Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Entity Embeddings and Ensemble Learning for Multi-Attribute Cyber-Attack Forecasting: A Benchmark and Evaluation Protocol for Merged Incident Repositories

Inbasat Fiza
International Islamic University, Islamabad
Umara Noor
umara.zahid@iiu.edu.pk

International Islamic University, Islamabad
Zahid Rashid
Seoul National University

Research Article

Keywords: Data integration, Data leakage, Evaluation methodology, Entity embeddings, Cyber-attack forecasting, Threat intelligence

Posted Date: November 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11259898/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Entity Embeddings and Ensemble Learning for Multi-Attribute Cyber-Attack Forecasting: A Benchmark and Evaluation Protocol for Merged Incident Repositories

Inbasat Fiza[1], Umara Noor[1,2*], Zahid Rashid[3]

[1] *Department of Computer Science, Faculty of Computing and Information Technology, International Islamic University, H-10, Islamabad 44000, Pakistan*

[2] *Department of Cyber Security and Forensic Computing, University of Prince Mugrin, Madinah 41499, Saudi Arabia*

[3] *Technology Management, Economics and Policy Program, College of Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Republic of Korea*

*Corresponding author: Umara Noor. E-mail: umara.zahid@iiu.edu.pk

Abstract

Integrating heterogeneous repositories into a single knowledge base, and completing each source's missing attributes by supervised imputation, is routine in data engineering, but its effect on downstream evaluation is rarely measured. We study it in cyber-attack forecasting, where individual incident repositories are either temporally dated or categorically shallow. To our knowledge, the resulting form of leakage has not been described for merged threat-intelligence corpora. Harmonising the VERIS Community Database with Hackmageddon timelines into 20,279 incidents under a common 64-attribute schema, we find attack-type macro F1 of 70.41% on records whose labels were reconstructed during merging against 26.58% on records whose labels were observed, a 43.83-point gap ($p < 0.0001$) that pooled metrics conceal. Because the sources cover disjoint periods, a masked-imputation control holding era and source constant isolates the effect: inflation persists at 16.86 points ($p = 0.012$) and scales inversely with imputer fidelity, so a weak imputation step distorts evaluation more than a strong one. We propose a source-agnostic, three-part protocol (provenance stratification, feature-group ablation and the masked-imputation control) applicable to any integrated dataset. Entity embeddings raise attack-type macro F1 from 14.00% to 32.57% at fixed architecture, and class cardinality rather than imbalance bounds performance: the 189-class country target reaches at most 7.32% macro F1 despite 66.79% accuracy. Under chronological evaluation, attack-type forecasting reaches 38.44% accuracy and 19.32% macro F1, and the deep model overtakes the tree-based baseline ($p < 0.003$). Code, taxonomy and corpus-construction scripts are released.

Keywords: Data integration · Data leakage · Evaluation methodology · Entity embeddings · Cyber-attack forecasting · Threat intelligence

1 Introduction

Operational cyber-defence is predominantly reactive. Intrusion detection systems, endpoint protection platforms and security information and event management tools recognise malicious activity as it manifests, so their value is bounded by the interval between compromise and detection. Organisations sustained an average of roughly 1,636 attacks per week during the second quarter of 2024 [1], and global cyber-crime costs are projected to reach USD 10.5 trillion annually [2]. At these volumes, advance characterisation of likely threats changes the defensive calculus in ways that incremental reductions in detection latency do not.

Forecasting serves this role in other risk-bearing domains: financial institutions reallocate exposure ahead of anticipated market movement [3], and health systems position capacity ahead of predicted demand [4]. Within cybersecurity, predictive methods have matured into a recognised subfield with its own surveys [5], [6], spanning short-horizon event prediction, attack-rate forecasting and longer-range trend anticipation. Two constraints nevertheless persist.

The first is scope. Published models predict attack volume [7], a single attack family [8], or activity within one organisation or sector [9], [10], rather than the conjunction of attributes a defender needs in order to act. Work that does forecast attack characteristics across organisations tends to predict a coarse outcome: a victim/non-victim label [12], an incident category, or a regional aggregate [15].

The second constraint is data. Network-traffic corpora such as the CIC-IDS family [17], [18] provide packet-level fidelity for controlled testbed traffic but record none of the strategic attributes on which anticipatory defence depends. Incident repositories are better aligned but individually deficient: the VERIS Community Database [19] applies a rigorous incident taxonomy, but its coverage effectively ends in 2017, while Hackmageddon [20] provides current timelines that are semi-structured and lack VERIS's actor, asset and motive dimensions.

The natural response, adopted in several studies [12], [14], is to merge repositories. Merging requires reconciling incompatible schemas, and where one source lacks a field the other records, the missing field is commonly reconstructed by a supervised model fitted on the source where it is present. This yields a corpus that is complete and internally consistent. It also yields a corpus in which a substantial fraction of labels are themselves model outputs derived from the very features later used as predictors. The problem is not specific to security: integrating sources with partially overlapping schemas, and completing the result by supervised imputation, is a routine step whenever a knowledge base or analytical dataset is assembled from heterogeneous information systems.

That such a construction risks leakage is not a new observation. Leakage, the introduction of information about the target that would not legitimately be available at prediction time, has been formalised for over a decade [23] and documented as a widespread cause of over-optimistic results across scientific fields [24]. Its effects have been quantified empirically in neuroimaging [25], catalogued into taxonomies [26], [27], and measured specifically within security machine learning, where constructing intrusion-detection datasets with and without preprocessing leakage was shown to inflate reported accuracy while degrading reliability [28].

What has not, to our knowledge, been described is the specific mechanism that arises when heterogeneous incident repositories are merged by cross-source label imputation, nor a diagnostic that surfaces it. This paper began as an attempt to forecast three attack attributes jointly from such a corpus.

The analysis reported in Section 7.5 subsequently showed one of the three, attack pattern, to be a derived taxonomy field rather than a forecastable event property; we therefore report forecasting results for attack type and targeted country and retain pattern as a diagnostic case. In evaluating that framework, we found pooled metrics substantially inflated by this mechanism, and that stratifying evaluation by label provenance exposed it immediately. Because merging incident repositories is common practice and provenance-stratified reporting appears absent from this literature, we treat the diagnostic as the paper's primary contribution and organise the work around it. Although cyber-attack forecasting is our application domain, the hazard and the protocol concern how integrated datasets are constructed and evaluated, and they therefore apply to data-mining studies built on any merged corpus whose labels were partly reconstructed during integration.

1.1 Contributions

- **A unified incident corpus and recovered taxonomy.** We harmonise VERIS (1971–2017) and Hackmageddon (2018–2023) into a 64-attribute schema of 20,279 incidents and consolidate 588 heterogeneous attack labels into 11 forecasting categories. Every label-to-category assignment is released with its provenance, together with scripts that rebuild the corpus from its upstream sources. We are not aware of an equivalent merged incident corpus at this temporal reach and attribute depth.
- **Identification and quantification of imputation-induced circularity in merged incident corpora.** We show that records whose labels were reconstructed during merging yield macro F1 inflated by 43.83 points for attack type and 10.03 points for pattern relative to records whose labels were observed, and we isolate the imputation-attributable portion at 16.86 points under a control that holds era and source constant. This is a domain-specific instance of a known failure mode [23], [24]; the contribution is its identification in merged threat-intelligence corpora, where it has not previously been reported, the measurement of its magnitude, and the finding that it scales inversely with imputer fidelity.
- **A provenance-aware evaluation protocol.** We propose three diagnostics (provenance stratification, feature-group ablation, and a masked-imputation control that holds era and source constant) and demonstrate all three. The protocol is source-agnostic, requires no data beyond what the merge already defines, and complements the checklist-style prevention advocated in the general leakage literature [24] with post-hoc detection applicable to corpora that have already been built.
- **A representation benchmark on incident-level data.** Replacing label encoding with learned entity embeddings [29] raises attack-type macro F1 from 14.00% to 32.57% at fixed architecture. Comparable gains are established for network-traffic and general categorical data [30], [31]; we confirm that the effect transfers to incident-level threat data, where high-cardinality nominal fields such as country and organisation dominate the feature space.
- **A cardinality benchmark.** No model exceeds 7.32% macro F1 on the 189-class country target despite 66.79% accuracy, and feature-group ablation shows the ceiling is not attributable to any removable feature group. This is consistent with the tail-label behaviour characterised in the extreme-classification literature [35], [36]; we quantify it for geographic forecasting and distinguish it from class imbalance, which is separately present in the corpus at a ratio of 306:1.
- **A forward-looking evaluation.** Under chronological partitioning the framework reaches 38.44% accuracy and 19.32% macro F1 on attack type, roughly a quarter below the random-split estimate,

and the deep model significantly outperforms the Random Forest baseline it loses to under random partitioning. We report both protocols and adopt the chronological figure as the headline forecasting result.

We additionally report a null result: neither a policy-gradient weighting layer of the kind studied in the dynamic-ensemble literature [38]–[40] nor the established accuracy-weighted alternatives, including CAWPE [41], improves on uniform averaging on any target in this corpus.

Section 2 reviews forecasting methods, incident datasets, the leakage literature, and representation and ensemble learning for categorical data. Section 3 describes corpus construction. Section 4 specifies the forecasting framework. Section 5 introduces the evaluation protocol. Section 6 details the experimental configuration, Section 7 reports results, and Sections 8–10 discuss, bound and conclude the work.

2 Related work

We review five strands: the datasets it draws on (section 2.1), incident-level forecasting (section 2.2–2.3), leakage in machine-learning-based science (section 2.4), representation learning for high-cardinality categorical data (section 2.5), and ensemble weighting (section 2.6). Section 2.7 states the resulting gaps.

2.1 Incident repositories and merged corpora

Cybersecurity datasets divide into network-traffic corpora and incident repositories. CSE-CIC-IDS2018 [17] and CIC-DDoS2019 [18] provide labelled flow records from controlled environments and have supported extensive intrusion-detection research, but they describe traffic rather than events and carry no information about target geography, victim sector or actor motive.

Incident repositories record real events. VERIS [19], [22] encodes threat actors, actions, affected assets, compromised attributes and actor motives under a formal schema, but its practical coverage terminates around 2017. Hackmageddon [20] compiles publicly reported incidents continuously, providing current visibility into techniques and targets, though its timelines vary in attribute set between releases and lack VERIS's categorical depth. The Web Hacking Incidents Database [21] is confined to media-reported web-application incidents and is heavily selection-biased.

Table 1 Incident-level and network-traffic datasets used in cyber-attack prediction research.

Dataset	Coverage	Granularity	Attribute depth	Fit for multi-attribute forecasting
CSE-CIC-IDS2018 [17]	2018 (testbed)	Packet / flow	Traffic features only	Low — no strategic attributes
CIC-DDoS2019 [18]	2019 (testbed)	Packet / flow	DDoS traffic only	Low — single attack family
VERIS / VCDB [19], [22]	1971–2017	Incident	Actors, actions, assets, motives, patterns	Moderate — dated; no technique label
Hackmageddon [20]	2018–present	Incident	Technique, target, sector, country	Moderate — shallow; variable schema
WHID [21]	Web incidents	Incident	Web-application events	Low — selection-biased
Unified corpus (this work)	1971–2023	Incident	64 harmonised attributes	Designed for the task

Because no single repository combines temporal reach with categorical depth, several studies merge them. Liu et al. [12] drew on VERIS, Hackmageddon and WHID jointly for breach forecasting;

Almahmoud et al. [14] assembled a monthly corpus of major incidents across 36 countries spanning 11 years from heterogeneous public sources. Merging necessarily involves schema reconciliation, and where a field is present in one source and absent in another, supervised imputation is a common remedy. We are not aware of prior work in this literature that reports performance separately for observed and reconstructed records, the omission that Section 5 addresses.

2.2 Trend and volume forecasting

The predictive-cybersecurity literature has been surveyed twice in recent years. Husák et al. [5] compare attack projection, predictive blacklisting and network security situation forecasting within a common alert-sharing environment, concluding that each reaches experimental deployment readiness but that forecasting methods differ sharply in the granularity of their output. Landauer et al. [6] review time-series analysis across cybersecurity analytics and characterise the field as spanning short-term low-level event ingestion through to long-term forecasts of high-level attack cases. Both surveys identify output granularity as the principal axis of variation, which is the axis on which the present work is positioned.

The earliest formulation treats forecasting as univariate time-series prediction over incident counts. Werner et al. [7] fitted an ARIMA model to a year of Hackmageddon data to predict daily attack volume, outperforming a series-mean baseline. Extensions add sophistication without broadening scope: a DDoS-focused ensemble [8] combined statistical models with change-point detection on CIC-DDoS2019 to forecast attack-flow volume within a single attack family, and a sector-oriented system [16] applied SARIMAX, Prophet and LSTM models to textual incident narratives to anticipate energy-sector trends. Almahmoud et al. [14] extend the horizon considerably, forecasting attack trends years in advance across 42 threat types using features drawn from research literature, news, blogs and social media, and deriving a five-phase threat cycle. These studies forecast trend and intensity rather than attack identity for a given date.

2.3 Event- and attribute-level forecasting

A second line forecasts events or their attributes. Ahmed et al. [10] predicted three event classes within 24-hour windows on CSE-CIC-IDS2018, reporting 90.4% accuracy. Wu et al. [11] used an attention-based LSTM over system logs to jointly predict the next security event and its time interval, reaching 92.15% precision, while noting the model cannot forecast events lacking an antecedent log signature. Okutan et al. [9] trained a Bayesian network on social-media and GDELT [42] signals to estimate attack probabilities for a single organisation, reporting 87%, 90% and 96% AUC for endpoint-malware, malicious-destination and malicious-email attacks respectively. Liu et al. [12] classified organisations as prospective breach victims at 90% accuracy using externally measurable network features, acknowledging that the evaluation retrodicts breaches within the observation window. Qasaimeh et al. [13] forecast attacks against financial institutions using deep neural networks, reporting 99.67% accuracy in testing that fell to 90.36% on live deployment, an instructive illustration of how sensitive reported figures are to evaluation conditions.

Closest to the present formulation is Mumtaz et al. [15], who classify and predict Significant Cyber Incidents from a curated incident dataset, applying data-mining feature extraction and comparing Naive Bayes, SVM, logistic regression and Random Forest, and reporting predictions of both incident type and the world region most likely to be affected. That work therefore addresses attack-type and

geographic prediction from incident-level data, as we do. It differs in three respects relevant here: predictions are made at continental rather than country granularity, are not indexed to a specific future date, and are produced from a single-source corpus in which no cross-source imputation occurs. The last difference is material, since it means the hazard identified in Section 5 does not arise for that study.

Table 2 Summary of cyber-attack forecasting literature.

Study	Data source	Method	Forecast output	Reported performance
Werner et al. [7]	Hackmageddon counts	ARIMA	Daily attack count	MSE 2.56; MAPE 74.8
Bitit et al. [8]	CIC-DDoS2019	Statistical ensemble + change points	DDoS occurrence	79.09% accuracy
Zacharis et al. [16]	Four public sources	SARIMAX, Prophet, LSTM	Sector threat trends	Expert-panel agreement
Ahmed et al. [10]	CSE-CIC-IDS2018	SMOreg, LSTM	Events per 24-h window	90.4% accuracy
Wu et al. [11]	System logs (4 networks)	Attention LSTM	Next event + interval	92.15% precision
Okutan et al. [9]	Social media, GDELT	Bayesian network	Attack probability	87–96% AUC
Liu et al. [12]	VERIS, Hackmageddon, WHID	Random Forest	Victim / non-victim	90% accuracy
Qasaimeh et al. [13]	Bank incidents	Deep neural network	Attack type	90.36% (live)
Almahmoud et al. [14]	Multi-source incident corpus, 36 countries	ML over literature, news, social features	Long-range trend across 42 threat types	Trend and threat-cycle validation
Mumtaz et al. [15]	Curated significant-incident dataset	NB, SVM, LR, RF	Incident type + affected region	Comparative accuracy by classifier
This work	Unified VERIS + Hackmageddon	Embedding ensemble	Attack type and country by date[a]	19.32% macro F1 (chronological)[b]

[a] *A third target, attack pattern, is reported in Section 7.5 as a diagnostic case rather than a forecasting result, on the evidence of feature-group ablation.* [b] *Reported under chronological evaluation (train ≤ 2020, test 2023). The comparators in this table were each evaluated under their own conditions, most using random or within-period partitions; the figure given here is forward-looking and is therefore not directly comparable with them. The corresponding within-period estimate for this work is 26.58% macro F1 (Section 7.3).*

2.4 Leakage in machine-learning-based science

Leakage was formalised by Kaufman et al. [23] as the introduction of information about the target that should not legitimately be available for mining, together with a general remedy the authors term learn–predict separation. Kapoor and Narayanan [24] surveyed leakage across 17 scientific fields, finding it documented in 294 papers and in some cases producing conclusions that did not survive correction, and proposed an eight-type taxonomy together with model info sheets as a preventive instrument. Complementary taxonomies distinguish leakage arising from feature selection, preprocessing, and violations of the independence assumption [26], [27].

A parallel strand quantifies leakage empirically rather than cataloguing it. Rosenblatt et al. [25] examined five forms of leakage across four neuroimaging datasets and three phenotypes, finding that leakage via feature selection and repeated subjects drastically inflates measured performance while other forms have minor effects, and that small datasets exacerbate the problem. Within security machine learning, Bouke and Abdullah [28] constructed versions of NSL-KDD, UNSW-NB15 and KDDCUP99 with and without preprocessing leakage, trained six classifiers on each, and showed that leakage yields higher but unreliable accuracy, with sensitivity varying by algorithm.

The present work sits in this second, quantitative strand. Two things distinguish it. First, the leakage vector we examine is not a preprocessing error committed during model fitting, but a property inherited from dataset construction: the labels themselves are partly model outputs, and no amount of care in the downstream pipeline removes this. Second, the affected records cannot be identified by inspecting the modelling code, which is where existing detection guidance directs attention [23], [24]; they can only be identified from provenance metadata generated at merge time. This motivates a detection procedure rather than a prevention checklist, which is what Section 5 provides. In data-engineering terms, label provenance is lineage metadata that integration pipelines generate at merge time but rarely carry forward into evaluation; the protocol makes that metadata part of the reporting.

2.5 Representation of high-cardinality categorical data

Comparative studies consistently find tree ensembles competitive with or superior to deep architectures on tabular data [32], [33]. Representation is a recognised contributor: high-cardinality categorical fields encoded as integers and then standardised impose a spurious ordinal geometry that tree splits ignore but dense layers cannot. Entity embeddings [29], which learn a dense vector per category jointly with the classifier, were introduced to address this, and their benefit over ordinal and one-hot encoding is established. Dahouda and Joe [30] report an F1 of 89% for learned embeddings against 71% for one-hot encoding under an LSTM on categorical benchmarks, while noting reduced memory and feature count. Within security, Almaslukh [31] applies entity embedding to transform raw sensor-network intrusion features into a more robust representation, comparing representations at fixed architecture and reporting macro F-score.

Alternatives to embeddings exist for very high cardinality, notably random-effects formulations that integrate mixed-model likelihoods into neural networks and report advantages over entity embeddings in low signal-to-noise regimes [34]. We do not evaluate these, and note the omission in Section 9.

Where the label space rather than the feature space is large, the governing literature is extreme classification. Datasets in this regime fit power-law distributions in which a large fraction of labels carry very few positive instances [35], and surveys of multi-label learning identify the resulting tail-label behaviour as a defining challenge [36]. Metric choice under multi-class imbalance has been studied separately, with macro-averaged measures preferred where minority-class competence matters [37]. Our country target, with 189 classes over roughly 13,000 training records, falls squarely in this regime, and Section 7.6 should be read as a domain instance of it rather than as a new phenomenon.

2.6 Ensemble weighting

Combining heterogeneous classifiers by learned rather than fixed weights is well studied. Large et al. [41] weight base classifiers by cross-validated accuracy estimates and demonstrate measurable benefit over alternative weighting, selection and meta-classifier approaches, establishing accuracy-weighted probabilistic combination as the standard reference point. Reinforcement learning has been applied to the same problem: Partalas et al. [38] use Q-learning to approximate an optimal inclusion policy for ensemble pruning, Savargiv et al. [39] use learning automata, a reinforcement-learning mechanism, to adapt how the base classifiers of a random forest are aggregated in response to environmental feedback, and Zhang et al. [40] apply a reinforcement mechanism to reweight classifiers under concept drift and class imbalance.

Reinforcement learning is itself well established within cybersecurity, though almost exclusively for control rather than prediction: mitigating DDoS flooding in software-defined networks [43], autonomous threat mitigation [44], securing industrial control planes [45], defending distributed energy resources [46], modelling false-data injection [47], adapting to novel cross-site scripting payloads [48], and countering reactive jamming [49]. In each case the agent selects a defensive action. Its use as an arbitration mechanism among predictive models is a distinct application, and one whose merit on this task we assess empirically rather than assume.

Our weighting layer (section 4.4) belongs to the ensemble-weighting family above and is not proposed as novel. We report it because its null result on this corpus is informative given that the approach is well motivated elsewhere.

2.7 Research gaps

Against this background, three gaps remain and define the scope of this study.

(1) Attribute-level forecasting indexed to a specific date remains rare. Existing work forecasts counts [7], trends [14], binary victim status [12], or type and region without date indexing [15]. Mumtaz et al. [15] are closest, predicting incident type and the affected world region; we are not aware of prior work forecasting attack type and country at national granularity for a specified future date. Date indexing and geographic granularity are therefore the substantive differences, not the number of attributes predicted.

(2) No public incident corpus combines multi-decade coverage with the categorical depth needed to support such a formulation [17]–[21].

(3) Studies that merge incident repositories by cross-source imputation [12], [14] do not report performance separately by label provenance, and no diagnostic for doing so has been proposed. The general leakage literature [23]–[28] provides prevention guidance aimed at the modelling pipeline, which does not reach a hazard introduced during corpus construction.

We note that the third gap is a specialisation of an established concern rather than an unremarked one.

3 The unified incident corpus

This section describes construction of the corpus. Figure 1 summarises the pipeline from the two source repositories through harmonisation to the forecasting models, and indicates where the evaluation protocol of Section 5 applies.

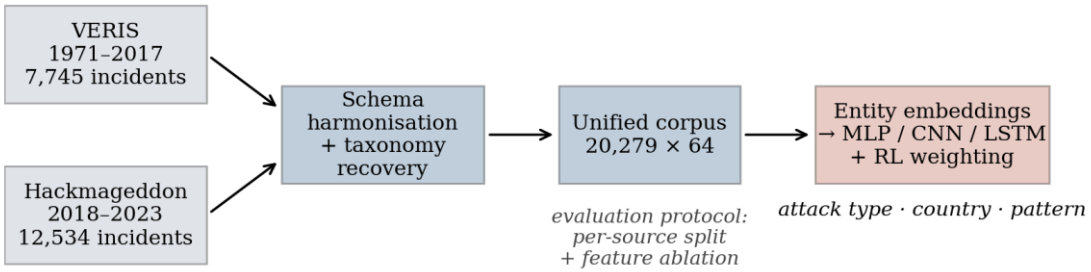


Fig. 1 Corpus construction and forecasting pipeline. Two incident repositories with disjoint temporal coverage are harmonised into a single 64-attribute schema; entity embeddings feed three encoder architectures whose outputs are combined by a learned weighting layer.

3.1 Sources and acquisition

Hackmageddon timelines covering 2018–2023 supplied 12,241 raw records. Records missing both country and organisation were removed, leaving 12,073; incidents recorded against several countries were decomposed into one record per country, raising the count to 12,534. Of three available date fields, only the reported date was retained, the occurred and discovered fields being absent from many timelines and sparsely populated where present.

One gap in coverage should be noted. Although the Hackmageddon timelines were collected for 2018–2023, the assembled corpus contains no records dated 2021. We report this because it constrains the chronological evaluation of Section 7.7, in which the training window necessarily ends in 2020, and because any reuse of the corpus for temporal modelling must account for the discontinuity.

The VERIS Community Database [19], [22] contributed 7,820 records described by 242 fields, reduced to 7,745 under the same completeness criterion. Sixty-four fields were retained: incident identifier, temporal components, victim country, organisation and industry, attack pattern, and 56 binary indicators encoding threat-actor category, actor motive, affected assets and compromised security attributes.

Table 3 Source characteristics before and after preprocessing.

Source	Period	Raw records	Raw fields	Retained records	Retained fields
VERIS / VCDB [19]	1971–2017	7,820	242	7,745	64
Hackmageddon [20]	2018–2023	12,241	13	12,534	64
Unified corpus	1971–2023	—	—	20,279	64

The Hackmageddon count rises relative to the raw corpus because multi-country incidents are decomposed into one record per country.

3.2 Schema harmonisation

The Hackmageddon reported-date field was decomposed into day, month and year to match the VERIS temporal representation; country codes were expanded to full names; missing categorical values in both sources were assigned the sentinel "Unknown" rather than dropped, preserving record completeness without fabricating categories. Field names were lower-cased and aligned to the VERIS convention. Because the two sources cover disjoint periods, concatenation introduces no duplicate incidents.

3.3 Attack taxonomy consolidation and recovery

The corpus contains 588 distinct raw attack labels, including 366 CVE-specific strings. This cardinality fragments the class distribution beyond usability and yields forecasts too granular to act upon, so labels were consolidated into 11 categories.

The consolidation was reconstructed in two tiers rather than devised afresh, so that it reproduces the scheme underlying the corpus. In the first tier, records were matched across an auxiliary release of the corpus on all 63 non-attack fields, and the consolidated label was read directly from the matched record. This recovered 181 raw labels covering 97.5% of records, with no ambiguity: every recovered label mapped to exactly one category. In the second tier, the residual 407 labels (2.5% of records) were assigned by case-insensitive keyword rules derived from the first-tier category definitions and applied in priority order, with unmatched labels falling to "other". Every assignment is released with its tier so the mapping can be audited independently.

Table 4 Consolidated attack categories and class distribution.

Category	Records	Share (%)	Category	Records	Share (%)
malware	6,734	33.2	other	858	4.2
hijacking	4,933	24.3	ddos	360	1.8
unknown	4,719	23.3	fake account / website	128	0.6
targeted attack	1,368	6.7	defacement	99	0.5
vulnerability	982	4.8	coordinated inauthentic behaviour	76	0.4
sqli	22	0.1			

Imbalance ratio between largest and smallest class: 306:1.

3.4 Cross-source attribute reconstruction

Harmonisation exposes a structural asymmetry: each repository records attributes the other omits. VERIS lacks the attack-technique label; Hackmageddon lacks the attack pattern and the 56 binary actor, motive, asset and attribute indicators, 57 fields in total. Restricting the schema to the intersection would discard precisely the categorical depth that motivated merging.

These fields were therefore reconstructed: a Random Forest classifier was trained on the source in which a field is populated, using attributes common to both repositories as predictors, and applied to the source in which the field is absent. Random Forest was chosen for its tolerance of mixed categorical and binary predictors and its stability under skewed class distributions. Missing date components were resolved by forward fill; residual missing categorical values were assigned "Unknown".

The consequence for evaluation is central to this paper and stated explicitly here. For the attack-type target, labels are observed for Hackmageddon records (2018–2023) and reconstructed for VERIS records (1971–2017). For the pattern target, the reverse holds. Any model trained on the pooled corpus is therefore fitted partly to observed labels and partly to the output of an earlier model that used the same features. Section 5 formalises the diagnostic this requires; Section 7.3 reports its result.

3.5 Corpus characteristics

The corpus spans 52 years and comprises 20,279 incidents described by 64 attributes, covering 189 countries, 11 attack categories and 11 attack patterns. Value ranges were verified for logical validity, categorical fields checked for type consistency, and duplicate incidents excluded by construction.

4 Forecasting framework

4.1 Problem formulation

Let $D = \{x_1, \ldots, x_n\}$ denote the corpus, each record x_i comprising a temporal component t_i and a feature vector f_i. We define three categorical targets:

$$y^{(1)} \in A,\ |A| = 11 \text{ (attack type)};\ \ y^{(2)} \in C,\ |C| = 189 \text{ (country)};\ \ y^{(3)} \in P,\ |P| = 11 \text{ (pattern)} \qquad (1)$$

and learn, for each target k, a mapping

$$F^{(k)} : (t, f) \mapsto \hat{y}^{(k)} = \arg\max_j P(y^{(k)} = j \mid t, f) \qquad (2)$$

Each target is modelled independently, so the framework returns a triple for a given date without assuming a factorisation over the joint distribution. Two of the three are forecasting targets in the operational sense. The third, attack pattern, proves on ablation (Section 7.5) to be largely determined by feature groups present at prediction time, and we retain it as a diagnostic case rather than reporting it as a forecast. The cardinality asymmetry between the two forecasting targets (11 classes against 189) proves decisive and is analysed in Section 7.6.

4.2 Input representation

Categorical fields are represented by learned entity embeddings [29] rather than integer codes. Each field of cardinality c receives an embedding table of dimension $\min(50, \lceil (c+1)/2 \rceil)$, trained jointly with the classifier. This avoids asserting an ordinal relation among nominal categories, an assertion that integer encoding followed by standardisation makes implicitly, and one to which dense layers, unlike tree splits, are sensitive.

Day and month are encoded cyclically as sine and cosine pairs so that adjacent values remain adjacent across period boundaries; year is scaled to the unit interval. The 56 binary indicators are passed through unchanged, being already in $\{0, 1\}$. Section 7.2 isolates the contribution of this representation by comparing against label encoding at identical architecture.

4.3 Encoder architectures

Three encoders share the embedding front end and differ in how they model interaction among embedded fields. Treating the embedded fields as a token sequence is what makes convolutional and recurrent operators meaningful here; applying them across raw feature indices, as is sometimes done, imposes an ordering the data does not possess.

- **MLP.** Concatenates all field embeddings with the dense block and applies two fully connected layers (512, 256) with batch normalisation, ReLU and dropout 0.3.
- **CNN.** Applies a one-dimensional convolution across the sequence of field-embedding tokens followed by adaptive max pooling, capturing locally grouped field interactions.

- **LSTM.** Passes a bidirectional recurrent layer (128 units per direction) over the field-embedding tokens.

4.4 Ensemble weighting layer

Encoder outputs are combined as a weighted mixture of class-probability vectors:

$$\hat{y} = arg\ max\ (\Sigma_{i=1}^{3} w_i P_i),\quad \Sigma_i w_i = 1,\quad w_i \geq 0 \qquad (3)$$

Weights are parameterised as a softmax over logits θ and trained on the validation partition by policy gradient. At each step an encoder m is sampled from the current weight distribution, a reward r = +1 is issued if that encoder classifies the instance correctly and r = −1 otherwise, and the logits are updated against a running-mean baseline b:

$$\theta \leftarrow \theta + \alpha\ (r - b)\ (e_m - w),\quad w = softmax(\theta) \qquad (4)$$

where e_m is the indicator vector for the sampled encoder. Because the update is applied independently per target, weights may differ across targets. We report in Section 7.1 that the learned weights do move away from uniform but that the resulting performance gain is not statistically distinguishable from uniform averaging on this task.

5 An evaluation protocol for merged corpora

Merging incident repositories creates two hazards that pooled metrics do not surface. Neither is unprecedented in kind: both are instances of leakage as formalised by Kaufman et al. [23], and both fall within the taxonomy of Kapoor and Narayanan [24]. What is specific here is where they originate and how they can be detected. Prevention guidance in the leakage literature is directed at the modelling pipeline: fit transforms on training data only, preserve learn–predict separation, and complete a model info sheet. That guidance does not reach a hazard already present in the corpus before any model is fitted, and no inspection of the modelling code will reveal it. The diagnostics below are therefore detection procedures, intended to be applied post hoc to corpora that have already been constructed, and to be reported alongside pooled results. Because they require only provenance information that an integration pipeline can record at merge time, they can be implemented as a standard step in data-mining workflows.

5.1 Hazard 1: imputation-induced circularity

When a field absent from one source is reconstructed by a supervised model fitted on the other, the reconstructed values encode that model's decision function over the shared features. A downstream forecasting model trained on the pooled corpus is then evaluated partly on records whose labels it can recover by approximating an earlier model rather than by learning the underlying phenomenon. Measured performance on those records reflects agreement between two models fitted to overlapping information, which is not evidence of predictive validity.

The effect is invisible in pooled reporting because it raises the aggregate without leaving a diagnostic signature: no unusual variance, no train–test gap, no implausible headline figure that would prompt suspicion. Empirical quantifications of other leakage vectors report inflation ranging from negligible to severe depending on mechanism and dataset size [25], [28], so the magnitude cannot be assumed and must be measured per corpus.

Diagnostic 1 (provenance stratification). Partition the test set by label provenance and report all metrics separately for observed and reconstructed records, together with the difference and a paired test across seeds. Where the strata coincide with source eras, as in our corpus, the partition is available without additional annotation; where they do not, a provenance flag written at merge time is sufficient.

5.2 Hazard 2: label derivability

Formal incident taxonomies often define one field as a deterministic or near-deterministic function of others. In the VERIS schema the attack pattern follows from the combination of action, actor, asset and attribute indicators by taxonomy rule. A model predicting pattern from those indicators recovers a rule rather than forecasting an event, and its accuracy will not transfer to settings where the indicators are unavailable at prediction time, which is the operational setting forecasting exists to serve.

Diagnostic 2 (feature-group ablation). Retrain with each schema-defined feature group withheld in turn and report the degradation. A target whose performance collapses when a single group is withheld is substantially rule recovery and should not be presented as a forecasting target on equal footing with the others; it may still be reported, but must be labelled as such. A target insensitive to every group is genuinely difficult, which is itself informative.

5.3 Diagnostic 3: the masked-imputation control

Diagnostic 1 measures the association between label provenance and predictive performance; it does not by itself establish that imputation is the cause. Where provenance is collinear with another variable (as in our corpus, in which the two source repositories cover disjoint eras), era-specific predictability and provenance cannot be separated by the stratified comparison alone. Any corpus assembled from sources with non-overlapping coverage will have this property, so the confound is a general feature of merged incident data rather than a peculiarity of ours. We therefore add a third diagnostic that breaks it.

Diagnostic 3 (masked-imputation control). Work within a single era, where every label is genuinely observed. Freeze a partition, designate a random fraction of records as masked, fit the same imputation model used at merge time on the unmasked training records only, and overwrite the target on masked records with its predictions. Train the forecasting model on the resulting mixed-provenance training set and evaluate on the held-out test set stratified into observed and reconstructed records. Era, source, taxonomy and period are identical across strata; the only difference is whether the label passed through the imputer, so any remaining gap is attributable to imputation alone.

Diagnostics 1 and 3 answer different questions and should be reported together. Diagnostic 1 gives the inflation actually present in the corpus as constructed, which is what a reader needs in order to discount published figures. Diagnostic 3 gives the portion attributable to imputation specifically, which is what establishes the mechanism. Where the two differ, the difference bounds the contribution of whatever else is collinear with provenance. Section 7.4 reports both for our corpus.

The control also yields a quantity worth reporting in its own right: the fidelity of the imputer, measured as the proportion of masked labels it reproduces exactly. As Section 7.4 shows, inflation is larger where

imputer fidelity is lower, which is what the circularity mechanism predicts: a downstream model can approximate an imputer's decision function whether or not that function approximates the truth.

5.4 What the protocol does not do

None of the three diagnostics removes the hazard; they only measure it. Removing Hazard 1 requires either restricting the schema to the intersection of the sources, at the cost of the categorical depth that motivated merging, or restricting evaluation to observed-label records, at the cost of test-set size. We adopt the latter in Section 7.3 and report the observed stratum as the defensible estimate. Which trade-off is appropriate will depend on the corpus and the claim being made.

None of the diagnostics requires abandoning merging or imputation, and none is expensive. All three are conditions on reporting: they oblige a study to state how much of its measured performance is attributable to construction artefacts, so that readers can calibrate accordingly.

6 Experimental setup

6.1 Partitioning

Two partitioning protocols are used, and they answer different questions.

- **Random partitioning.** Training (64%), validation (16%) and test (20%) subsets drawn at random. Partition indices are generated once per seed and reused by every configuration, so all comparisons rest on identical splits. This protocol is used for Sections 7.1–7.6. It is required for the provenance diagnostic of Section 7.3, because the observed and reconstructed strata are era-defined and a chronological split would place them in different partitions rather than allowing them to be compared within the test set.
- **Chronological partitioning.** Training on records dated 2020 or earlier, validating on 2022 and testing on 2023. The corpus contains no 2021 records (Section 3.1), so the training window ends at 2020. This protocol is used for Section 7.7 and is the one that answers whether the framework forecasts, since the test window lies wholly beyond the training window.

We report both rather than substituting one for the other. Random partitioning permits interpolation within the observation window and therefore overstates forward-looking capability; chronological partitioning removes that advantage but forecloses the provenance analysis. Presenting only the first would inflate the forecasting claim, and presenting only the second would remove the evidence for the paper's principal contribution.

In both protocols the validation subset supplies the early-stopping criterion and, where applicable, the reward signal for the weighting layer, so ensemble weights are fitted without reference to test data. Feature standardisation is fitted on the training partition only.

6.2 Training configuration

Table 5 Training configuration.

Parameter	Setting
Optimiser	AdamW, learning rate 1×10^{-3}, weight decay 1×10^{-4}
Batch size	256

Parameter	Setting
Maximum epochs	60 (encoders); early stopping patience 10
Model selection	Best validation macro F1, weights restored
Loss	Cross-entropy
Dropout	0.30
Embedding dimension	min(50, ⌈(cardinality+1)/2⌉); 32 in ensemble configuration
Weighting layer	Policy gradient, $\alpha = 0.05$, 8 passes over validation set
Random Forest baseline	100 trees, default depth
Seeds	5 (0–4); all results reported as mean ± SD

6.3 Metrics

We report accuracy together with macro-averaged precision, recall and F1. Macro averaging is essential rather than optional here: on the country target the majority class alone yields 48.89% accuracy, so accuracy is largely uninformative about per-class competence. Where accuracy and macro F1 diverge sharply, the divergence is itself the result. A majority-class predictor is included throughout as a reference floor.

6.4 Baselines and ablations

Comparisons include each encoder in isolation, a uniform-weight ensemble, the learned-weight ensemble, a Random Forest baseline under identical preprocessing and partitioning, and the majority-class floor. The representation ablation of Section 7.2 additionally reports the same architectures trained on label-encoded and standardised inputs, holding capacity, optimiser and schedule fixed.

6.5 Implementation

Models were implemented in PyTorch 2.13 with scikit-learn 1.8 under Python 3.12, and trained on CPU. Complete code, the recovered taxonomy with per-label provenance, and per-seed result files are released, together with scripts that rebuild the corpus from its upstream sources; the corpus itself is not redistributed (see Statements and Declarations).

6.6 Use of large language models

A large language model, Claude (Anthropic), was used to assist with drafting and editing the manuscript text. All text produced with its assistance was reviewed and edited by the authors, who take full responsibility for the content of the article.

7 Results

One caveat governs how the tables below should be read. Section 7.5 shows that the pattern target is largely derivable from feature groups present at training time, so its high scores measure rule recovery rather than forecasting. We report pattern throughout for completeness and because it is the demonstration case for Diagnostic 2, but its figures are not comparable with those for attack type and country and should not be aggregated with them.

All results are means over five seeds with standard deviations, evaluated on held-out real records. Section 7.1 reports the benchmark and the ensemble-weighting comparison, 7.2 the representation ablation, 7.3 the provenance diagnostic, 7.4 the masked-imputation control, 7.5 the feature-group ablation, 7.6 the cardinality analysis, and 7.7 the chronological evaluation.

7.1 Benchmark

Table 6 Forecasting performance on the unified corpus (mean ± SD over five seeds). Best macro F1 per target in bold. Pattern figures are shown for completeness but are not a forecasting result, see Section 7.5.

Target	Model	Accuracy (%)	Macro F1 (%)
Attack type	MLP	60.87 ± 0.94	33.41 ± 1.94
	CNN	59.43 ± 0.31	30.77 ± 1.35
	LSTM	60.40 ± 0.63	31.84 ± 1.94
	Ensemble (uniform)	61.08 ± 0.45	32.25 ± 2.30
	Ensemble (learned weights)	60.94 ± 0.98	32.57 ± 2.62
	Random Forest	60.79 ± 0.57	35.46 ± 1.07
	Majority class	32.87 ± 0.30	4.50 ± 0.03
Pattern	MLP	95.48 ± 0.32	88.12 ± 3.37
	CNN	95.24 ± 0.45	87.06 ± 2.24
	LSTM	95.62 ± 0.47	88.98 ± 3.36
	Ensemble (uniform)	95.60 ± 0.39	88.35 ± 2.58
	Ensemble (learned weights)	95.60 ± 0.47	88.95 ± 3.36
	Random Forest	95.74 ± 0.27	87.17 ± 3.15
	Majority class	38.86 ± 0.60	5.43 ± 0.32
Country	MLP	66.61 ± 0.47	5.14 ± 0.46
	CNN	66.02 ± 0.49	4.64 ± 0.64
	LSTM	66.59 ± 0.42	4.96 ± 0.56
	Ensemble (uniform)	66.81 ± 0.47	4.99 ± 0.65
	Ensemble (learned weights)	66.61 ± 0.37	4.95 ± 0.63
	Random Forest	66.79 ± 0.59	7.32 ± 1.03
	Majority class	48.89 ± 0.24	0.53 ± 0.00

The deep ensemble and the Random Forest baseline are closely matched. On attack type and country the baseline leads by 2.89 and 2.37 macro F1 points respectively; on pattern the ensemble leads by 1.78 points, though as noted that target does not measure forecasting. Paired tests across seeds (Table 7) find only the country difference significant at the conventional threshold. On the two targets that do measure forecasting, then, the proposed framework does not improve on a Random Forest baseline under random partitioning. Section 7.7 shows this ordering reverses under a chronological split, which bears on how the comparison should be read.

Table 7 Paired t-tests on macro F1 across five seeds.

Comparison	Target	Difference (pp)	t	p
Learned weights vs uniform	Attack type	+0.32	0.58	0.593
	Pattern	+0.60	0.90	0.416
	Country	−0.04	−0.36	0.733
Ensemble vs Random Forest	Attack type	−2.89	−2.75	0.053
	Pattern	+1.78	1.90	0.126
	Country	−2.37	−4.06	0.022

Because a null result for a learned weighting scheme may reflect the scheme rather than the task, we compared the policy-gradient layer against the established alternatives (Table 8). Weights were estimated on the validation partition in every case, so all schemes receive the same information.

Table 8 Ensemble weighting schemes, all applied to the same three trained encoders (macro F1 %, mean ± SD over five seeds). CAWPE follows Large et al. [41] with α = 4.

Weighting scheme	Attack type	Country	p vs uniform (attack)
Uniform (reference)	32.25 ± 2.30	4.99 ± 0.65	—
Validation accuracy	32.35 ± 2.33	4.99 ± 0.66	0.355
CAWPE (accuracy^4) [41]	32.35 ± 2.36	5.00 ± 0.67	0.333
CAWPE (macro F1^4)	32.21 ± 2.34	5.05 ± 0.60	0.848
Best single member	32.18 ± 1.47	5.13 ± 0.47	0.923
Policy gradient (ours)	32.57 ± 2.62	4.95 ± 0.63	0.593

No weighting scheme improves on uniform averaging on either target. Differences from the uniform reference span −0.08 to +0.32 macro F1 points for attack type and −0.04 to +0.14 for country, and no comparison approaches significance (all $p > 0.25$). CAWPE, which outperforms alternative weighting and selection approaches across a large benchmark suite in its original evaluation [41], is indistinguishable from equal weights here.

This makes the null result interpretable. Had CAWPE succeeded where the policy-gradient layer failed, the correct conclusion would have been that our implementation was inadequate. That it does not indicates a property of the task: the three encoders are too similar in competence and too correlated in their errors for any weighting rule to exploit. Validation accuracies differ by under two percentage points, and the members share an embedding front end, so the diversity that weighted combination depends upon is largely absent. We report this as a negative result about ensemble weighting on this corpus rather than as a deficiency of any scheme.

We report the weighting-layer result as a null finding. The learned weights do depart from uniform averaging (0.60/0.25/0.16 across MLP, CNN and LSTM on attack type and 0.23/0.23/0.55 on pattern), and the selected encoder varies by target, which is the behaviour the mechanism is designed to produce. The resulting gains over uniform averaging, however, are 0.32, 0.60 and −0.04 macro F1 points, none statistically distinguishable from zero. Inspection across seeds shows the layer selecting different encoders on different partitions, consistent with fitting validation noise rather than stable competence differences. On this task, learned ensemble weighting does not improve on uniform averaging by any rule we tested, and we do not claim otherwise.

7.2 Representation ablation

Holding architecture, capacity, optimiser and schedule fixed, we compare learned entity embeddings against label encoding followed by standardisation, the representation used in much prior cybersecurity forecasting work.

Table 9 Effect of input representation on macro F1 (%), architecture held constant.

Target	Label encoding + scaling	Entity embeddings	Difference (pp)
Attack type	14.00 ± 1.59	32.57 ± 2.62	+18.57
Pattern	81.51 ± 4.21	88.95 ± 3.36	+7.44
Country	1.79 ± 0.17	4.95 ± 0.63	+3.16

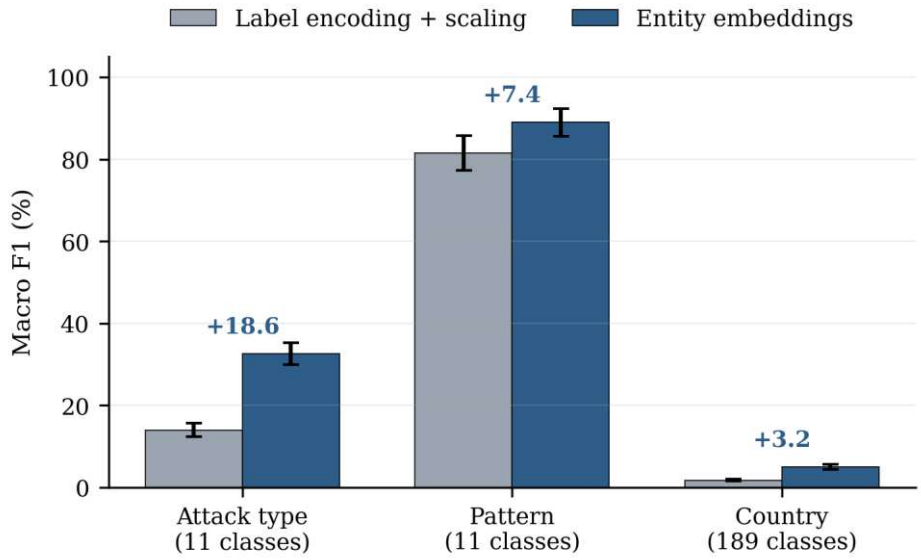


Fig. 2 Macro F1 under label encoding versus learned entity embeddings, at identical architecture and capacity. Error bars are standard deviations over five seeds.

The effect is large where it matters most. On attack type, embeddings raise macro F1 from 14.00% to 32.57%, an 18.57-point gain that converts a model performing barely above the majority-class floor into one competitive with the Random Forest baseline. Under label encoding the deep models trail the baseline by more than 20 macro F1 points; under embeddings the gap narrows to 2.89 points and is no longer significant at $\alpha = 0.05$.

This bears directly on how comparative results in this literature should be read. Studies reporting that deep architectures underperform tree ensembles on incident data, where those studies encode high-cardinality categorical fields as scaled integers, may be measuring a representation deficit rather than an architectural one. The distinction matters because the remedies differ entirely.

7.3 Provenance stratification

Applying Diagnostic 1, the test set is partitioned by label provenance. For attack type, labels are observed for Hackmageddon-era records and reconstructed for VERIS-era records; for pattern the reverse holds.

Table 10 Performance stratified by label provenance (mean ± SD over five seeds).

Target	Label provenance	n	Accuracy (%)	Macro F1 (%)
Attack type	Observed	~2,509	45.92 ± 0.99	26.58 ± 1.32
	Reconstructed	~1,546	86.03 ± 1.07	70.41 ± 5.84
	Inflation		+40.11	+43.83
Pattern	Observed	~1,546	90.85 ± 0.99	84.24 ± 3.10
	Reconstructed	~2,509	98.26 ± 0.20	94.27 ± 5.05
	Inflation		+7.41	+10.03

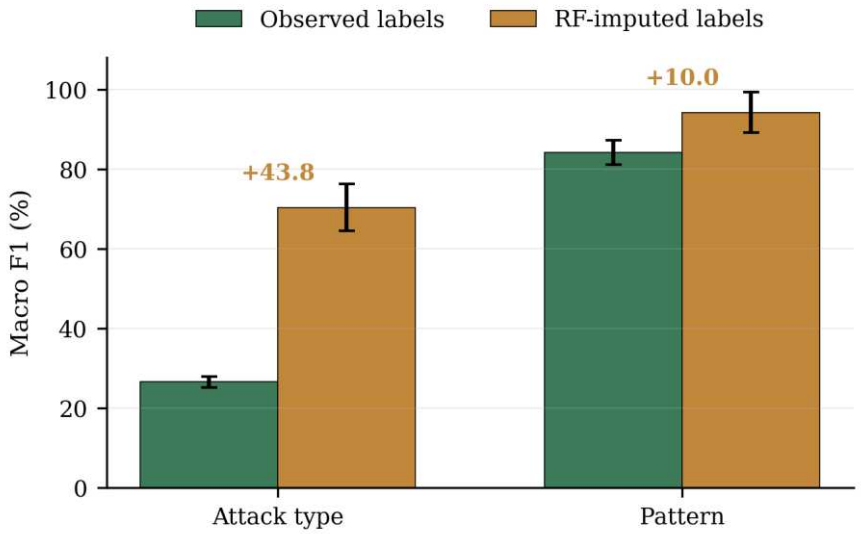


Fig. 3 Macro F1 on records with observed labels versus records whose labels were reconstructed during merging. Differences are +43.83 points for attack type (paired t = 19.40, p < 0.0001) and +10.03 for pattern (t = 3.34, p = 0.029).

The attack-type gap is 43.83 macro F1 points (paired t = 19.40, p < 0.0001), and the pattern gap 10.03 points (t = 3.34, p = 0.029). Both are significant; the attack-type effect is very large and highly stable, with a standard deviation of 1.32 on the observed stratum across five seeds.

The interpretation is direct. On reconstructed records the forecasting model is not predicting attack type from incident characteristics; it is approximating the decision function of the Random Forest that generated those labels from the same features. Agreement between two models fitted to overlapping information is not evidence of predictive validity.

The consequence for reporting is that the pooled figure of 32.57% macro F1 in Table 6 overstates forecasting capability. The defensible estimate is the observed stratum: 45.92% accuracy and 26.58% macro F1. We adopt this as the headline result. Every study merging incident repositories with cross-source imputation is exposed to an analogous discrepancy, and pooled metrics will not reveal it. This comparison cannot, however, establish that imputation is the cause, because the two strata also differ in era; Section 7.4 isolates the imputation-attributable component.

7.4 Masked-imputation control

The stratification in Section 7.3 cannot by itself distinguish circularity from era-specific predictability, because the two source repositories cover disjoint periods. Applying Diagnostic 3 removes that ambiguity. For attack type we restrict the corpus to the Hackmageddon era (2018–2023), in which every attack label is genuinely observed; for pattern we restrict to the VERIS era (≤2017), in which every pattern label is observed. Within each restricted corpus we mask 40% of records (approximately the proportion of the full corpus carrying a reconstructed attack label), fit the same Random Forest imputer on the unmasked training records only and overwrite the masked labels with its predictions.

Table 11 Masked-imputation control, with era, source and taxonomy held constant (mean ± SD over five seeds). Imputer fidelity is the proportion of masked labels the imputer reproduced exactly.

Target	Era	Imputer fidelity	Stratum	n	Accuracy (%)	Macro F1 (%)
Attack type	Hackmageddon 2018–2023	42.2%	Observed	~1,513	43.74 ± 2.49	24.03 ± 2.07
			Reconstructed	~993	64.10 ± 3.80	40.89 ± 7.16
			Inflation		+20.36	+16.86
Pattern	VERIS ≤2017	91.3%	Observed	~939	90.56 ± 0.66	82.57 ± 3.68
			Reconstructed	~609	93.96 ± 1.88	90.05 ± 5.28
			Inflation		+3.40	+7.49

Inflation persists with era held constant. For attack type the reconstructed stratum exceeds the observed stratum by 20.36 accuracy points (paired $t = 9.61$, $p = 0.0007$) and 16.86 macro F1 points ($t = 4.40$, $p = 0.0117$). For pattern the accuracy difference is +3.40 ($t = 4.80$, $p = 0.0086$) and the macro F1 difference +7.49 ($t = 2.56$, $p = 0.063$). Because source, taxonomy, period and model are identical across strata, and the partition into strata is random, these differences are attributable to imputation and to nothing else. Era-specific predictability is eliminated as an explanation.

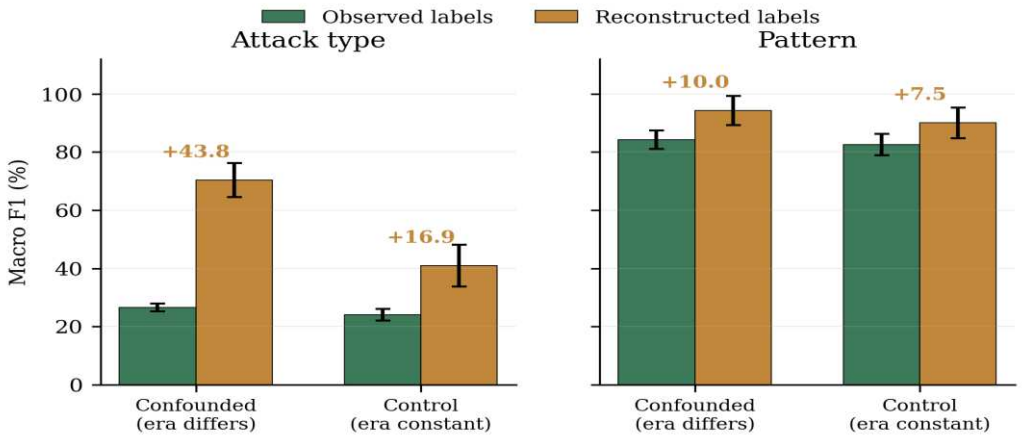


Fig. 4 Provenance-stratified macro F1 under the confounded comparison of Section 7.3, in which provenance and era covary, and under the masked-imputation control of this section, in which era is held constant. Inflation persists in the control but is smaller, indicating that the confounded figure combines circularity with era effects.

Two further observations follow, and both matter for how the results of Section 7.3 should be read.

First, circularity accounts for part but not all the pooled effect. The controlled inflation for attack type, 16.86 macro F1 points, is approximately 38% of the 43.83 points measured under the confounded

comparison. The residual is attributable to whatever else covaries with provenance in the full corpus: era-specific predictability, and differences between the two sources in label distribution, since the consolidated taxonomy was derived largely from Hackmageddon vocabulary. We therefore report the confounded figure as the inflation present in the corpus as constructed, and the controlled figure as the portion attributable to imputation, and we do not claim the former is wholly circular.

Second, inflation scales inversely with imputer fidelity. The attack-type imputer reproduced only 42.2% of masked labels exactly, yet the forecasting model recovered its outputs far more accurately than it recovered the observed labels. The pattern imputer was substantially more faithful at 91.3%, and inflation was correspondingly smaller and, for macro F1, not significant at the conventional threshold. This is what the mechanism predicts: a downstream model can learn to approximate an imputer's decision function irrespective of whether that function approximates reality, and the more the imputer departs from reality the more distinctive and hence the more learnable its output becomes. A well-performing imputer is therefore less dangerous to downstream evaluation than a poor one, which inverts the intuition that a weak imputation step is a minor problem.

7.5 Feature-group ablation

Applying Diagnostic 2, each schema-defined feature group is withheld in turn.

Table 12 Feature-group ablation (macro F1 %, mean ± SD).

Features withheld	Pattern	Country
None (all features)	87.05 ± 4.48	5.73 ± 0.76
action.*	81.46 ± 4.31	5.28 ± 1.01
action.*, actor.*, asset.*, attribute.*	43.30 ± 1.34	4.38 ± 0.56
Degradation (pp)	−43.75	−1.35

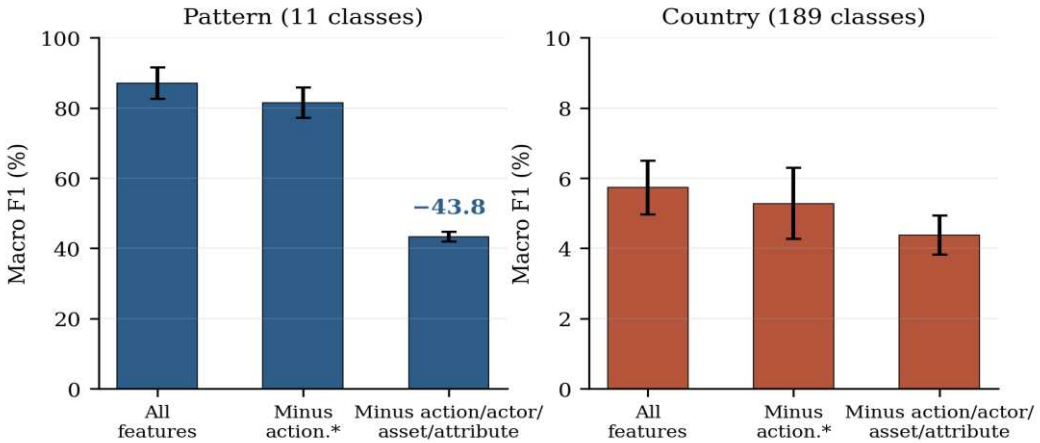


Fig. 5 Feature-group ablation. Pattern performance collapses when the VERIS action, actor, asset and attribute groups are withheld; country performance is insensitive to every group. Note the differing vertical scales.

The two targets behave in opposite ways, and both results are informative. Pattern loses 43.75 macro F1 points when the VERIS indicator groups are withheld, falling from 87.05% to 43.30%. This confirms that pattern prediction is substantially the recovery of a taxonomy rule: in the VERIS schema the pattern follows from the combination of action, actor, asset and attribute values. High pattern accuracy should

therefore not be presented as evidence of forecasting capability and will not transfer to deployment settings where those indicators are unavailable at prediction time.

A second consideration compounds this. The action, actor, asset and attribute indicators are themselves reconstructed for the Hackmageddon portion of the corpus (Section 3.4). On those records, pattern prediction therefore recovers a taxonomy rule applied to imputed inputs, a second-order form of the hazard identified in Section 5.1, in which neither the label nor the features it is derived from were observed. The provenance stratification in Table 10 captures the first-order effect for pattern but not this compounding, and the pattern figures should be discounted accordingly.

We therefore do not treat pattern as a forecasting target. It is retained in the tables because it is the demonstration case for Diagnostic 2 and because its contrast with country is informative, but no claim about forecasting capability rests on it.

Country, by contrast, is insensitive to every group, losing only 1.35 points when all four are removed. Its low performance is therefore not caused by missing or misallocated signal; no feature group in the schema carries appreciable geographic information. This is a genuine difficulty result rather than an artefact, and it motivates the analysis that follows.

7.6 Cardinality analysis

The three targets differ sharply in the relationship between accuracy and macro F1, and the ordering follows class cardinality rather than class imbalance.

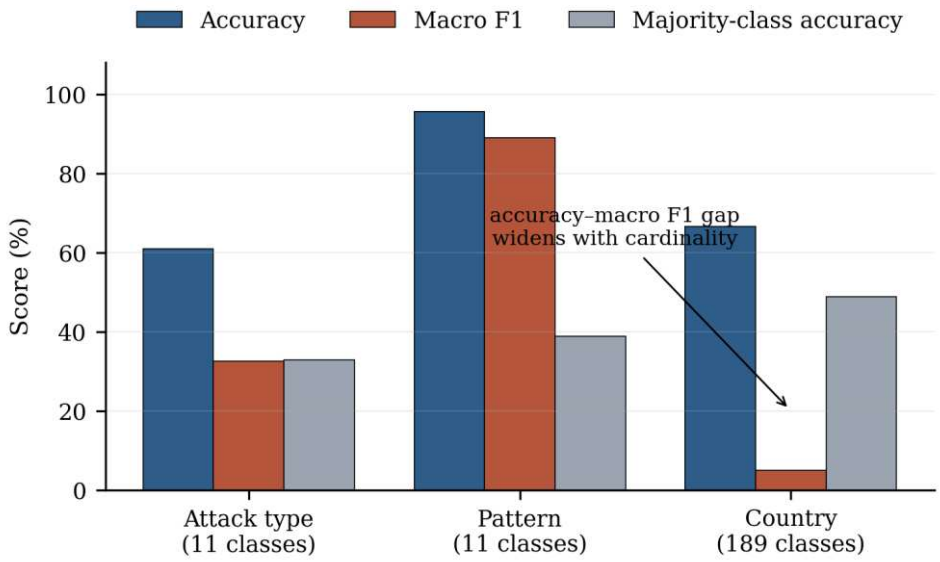


Fig. 6 Accuracy, macro F1 and the majority-class accuracy floor for each target. The accuracy–macro F1 gap widens with class cardinality; on the 189-class country target accuracy remains high while macro F1 approaches the floor.

On the 189-class country target the best model reaches 66.81% accuracy but at most 7.32% macro F1, against a majority-class floor of 48.89% accuracy. The interpretation is that the model predicts the dominant country and is correct about two-thirds of the time, while exhibiting almost no per-class discrimination. Reporting accuracy alone would present this as moderate success; macro F1 shows it is close to the floor.

This is not a class-imbalance problem in the usual sense. Under identical conditions (the same models, the same random partition), the attack-type target carries a 306:1 imbalance ratio yet reaches 35.46% macro F1 for the Random Forest baseline and 32.57% for the deep ensemble, while the country target reaches 7.32% and 4.95% respectively. The ordering persists under chronological evaluation (Section 7.7), where the two targets reach 19.32% and 3.65% macro F1, so the contrast is a property of the targets rather than of a particular protocol. What distinguishes them is the number of classes relative to available discriminative signal. With 189 classes and roughly 13,000 training records, many countries are represented by single-digit counts, and the ablation of Section 7.5 shows that no feature group supplies the geographic signal that would be needed to separate them. Techniques addressing imbalance (reweighting, resampling, synthetic augmentation) redistribute prior mass but do not create discriminative information, and cannot be expected to resolve this.

7.7 Chronological evaluation

Sections 7.1–7.6 use random partitioning, which permits a model to interpolate within the observation window. To assess forecasting proper we retrain under a strictly chronological split: training on records dated 2020 or earlier, validating in 2022, testing in 2023. Two training configurations are compared. The first uses the whole corpus up to the cutoff, so most attack-type training labels are reconstructed; the second restricts training to records whose attack label was observed. In both, every test label is observed, because the 2023 window lies wholly within Hackmageddon coverage.

Table 13 Chronological evaluation for the attack-type and country targets (mean ± SD over five seeds). Training ≤2020, validation 2022, test 2023 (n = 1,705).

Target	Training set	Model	Accuracy (%)	Macro F1 (%)
Attack type	Full corpus (n = 15,354)	Deep model	37.15 ± 1.32	18.29 ± 1.00
		Random Forest	30.78 ± 0.64	13.58 ± 0.30
	Observed labels only (n = 7,609)	Deep model	38.44 ± 1.18	19.32 ± 1.59
		Random Forest	32.57 ± 0.55	14.51 ± 0.44
	—	Majority class	36.66	5.96
Country	Full corpus (n = 15,354)	Deep model	51.30 ± 0.17	3.65 ± 0.20
		Random Forest	51.96 ± 0.37	3.60 ± 0.11
	—	Majority class	30.03	0.46

Three results follow, and each qualifies a claim made earlier in this section.

First, forward-looking performance is substantially lower than the random-split estimate, and accuracy in particular falls to the majority-class floor. The best configuration reaches 38.44% accuracy against a floor of 36.66%, a difference of under two points. Macro F1 remains well clear of the floor at 19.32% against 5.96%, so the model retains genuine per-class competence, but no claim about forward-looking accuracy is supportable. Relative to the observed-stratum estimate of Section 7.3 (45.92% accuracy, 26.58% macro F1), chronological evaluation reduces macro F1 by roughly a quarter. We adopt 38.44% accuracy and 19.32% macro F1 as the headline forecasting result, since it is the only figure produced under a protocol in which the test window lies beyond the training window.

Second, the ordering between the deep model and the Random Forest baseline reverses. Under random partitioning the baseline led on attack type by 2.89 macro F1 points (Table 7, $p = 0.053$); under

chronological partitioning the deep model leads by 4.70 points on the full corpus (paired t = 9.27, p = 0.0008) and by 4.81 points on observed labels only (t = 6.47, p = 0.0029). Both are significant, and the effect is larger and better separated than the random-split comparison it reverses. The most plausible reading is that the tree ensemble exploits within-period regularities (particular organisations, sectors and technique vocabularies prevalent during the observation window) that do not persist into 2023, whereas the learned embedding representation generalises across that shift. This is consistent with the representation result of Section 7.2 and, we suggest, is a more informative comparison than the random-split ordering, since distributional shift between training and deployment is the condition under which a forecasting model is used.

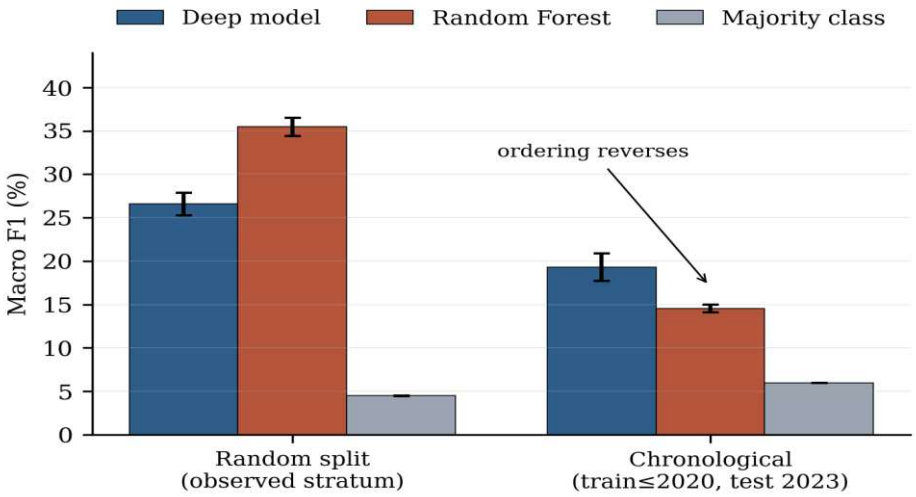


Fig. 7 Attack-type macro F1 under random partitioning (observed-label stratum) and under chronological partitioning. Absolute performance falls under the stricter protocol and the ordering between the deep model and the Random Forest baseline reverses.

Third, training on observed labels only outperforms training on the full corpus, by 1.03 macro F1 points, though the difference does not reach significance at five seeds (t = 1.67, p = 0.170). The direction is consistent with Sections 7.3 and 7.4: reconstructed labels do not merely inflate measured performance; they appear to carry a small cost when the evaluation target is real and future. Halving the training set while improving forward performance is a result we report with appropriate caution given the significance level, but it warrants replication.

The pattern target is omitted from Table 13. Observed pattern labels exist only for the VERIS era, which ends in 2017, while the chronological test window falls in 2023 where every pattern label is reconstructed. A chronological evaluation of pattern would therefore measure a model recovering an imputer's output from imputed inputs and would be uninterpretable. The corpus cannot support forward-looking evaluation of that target at all, which reinforces rather than qualifies the conclusion of Section 7.5.

8 Discussion

8.1 Implications for merged-corpus research

The provenance result generalises beyond this corpus. Any study that merges heterogeneous incident repositories and reconstructs missing fields by supervised imputation will produce a dataset in which

some labels are model outputs derived from the shared feature space. Evaluating on the pooled test set then mixes two quantities: predictive validity on observed labels, and inter-model agreement on reconstructed ones. The mixture is reported as a single number, and the second component inflates it.

Our measurements suggest the inflation can be very large: 43.83 macro F1 points on the attack-type target as the corpus is constructed, of which 16.86 points survive a control holding era and source constant. Both numbers matter, and for different readers. A reader discounting a published figure from a merged corpus needs the first, because that is the distortion present. A reader deciding whether imputation is safe in their own pipeline needs the second, because that is the part attributable to the mechanism rather than to correlated properties of the sources.

The relationship between imputer fidelity and inflation deserves emphasis because it is counter-intuitive. A merge-time imputer that performs poorly might be assumed to introduce noise, degrading downstream performance and thereby erring on the side of pessimism. Our results indicate the opposite: the weaker imputer produced the larger inflation. A poor imputer imposes a more idiosyncratic and hence more learnable structure on the labels it generates, and the downstream model learns that structure in place of the phenomenon. Merge pipelines that report imputation accuracy as a quality check are therefore reporting a quantity that, on this evidence, is inversely related to the evaluation risk they face.

None of this is an argument against merging, which remains the only practical route to a corpus with both temporal reach and categorical depth. It is an argument for stratified reporting, which costs nothing beyond a partition the merge already defines, and for the control of Section 5.3 where sources do not overlap in time. For knowledge and information systems more generally, the practical recommendation is that integration pipelines record label provenance as first-class metadata, so that any downstream evaluation can be stratified by it.

8.2 Representation versus architecture

The finding that entity embeddings recover 18.57 macro F1 points on attack type, at fixed architecture, complements the tabular deep learning literature [29], [32], [33] and has a specific implication for cybersecurity forecasting. Incident data is dominated by high-cardinality nominal fields (country, organisation, industry) for which integer encoding is a poor representation and for which tree-based methods are structurally advantaged. Comparisons that hold representation fixed at label encoding therefore confound representation with architecture. Once embeddings are used, the deep ensemble and Random Forest become statistically indistinguishable on two of three targets under random partitioning, and the deep model overtakes the baseline outright under chronological evaluation (Section 7.7). The latter comparison is the more relevant one for deployment, and it suggests the advantage of tree ensembles on incident data reported elsewhere may be partly an artefact of evaluating within rather than beyond the observation window.

8.3 What forecasting can and cannot deliver here

Taken together, the results support a measured account of what this class of model provides. Attack-type forecasting under the chronological protocol reaches 38.44% accuracy and 19.32% macro F1 over 11 classes. Accuracy at that level is indistinguishable from always predicting the most common class, so the usable signal lies entirely in the macro-averaged figure, which is more than three times the 5.96% majority-class floor. That signal may support coarse resource allocation, though we have not evaluated

that use and make no quantified claim about its operational value; it is certainly not reliable at incident level. Pattern scores highly but is substantially rule recovery, compounded on part of the corpus by reliance on reconstructed inputs, and is not an operational forecasting capability at all. Geographic forecasting, on this feature representation, does not work. The honest summary is that one of the three targets yields a usable if modest signal, one yields none, and one is not a forecasting target.

We regard this as more useful to practitioners than a higher headline figure would be, because it indicates where the capability is real and where it is not.

8.4 Comparison with prior reported performance

Direct comparison with Table 2 is not meaningful, since no prior study forecasts the same targets on the same data, and the studies listed differ in class count, scope and evaluation condition. We note only that reported accuracies in this literature are commonly in the 90% range for tasks with between one and seven classes within a single network or sector, whereas the present task spans 11 classes across sectors and regions. The instructive precedent is Qasaimeh et al. [13], where 99.67% in testing became 90.36% on live deployment: evaluation conditions, not model class, accounted for the difference.

9 Limitations

- **Observation window.** The corpus ends in 2023, and the chronological evaluation of Section 7.7 therefore trains on records to 2020 and tests in 2023. The threat landscape has continued to move since: attack-technique distributions reported by the same source for 2025 and 2026 differ appreciably from those observed in the test window. Two consequences should be distinguished. The empirical forecasting figures reported here (38.44% accuracy and 19.32% macro F1 on attack type) are estimates for the period sampled and should not be read as current operational performance; a model trained on 2023 data would very likely perform differently against 2026 incidents, and we make no claim otherwise. The methodological findings, by contrast, do not depend on the observation window. Imputation-induced circularity is a property of how a corpus is assembled and evaluated rather than of the incidents it contains: the mechanism arises whenever a supervised model supplies labels that a downstream model is later scored against, and the masked-imputation control of Section 7.4 isolates it within a single era precisely so that the result does not rest on any particular period. The same applies to the representation effect of Section 7.2 and the cardinality ceiling of Section 7.6, both of which concern the structure of the learning problem rather than the content of the data. The protocol of Section 5 is likewise period-agnostic and applies unchanged to a corpus extended to the present.
- **Reporting bias in the sources.** Both repositories record publicly disclosed incidents. Undetected, unreported and non-disclosed attacks are absent, and disclosure practice varies by jurisdiction and sector. The corpus describes reported attacks rather than attacks, and the concentration of records on a small number of countries partly reflects this.
- **A single chronological test window.** The chronological evaluation tests on one year, 2023, comprising 1,705 records. Whether the observed ordering between model families holds across other cutoffs is untested, and the absence of 2021 records from the corpus prevents a contiguous multi-year rolling evaluation that would settle it.
- **Pattern cannot be evaluated forward.** Because observed pattern labels end in 2017 and the chronological test window falls in 2023, no forward-looking estimate of pattern forecasting is

obtainable from this corpus. The pattern figures reported under random partitioning should not be read as forecasting performance under any protocol.

- **Ensemble diversity not engineered.** The null weighting result (Table 8) holds for three encoders sharing an embedding front end and differing only in how they model field interactions. Their validation accuracies differ by under two points. A more diverse pool mixing tree ensembles, linear models and networks might give weighting something to exploit, and we did not test this. The result should be read as applying to ensembles of this construction, not to ensemble weighting generally.
- **Single masking fraction and imputer configuration.** The control of Section 7.4 masks 40% of records and uses one Random Forest imputer configuration. Whether inflation scales with the masking fraction, or varies with imputer family, is untested. The inverse relationship we observe between imputer fidelity and inflation rests on two targets rather than a systematic sweep and should be treated as indicative.
- **Residual gap not fully attributed.** The control isolates 16.86 of the 43.83 macro F1 points measured under the confounded comparison. We attribute the remainder to era-specific predictability and to label-distribution differences between the sources, but we do not decompose it further, and the relative contribution of those two factors is unresolved.
- **Second-tier taxonomy assignments.** Of 588 raw attack labels, 407 (2.5% of records) were assigned by keyword rule rather than recovered empirically. These are released for inspection, but they were not independently annotated, and no inter-annotator agreement is reported.
- **One forecast per record.** The framework returns a single most-probable attribute triple. Real periods contain multiple concurrent incidents, so the output characterises the modal event rather than the activity profile.
- **No interpretability analysis.** The framework does not expose the basis of individual predictions, which limits operator trust and impedes error diagnosis in a setting where forecasts inform resource commitments.

10 Conclusion and future work

We constructed a unified corpus of 20,279 cyber-incident records spanning 1971–2023 by harmonising the VERIS Community Database with Hackmageddon timelines, recovered a 588-to-11 attack taxonomy with per-label provenance, and benchmarked deep and classical models on the joint forecasting of attack type, targeted country and attack pattern.

The principal finding concerns evaluation rather than modelling. Cross-source imputation, used routinely to complete merged corpora, produces labels that are themselves model outputs over the shared feature space, and evaluating on the pooled test set inflates measured performance substantially: by 43.83 macro F1 points for attack type and 10.03 for pattern in our corpus. A control holding era, source and taxonomy constant attributes 16.86 of those points to imputation itself, confirming the mechanism while showing that the pooled figure also reflects differences between the source repositories. We proposed and demonstrated a three-part protocol (provenance stratification, feature-group ablation, and a masked-imputation control) that surfaces this hazard, the related hazard of taxonomy-derived labels, and the confounding that arises when sources cover disjoint periods. Applying it here shows that pattern prediction is largely rule recovery, and that the defensible attack-type estimate under random partitioning is 45.92% accuracy and 26.58% macro F1 rather than the pooled 60.94%

and 32.57%. Under a strictly chronological protocol, which is the condition a deployed forecaster faces, the figure falls further to 38.44% accuracy and 19.32% macro F1, with accuracy at the majority-class floor and the usable signal residing in the macro-averaged measure.

Three further results follow. Learned ensemble weighting, including CAWPE and accuracy-proportional schemes as well as our policy-gradient layer, does not improve on uniform averaging on any target, which we attribute to insufficient diversity among encoders sharing a representation. Learned entity embeddings recover 18.57 macro F1 points over label encoding at fixed architecture, indicating that deep-model underperformance on incident data is substantially a representation effect. And class cardinality, not class imbalance, bounds the country target: no model exceeds 7.32% macro F1 over 189 classes, and no feature group carries the missing signal.

Several directions follow directly. The country target should be reformulated hierarchically, predicting region or economic bloc before country, and evaluated against the flat formulation. A systematic sweep over masking fraction and imputer family would establish whether the inverse relationship between imputer fidelity and inflation holds generally, which would turn a two-point observation into a usable rule of thumb for merge pipelines. Extending the corpus beyond 2023, and filling the 2021 gap, would permit a rolling multi-year chronological evaluation and establish whether the reversal in model ordering we observe is stable across cutoffs. Finally, the provenance diagnostic should be applied retrospectively to existing merged-corpus studies, where we expect it to revise reported figures. More broadly, the protocol is domain-agnostic, and we expect it to be useful wherever data-mining results are reported on datasets integrated from heterogeneous sources.

Acknowledgements

The authors thank the Department of Computer Science, Faculty of Computing and Information Technology, International Islamic University, Islamabad, for institutional support.

Statements and Declarations

Funding

No funding was received for conducting this study.

Competing interests

The authors have no relevant financial or non-financial interests to disclose.

Ethics approval and consent to participate

Not applicable. The study uses publicly available incident records and involves no human participants.

Consent for publication

Not applicable.

Data availability

The code, attack taxonomy and per-seed result files supporting this study are openly available at https://github.com/UmaraNoor/cyber-attack-forecasting-repo. The repository contains data-acquisition and corpus-construction scripts, the recovered attack taxonomy mapping all 588 raw labels to 11 categories with each assignment tagged as empirically recovered or rule-assigned, the harmonisation, modelling and diagnostic code, per-seed result files for every table in this paper, and the figure-generation scripts. A documented command is provided for each table and figure together with the expected values, so that individual results can be verified without running the complete suite. The suite executes in approximately 90 minutes on a single CPU core and requires no GPU.

The unified corpus is not redistributed. It is derived from Hackmageddon [20], accessed 7 August 2023, and the VERIS Community Database [19], [22]; researchers should obtain both directly from their original sources and comply with the applicable terms of use and licensing conditions. The repository provides an acquisition script that retrieves the VERIS Community Database from its public repository and documents how to obtain the Hackmageddon timelines, a construction script that rebuilds the 20,279-record corpus from both, and a validation script that compares a local rebuild against the published corpus statistics (row counts, schema, era split and category counts) so that any divergence is quantified before results are reported. Because Hackmageddon is updated continuously, timelines accessed after August 2023 will yield a corpus that differs from the one analysed here; the validation script reports by how much.

Users of the corpus should note that label provenance varies by target and era, as described in Section 3.4, and that any evaluation performed on it should report metrics stratified accordingly.

Code availability

The complete code, including corpus-construction, modelling and diagnostic scripts, is available at https://github.com/UmaraNoor/cyber-attack-forecasting-repo.

Author contributions

Inbasat Fiza: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization.

Umara Noor: Conceptualization, Methodology, Supervision, Writing – review & editing, Project administration.

Zahid Rashid: Conceptualization, Methodology, Supervision, Writing – review & editing, Project administration.

References

1. Check Point Research (2024) Check Point Research reports highest increase of global cyber attacks seen in last two years: a 30% increase in Q2 2024 global cyber attacks. https://blog.checkpoint.com/research/. Accessed 10 Nov 2024
2. Embroker (2024) 2024 must-know cyberattack statistics and trends. https://www.embroker.com/blog/cyber-attack-statistics/. Accessed 10 Nov 2024
3. Carta S, Corriga A, Ferreira A, Podda AS, Recupero DR (2021) A multi-layer and multi-ensemble stock trader using deep learning and deep reinforcement learning. Appl Intell 51(2):889–905. https://doi.org/10.1007/s10489-020-01839-5

4. Grøntved S, Kirkeby MJ, Johnsen SP, Mainz J, Valentin JB, Jensen CM (2024) Towards reliable forecasting of healthcare capacity needs: a scoping review and evidence mapping. Int J Med Inform 189:105527. https://doi.org/10.1016/j.ijmedinf.2024.105527

5. Husák M, Bartoš V, Sokol P, Gajdoš A (2021) Predictive methods in cyber defense: current experience and research challenges. Future Gener Comput Syst 115:517–530. https://doi.org/10.1016/j.future.2020.10.006

6. Landauer M, Skopik F, Stojanović B, Flatscher A, Ullrich T (2025) A review of time-series analysis for cyber security analytics: from intrusion detection to attack prediction. Int J Inf Secur 24(1):3. https://doi.org/10.1007/s10207-024-00921-0

7. Werner G, Yang S, McConky K (2017) Time series forecasting of cyber attack intensity. In: Proceedings of the 12th annual conference on cyber and information security research (CISRC '17). ACM, New York, pp 1–3. https://doi.org/10.1145/3064814.3064831

8. Bitit R, Derhab A, Guerroumi M, Khan FA (2024) DDoS attack forecasting based on online multiple change points detection and time series analysis. Multimed Tools Appl 83(18):53655–53685. https://doi.org/10.1007/s11042-023-17637-3

9. Okutan A, Werner G, Yang SJ, McConky K (2018) Forecasting cyberattacks with incomplete, imbalanced, and insignificant data. Cybersecurity 1(1):15. https://doi.org/10.1186/s42400-018-0016-5

10. Ahmed Y, Azad MA, Asyhari T (2024) Rapid forecasting of cyber events using machine learning-enabled features. Information 15(1):36. https://doi.org/10.3390/info15010036

11. Wu S, Wang B, Wang Z, Fan S, Yang J, Li J (2022) Joint prediction on security event and time interval through deep learning. Comput Secur 117:102696. https://doi.org/10.1016/j.cose.2022.102696

12. Liu Y, Sarabi A, Zhang J, Naghizadeh P, Karir M, Bailey M, Liu M (2015) Cloudy with a chance of breach: forecasting cyber security incidents. In: Proceedings of the 24th USENIX security symposium. USENIX Association, Washington, DC, pp 1009–1024

13. Qasaimeh M, Abu Hammour R, Bani Yassein M, Al-Qassas RS, Lara Torralbo JA, Lizcano D (2022) Advanced security testing using a cyber-attack forecasting model: a case study of financial institutions. J Softw Evol Process 34(11):e2489. https://doi.org/10.1002/smr.2489

14. Almahmoud Z, Yoo PD, Alhussein O, Farhat I, Damiani E (2023) A holistic and proactive approach to forecasting cyber threats. Sci Rep 13:8049. https://doi.org/10.1038/s41598-023-35198-1

15. Mumtaz G, Akram S, Iqbal MW, Ashraf MU, Almarhabi KA, Alghamdi AM, Bahaddad AA (2023) Classification and prediction of significant cyber incidents (SCI) using data mining and machine learning (DM-ML). IEEE Access 11:94486–94496. https://doi.org/10.1109/ACCESS.2023.3249663

16. Zacharis A, Katos V, Patsakis C (2024) Integrating AI-driven threat intelligence and forecasting in the cyber security exercise content generation lifecycle. Int J Inf Secur 23(4):2691–2710. https://doi.org/10.1007/s10207-024-00860-w

17. Canadian Institute for Cybersecurity (2018) CSE-CIC-IDS2018 dataset. University of New Brunswick. https://www.unb.ca/cic/datasets/ids-2018.html. Accessed 10 Nov 2024

18. Canadian Institute for Cybersecurity (2019) CIC-DDoS2019 dataset. University of New Brunswick. https://www.unb.ca/cic/datasets/ddos-2019.html. Accessed 16 Dec 2024

19. Verizon (2024) VERIS: vocabulary for event recording and incident sharing; Data breach investigations report. https://www.verizon.com/business/resources/reports/dbir/. Accessed 10 Nov 2024

20. Passeri P. HACKMAGEDDON: information security timelines and statistics. https://www.hackmageddon.com/. Accessed 7 Aug 2023

21. Web Application Security Consortium. Web Hacking Incident Database (WHID). http://projects.webappsec.org/w/page/13246995/Web-Hacking-Incident-Database. Accessed 16 Dec 2024

22. VERIS Community Database (VCDB). GitHub repository. https://github.com/vz-risk/VCDB. Accessed 17 Dec 2024

23. Kaufman S, Rosset S, Perlich C (2011) Leakage in data mining: formulation, detection, and avoidance. In: Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining (KDD '11). ACM, New York, pp 556–563. https://doi.org/10.1145/2020408.2020496

24. Kapoor S, Narayanan A (2023) Leakage and the reproducibility crisis in machine-learning-based science. Patterns 4(9):100804. https://doi.org/10.1016/j.patter.2023.100804

25. Rosenblatt M, Tejavibulya L, Jiang R, Noble S, Scheinost D (2024) Data leakage inflates prediction performance in connectome-based machine learning models. Nat Commun 15:1829. https://doi.org/10.1038/s41467-024-46150-w

26. Apicella A, Isgrò F, Prevete R (2025) Don't push the button! Exploring data leakage risks in machine learning and transfer learning. Artif Intell Rev 58(11):339. https://doi.org/10.1007/s10462-025-11326-3

27. Sasse L, Nicolaisen-Sobesky E, Dukart J, Eickhoff SB, Götz M, Hamdan S, Komeyer V, Kulkarni A, Lahnakoski JM, Love BC, Raimondo F, Patil KR (2025) Overview of leakage scenarios in supervised machine learning. J Big Data 12:135. https://doi.org/10.1186/s40537-025-01193-8

28. Bouke MA, Abdullah A (2023) An empirical study of pattern leakage impact during data preprocessing on machine learning-based intrusion detection models reliability. Expert Syst Appl 230:120715. https://doi.org/10.1016/j.eswa.2023.120715

29. Guo C, Berkhahn F (2016) Entity embeddings of categorical variables. arXiv preprint arXiv:1604.06737. https://doi.org/10.48550/arXiv.1604.06737

30. Dahouda MK, Joe I (2021) A deep-learned embedding technique for categorical features encoding. IEEE Access 9:114381–114391. https://doi.org/10.1109/ACCESS.2021.3104357

31. Almaslukh B (2021) Deep learning and entity embedding-based intrusion detection model for wireless sensor networks. Comput Mater Contin 69(1):1343–1360. https://doi.org/10.32604/cmc.2021.017914

32. Grinsztajn L, Oyallon E, Varoquaux G (2022) Why do tree-based models still outperform deep learning on typical tabular data? In: Advances in neural information processing systems 35 (NeurIPS 2022). Curran Associates, Red Hook, pp 507–520

33. Borisov V, Leemann T, Seßler K, Haug J, Pawelczyk M, Kasneci G (2024) Deep neural networks and tabular data: a survey. IEEE Trans Neural Netw Learn Syst 35(6):7499–7519. https://doi.org/10.1109/TNNLS.2022.3229161

34. Avanzi B, Taylor G, Wang M, Wong B (2024) Machine learning with high-cardinality categorical features in actuarial applications. ASTIN Bull 54(2):213–238. https://doi.org/10.1017/asb.2023.29

35. Babbar R, Schölkopf B (2017) DiSMEC: distributed sparse machines for extreme multi-label classification. In: Proceedings of the tenth ACM international conference on web search and data mining (WSDM '17). ACM, New York, pp 721–729. https://doi.org/10.1145/3018661.3018741

36. Liu W, Wang H, Shen X, Tsang IW (2022) The emerging trends of multi-label learning. IEEE Trans Pattern Anal Mach Intell 44(11):7955–7974. https://doi.org/10.1109/TPAMI.2021.3119334

37. Tanha J, Abdi Y, Samadi N, Razzaghi N, Asadpour M (2020) Boosting methods for multi-class imbalanced data classification: an experimental review. J Big Data 7:70. https://doi.org/10.1186/s40537-020-00349-y

38. Partalas I, Tsoumakas G, Vlahavas I (2009) Pruning an ensemble of classifiers via reinforcement learning. Neurocomputing 72(7–9):1900–1909. https://doi.org/10.1016/j.neucom.2008.06.007

39. Savargiv M, Masoumi B, Keyvanpour MR (2021) A new random forest algorithm based on learning automata. Comput Intell Neurosci 2021:5572781. https://doi.org/10.1155/2021/5572781

40. Zhang H, Liu W, Wang S, Shan J, Liu Q (2022) Reinforcement online active learning ensemble for drifting imbalanced data streams. IEEE Trans Knowl Data Eng 34(8):3971–3983. https://doi.org/10.1109/TKDE.2020.3026196

41. Large J, Lines J, Bagnall A (2019) A probabilistic classifier ensemble weighting scheme based on cross-validated accuracy estimates. Data Min Knowl Discov 33:1674–1709. https://doi.org/10.1007/s10618-019-00638-y

42. The GDELT Project. Global database of events, language and tone (GDELT). https://www.gdeltproject.org/data.html. Accessed 16 Dec 2024

43. Liu Y, Dong M, Ota K, Li J, Wu J (2018) Deep reinforcement learning based smart mitigation of DDoS flooding in software-defined networks. In: 2018 IEEE 23rd international workshop on computer aided modeling and design of communication links and networks (CAMAD). IEEE, pp 1–6. https://doi.org/10.1109/CAMAD.2018.8514971

44. Akbari I, Tahoun E, Salahuddin MA, Limam N, Boutaba R (2020) ATMoS: autonomous threat mitigation in SDN using reinforcement learning. In: NOMS 2020 – 2020 IEEE/IFIP network operations and management symposium. IEEE, pp 1–9. https://doi.org/10.1109/NOMS47738.2020.9110426

45. Wang J, Liu J, Guo H, Mao B (2022) Deep reinforcement learning for securing software-defined industrial networks with distributed control plane. IEEE Trans Ind Inform 18(6):4275–4285. https://doi.org/10.1109/TII.2021.3128581

46. Roberts C, Ngo S-T, Milesi A, Peisert S, Arnold D, Saha S, Scaglione A, Johnson N, Kocheturov A, Fradkin D (2020) Deep reinforcement learning for DER cyber-attack mitigation. In: 2020 IEEE international conference on communications, control, and computing technologies for smart grids (SmartGridComm). IEEE, pp 1–7. https://doi.org/10.1109/SmartGridComm47815.2020.9302997

47. Wang Z, Chen Y, Liu F, Xia Y, Zhang X (2018) Power system security under false data injection attacks with exploitation and exploration based on reinforcement learning. IEEE Access 6:48785–48796. https://doi.org/10.1109/ACCESS.2018.2856520

48. Tariq I, Sindhu MA, Abbasi RA, Khattak AS, Maqbool O, Siddiqui GF (2021) Resolving cross-site scripting attacks through genetic algorithm and reinforcement learning. Expert Syst Appl 168:114386. https://doi.org/10.1016/j.eswa.2020.114386

49. Lichtman M, Reed J (2014) Reinforcement learning for reactive jamming mitigation. J Cyber Secur Mobil 3:213–230. https://doi.org/10.13052/jcsm2245-1439.325