UDC 004.05

Forecasting Cyber Threat Intelligence with Memory Augmented Transformer

Anatolii Feher[1]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic University", Kyiv, Ukraine*

Abstract

Cyber threat intelligence data are volatile, irregular, and shaped by abrupt regime shifts, making accurate forecasting particularly challenging. Motivated by this, we explore the potential of a memory-augmented Transformer forecaster that integrates an evolving memory mechanism and confidence-regulated attention. Introducing complementary design that enables the model to balance adaptability with stability, remaining robust under noise and structural changes in the threat landscape. Building on and re-architecting the original ACWA-based approach, the resulting ChronoTensor introduced enhanced model achieves parity with state-of-the-art forecasting methods while introducing transparent memory and attention pathways that enhance the interpretability and explainability of its predictions.

Keywords: time series forecasting, adaptive memory mechanisms, cyber threat intelligence, OSINT

Introduction

Through years analytics in cybersecurity has become a fundamental component of modern threat intelligence and operational risk management, as global digital infrastructure continues to expand, the frequency, complexity, and impact of cyber incidents intensify progressively, producing dynamic threat landscape. Where that exact dynamic nature of it, could be complementary assessed with analytical instruments such as forecasting addressed to define proactive defense strategies, optimizing the allocation of response resources, and improving situational awareness across many related applications.

However, forecasting cybersecurity behavior remains fundamentally challenging, where threat landscapes exhibit long-term dependence punctuated by abrupt shifts, irregular volatility driven by global events, and heterogeneous signals extracted from noisy sources. Thus traditional statistical methods such as Autoregressive Integrated Moving Average (ARIMA) have served as baselines in outlined tasks because of their stability and conservative treatment of noise [1], yet they limited for capturing complex and non-linear dependencies when regime transitions characteristic implies.

Neural sequence models, particularly recurrent networks and Transformer-based architectures, offer greater expressive power but often exhibit over-sensitivity to noise, overfitting risks, and an inability to adjust to evolving temporal structures once training has concluded.

These limitations arise from a common assumption shared across both statistical and deep learning approaches, that memory is either implicit, static, or fixed at training time, and attention when present does not dynamically regulate its forecasting behavior according to the historical relevance of observed patterns [2]. To address such shortcomings, we propose ChronoTensor, a forecasting architecture that integrates Transformer-based sequence modeling with a dynamic memory mechanism designed to operate under the constraints of long-range, non-stationary, and event-driven cybersecurity time series.

Where for validation and testing of described architecture been made on the recollected with Open Source Intelligence (OSINT) data upon the given topic with validated long-term temporal dependence, with a Hurst exponent of H=0.69, while simultaneously containing high-variance, heavy-tailed fluctuations. Which ideally fit given research objective under complicated non-stationarity, periodic regime changes, and uneven event bursts.

1. Methodology

The forecasting architecture employed in the preliminary phase of this work was based on a hybrid integration of a simplified Transformer encoder with the Adaptive Contextual Weighted Average (ACWA) mechanism [3]. In this design, the model received two inputs – the normalized series value and the ACWA-generated contextual prediction, which were embedded jointly and enriched with sinusoidal positional encodings.

Such architecture modification been inspired by the structural idea of Natural Language Processing (NLP) tokenization from Generative Pre-trained Transformer (GPT) models, leading to the introduction of a custom attention module at the first processing stage [4]. In this module, canonical self-attention scores were multiplicatively modulated by ACWA-derived pattern weights, prioritizing historically recurrent configurations over less informative segments of the sequence. However, this block lacked residual connections, normalization layers, and a feed-forward subnet, which impeded gradient stability and limited the representational capacity of the architecture.

Following this initial ACWA-augmented module, the sequence was processed by a lightweight Transformer encoder comprising two layers with multi-head self-attention, Add-Norm residual pathways, and position-wise feed-forward transformations. The final output embedding corresponding to the last timestep was mapped through a linear projection to produce the forecast. While the approach proved functionally viable, several structural limitations became evident the ACWA mechanism influenced only a single layer, its contribution was partially duplicated via the input stream, and its pattern weights remained externally computed and non-differentiable. As a result, the model lacked adaptability to benefit from ACWA's inductive bias consistently throughout the network depth. These constraints motivated the experimentation on memory-augmentation into unified Transformer architecture.

Such motivation was born through researching latest works on how memory and attention interact in modern Large Language Models (LLMs), specifically the Mistral Sliding Window Attention which showed that attention doesn't need to be uniformed and Transformer-XL with MemGPT outlining ability of treating persistent memory alongside training process [5, 6, 7].

For comprehensive empirical evaluation of the proposed methodology was conducted on a decade-long dataset derived from the GDELT Global Knowledge Graph (GKG) version 1.0, an open-source system that monitors global news media in more than one hundred languages. The corpus been extracted to achieve daily files spanning 1 January 2015 to 1 January 2025, each containing tens of thousands of extracted "namesets" representing unique co-occurrences of persons, organizations, locations, and thematic tags within individual news articles.

For each date, the final value is the sum of NUMARTS across both aggregation tracks, to extract a coherent signal of global cybersecurity discourse, a dual-track aggregation was applied [8]. The first track selected records tagged with standardized cybersecurity themes (e.g., CYBER_ATTACK, TAX_FNCACT_HACKER, SOC_RANSOM, WB_2457_CYBER_CRIME, WB_670_ICT_SECURITY), summing their NUMARTS values to obtain a daily count of articles explicitly categorized as cyber-related by GDELT's NLP pipeline.

The second track compensates for gaps in GDELT's theme tagging, since many reports on cyber operations or threat actors lack explicit cybersecurity labels. To capture these, substring matching was applied across THEMES, ORGANIZATIONS, SOURCES, and SOURCEURLS for roughly seventy known threat actors and operational keywords. This set included nation-state groups (e.g., APT28, APT29, APT41, Fancy Bear, Sandworm, Lazarus), ransomware operators (REvil, LockBit, Clop, BlackCat/ALPHV), hacktivist collectives (Anonymous, LulzSec, GhostSec, Killnet), among other actors and general terms such as cyberattack, hacktivist, and data breach.

To mitigate contamination from entertainment-driven references, all records containing lexical indicators of fictional or media-product contexts such as movie, film, videogame, hackathon, among others were systematically filtered.

The final dataset comprises 3,654 daily observations, with typical volumes ranging from 500 to 3,500 articles, where last 30 days (month) were cut for evaluating forecast performance. Achieved time-series exhibits a clear weekday-weekend seasonal pattern and a long-tailed distribution marked by episodic spikes during major cyber incidents, vulnerability disclosures, or geopolitical escalations.

2. Framework Evolution

The ChronoTensor architecture evolved from an earlier, simpler Transformer model that was designed to recognize with tokens recurring patterns in time series data. The initial Hybrid ACWA-Transformer, designed to combine a standard Transformer encoder with an additional ACWA module, which allowed to convert the time-series into discrete states based on value ranges and stored commonly observed patterns along with what typically followed them [3].

These stored patterns were then used to slightly adjust the attention scores in the first encoder layer, nudging the model toward behaviors that had appeared frequently in the past, where the final prediction was produced by combining the Transformer's output with the ACWA forecast using a linear regression model trained on a separate validation segment.

Although such design introduced basic pattern awareness, it had several clear drawbacks, such as pattern dictionary was fixed through the dataset, and weren't designed for dynamic adaptation. Along with pattern influence, which was also limited to a single attention layer, and the adjustment mechanism itself was external to the learning process and not trainable end-to-end.

In addition, the custom attention block lacked tuned components such as residual connections, normalization, and feed-forward layers, which reduced training stability and expressive power. To address these limitations been explored potential framework evolution that would not only cover them, but would introduce complementary synergy within the ACWA tokenized aggregation.

For such transition been researched the recent progress in LLMs, where researched that attention and memory could be treated not as separate mechanisms but as tightly coupled systems that guide each other [5,6]. Studies of sparse and windowed attention models such as Longformer, Informer and BigBird highlighted how models learn to focus computation on relevant regions of long sequences [9, 10].

Where complementary addition to such was works on modern memory systems, specifically including retrieval-augmented generation [11], hierarchical type [12] and summarization type of memory [13], and long-range streaming memory, showed how models preserve and reuse information across extended contexts.

Across these approaches, a consistent idea emerged memory guides attention by highlighting previously relevant situations, while attention identifies which new information is salient enough to retain. This bidirectional interaction is mediated by internal confidence signals that indicate whether the model is encountering familiar regimes or novel conditions, conceptually echoing the gating principles in recurrent architectures such as Long Short-Term Memory (LSTMs) [14] and Gated Recurrent Units (GRUs) [15], where learned gates regulate the flow, retention, and updating of memory based on contextual relevance [16]. Building on these observations, the exploration of attention–memory interaction led to the development of the Memory-Augmented Adaptive Attention (MAAA) framework, which formalizes this relationship and outlines principles for applying it to time-series forecasting.

In ChronoTensor, the MAAA framework is realized through two fundamental architectural changes that transform ACWA from a fixed heuristic into an adaptive, fully trainable system.

The first change concerns the redesign of the attention mechanism itself. The original ACWA-based attention module was restructured as a proper Transformer block by introducing residual connections, layer normalization, and feed-forward layers. Modification brought the component in line with established Transformer practice, improving gradient stability and providing sufficient representational capacity for downstream layers integration.

Beyond structural alignment, the role of ACWA within attention was fundamentally redefined. The earlier hard-coded modification of attention scores was replaced with a dual-branch attention formulation, where one branch computes standard self-attention over the encoded input, while the second incorporates pattern-based signals derived from ACWA. To mediate the interaction between these branches, the computation of pattern weights was extended to yield explicit time-step–level confidence scores. With these confidence signals condition a learnable gating mechanism that blends memory-driven and standard attention, allowing the model to regulate its reliance on historical pattern knowledge when encountering familiar contexts, and to favor broader, exploratory attention when such confidence is low. Through this mechanism, memory, attention, and

confidence become tightly coupled within a single, end-to-end trainable forecasting process.

The second architectural change addresses how patterns are stored, updated, and forgotten over time. Instead of a static pattern dictionary extracted once from the training data and kept fixed, a streaming memory system is employed that allows pattern representations and their associated confidence levels to evolve continuously during training. Patterns that recur frequently are progressively reinforced through increasing confidence, while patterns that diminish in relevance decay and are eventually pruned, which introduce dynamic memory behavior that enables ChronoTensor to adapt to regime shifts in the underlying time-series, preserving useful historical structure without overcommitting to outdated patterns.

The training procedure was adapted to support this evolving memory dynamics, where after each training batch, the streaming memory update rule recalculates pattern confidences to ensure that the model's inductive bias remains aligned with the current temporal regime. Additional stabilization measures, including gradient clipping, weight decay, a reduce-on-plateau learning rate schedule, and an extended early-stopping horizon, were incorporated to handle transitions in memory composition during training. During inference, the memory is frozen to guarantee deterministic and reproducible predictions and to prevent unintended drift.

Taken together, these changes transform the original Hybrid ACWA-Transformer into a unified and internally consistent architecture, in which attention, memory, confidence, and temporal structure are no longer loosely connected components but operate as a single computational framework.

At a behavioral level, this design enables patterns to be continuously reinforced when they recur and gradually down-weighted when they cease to appear, allowing the model's inductive bias to evolve in step with the data. This evolutionary update process, combined with confidence-based gating between memory and attention, allows ChronoTensor to adapt effectively to non-stationary temporal dynamics while maintaining a controlled balance between historical knowledge and newly observed information. Where such adaptability is particularly critical for real-world cybersecurity time-series analysis, where underlying processes shift over time, threat activity exhibits regime changes, and reliance on outdated patterns can quickly lead to misleading forecasts or delayed detection of emerging behaviors.

2.1. Streaming Memory with Decay

In the original ACWA formulation, the pattern dictionary P was constructed once using the full training dataset. Each observed token sequence:

$$p = (\tau_t, \tau_{t+1}, \dots, \tau_{t+k}), \quad (1)$$

was assigned a fixed empirical frequency and associated with all observed next-token outcomes. As a result, both pattern weights and their implied confidence values remained static throughout training.

ChronoTensor replaces this static formulation with a streaming memory system, in which the confidence assigned to each stored pattern evolves continuously during training, let:

$$c_p{}^{(t)} \in [0,1], \quad (2)$$

denote the confidence of pattern p at training step t.

Whenever a pattern is observed in the input sequence, its confidence is reinforced according to:

$$c_p{}^{(t+1)} = min(1, c_p{}^{(t)}(1+\alpha)), \quad (3)$$

where α>0 is a reinforcement coefficient controlling the rate of confidence increase.

Between occurrences, pattern confidence decays exponentially:

$$c_p{}^{(t+1)} = c_p{}^{(t)}\gamma^{\Delta t}, \quad (4)$$

where $\gamma \in (0,1)$ is a decay factor and Δt denotes the number of steps since the last occurrence of pattern p.

Patterns whose confidence falls below a minimum threshold c_{min} are removed from memory:

$$P \leftarrow \{p \in P \mid c_p{}^{(t)} \geq c_{min}\}, \quad (5)$$

This streaming consolidation mechanism ensures that memory reflects the current temporal regime by reinforcing frequently recurring structures while allowing obsolete patterns to gradually vanish. Such behavior is particularly important in cybersecurity time series, where the relevance of historical patterns can change substantially over multi-month horizons as threat activity evolves.

2.2. Gating with Attention Confidence

Given an input window of discrete tokens $\tau 1{:}T$, the memory system produces a pattern-conditioned modulation signal that reflects which stored patterns match the recent context and how confident the model currently is in those patterns.

For each timestep j, the pattern weight is defined as:

$$w_j = \frac{1}{Z}\sum_{p\in P} c_p{}^{(t)}\, 1\{p \subset (\tau_{j-k+1},\dots,\tau_j)\}, \quad (6)$$

where $1\{\cdot\}$ indicates a pattern match, $cp(t)$ is the current streaming confidence, and Z normalizes the weights such that $\sum jwj$=1.

These weights are incorporated into the attention computation additively, rather than multiplicatively, to avoid suppressing attention scores when pattern confidence is low. Specifically, the modified attention logits become:

$$\hat{A} = Softmax\left(\frac{QK^T}{\sqrt{d_k}} + W\right), \quad (7)$$

$$W_{ij} = \beta w_j,$$

where β controls the strength of memory-based modulation.

This additive formulation preserves differentiability and prevents the attention collapse observed in earlier multiplicative ACWA variants, allowing memory signals to bias attention without dominating it.

While memory-weighted attention provides useful inductive bias, its influence should depend on contextual reliability. ChronoTensor therefore introduces a dual-branch attention mechanism with an explicit gating function.

Given an input representation $X \in RT \times d$, the standard attention branch computes:

$$A_{std}(X) = = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) V, \quad (8)$$

the memory-weighted branch computes:

$$A_{mem}(X) = = softmax\left(\frac{QK^T}{\sqrt{d_k}} + W\right) V, \quad (9)$$

where W incorporates the streaming memory weights defined above.

To regulate the contribution of these branches, a learnable gate is introduced:

$$g = \sigma(W_g\,[\,X_\ell \parallel w] + b_g), \quad (10)$$

where $X\ell$ is the local representation at the final timestep, w is the vector of ACWA confidence scores, and $\sigma(\cdot)$ denotes the logistic sigmoid.

The fused attention output is then given by:

$$A_{fused} = g\,A_{mem} + (1-g)\,A_{std}, \quad (11)$$

This gating mechanism allows the model to emphasize memory-driven attention when strong, reliable patterns are present, while smoothly reverting to standard self-attention in novel or weakly structured contexts. Unlike the linear regression blending used in the original Hybrid ACWA-Transformer, gating is fully differentiable and context-dependent, enabling gradients to shape how and when memory is used.

During inference, the streaming memory is frozen ((t)=const), ensuring deterministic predictions and preventing unintended drift.

2.3. Encoder Attention under Memory

The final ChronoTensor encoder block integrates the streaming memory and confidence-based gated attention into an otherwise standard Transformer structure. Given an input X, the encoder block is defined as:

$$X' = LayerNorm(X + A_{fused}(X)), \quad (12)$$

$$X'' = LayerNorm(X' + FFN(X')), \quad (13)$$

Predictions are computed from the representation at the final timestep:

$$\hat{x}_{T+1} = W_o\,X_T{}'' + b_o, \quad (14)$$

Thus MAAA formulation retains only the two components that demonstrated consistent empirical benefit on cybersecurity datasets, a streaming memory with reinforcement, decay, and pruning, and confidence-based gated attention regulating the interaction between memory and learned representations.

2.4. Resulting Architecture

ChronoTensor preserves the core structure of the Transformer encoder while extending it with confidence-regulated memory mechanism derived from ACWA. Standard self-attention, residual connections, layer normalization, and feed-forward sublayers remain intact, ensuring architectural compatibility and stable training.

Within this framework, ACWA functions as a streaming pattern memory whose influence on attention is mediated by a learnable gate conditioned on contextual confidence.

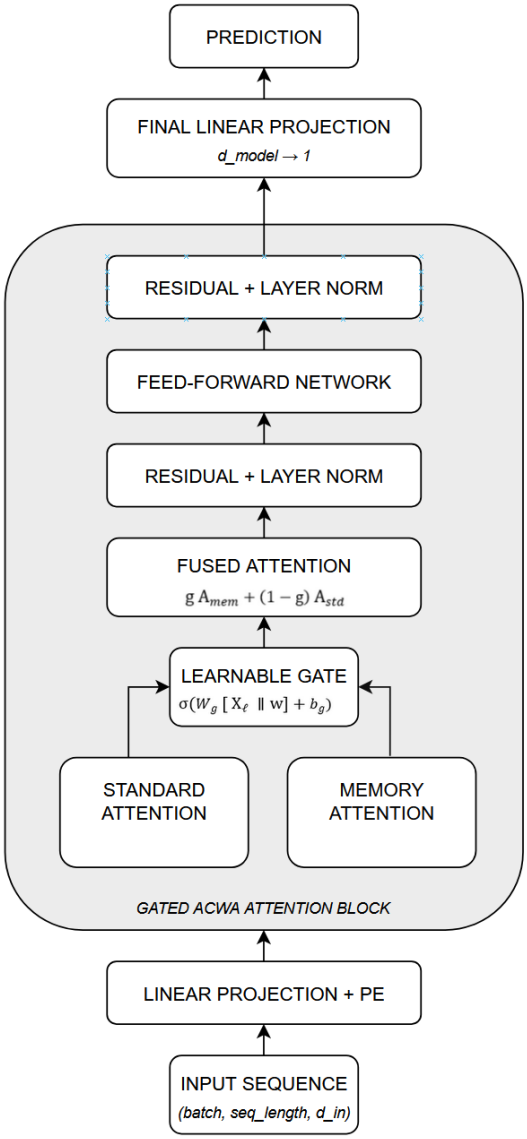

Figure 1: Complete ChronoTensor architecture of encoder block with confidence-gated and memory-conditioned attention.

3. Results and Discussion

The performance of the proposed memory-augmented ChronoTensor architecture was evaluated against seven established forecasting methods, including classical statistical baselines, recurrent neural networks, feed-forward deep learning models, and recent Transformer-based designs. The forecasting task was conducted on a ten-year OSINT cybersecurity dataset, where last month (30 days) of it were cut for the prediction efficiency assessment. The quantitative results are summarized in Table 1 and Table 2.

Table 1
Error Metrics Comparison

	RMSE	MAE	MAPE	SMAPE
NBEATS	1027.3	723.0	46.2%	30.7%
LSTM	1016.0	809.5	57.4%	37.6%
ARIMA	986.3	707.5	46.6%	31.7%
PatchTST	807.5	538.1	33.1%	24.4%
ChronoTensor	782.4	572.1	38.5%	29.0%

Table 2
Quality Metrics Comparison

	R^2	Bias	Dir. Acc
NBEATS	0.093	424.3	75.9%
LSTM	0.113	551.7	51.7%
ARIMA	0.164	254.3	62.1%
PatchTST	0.439	111.6	69.0%
ChronoTensor	0.474	31.7	69.0%

ChronoTensor achieves the lowest RMSE within the entire comparison set, demonstrating a 20.7% improvement over ARIMA, 23.0% over LSTM, 23.8% over Neural Basis Expansion Analysis for Time Series (N-BEATS). With respect to the 2023 state-of-the-art PatchTST model, ChronoTensor attains a 3.1% RMSE improvement while also achieving the highest explained variance ($R2$=0.474), indicating parity capabilities in modeling both the persistent and event-driven components of the underlying series.

Although PatchTST attains lower MAE, MAPE, and SMAPE, where these metrics emphasize proportional or local deviations rather than global forecast error. In contrast, RMSE and $R2$ capture performance in the presence of large-scale fluctuations precisely the behavior exhibited by cybersecurity OSINT time series due to episodic spikes associated with threat disclosures, zero-day exploitation surges, or coordinated attack campaigns.

In this regard, ChronoTensor demonstrates a stronger capacity to recover the underlying structure of the time series while mitigating sensitivity to noise-driven outliers.

Traditional pattern-based forecasting methods rely on static pattern dictionaries or fixed statistical assumptions about seasonality and trend. In contrast, the streaming memory mechanism continuously updates pattern confidence values according to reinforcement and decay dynamics:

$$c_p^{(t+1)} = \begin{cases} c_p^{(t)} \gamma^{\Delta t} & , \\ min\left(1, c_p^{(t)}(1+\alpha)\right), \end{cases} \quad (15)$$

where α denotes reinforcement strength, γ the decay coefficient, and Δt the absence interval, and if pattern occurs – otherwise.

This mechanism enables the model to strengthen recurrent temporal structures while gradually diminishing the influence of outdated patterns. In cybersecurity applications where the operational environment is shaped by rapidly changing geopolitical conditions, emergent vulnerabilities, and evolving threat-actor behavior such adaptivity is essential for maintaining temporal relevance.

Among the baselines, N-BEATS achieved the highest directional accuracy, reflecting its ability to capture turning points. However, its aggressive decomposition tends to amplify short-term fluctuations rather than suppress them, which reduces robustness in highly volatile sequences. LSTM, despite its recurrent formulation, struggles to maintain meaningful dependencies across decade-long series and exhibits susceptibility to overfitting noise-driven variance. More broadly, static approaches such as ARIMA, LSTM, and conventional Transformer variants lack the ability to adjust their internal representation of historical relevance once training is complete, where introduced ChronoTensor is continuously re-weighted historical prior, enabled by streaming memory, empirically improves generalization across evolving temporal regimes.

ChronoTensor's learned representations exhibit smoothness and robustness forecasts follow the dominant long-term signal rather than reacting to stochastic perturbations. This directly contributes to the model's superior RMSE performance. Notably, ARIMA traditionally a strong baseline for noise-dominated series performs better than most deep learning models due to its conservative smoothing behavior. Yet, its performance deteriorates under non-stationary conditions and in the presence of event-driven regime changes, which exceed the assumptions of differenced linear models.

To address these limitations, ChronoTensor incorporates a confidence-based gating mechanism that combines memory-weighted and standard attention:

$$A_{fused} = g\, A_{mem} + (1-g)\, A_{std}\,, \quad (16)$$

where the gating term $g \in (0,1)$ is produced by a sigmoid-activated transformation of the local representation and corresponding memory confidence values.

High-confidence contexts emphasize memory-guided attention, yielding stable forecasts aligned with reinforced historical patterns. Low-confidence contexts reduce reliance on memory and broaden the contribution of standard attention, improving adaptability in the presence of novel or weakly expressed structures.

This gating mechanism effectively encodes a learned forecasting risk model under uncertainty, the model behaves conservatively, while strong pattern confidence encourages the exploitation of structured regularities. The resulting behavior reflects the classical statistical philosophy of selectively smoothing the series while preserving long-term directional structure closely resembling the forecasting principles embodied in ARIMA yet realized here through a neural architecture. This property explains why ChronoTensor attains the highest $R2$ value among evaluated methods it does not merely fit pointwise deviations but captures structural variance of threat landscape.

PatchTST remains a strong baseline and performs particularly well on proportional error metrics (MAE, MAPE, SMAPE), which emphasize short-term pointwise accuracy. These metrics tend to favor models responsive to local fluctuations, even when such fluctuations are partially noise-driven. However, for datasets characterized by extreme spikes, heavy-tailed deviations, and irregular event cascades conditions typical of cyber threat reporting global error measures such as RMSE and explained variance ($R2$) provide a more meaningful assessment of forecasting robustness, where specifically on these metrics, ChronoTensor demonstrates improved

performance because its architecture prioritizes reliable long-term signal extraction over sensitivity to short-term volatility, shown in Figure 2.

structural trends while mitigating the influence of noise-driven fluctuations behavior aligned with the principles of classical statistical forecasting, yet realized here through a neural architecture.

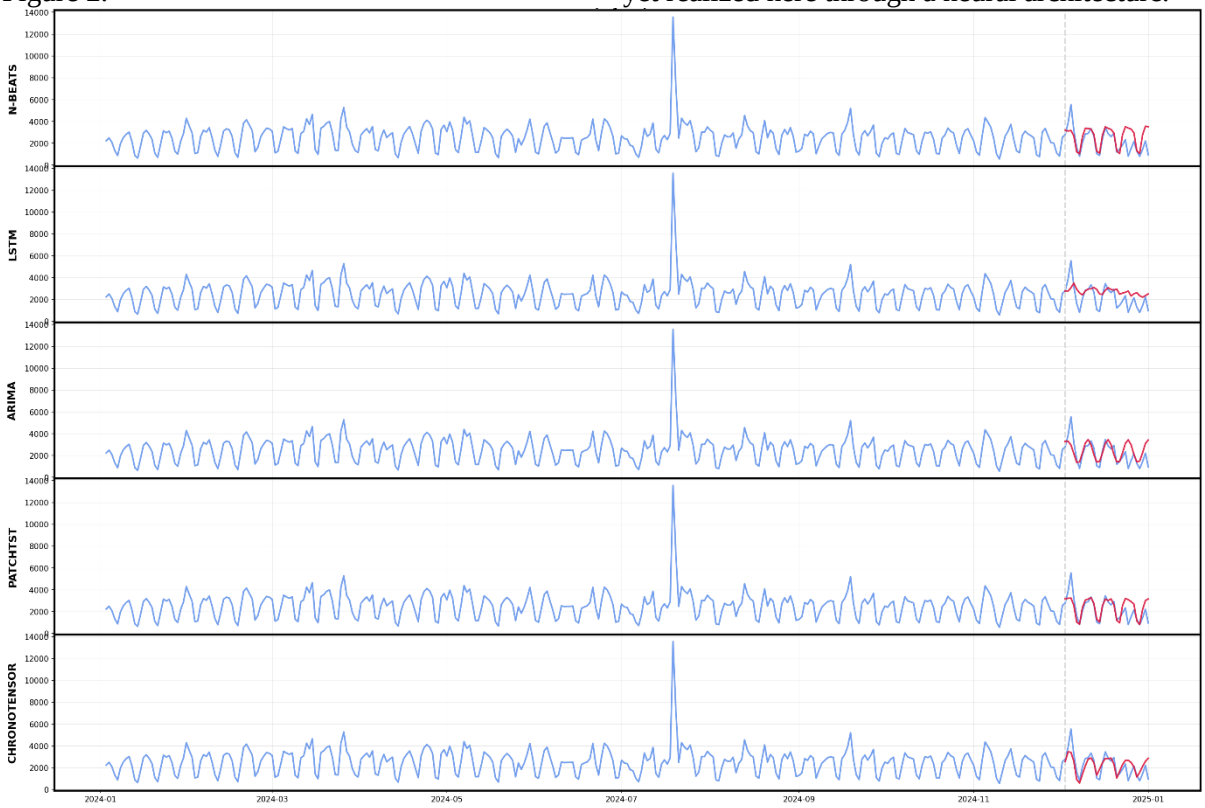

Figure 2: The rows show applied forecast methods by chosen dataset: actual time series – blue, predicted – red, last month segment – grey cut, and truncated previous years for better representation – light blue.

Conclusions

This work introduced ChronoTensor, a memory-augmented Transformer designed for forecasting long-horizon, high-volatility cybersecurity activity derived from OSINT-based threat intelligence time series. The architecture departs from conventional forecasting paradigms by integrating an explicit, evolving memory mechanism directly into the attention computation, enabling effective operation under the non-stationary and event-driven structure of cyber threat intelligence data.

Extensive evaluation on a decade-long dataset shows that ChronoTensor reaches competitive parity with state-of-the-art level performance, improving RMSE over ARIMA, LSTM, N-BEATS, and even PatchTST, while achieving the highest explained variance. Despite its compact trained size (approximately 200k parameters), ChronoTensor consistently captures underlying

Where distinctive advantage of ChronoTensor lies actually in its explainability, an area where most modern deep learning forecasters offer limited transparency. The streaming memory subsystem provides explicit insight into which historical patterns remain influential, how their confidence evolves over time, and how these patterns contribute to the final prediction. The gating mechanism further enables inspection of when the model prioritizes memory-derived structure versus when it relies on raw attention to accommodate novel or uncertain conditions. This yields a transparent decision process that can be examined at each timestep, in contrast to models such as PatchTST or N-BEATS, whose internal representations lack interpretable counterparts. As a result, ChronoTensor offers not only improved accuracy but also enhanced auditability, an essential property for threat intelligence and cybersecurity applications.

Taken together, the findings show that, ChronoTensor bridges classical statistical intuition with the flexibility of deep learning, demonstrating that reliability and adaptability can coexist.

Future work may extend this framework to present multivariate cybersecurity indicators incorporating additional contextual and semantic metadata from threat intelligence sources [17] for cross-modal forecasting exploration.

References

[1] Time Series Analysis: Forecasting and Control / G. E. P. Box, G. M. Jenkins, G. C. Reinsel, G. M. Ljung. - 5th. - 2015. - 712 p.

[2] Attention is All You Need / A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin // Advances in Neural Information Processing Systems (NeurIPS). - 2017. - P. 5998–6008.

[3] Anatolii Feher / Forecasting Information Operations with Hybrid Transformer Architecture // Vol. 6 No. 2: Theoretical and Applied Cybersecurity. -2024. -P 1-5.

[4] Radford A., Narasimhan K., Salimans T., Sutskever I. Improving Language Understanding by Generative Pre-Training // OpenAI Technical Report. — 2018.

[5] Jiang A. Q., Sablayrolles A., Mensch A., Bamford C., Chaplot D. S., de las Casas D., Bressand F., Lengyel G., Lample G., Saulnier L., Lavaud L. R., Lachaux M.-A., Stock P., Le Scao T., Lavril T., Wang T., Lacroix T., El Sayed W. Mistral 7B // arXiv preprint arXiv:2310.06825. – 2023.

[6] Dai Z., Yang Z., Yang Y., Carbonell J., Le Q. V., Salakhutdinov R. Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL). – 2019. – arXiv:1901.02860.

[7] Packer C., Wooders S., Lin K., Fang V., Patil S. G., Stoica I., Gonzalez J. E. MemGPT: Towards LLMs as Operating Systems // arXiv preprint arXiv:2310.08560. – 2024.

[8] Agrawal G., Pal K., Deng Y., Liu H., Baral C. AISecKG: Knowledge Graph Dataset for Cybersecurity Education // Proceedings of the AAAI 2023 Spring Symposium on Challenges Requiring the Combination of Machine Learning and Knowledge Engineering (AAAI-MAKE 2023). – San Francisco, USA. – 2023.

[9] Li Y., Wehbe R. M., Ahmad F. S., Wang H., Luo Y. Clinical-Longformer and Clinical-BigBird: Transformers for Long Clinical Sequences // arXiv preprint arXiv:2201.11838. – 2022. – DOI: 10.48550/arXiv.2201.11838.

[10] Zhou H., Zhang S., Peng J., Zhang S., Li J., Xiong H., Zhang W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting // Proceedings of the AAAI Conference on Artificial Intelligence. — 2021. — P. 11106–11115.

[11] Lewis P., Perez E., Karpukhin V., et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks // Advances in Neural Information Processing Systems (NeurIPS). – 2020.

[12] He Z., Cao Y., Qin Z., Prakriya N., Sun Y., Cong J. HMT: Hierarchical Memory Transformer for Efficient Long Context Language Processing // Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT). – 2025. – P. 8068–8089.

[13] Xiao G., Tian Y., Chen B., Han S., Lewis M. Efficient Streaming Language Models with Attention Sinks // Proceedings of the International Conference on Learning Representations (ICLR). – 2024. – arXiv:2309.17453.

[14] Hochreiter S., Schmidhuber J. Long Short-Term Memory // Neural Computation. – 1997. – Vol. 9, No. 8. – P. 1735–1780. – DOI: 10.1162/neco.1997.9.8.1735.

[15] Chung J., Gulcehre C., Cho K., Bengio Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling // arXiv preprint arXiv:1412.3555. – 2014. – Presented at the NIPS 2014 Deep Learning and Representation Learning Workshop. – DOI: 10.48550/arXiv.1412.3555.

[16] Furizal F., Fawait A. B., Maghfiroh H., Ma'arif A., Firdaus A. A., Suwarno I., Hide. Long Short-Term Memory vs Gated Recurrent Unit: A Literature Review on the Performance of Deep Learning Methods in Temperature Time Series Forecasting // International Journal of Robotics and Control Systems. – 2024. – Vol. 4, No. 3. – P. 1506–1526. – DOI: 10.31763/ijrcs.v4i3.1546.

[17] D. Lande L. Strashnoy GPT Semantic Networking: A Dream of the Semantic Web - The Time is Now. - ISBN 978-966-2344-94-3. - 2023. - 168 p.