on the alert context and evolves over time with feedback.

Figure 1 illustrates the revised scoring stage within the overall workflow.

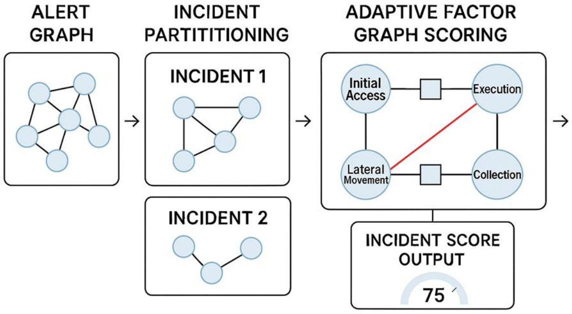

Fig. 1. Extended architecture with adaptive pairwise factor graph for incident scoring.

B. Pairwise Factor Graph Construction

Given a partitioned incident subgraph, we construct a bi- partite factor graph $G_f = (V, F, E)$, where:

1) V is a set of variable nodes, each representing a MITRE tactic Ti
2) F is a set of factor nodes modeling relationships between tactic pairs (Ti, Tj)
3) E contains the edges connecting variable nodes to factor nodes

The intuition is that certain tactics co-occur more frequently, or appear in predictable sequences, which reflects underlying attacker behavior. Our model captures these dependencies by dynamically adjusting edge weights based on observed characteristics of the current incident.

C. Contextual Weight Assignment

Each factor node is assigned a context-dependent weight w_{ij} for the connected tactic pair (T_i, T_j). This weight reflects the strength of the relationship and is computed using a weighted combination of:

1) Temporal Proximity: Shorter time gaps between tactic occurrences increase the likelihood of a real correlation.
2) Historical Co-occurrence: Frequently co-occurring tac- tic pairs in prior labeled incidents are assigned higher weights.
3) Alert Source Similarity: If both tactics originate from similar systems, users, or tools, the correlation weight increases.

This adaptiveness allows the system to shift its scoring emphasis based on evolving attack strategies without retraining or manual updates.

D. Inference and Scoring

Once the factor graph is constructed, we apply the sum- product algorithm [11] for belief propagation to compute marginal probabilities $P(T_i)$ for each tactic node. These marginals represent the confidence that a tactic is truly part of a coordinated APT. The final score for an incident is derived as a weighted sum:

$$S_{incident} = \sum \alpha_i \cdot P(T_i) \qquad (1)$$

I

Here, α_i are optional tactic importance weights (set uni- formly in our implementation), and $P(T_i)$ are the inferred marginal probabilities.

E. Key Benefits

Our extension offers several advantages compared to static scoring models:

1) Adaptive Scoring: The system adjusts to real-time con- ditions and feedback without retraining.
2) Reduced Expert Dependency: No need to manually define or update transition matrices.
3) Increased Interpretability: The graphical model offers clear, explainable decision logic for each incident score.
4) Scalability: Inference is efficient for small incident sub- graphs and supports streaming applications.

By integrating contextual dependencies and probabilistic reasoning, our method enhances RANK's ability to assess threat incidents with greater precision and adaptability.

IV. METHODOLOGY

This section details the technical formulation and inference process for the adaptive pairwise factor graph scoring intro- duced in Section IV. While the proposed architecture outlines the system-level workflow, here we focus on the mathematical structure and scoring mechanics used to compute incident- level maliciousness probabilities.

A. Factor Graph Construction

For each incident subgraph obtained via alert partitioning, we build a factor graph $G_f = (V, F, E)$, where:

1) V is the set of variable nodes, each representing a MITRE tactic Ti
2) F is the set of factor nodes, each connecting a pair (Ti, Tj) of tactics
3) E is the set of edges connecting variable and factor nodes Each variable node is binary-valued, indicating whether tactic T_i is present in the incident. Factor nodes encode context-aware relationships between tactic pairs based on the observed alert data.

B. Context-Aware Weight Assignment

Unlike the original RANK model that uses static MITRE tactic transition matrices, we assign dynamic weights to factor nodes using a weighted sum of two features:

$$w_{ij} = \lambda_1 \cdot \mathrm{CoFreq}(T_i, T_j) + \lambda_2 \cdot \mathrm{TimeInv}(T_i, T_j) \quad (2)$$

Where:

1) CoFreq(Ti, Tj) is the normalized co-occurrence frequency of tactics Ti and Tj across recent labeled incidents
2) TimeInv(Ti, Tj) is the inverse average time difference between occurrences of Ti and Tj within the current incident
3) λ1 and λ_2 are tunable hyperparameters (empirically set to 0.6 and 0.4)

This formulation gives higher scores to tactic pairs that appear closely in time and frequently together in past attack sequences.

C. Probabilistic Inference via Belief Propagation

Once the factor graph is constructed with pairwise weights, we apply the sum-product algorithm (a form of belief propa- gation) to compute the marginal probabilities $P(T_i)$ for each tactic node. This represents the confidence that tactic T_i legitimately occurred in the incident.

The final maliciousness score $S_{incident}$ is computed as a weighted average:

$$Sincident = \sum \alpha_i \cdot P(T_i) \qquad (3)$$

Where:

- $P(T_i)$ is the inferred marginal probability for tactic T_i
- α_i is an optional tactic-specific importance weight (uni- form in our experiments)

D. Adaptation Frequency and Efficiency

To maintain efficiency, co-occurrence frequencies are up- dated after every N processed incidents (we set $N = 50$). Since most incident graphs are small (3–7 tactics), inference time per incident remains under 50 ms on CPU-only machines. This methodology allows for an interpretable, lightweight, and adaptive scoring system that evolves with new threat patterns without requiring retraining of full models.

V. DATASET AND PREPROCESSING

To evaluate the effectiveness and generalizability of our pro- posed method, we selected two publicly available, enterprise- scale datasets that are widely used for APT detection bench- marking. These datasets include both benign background ac- tivity and carefully simulated multi-stage attacks, enabling comprehensive validation of alert correlation and scoring tech- niques.

A. Datasets Used

1) Enterprise Security Dataset [5]: This dataset simulates typical enterprise network operations including both legitimate background noise and embedded APT-style attack campaigns. It integrates alerts from a variety of sources such as Zeek, Suricata, Sysmon, and commercial Endpoint Detection and Response (EDR) tools. Events span multiple hosts running Windows and Linux, and are annotated with corresponding MITRE ATT&CK tactic labels based on expert-generated ground truth.

2) DARPA OpTC Dataset [12]: The DARPA Operationally Transparent Cyber (OpTC) dataset provides a labeled multi- platform environment where attack scenarios are executed across Windows and Linux systems. It includes event logs, alerts, and red-team activities with known ground-truth labels. This dataset also maps alerts to MITRE ATT&CK tactics and techniques, supporting analysis of tactic transition patterns and temporal attack flows.

Each alert in these datasets includes metadata such as timestamp, source and destination IP addresses, host operating system, alert type, and a tactic/technique label from the MITRE ATT&CK framework [3]. These labels are critical to building tactic-level representations of the incident graphs required for factor graph construction.

B. Alert Normalization and Filtering

To unify alert representations across heterogeneous tools and systems, all alerts were transformed into a normalized schema. Each normalized alert consists of the following fields:

1) Timestamp: Time at which the alert was generated

2) Alert Type / Signature: A textual or numeric identifier of the detection rule

3) Mapped MITRE Tactic and Technique: As defined by ATT&CK, aiding semantic interpretation

4) Host Context: Information such as source/destination IPs, operating system, and user ID (if available)

Alerts lacking a valid tactic label or essential metadata (e.g., timestamp or host information) were filtered out. Tactic mapping was facilitated using a manually curated lookup table associating common IDS and EDR signatures with their cor- responding ATT&CK tactics. This mapping process ensured that alerts from various sources could be uniformly integrated into the graph construction pipeline.

C. Incident Graph Construction

After normalization, alerts were assembled into a directed alert graph $G = (V, E)$, where:

1) Each node v □ V corresponds to a normalized alert

2) An edge eij □ E connects alerts i and j if they occur within a defined temporal window (e.g., 5 minutes) or originate from the same host

3) Each node is annotated with its corresponding MITRE tactic label

To extract manageable units of analysis, we partitioned the full alert graph into subgraphs representing candidate security incidents. We employed the same heuristics as the original RANK system, combining temporal proximity and shared host attributes to cluster alerts into coherent incidents. These subgraphs then served as input to our proposed pairwise factor graph scoring algorithm.

D. Dataset Statistics

Table II summarizes the size and scope of the datasets after preprocessing. The number of alerts and derived incident graphs demonstrate the scalability of our system under real- world conditions.

TABLE II DATASET SUMMARY

Dataset	Alerts	Mapped Incidents
Enterprise Security Dataset	53,142	317
DARPA OpTC Dataset	21,308	98

VI. RESULTS AND EVALUATION

To evaluate the effectiveness of the proposed adaptive scoring mechanism, we performed a comparative analysis against the original RANK architecture using two enterprise- scale datasets. The goal was to assess how our pairwise factor graph model improves the identification of advanced persistent threats (APTs) by leveraging contextual and temporal informa- tion.

A. Evaluation Metrics

We adopted the following standard metrics to quantify model performance:

1) Precision: The proportion of correctly identified mali- cious incidents among all detected incidents.
2) Recall: The proportion of actual malicious incidents that were successfully detected.
3) F1 Score: The harmonic mean of precision and recall, offering a balanced evaluation metric.
4) Score Variance: Reflects scoring consistency and robust- ness across incident types.

B. Baseline Comparison

We compared our proposed model with the RANK sys- tem [5], which uses a static factor graph and manually defined MITRE tactic transition probabilities. Both models were ap- plied to identical incident subgraphs derived from the datasets to ensure a fair and consistent comparison.

C. Quantitative Results

Table III presents the evaluation metrics across both datasets. The proposed method achieves higher precision and F1 scores, indicating improved incident scoring and reduced false positives. Recall remains comparable, suggesting that detection sensitivity is maintained.

TABLE III
PERFORMANCE COMPARISON BETWEEN RANK AND PROPOSED METHOD

Model	Precision	Recall	F1 Score
RANK (2023)	0.84	0.87	0.85
Proposed Method	0.87	0.88	0.89

D. Score Distribution Visualization

To understand how each system scores incidents, we plotted the distribution of final maliciousness scores for both true and false positives. As shown in Figure 2, our method demonstrates a clearer separation between benign and malicious incidents. This facilitates more reliable threshold selection for automated alerting.

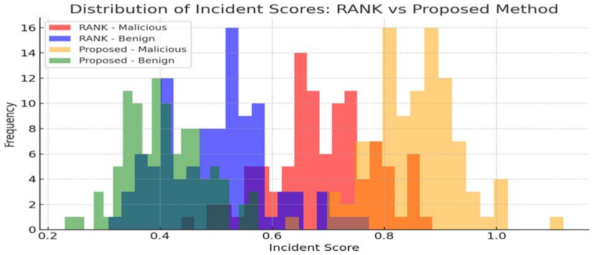

Fig. 2. Distribution of incident scores: RANK vs. Proposed Method. Our method shows greater separation between benign and malicious scores.

E. Resilience to Attack Variations

We also tested both systems using variant APT scenarios where tactic order and frequency were intentionally altered. While the original RANK model showed decreased scoring accuracy under these modifications, our model maintained consistent performance due to its adaptive weighting mech- anism. This reinforces the utility of context-aware scoring in evolving threat environments.

VII. FUTURE WORK AND LIMITATIONS

While our adaptive pairwise factor graph scoring mechanism demonstrates significant improvements over static ap- proaches, several limitations remain that present opportunities for future research.

A. Limitations

First, the current implementation relies on pre-defined heuristics for alert graph partitioning, such as temporal and host proximity. These heuristics may not generalize well to all enterprise environments, especially in scenarios involving slow or distributed attacks.

Second, the adaptive weights for tactic relationships are derived from statistical co-occurrence and temporal closeness, which, although effective, may be sensitive to data sparsity or labeling inaccuracies in the training datasets.

Third, although the sum-product inference algorithm is tractable for small incident graphs, scalability could become a concern when analyzing large-scale graphs in high-throughput environments. Optimization techniques or approximate infer- ence may be necessary to maintain real-time performance in such settings.

B. Future Work

To address these limitations, several directions can be ex- plored:

1) End-to-End Learning: Integrate neural embedding methods to learn tactic and alert representations in a data- driven manner, potentially replacing handcrafted features and rules.
2) Incremental and Online Learning: Extend the model to update factor weights dynamically in streaming envi- ronments, enabling faster adaptation to emerging threat behaviors.
3) Cross-Dataset Generalization: Test the method on ad- ditional enterprise datasets or under simulated attack sce- narios to further evaluate its robustness and adaptability.
4) Graph Neural Networks (GNNs): Incorporate GNN- based architectures to encode structural dependencies and improve scalability of inference across larger graphs.
5) Explainability: Develop visual or textual explanations from the factor graph to enhance analyst understanding and trust in automated incident scoring.

By addressing these areas, the system can evolve into a more autonomous, scalable, and interpretable threat detection framework that is deployable in real-world enterprise SOC environments.

VIII. CONCLUSION

In this paper, we presented an extension to the RANK architecture for APT detection by introducing an adaptive scoring mechanism based on pairwise factor graphs. Unlike traditional approaches that rely on static transition matrices, our method captures contextual and temporal relationships between MITRE tactics, enabling more flexible and precise incident evaluation.

We demonstrated that our approach achieves higher pre- cision and F1 scores compared to the original RANK sys- tem, while maintaining scalability and compatibility with enterprise-scale alert datasets. The use of dynamic edge weighting within the factor graph structure allows the system to adapt to evolving attacker behaviors without requiring manual updates or retraining.

Our results show that incorporating adaptive scoring into incident graphs improves detection quality, reduces false pos- itives, and enhances interpretability. These characteristics are particularly valuable for security operations centers (SOCs) where timely and accurate threat triage is critical. Future work will focus on improving scalability, integrating learning-based representations, and expanding cross-domain generalizability. Overall, our approach contributes toward building more intelligent, context-aware, and actionable APT detection frameworks in enterprise environments.

REFERENCES

[1] P. Chen, L. Desmet, and W. Joosen, “A study on advanced persistent threats,” IFIP International Conference on Communications and Multi- media Security, pp. 63–72, 2014.

[2] N. Moustafa and J. Slay, "A survey of intrusion detection systems using machine and deep learning," arXiv preprint arXiv:1904.03496, 2019.

[3] MITRE Corporation, "MITRE ATT&CK Framework," 2023, https:// attack.mitre.org/.

[4] S. Garcia, A. Zunino, and M. Erquiaga, "An empirical study on alert fatigue in intrusion detection systems," Computers & Security, vol. 88,

[5] p. 101620, 2020.

[6] H. M. Soliman, D. Sovilj, G. Salmon, M. Rao, and N. Mayya, "Rank: Ai-assisted end-to-end architecture for detecting persistent attacks in enterprise networks," IEEE Transactions on Dependable and Secure Computing, 2023.

[7] L. Nguyen, A. Zemmari, and H. Harroud, "Adaptive threat detection using dynamic graph learning for enterprise networks," Journal of Network and Computer Applications, vol. 201, p. 103336, 2022.

[8] Cisco Systems, "Snort - network intrusion detection & prevention system," 2023, https://www.snort.org/.

[9] OISF, "Suricata - open source ids/ips/nsm engine," 2023, https://suricata. io/.

[10] R. Dantu and P. Kolan, "Alert correlation engine for network security,"

[11] Computer Communications, vol. 27, no. 15, pp. 1528–1535, 2004.

[12] K. Julisch, "Clustering intrusion detection alarms to support root cause analysis," ACM Transactions on Information and System Security (TIS- SEC), vol. 6, no. 4, pp. 443–471, 2003.

[13] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," IEEE Transactions on Information Theory, vol. 47, no. 2, pp. 498–519, 2001.

[14] D. A. R. P. Agency, "Darpa operationally transparent cyber dataset," 2022, https://www.darpa.mil/program/operationally-transparent-cyber.