BDAA 2026 Gran Canaria

Big Data Analytics & Applications

Proceedings of the 2nd International Conference on Big Data Analytics & Applications (BDAA' 2026)

28-30 October 2026
Las Palmas de Gran Canaria, Spain

IFSA

Big Data Analytics & Applications

Proceedings of the 2nd International Conference on Big Data Analytics & Applications (BDAA' 2026)

28-30 October 2026

Las Palmas de Gran Canaria, Spain

Edited by Sergey Y. Yurish

IFSA Publishing

Sergey Y. Yurish, *Editor*
BDAA' 2026 Conference Proceedings

Copyright © 2026
by International Frequency Sensor Association (IFSA) Publishing, S. L.

E-mail (for orders and customer service enquires): ifsa.books@sensorsportal.com

Visit our Home Page on http://www.sensorsportal.com

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (IFSA Publishing, S. L., Barcelona, Spain).

Neither the authors nor International Frequency Sensor Association Publishing accept any responsibility or liability for loss or damage occasioned to any person or property through using the material, instructions, methods or ideas contained herein, or acting or refraining from acting as a result of such use.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identifying as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

ISBN: 978-84-09-93454-6
BN-20261020-XX
BIC: GPH

Contents

Foreword **4**

Comparison of Open-Weight Text-to-Image Model Deployment Configurations for Large-Scale Local Image Generation **5**
Y. S. Lee, H. Bothe and M. Geierhos

Big Data in the Process of Graphene Commercialization in Medical Applications: A Case Study **8**
A. Żołnierski, J. Gulczyński and D. Jaruga

Forecast-Bounded Multidimensional Rate Limiting: A Self-Hosted Control for the Detection Gap in Commercial Bot Management **12**
S. Brahmbhatt and V. Gangadhara

A Multi-Tier Framework for Adversarial-Resilient Detection of Unwanted Business Communication at Scale **18**
Kamya Batra

Enhanced Privacy and Fidelity Evaluation of LLM-Generated Cybersecurity Tabular Data **22**
G. Giacomello, O. Preka

Foreword

It is my great pleasure to introduce the Proceedings of the 2nd International Conference on Big Data Analytics & Applications (BDAA' 2026), held on 28–30 October 2026 in Las Palmas de Gran Canaria, Spain. The conference provides an international forum for researchers, engineers, data scientists, and industry professionals to exchange ideas and discuss the practical and scientific challenges of turning large-scale data into reliable knowledge and effective decisions.

The importance of big data analytics extends far beyond the ability to collect and store vast quantities of information. Its real value lies in extracting meaningful patterns from heterogeneous, rapidly changing, and often incomplete data, while ensuring that analytical results remain accurate, reproducible, secure, and useful. The continued development of artificial intelligence, machine learning, statistical methods, and scalable computing infrastructures creates new possibilities for research and innovation, but also raises demanding questions about computational efficiency, data governance, robustness, and responsible implementation.

The papers presented in these Proceedings demonstrate the breadth of contemporary big data applications. One contribution examines the infrastructure requirements and performance trade-offs of open-weight text-to-image models for large-scale local image generation. Another explores the use of artificial intelligence and heterogeneous information resources to guide research and innovation, focusing on the commercialization potential of graphene-based materials for medical applications. Two further studies address important challenges in digital security: forecast-based, multidimensional rate limiting to mitigate automated traffic, and a multi-tier analytical framework for detecting unwanted business communication at scale.

Although these studies concern different application domains, they share a central objective: moving from theoretical methods to systems capable of supporting real operational needs. Their approaches highlight the significance of carefully defined evaluation criteria, appropriate model selection, interpretable results, and the integration of analytical methods with domain-specific expertise. They also remind us that performance cannot be evaluated through a single measure. Accuracy, throughput, memory requirements, responsiveness, resilience to adversarial behaviour, and the consequences of incorrect decisions may all influence the suitability of a solution for practical deployment.

Another important message of BDAA' 2026 is that technological progress must be accompanied by scientific and professional responsibility. The use of large datasets and increasingly autonomous analytical systems calls for continuous attention to privacy, transparency, validation, and security. Genuine progress depends not only on more powerful algorithms, but also on a clear understanding of their assumptions, limitations, and broader implications for industry, research, and society.

On behalf of the Organizing Committee, I would like to thank all authors for their valuable contributions and for sharing their research with the international community. I am equally grateful to the reviewers and members of the International Program Committee for their careful assessments, and to the colleagues and supporting organizations whose dedication made the preparation of this conference and its Proceedings possible.

I hope that this volume will stimulate further scientific discussion, encourage new interdisciplinary collaborations, and contribute to the development of big data technologies that are efficient, trustworthy, and genuinely beneficial in real-world applications.

Prof., Dr. Sergey Y. Yurish
BDAA' 2026 Chairman

(016)

Comparison of Open-Weight Text-to-Image Model Deployment Configurations for Large-Scale Local Image Generation

Y. S. Lee, H. Bothe and M. Geierhos
University of the Bundeswehr Munich, RI CODE, Werner-Heisenberg-Weg 39,
85579 Neubiberg, Germany
E-mail: {yeongsu.lee, hendrik.bothe, michaela.geierhos}@unibw.de

Summary: This paper compares the practical deployment configurations of four open-weight text-to-image models – SDXL Base (SDXL), Stable Diffusion 3.5 Large (SD3.5), FLUX.2-klein-9B (FLUX), and Qwen-Image (QWEN) – for large-scale local image generation. Using 7992 structured portrait prompt instances per model, derived from 999 structured facial descriptions, we measured inference latency, throughput, peak allocated GPU memory, and cumulative inference time on a single NVIDIA A100 80 GB GPU. The models were evaluated at fixed model-specific inference configurations; consequently, the results characterize the tested deployment configurations rather than intrinsic model efficiency. Under these configurations, FLUX, executed with four inference steps, achieved the highest throughput (28.76 images per minute) and the lowest cumulative inference time (4.63 hours), while SDXL, executed with 30 inference steps, required the least peak allocated GPU memory (10.5 GB). SD3.5 and QWEN, both executed with 30 inference steps, required approximately 34 hours of cumulative inference time, with QWEN reaching a peak allocated GPU memory of 57.9 GB. These findings illustrate substantial latency-memory trade-offs and emphasize the importance of infrastructure characteristics when selecting text-to-image foundation model configurations for large-scale local deployment.

Keywords: Text-to-Image generation, Inference latency, Throughput, GPU memory, Local deployment.

1. Introduction

Open-weight text-to-image foundation models are becoming increasingly photorealistic. However, their computational requirements differ substantially. When it comes to large-scale local deployment, selecting a model depends on more than just generation quality. Infrastructure characteristics, such as inference latency, throughput, GPU memory consumption, and cumulative processing time, also play a role.

This study presents a controlled empirical evaluation of four contemporary open-weight text-to-image foundation models under identical hardware conditions, with a particular focus on infrastructure requirements. The aim is not to determine an absolute ranking of the models or to propose a general methodology for text-to-image efficiency evaluation, but rather to quantify the computational costs of deployment configurations and pinpoint practical trade-offs for large-scale local image generation.

2. Experimental Setup

The controlled experimental workload is based on a corpus of 999 structured facial descriptions established in our previous study [1]. Each description consists of seven sentences representing predefined semantic components. From each description, eight prompt variants were constructed: the original description (V1), six ablated variants (V2-V7) obtained by removing one of the sentences at a time, and a minimal variant (V8) containing only the first sentence that includes demographic information and emotional expression. This yields eight systematically related prompts per description and, consequently, 7992 structured prompt instances per model. Across the four evaluated text-to-image configurations, the experimental workload comprises 31968 generation instances in total. The semantic components and the construction of the individual prompt variants are described in detail in our related study [2]. An example of a structured facial description is shown in Fig. 1.

> *A profile portrait of a 34-year-old woman, white, with a happy expression. The hair appears as deep brown hair, complemented by bangs. Grooming details indicate clean-shaven. Facial features include slightly full lips, softly contoured face shape, slim facial contour, balanced nose proportions, no double chin, and not bushy eyebrows, while more subtle traits such as subtle cheekbone definition, subtle pointy nose, subtle under-eye bags, and slightly arched eyebrows are also present. Skin and complexion details show subtle rosy cheeks and subtle paleness with bold lipstick and heavy makeup, and the overall expression is characterized by mouth firmly closed and subtle smile. Accessories include bare face, uncovered head, wearing a necklace, and without a tie.*

Fig. 1. Example of a structured facial description.

All images were generated at a resolution of 1024 × 1024 pixels under the hardware and software configuration summarized in Table 1, which is also used in our related study [2]. The two studies address complementary research questions: our related study investigates how semantic component ablation affects

portrait generation quality, whereas the present study focuses on the computational and infrastructure implications of the resulting large-scale workload, including inference latency, throughput, GPU memory allocation, and cumulative inference time.

Table 1. Experimental hardware and software configurations.

Component	Specification
GPU	NVIDIA A100 80 GB
CUDA	13.0
PyTorch	2.11
Diffusers	0.39.0
Precision	bfloat16 / float16
Batch size	1
Random seed	42

The evaluated model configurations are SDXL [3], SD3.5 [4], FLUX [5], and QWEN [6]. All model weights were loaded locally, and inference was performed using the Hugging Face Diffusers framework [7]. SDXL, SD3.5, and QWEN were executed with 30 inference steps, whereas FLUX used four steps according to the evaluated inference configuration. Note that Stability AI's official SD3.5-Large example uses 28 steps and the Diffusers Qwen-Image pipeline defaults to 50 steps. For both SD3.5 and Qwen-Image, 30 steps were chosen to ensure comparability with SDXL and to limit computational intensity. The analysis considers inference latency, generation throughput, peak allocated GPU memory, and cumulative inference time. Peak allocated GPU memory was measured using torch.cuda.max_memory_allocated() and was recorded per model run. This metric captures peak tensor memory allocated by PyTorch and does not include all CUDA context and caching overheads. Additionally, the V1, V4, and V8 variants were selected as representative input configurations to examine whether substantial reductions in prompt content and length affect inference latency. The V1 represents the complete original description, the V4 the strongest single-component reduction through removal of the fourth sentence, the longest semantic component, and the V8 the minimal first sentence description. Thus, the three variants provide full, intermediate, and minimal textual workloads while avoiding an exhaustive latency comparison across all eight semantically related variants.

3. Results and Discussion

Fig. 2 summarizes the infrastructure characteristics of the evaluated configurations.

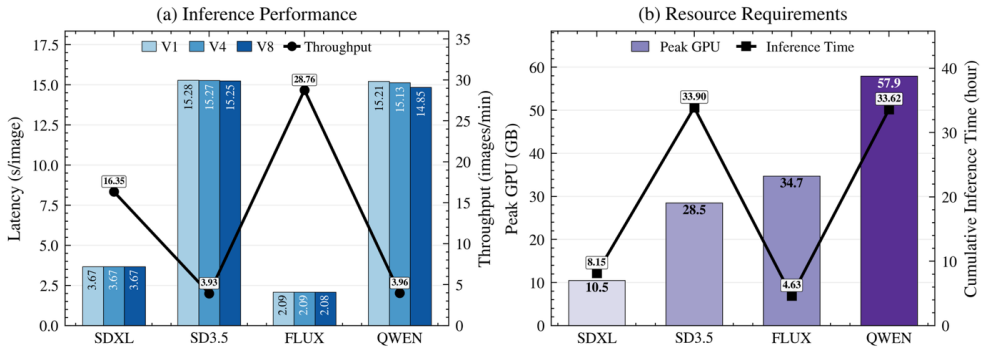


Fig. 2. Infrastructure characteristics of the evaluated text-to-image model configurations: (a) inference performance for representative description variants and throughput; (b) peak allocated GPU memory and cumulative inference time.

Fig. 2(a) shows the mean inference latency of the complete description (V1), the ablated description (V4), and the minimal reference description (V8). Within each model, latency varies slightly across these variants. This suggests that variation in the textual input contributes relatively little to total generation latency compared to the differences between the generation pipelines within the evaluated description range. Under the deployment configurations, FLUX, executed with four inference steps, achieved a mean latency of approximately 2.09 seconds per image and the highest throughput of 28.76 images per minute. SDXL, executed with 30 inference steps, required approximately 3.67 seconds per image and achieved a throughput of 16.35 images per minute, whereas SD3.5 and QWEN, both executed with 30 inference steps, required approximately 15.2 seconds per image, corresponding to a throughput of about 3.9 images per minute. Fig. 2(b) illustrates the allocated GPU memory and cumulative inference times. SDXL had the lowest peak allocated GPU memory, at 10.5 GB. Next was SD3.5, at 28.5 GB, followed by FLUX, at 34.7 GB. QWEN required 57.9 GB. Processing all 7992 generation requests required approximately 4.63 hours of cumulative inference time with FLUX and 8.15 hours of cumulative inference time with SDXL, compared to about 34 hours of cumulative inference time with SD3.5 and QWEN. Under the deployment configurations, Fig. 2 emphasizes the latency-memory trade-off: FLUX provided the lowest

latency and total execution time, while SDXL provided a substantially lower memory footprint. SD3.5 and QWEN imposed considerably higher cumulative inference costs; QWEN required the highest peak allocated GPU memory.

Furthermore, Fig. 3 provides qualitative context by presenting portraits generated by all four model configurations (SDXL, SD3.5, FLUX, and QWEN) based on an identical structured facial description. The visible differences between the generated outputs demonstrate that computational efficiency alone cannot determine a model's overall suitability. These illustrations were provided for comparison purposes only and did not represent a systematic evaluation of generation quality.



Fig. 3. Exemplary portraits generated from model configurations using the prompt in Fig. 1 at a resolution of 1024 × 1024 pixels.

Beyond the performance considerations discussed above, facial portrait generation also raises potential ethical and privacy concerns. Although the present study uses structured descriptions as generation inputs and focuses exclusively on infrastructure performance, synthetic facial outputs may reflect biases in the underlying models and should not be assumed to be free of privacy implications. The experiments are therefore intended for controlled comparative evaluation rather than unrestricted facial-data generation.

4. Conclusion and Future Work

The evaluation revealed substantial differences in the infrastructure requirements of contemporary open-weight, locally deployable text-to-image foundation model configurations. Under the evaluated configurations, FLUX produced the lowest measured inference latency and the least cumulative inference time. In contrast, SDXL had the lowest peak allocated GPU memory. While these differences appear moderate at the single-image level, they accumulate considerably across thousands of generation requests. This demonstrates the relevance of infrastructure characteristics for large-scale, single-GPU image generation under the evaluated configurations.

However, computational efficiency represents only one dimension of practical model selection. Future work will therefore expand the analysis to include a joint quality-efficiency evaluation. This evaluation will combine latency, throughput, memory consumption, and computational cost with systematic measures of perceptual image quality and text-image correspondence. This analysis could provide a more comprehensive evaluation of quality-efficiency trade-offs in large-scale text-to-image applications. Additional experiments across different GPU architectures, batch sizes, quantized inference configurations, and energy measurements will further investigate scalability and deployment efficiency. Furthermore, the interaction between prompt complexity and computational cost warrants systematic investigation, as the present results suggest that variation in semantic input content has only a minor effect on inference latency compared to the differences between the evaluated model configurations.

Acknowledgements

This research is partly funded by dtec.bw – Digitalization and Technology Research Center of the Bundeswehr. dtec.bw is funded by the European Union – NextGenerationEU.

References

[1]. Y. S. Lee, H. Bothe, M. Geierhos, Template-driven multimodal face pseudonymization for privacy-preserving big data analytics, *Algorithms*, Vol. 19, Issue 3, 2026, 176.

[2]. Y. S. Lee, H. Bothe, M. Geierhos, The impact of semantic components on the generation of text-to-image portraits, in *Proceedings of the Asia Symposium on Image and Graphics (ASIG)*, 2026 (in print).

[3]. D. Podell, Z. English, K. Lacey, A. Blattmann, et al., SDXL: Improving latent diffusion models for high-resolution image synthesis, in *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024.

[4]. P. Esser, S. Kulal, A. Blattmann, R. Entezari, et al., Scaling rectified flow transformers for high-resolution image synthesis, in *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024, pp. 12606-12633.

[5]. FLUX.2 [klein]: Towards interactive visual intelligence, Black Forest Labs, 2025, https://bfl.ai/blog/flux2-klein-towards-interactive-visual-intelligence

[6]. Qwen Team, Qwen-Image technical report, *arXiv*, 2025, arXiv:2508.02324.

[7]. Hugging Face Diffusers documentation, Hugging Face, https://huggingface.co/docs/diffusers/v0.39.0/en/

(017)

Big Data in the Process of Graphene Commercialization in Medical Applications: A Case Study

A. Żołnierski [1], J. Gulczyński [2] and D. Jaruga [3]
[1] Institute of Economics of the Polish Academy of Sciences, Nowy Świat 72, 00-330 Warsaw, Poland
[2] Division of Pathology and Neuropathology, Medical University of Gdansk,
Nowe Ogrody 1-6, 80-803 Gdansk, Poland
[3] Department of Information Technology, Faculty of Political Science and International Studies,
University of Warsaw, Krakowskie Przedmieście 26/28, Warsaw Poland
Tel.: + 48 226572707
E-mail: aleksander.zolnierski@inepan.waw.pl

Summary: This paper analyses the role of artificial intelligence (AI) and Big Data (BD) in supporting medical R&D activities through advanced analytics of large, heterogeneous, and predominantly unstructured datasets. It presents a methodological framework integrating natural language processing, machine learning, and statistical inference to transform dispersed textual and numerical information into actionable knowledge for strategic R&D management. The approach enabled the identification of promising technological pathways, including the biomedical potential of graphene-based materials, and contributed to the development of a patented invention aimed at potential cancer therapy applications. The study demonstrates that AI-supported analysis can improve decisions within innovation process. Patent analytics further strengthens analysis by supporting assessments of technological maturity, competitive positioning, patentability, and intellectual property risks. The most effective model is integration of computational intelligence with domain-specific human expertise rather than its replacement.

Keywords: Artificial intelligence, Big data analytics, Medical R&D, Patent analytics, Graphene-based biomedical applications, R&D management, Evidence-based innovation.

1. Introduction

For nearly a decade, methods and techniques have been developed to effectively generate useful information for institutions supporting research and development processes. New methods for refining information from dispersed and heterogeneous sources enable the use of up-to-date data in innovation processes, the development of inventions, and the discovery of new applications. They also enable the characterization of selected aspects of reality at a given point in time. This can support the development of new technologies, including those used in medicine and therapeutics, and improve the quality of scientific knowledge.

Our paper examines artificial intelligence (AI) and Big Data (BD) role in analytics for supporting research and development processes in medicine. We present a case study of AI and BD applications that were used at an earlier stage to identify technology trends emerging from large, heterogeneous, and largely unstructured information resources. Our case study uses an approach that integrates natural language processing, machine learning, and statistical inference to transform distributed text and numerical resources into operational knowledge relevant to strategic R&D management [2].

Applications of artificial intelligence are driving advances in nanomedicine. These encompass research drawing on the results of autonomous experimentation, predictive tools, and improvements in decision-making concerning research, development, and technology [5].

The integration of artificial intelligence and machine learning into medical practice has fundamentally transformed healthcare delivery worldwide. Artificial intelligence and Big Data analytics have facilitated clinical decision support, population health management, and improvements in healthcare operations. They have enabled faster and more accurate diagnosis and individualized treatment, particularly in atypical cases. They also support the efficient management of the growing volume of clinical information generated during diagnosis and treatment [11].

Artificial intelligence has applications across several areas of medical practice. These include medical research and drug discovery, clinical decision-making based on large datasets, image analysis, and diagnostics.

2. Methodology

The analytical project combined information refinement, natural language processing and big data analytics to identify technological trends and support research and development decisions. The workflow comprised research scope definition, source selection, automated data acquisition, text preprocessing, statistical analysis and reporting. In the present application, the analysis focused on the potential practical applications of graphene.

At the outset, the research problem and theoretical framework informed the definition of source characteristics and search criteria. These criteria addressed both the research domain and the types of information required. Sources were selected with researcher involvement to align the collected material with the study objectives. The source framework encompassed scientific publications and patent records. For the graphene application, the scientific literature component included nearly one thousand articles published in journals with high impact factors.

Data acquisition was performed using a modular information system comprising source identification, metadata analysis, data collection, monitoring and reporting modules. Each agent maintained an internal database of previously acquired records, limiting redundant retrieval and improving collection efficiency.

The retrieved documents were transferred to an analytical data warehouse hosted across multiple servers. Each record contained the textual material and associated metadata, including source identification, publication date and language. Where available, additional metadata described authorship, publication scope and document modification dates. Data replication and backups supported integrity and operational continuity, while dedicated software updated the warehouse with newly acquired material. A layered architecture and modular design enabled the system to accommodate additional sources and increasing data volumes.

Before analysis, documents were converted into machine-readable text, encoded in UTF-8 and checked for completeness. Text preprocessing comprised tokenisation, lemmatisation and keyword extraction. Natural language processing and supervised and unsupervised machine learning were used to support document classification. These procedures transformed unstructured textual material into structured datasets suitable for quantitative analysis.

The resulting datasets were analysed using R-based tools. Descriptive statistics, cluster analysis and predictive modelling were employed to examine patterns in the collected material. Temporal changes in publication frequencies and sentiment were analysed to identify technological trends and deviations from prevailing patterns. Visualisations supported the interpretation and reporting of findings in relation to the potential applications and technological development of graphene.

Throughout the project, continuous monitoring assessed data acquisition, computing resource use and system operation. This monitoring, together with replication and backup procedures, supported the continuity of the analytical workflow. The resulting framework integrated researcher-guided source selection with automated collection and analysis, providing a methodological basis for examining technological development through heterogeneous textual sources.

3. Results

The analytical workflow of our research made it possible to identify promising technological pathways and possible targets and assess their relevance for medical and therapeutic applications [11].

The system is designed to predict emerging areas of technological convergence [14].

A set of deep learning tools analyzing scientific texts, patents, and bibliometric information enabled the identification of technological domains. Identifying and implementing promising technologies at an early stage of development can provide substantial value to scientific institutions, as well as to society and the economy as a whole. This can strengthen the competitive advantage of scientific organizations and inform national technology policy and R&D strategies [3].

The methodology supported the recognition of the biomedical potential of graphene-based materials and contributed to the development of a composition combining graphene and one of the substances with chemotherapeutic activity and leds to patented invention "GO-2-ME and rGO-2-ME hybrids, method of their preparation and their use", protected under patent procedures (Patent Application EP22856321.9 [EPO] and Publication no. EP4384222A1 [EPO], Polish Patent No. 243445 / application P.438737). This invention illustrates how computationally assisted technological knowledge mining can contribute to the discovery of solutions for further research related to applications in cancer therapy [12].

The implementation of the system represented an important advancement in the use of Big Data methods for R&D intelligence [13]. At the time of its initial deployment, the combined use of automated data refinement, large-scale analysis of unstructured information and AI-assisted identification of medical applications of nanomaterials constituted a novel approach.

A central feature of the designed tools was not merely the identification of applications for nanomatherials, but also the iterative definition and optimization of the search strategy itself. During implementation, several barriers to the wider use of such systems were identified. The most important concerned insufficient knowledge of available information resources and limited capacity among researchers, innovators and practitioners to use advanced analytical tools effectively. However, the evaluation of the analytical model demonstrated that AI-supported classifications may achieve higher precision and consistency than expert-only assessments, particularly when large and heterogeneous datasets are involved. Unlike purely human assessments, algorithmic analysis is less susceptible to temporary intellectual fashions, institutional pressures, political expectations or selective attention to dominant research agendas [10, 14]. This does not eliminate the need for expert judgement, but it changes its function: experts become

interpreters, validators and strategic users of analytically generated knowledge [8].

What is worth mentioning is the fact that implementation of graphene in several areas of medical applications was being noted by researches some time ago, nevertheless in papers from a decade before, health aspects and pharmaceutical applications were planned much later. Which leads to conclusion that proper and targeted analysis supported by AI let us identify areas of interest, conduct the research leading to full implementation [6, 9].

4. Conclusions

Global challenges in oncology related to cancer incidence and mortality also require advanced analytical and computational methods though these datasets are still under construction, structurisation and the output data not necessarily include all vital and reliable information for further analyses, so far. Contemporary data analysis in science and industry increasingly extends beyond stand-alone statistical models. It is moving towards integrated solutions that combine deep learning, knowledge graphs, ontologies, semantic reasoning, and distributed infrastructure.

The mechanisms underlying tumor development and the prognostic information required for individualized treatment planning are investigated through tissue microenvironment analysis and molecular profiling. Artificial intelligence techniques enable the identification of key factors within molecular datasets. These techniques include deep learning algorithms combined with hybrid optimization methods. In the future, the system we designed may also support such personalized treatment through individualized approaches to drug administration in cancer therapy.

An important aspect of developing personalized medicines is the management of data required by regulatory authorities and the exchange of information. These activities facilitate patient engagement, improve communication, and strengthen connections among stakeholders. The use of artificial intelligence enables interdisciplinary collaboration among patients, health authorities, healthcare providers, and the biopharmaceutical industry [1].

In this context, our system has several limitations. Its application to supporting the management of research, development, and innovation in healthcare through AI-assisted prediction and Big Data analytics raises several key challenges. These include potential algorithmic bias, data privacy risks specific to medical analytics, and the limited interpretability of models operating as “black boxes” [4].

For this reason, the research process should combine the capacity to analyze large, often unstructured datasets with openness to integrating different methods, while maintaining human oversight of the outcomes. This requires a new approach to data sources and analytical methods that incorporates research ethics. New skills and knowledge are needed at every level, from foundational technologies to advanced and innovative algorithms.

Our case study highlights the growing importance of analytics for technological progress. Patent information constitutes a strategic source of knowledge about technological maturity, competitive positioning, convergence processes and potential infringement risks. This is particularly relevant when innovative projects are developed by interdisciplinary teams combining scientific, engineering and clinical expertise. In this case, Big Data methods contributed to the development of knowledge essential for strategic R&D planning and intellectual property protection. The broader significance of our case study lies in its contribution to evidence-based policy and open data-driven governance in R&D. At the same time, it demonstrates how the competencies of teams representing different fields of expertise can be combined in innovative ways.

AI and Big Data tools can accelerate material discovery, support drug development, improve patent intelligence and enhance the design of public intervention instruments [7]. Their effective use, however, requires institutional competence, methodological transparency and sustained expert oversight [15]. Our case study confirmed that integrating computational intelligence with domain knowledge is a valuable model. Such integration creates conditions for more objective, timely and strategically relevant decisions in the governance of research, innovation and technological development [5].

References

[1], M. Algorri, N. S. Cauchon, T. Christian, C. O’Connell, et al., Patient-centric product development: A summary of select regulatory CMC and device considerations, *Journal of Pharmaceutical Sciences*, Vol. 112, Issue 4, 2023, pp. 922-936.

[2], W. Cetera, W. Gogołek, A. Żołnierski, et al., Potential for the use of large unstructured data resources by public innovation support institutions, *Journal of Big Data*, Vol. 9, 2022, 46.

[3], S. Chai, A. Menon, H. Tabakovic, C. Lee, Using machine learning to predict technology generality, *Academy of Management Proceedings*, Vol. 2021, Issue 1, 2021, 11418.

[4], S. S. Dhanda, D. Panwar, C. C. Lin, et al., Advancement in public health through machine learning: A narrative review of opportunities and ethical considerations, *Journal of Big Data*, Vol. 12, 2025, 154.

[5], L. Figueiró, J. Paulo, P. E. N. de Souza, P. C. Morais, Artificial intelligence and machine learning in nanomedicine. What do we expect for 2030?, *Nanomedicine*, Vol. 18, Issue 16, 2023, pp. 1041-1043.

[6], The Graphene Flagship technology and innovation roadmap: Biomedical applications, Graphene Flagship, 2017, https://graphene-flagship.eu/

[7], I. M. Hezam, A. M. Ali, A. M. Alshamrani, et al., Artificial intelligence’s impact on drug delivery in healthcare supply chain management: Data, techniques,

analysis, and managerial implications, *Journal of Big Data*, Vol. 11, 2024, 177.

[8], S. Jagan, H. Parthasarathi, G. V. Londhe, K. Srinivas, et al., Development of precision medicine for hepatic cancer evolving molecular profiling techniques through artificial intelligence, *Journal of the Balkan Tribological Association*, Vol. 30, Issue 1, 2024, pp. 81-91.

[9], J. Kinaret, V. Falko, A. Ferrari, A. Helman, et al., Coordination action for graphene-driven revolutions in ICT and beyond: WP6 project management of the CA project, Technical Report, *European Commission*, Brussels, 2012.

[10], W. J. Lee, W. K. Lee, S. Y. Sohn, Patent network analysis and quadratic assignment procedures to identify the convergence of robot technologies, *PLoS ONE*, Vol. 11, Issue 10, 2016, e0165091.

[11], K. Shahzad, S. A. Khan, M. Latif, et al., Big data analytics in healthcare: Current practices, innovations, and future prospects, *Journal of Big Data*, Vol. 12, 2025, 242.

[12], K. Uzdrowska, N. Knap, L. Konieczna, A. Kamm, et al., Combined graphene oxide with 2-methoxyestradiol for effective anticancer therapy in-vitro model, *International Journal of Nanomedicine*, Vol. 20, 2025, pp. 933-950.

[13], A. Uzundurukan, M. Nelson, C. Teske, M. S. Islam, et al., Metaanalysis and review of in silico methods in drug discovery part 1: Technological evolution and trends from big data to chemical space, *The Pharmacogenomics Journal*, Vol. 25, Issue 3, 2025, 8.

[14], H. Zhang, Y. Chen, Technology feature fusion prediction method based on multidimensional information extraction and deep neural networks, *Knowledge and Information Systems*, Vol. 67, 2025, pp. 9051-9084.

[15], A. Żołnierski, W. Cetera, D. Jaruga, J. Grzegorek, et al., Graphene and its applications. Study on the development trends in research and on the implementation potential using big data and information refining methods, *Journal of Nanotechnology Research*, Vol. 4, 2022, pp. 117-124.

(021)

Forecast-Bounded Multidimensional Rate Limiting: A Self-Hosted Control for the Detection Gap in Commercial Bot Management

S. Brahmbhatt and V. Gangadhara
Akamai Technologies, Inc., 145 Broadway, Cambridge, MA 02142, United States
Tel.: +1 617 444 3000
E-mail: sbrahmbh@akamai.com, vgangadh@akamai.com

Summary: Commercial bot-management platforms adapt to new automation through a slow path: a previously unseen signature must be observed, labelled and distributed before it is enforced, and volumetric attacks cause most of their damage inside that interval. This paper describes a self-hosted control that covers the interval. It composes forecast-derived traffic ceilings that are empirically calibrated rather than taken from nominal model intervals, a two-window aggregate detector with precisely defined rolling windows, a context-aware gatekeeper with hysteresis, on-demand attribution of the excess across behavioral dimensions using reconciled expectations, and blast-radius-limited synthesis of short-lived rules whose limits carry an explicit exceedance probability. Extending an earlier practitioner article, the paper adds a formal specification, a corrected statistical treatment, a per-branch analysis of time to mitigation, and a validation protocol based on shadow operation, incident replay and synthetic injection.

Keywords: Adaptive rate limiting, Bot mitigation, Anomaly detection, Time-series forecasting, Prediction-interval calibration, TLS fingerprinting, Web application security.

1. Introduction

Commercial bot-management platforms are effective classifiers of automated web traffic, but they adapt to new automation through a slow path: a previously unseen signature must be observed, labelled and distributed to a global enforcement fleet before it is enforced. An attacker with a fresh residential proxy pool [1] and a headless browser that reproduces the target environment presents a combination the classifier has not seen, and measurements show that well-constructed automation can bypass commercial anti-bot services [2]. Volumetric attacks cause most of their damage inside this interval. In the incidents observed in our estate, vendor-side adaptation took on the order of hours, whereas origin degradation began within minutes; this observation describes one estate and one vendor configuration and is not claimed as a general property of commercial platforms.

Two further properties widen the interval. The escalation path for a live bypass is a support queue whose output, typically a regular expression or a prefix list, stops matching when the attacker rotates the matched attribute. The strongest evidence is also often subject to disclosure constraints: a fraud team can identify gift-card enumeration within minutes by joining request logs against account identifiers and transaction history, and that join constitutes personal data. The General Data Protection Regulation [3] does not prohibit sharing it with a contracted processor under Article 28, but any disclosure must rest on a lawful basis (Article 6), respect data minimisation (Article 5(1)(c)), be covered by processor terms and be appropriately secured (Article 32). Organizational policy frequently restricts such exports during an incident, and when the evidence stays inside the perimeter, the enforcement decision is most readily made there as well.

This paper describes a self-hosted control that covers the interval; it does not replace a bot-management platform. An earlier practitioner article by the first author [4] described the architecture informally. The contribution of the present paper is not any single component, since forecast-derived thresholds and fingerprint-based counting keys are each known, but an operational composition of forecast-calibrated ceilings, on-demand multidimensional attribution, context-aware gating, blast-radius control, short-lived rule synthesis, and shadow and replay validation. Relative to [4], the following elements are new: (i) a formal specification of each stage; (ii) empirical calibration of the forecast ceilings and an empirical run-length criterion for consecutive breaches, replacing nominal interval arithmetic and an independence assumption (Section 4); (iii) precise semantics for the rolling detection windows (Section 4.3); (iv) reconciliation of slice forecasts to the aggregate and a corrected definition of excess coverage (Section 6); (v) an overdispersion-aware bound for sparse slices (Section 6.2); (vi) rule limits defined as calibrated predictive quantiles with an explicit exceedance probability (Section 7); and (vii) a per-branch analysis of time to mitigation within a validation protocol of shadow operation, incident replay and synthetic injection (Section 8).

2. Background and Related Work

The conventional rate limiter counts requests per source address against a fixed threshold, which has no

effect on a botnet distributed over 200 000 residential addresses that each emit four requests per minute. The objection conflates two elements. The counter is mature: a token bucket, equivalent to the generic cell rate algorithm standardized for ATM traffic policing [5], is adequate, and the semantics of a refused request are specified [6]. The failure lies in the key and the threshold, since a source address is a poor key against a distributed adversary and a constant is a poor threshold for a series that varies sixfold within a day.

Automated clients have been characterized by browser-fingerprint inconsistencies [7] and by the artefacts of advanced evasion [8]. At the transport layer the TLS ClientHello identifies the client implementation, and JA4 [9] provides a format that is robust to extension reordering. Such fingerprints can be forged [10], but only through engineering effort in the TLS stack. We do not classify clients: fingerprints serve only as counting keys, because an attacker who must hold a fingerprint constant to sustain throughput provides a stable grouping variable.

Volume anomalies have been detected with wavelet decomposition [11] and subspace methods [12], which are more powerful than the additive decomposition model [13] we deploy; we chose the latter because its components can be inspected during an incident and holiday effects are declared rather than engineered. Forecasting practice follows [14], and two of its elements are central here: the nominal level of a model-based interval does not guarantee its empirical coverage [15], and forecasts produced independently for the components of an aggregate need not sum to it, which motivates reconciliation [16].

3. Threat Model and System Overview

We assume an adversary whose objective is throughput: scraping, credential stuffing, inventory scalping and application-layer flooding must finish before the defender reacts. The adversary controls every client-chosen field, including the User-Agent string, cookies and path distribution, and can forge the TLS ClientHello and HTTP/2 header ordering at materially higher cost. The design is treated as public; models, ceilings and thresholds are not. Low-and-slow automation that stays within human volume produces no volumetric signal by construction and is out of scope. The volumetric family is the one that causes origin outages and stock-outs.

The system comprises two detection tiers and an enforcement stage. The edge writes per-minute counters to Redis and daily Parquet files to object storage, with buckets formed in UTC and absent buckets zero-filled. A nightly job fits and calibrates the forecasting models and publishes 36 hours of expected values and ceilings to Redis, so that a failed run degrades into stale but usable ceilings; an alarm fires when the newest published key is older than 26 hours. Every 60 s, tier one compares a few aggregate series with the precomputed ceilings, so no model artefact is loaded on the hot path. Tier two attributes the excess across approximately a thousand sliced series and runs only when tier one and a contextual gatekeeper agree, which in our deployment happens a few dozen times per month. The design rests on an empirical property that operators should verify on their own logs: legitimate traffic is highly regular, and the shares of its slices are stable. Linux traffic in our estate, for example, stays near 2 % from hour to hour, so a rise to 9 % within ten minutes indicates automation rather than new users.

4. Tier One: Calibrated Forecast Ceilings

4.1. Model and Training History

One decomposable additive model (Prophet) [13] is fitted per hostname and window length on sixty days of history in about forty seconds. The response is modelled as $\ln(1 + y)$, so that the interval scales with volume (Fig. 1). Growth is flat, yearly seasonality is disabled, weekly seasonality is enabled, and the daily term uses Fourier order twelve, because the default order of four rounds off the morning ramp. The model was preferred to potentially more accurate neural forecasters because a suspect ceiling can be decomposed into trend, weekly and daily components during an incident. Past attacks are removed before fitting, since every attack left in the history widens the interval and raises the next ceiling: known mitigation windows are excluded, and buckets deviating from the rolling daily median by more than six scaled median absolute deviations [17] are replaced by that median.

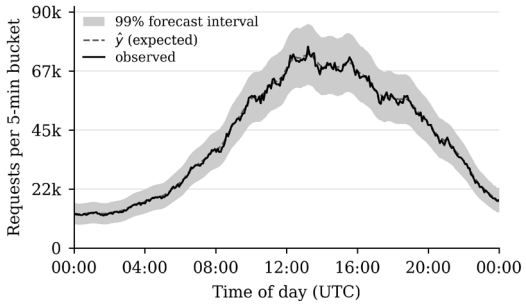


Fig. 1. One weekday of request volume in five-minute buckets against the nominal 99 % interval of the model. The interval scales with volume: an excess of 8000 requests per five-minute bucket lies far outside it at 04:00 but inside it at 14:00. An earlier version of this figure appeared in [4].

4.2. Empirical Calibration

The earlier version [4] set the interval width to 0.99 and inferred $0.005 \times 288 \approx 1.4$ upper breaches per series per day. That inference holds only if the interval has calibrated coverage, and the model documentation cautions against expecting it [15]: by default the interval omits seasonal uncertainty, and with flat growth it reflects observation noise treated as independent and homoscedastic, whereas the residuals of web traffic are autocorrelated and heteroscedastic

across the day. We therefore treat the nominal upper bound $\hat{u}_t$ (in log space) as an uncalibrated score. The ceiling is given by (1), where the offset δ_α is the split-conformal [18] order statistic (2) of the residuals $r_t = \ln(1 + y_t) - \hat{u}_t$ over n held-out, attack-free windows.

$$C_t = \exp(\hat{u}_t + \delta_\alpha) - 1, \quad (1)$$

$$\delta_\alpha = r_{(\lceil (n+1)(1-\alpha) \rceil)} \quad (2)$$

The calibration set comprises the most recent fourteen days of rolling-origin forecasts, excluding mitigation windows, with separate offsets for four six-hour strata of the day. We set $\alpha = 0.005$, which preserves the intended budget of about 1.4 breaches per series-day. Because autocorrelated traffic violates exchangeability, the resulting rate is validated on further held-out attack-free days rather than assumed, as reported in Section 8.

4.3. Window Semantics and the Two-Window Detector

Counters are kept per UTC minute. The detector runs at each minute boundary τ after a short grace period for late writes and reads only completed minutes. $W_5(\tau)$ and $W_{60}(\tau)$ are trailing sums over $(\tau - 5 \text{ min}, \tau]$ and $(\tau - 60 \text{ min}, \tau]$, recomputed every minute. Ceilings are published for aligned slots, and the ceiling of a trailing window is interpolated by overlap as in (3), where s is the slot containing τ and f the fraction of the window in s; calibration is performed on this same interpolated statistic.

$$C_W(\tau) = (1 - f)C_{s-1} + fC_s \quad (3)$$

The ratios $\rho = W/C_W$ of both windows are combined by Algorithm 1, where κ counts consecutive aligned, non-overlapping five-minute slots in breach. The earlier argument [4] that three consecutive false breaches are negligible assumed independence, which would give $288 \times 0.005^3 \approx 3.6 \times 10^{-5}$ runs per series-day; forecast errors are positively autocorrelated, so runs are far more frequent. The persistence threshold is therefore the smallest κ whose empirical run rate on attack-free held-out traffic is within the tolerated false-alert budget; the deployment uses $\kappa = 3$. The branches differ in minimum latency (Section 8): the 3× branch can fire at the first minute boundary after the breach, the dual-window branch waits for W_{60} to breach, which takes tens of minutes for a gradual ramp, and the persistence branch needs at least fifteen minutes by construction. Fig. 2 shows one incident through both windows.

5. Contextual Gating

A tier-one alert is not by itself a reason to change the edge configuration, since challenges reduce conversion and every rule is a change to be reasoned about during the incident. A scoring stage combines the tier-one state with infrastructure load (origin CPU, 5xx rate and upstream latency), the business calendar, geography and cooldown state into a weighted score on a 0–100 scale, so that every term can be examined in a post-incident review. A scraper at four times normal volume against an idle origin score about 60 and produces a notification only. To prevent oscillation, the gatekeeper activates at 70, releases below 40, holds a mitigation for at least ten minutes and applies a twenty-minute cooldown during which re-entry requires a score of 80. Every transition is logged with its full context.

Algorithm 1. Combining the two windows into a tier-one state.

input: ρ_5, ρ_{60} at tick τ; κ, consecutive breached five-minute slots
if $\rho_5 \geq 3$ **then return** ALERT (3× ceiling)
if $\rho_5 > 1$ **and** $\rho_{60} > 1$ **then return** ALERT (both windows)
if $\rho_5 > 1$ **and** $\kappa \geq 3$ **then return** ALERT (persistence)
if $\rho_5 > 1$ **or** $\rho_{60} > 1$ **then return** WATCH
return OK

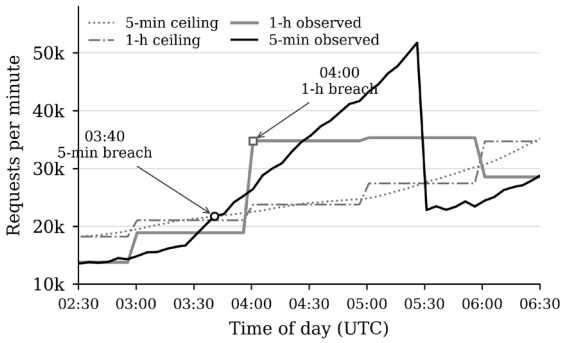


Fig. 2. A single incident viewed through both windows, 02:30–06:30 UTC. The attack ramps from 03:30 and the five-minute series crosses its ceiling at 03:40; the hourly series, smoother and with a higher ceiling, crosses later. The drop at 05:30 marks the mitigation applied during the incident. An earlier version of this figure appeared in [4].

6. Tier Two: Attributing the Excess

Cardinality multiplies in web telemetry: about 90 operating-system versions, 350 browser builds, 1500 to 4000 JA4 fingerprints and 90 countries yield billions of candidate keys. Tier two therefore evaluates six single dimensions (operating system, browser build, JA4, country, autonomous system and a header-order hash) and four pairs, such as JA4 with operating system, which exposes impossible combinations like a macOS User-Agent presenting the ClientHello of a Go HTTP client. The top $N = 1000$ values across dimensions are modelled individually, and the remainder of each dimension is aggregated into ano*ther* series.

6.1. Reconciled Expectations and Coverage

Slice forecasts are fitted independently, so their sum over a dimension need not equal the aggregate expectation $\hat{E}$ of tier one. The raw expectations $e(v)$ are therefore reconciled proportionally as in (4), where V_d contains the modelled values, the *other* series and the sparse values of Section 6.2; trace-minimization reconciliation [16] is a more general alternative. The aggregate excess E and the surplus $s(v)$ of each value are given by (5), and the excess coverage $k(v)$ by (6).

$$\tilde{e}(v) = \frac{e(v)\hat{E}}{\sum_{u \in V_d} e(u)}, \tag{4}$$

$$E = O - \hat{E}, \quad s(v) = \max(o(v) - \tilde{e}(v), 0), \tag{5}$$

$$k(v) = \min\left(1, \frac{s(v)}{E}\right) \tag{6}$$

Because the reconciled expectations sum to $\hat{E}$, signed deviations sum to E and the clipped surpluses sum to at least E; coverages are therefore not a partition and may sum to more than one. The earlier formulation [4] normalized each surplus by the sum of positive surpluses within its dimension, a denominator that differs between dimensions, so the resulting shares were not comparable across dimensions as the rule builder requires. Values are reported with $k(v)$ and $h(v)$, their share of normal traffic over the trailing thirty attack-free days (Table 1).

Table 1. Attribution output for one scraping incident.

Dimension value	$o(v)$	$\tilde{e}(v)$	$k(v)$	$h(v)$
ua_browser				
Chrome/124.0.0.0	51 420	11 410	0.955	0.061
Chrome/123.0.0.0	9 830	9 120	0.017	0.049
HeadlessChrome/124	1 220	40	0.028	0.000
ja4				
t13d1516h2_8daaf6152771	41 490	310	0.983	0.002
t13d1517h2_5b57614c22b0	12 410	12 600	0.000	0.071

Window 03:40–03:55 UTC; E = 41 900 requests per minute; volumes are mean requests per minute. o: observed; $\tilde{e}$: reconciled expectation (4); k: coverage (6); h: normal share.

6.2. Sparse Slices

A series that is zero for eleven hours and then records forty requests cannot be forecast meaningfully. The earlier version [4] bounded such slices with a Poisson quantile, but web request counts are overdispersed and clustered in page-load bursts, so a Poisson bound under-covers. Where the attack-free variance σ^2 of a slice exceeds its mean μ, we use the moment-matched negative binomial bound (7) [19]; otherwise the Poisson bound is retained.

$$\varphi = \frac{\mu^2}{\sigma^2 - \mu}, \qquad C_{\mathrm{sp}} = \max\left(25, F^{-1}_{\mathrm{NB}(\mu,\varphi)}(1 - \alpha_{\mathrm{s}})\right) \tag{7}$$

The level $\alpha_{\mathrm{s}} = 10^{-5}$ is a multiplicity correction: with M sparse slices per invocation, a calibrated bound yields $M\alpha_{\mathrm{s}}$ false slice alarms per invocation, for example 0.03 for M = 3000, and it is verified in aggregate across slices rather than per slice. The *other* series closes the remaining gap: an attacker rotating through 4000 rare User-Agent strings produces 4000 unremarkable slices but moves the *other* share from 1.2 % to 22 %.

7. Rule Synthesis and Enforcement

A candidate predicate must cover at least 60 % of the excess. Among candidates whose normal share is at most 5 %, the one with the highest leverage (8) is selected and throttled. If none qualifies, the covering candidate with the smallest normal share is challenged rather than blocked, since genuine browsers complete a challenge and most automation does not; if no candidate covers the excess, an operator is paged.

$$L(v) = \frac{k(v)}{\max\left(h(v), 10^{-4}\right)} \tag{8}$$

In Table 1 the JA4 candidate has leverage 0.983/0.0021 = 468 (unrounded normal share) and the browser candidate 0.955/0.061 = 16, so the selection follows from the arithmetic. The earlier version [4] set the limit to 1.25 times the expected rate and stated that this admits all legitimate traffic. That statement is too strong: legitimate traffic in small or burst slices exceeds 1.25 times its expectation with non-negligible probability, and legitimate and automated requests matching the predicate compete for the same tokens. The limit is therefore the calibrated upper predictive quantile (9), where $C_{1-\beta}(v)$ is computed as in (1)–(2) for modelled slices and as in (7) for sparse ones, and the floor of 60 requests per minute avoids degenerate limits.

$$R(v) = \max\left(60, C_{1-\beta}(v)\right) \tag{9}$$

With β = 0.01, the probability that the legitimate traffic of the slice alone exceeds the limit in a window is 1 %. The limit does not select which requests are admitted, so collateral exposure during an attack is governed by $h(v)$, which is why selection minimizes it. Rule identifiers are hashes of the predicate, so repeated detection updates a single rule, and every rule expires after fifteen minutes unless renewed. Protected predicates such as corporate egress and partner integrations are never hard-limited. A blast-radius gate downgrades any rule matching more than 5 % of normal traffic to a challenge and refuses to deploy above 20 %, paging an operator instead; this guards

against an attacker who hides in the most common User-Agent slice. Rules are pushed to the CDN rule API, where propagation takes 10 to 60 s, or enforced behind the CDN by an atomic token bucket in Redis; refused requests receive status 429 with a Retry-After header [6].

8. Evaluation

A system that can throttle paying customers must be validated before it enforces. The evaluation uses one production estate operated for more than a year, including 48 labelled incidents, and three exercises.

Shadow operation. The full pipeline runs for at least four weeks with the action fixed to log-only, and every proposed rule is reviewed by an analyst. The key metric is the number of rules proposed per quiet week, because proposals triggered by marketing mail-outs, application releases and partner batch jobs erode on-call trust; the deployment holds it below one (Table 2).

Table 2. Operational metrics.

Metric	Value	Basis
Time to mitigation, p50 and p90	Under 3 min, propagation included	Incident replay
Rules proposed per quiet week	Below 1	Shadow operation
Collateral share h of deployed rules, median	Under 1 %	Deployed rules
Sensitivity floor at daily trough	≈1.4× ceiling	Synthetic injection
Calibrated five-minute exceedance	0.598 % (target 0.5 %)	Held-out attack-free days
Persistence runs ($\kappa \geq 3$)	0.16 per series-day	Held-out attack-free days
Ceiling staleness alarm	26 h	Configuration

Replay. Raw counters from each labelled incident are replayed through the live detector with origin CPU, calendar state and geographic mix reconstructed. Time to mitigation is decomposed as $T = T_{\text{det}} + T_{\text{gate}} + T_{\text{attr}} + T_{\text{prop}}$: onset to tier-one alert, alert to gatekeeper activation, attribution and rule synthesis (about 2 s), and measured CDN propagation. Table 3 gives the minimum T_{det} of each branch of Algorithm 1. The replayed end-to-end time was under three minutes at both p50 and p90, propagation included. Both figures are below the minimum latency of the dual-window and persistence branches, so they characterize abrupt attacks detected by the 3× branch; gradual ramps are detected later by construction.

Synthetic injection. An attack-free day is selected and an attack of known shape, magnitude and time is added. Sweeping the magnitude downward yields the sensitivity floor: at the daily through the detector fires at approximately 1.4 times the ceiling, whereas at the afternoon plateau the band is thousands of requests wide and the floor is correspondingly higher, so sweeps must cover time of day as well as magnitude. Table 2 summarizes the operational metrics; they depend on traffic shape and are an example of what to measure rather than targets.

Table 3. Minimum detection latency by branch of Algorithm 1.

Branch	Condition	Minimum T_{det}
3× ceiling	$\rho_5 \geq 3$	First completed minute
Both windows	$\rho_5 > 1$, $\rho_{60} > 1$	Tens of minutes for a gradual ramp
Persistence	$\rho_5 > 1$, $\kappa \geq 3$	15 min

9. Limitations

The system cannot detect low-and-slow automation: thirty login attempts per minute across 50 000 residential addresses produce no volumetric signal and require session-level behavioral signals [7, 8]. Most counting keys are attacker-controlled, which is why the blast-radius gate exists and why TLS and header-order fingerprints, which are costlier to forge [10], justify their integration cost. Detection is confined to the layer the counters describe; where the CDN serves most requests from cache, traffic should be modelled at the origin or per route family. Volume cannot distinguish a flash crowd from a flood, and the calendar and infrastructure terms reduce but do not remove this risk. Calibration assumes that the recent past is representative and degrades after abrupt legitimate changes until the calibration window absorbs them. Finally, the evaluation covers one estate and one vendor configuration.

10. Conclusions

This paper described a self-hosted rate-limiting control that covers the interval between the onset of a volumetric bot attack and the adaptation of an external bot-management platform. Its contribution is the operational composition of calibrated forecast ceilings, on-demand attribution with reconciled expectations, context-aware gating, blast-radius control and short-lived rules whose limits carry an explicit exceedance probability. Relative to [4], it replaces nominal interval arithmetic with empirical calibration and run-length measurement, defines the detection windows, makes attribution coherent, accounts for overdispersion and analyses time to mitigation per detection branch. The design can be deployed incrementally; a first shadow deployment comprising the counters, one calibrated model and the two-window detector took approximately three weeks of work for one engineer.

References

[1]. X. Mi, et al., Resident evil: Understanding residential IP proxy as a dark service, *in Proceedings of the IEEE Symposium on Security and Privacy (SP)*, 2019, pp. 1185-1201.

[2]. B. Amin Azad, O. Starov, P. Laperdrix, N. Nikiforakis, Web Runner 2049: Evaluating third-party anti-bot services, *Lecture Notes in Computer Science*, Vol. 12223, 2020, pp. 135-159.

[3]. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation), *Official Journal of the European Union*, Vol. L 119, 2016, pp. 1-88.

[4]. S. Brahmbhatt, Smart rate limiting: The fail-safe your bot vendor cannot build for you, HackerNoon, 2026, https://hackernoon.com/smart-rate-limiting-the-fail-safe-your-bot-vendor-cannot-build-for-you

[5]. Traffic management specification version 4.0, af-tm-0056.000, *The ATM Forum*, 1996.

[6]. M. Nottingham, R. Fielding, Additional HTTP status codes, Request for Comments 6585, *Internet Engineering Task Force (IETF)*, 2012.

[7]. A. Vastel, P. Laperdrix, W. Rudametkin, R. Rouvoy, FP-scanner: The privacy implications of browser fingerprint inconsistencies, in *Proceedings of the 27th USENIX Security Symposium (USENIX Security)*, 2018, pp. 135-150.

[8]. C. Iliou, T. Kostoulas, T. Tsikrika, V. Katos, et al., Towards a framework for detecting advanced web bots, in *Proceedings of the 14th International Conference on Availability, Reliability and Security (ARES)*, 2019, 17.

[9]. J. Althouse, JA4+ network fingerprinting, FoxIO, 2023, https://github.com/FoxIO-LLC/ja4

[10]. S. Frolov, E. Wustrow, The use of TLS in censorship circumvention, in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2019.

[11]. P. Barford, J. Kline, D. Plonka, A. Ron, A signal analysis of network traffic anomalies, *in Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement (IMW)*, 2002, pp. 71-82.

[12]. A. Lakhina, M. Crovella, C. Diot, Diagnosing network-wide traffic anomalies, in *Proceedings of the ACM SIGCOMM Conference (SIGCOMM)*, 2004, pp. 219-230.

[13]. S. J. Taylor, B. Letham, Forecasting at scale, *The American Statistician*, Vol. 72, Issue 1, 2018, pp. 37-45.

[14]. R. J. Hyndman, G. Athanasopoulos, Forecasting: Principles and Practice (3rd ed.), *OTexts*, Melbourne, 2021.

[15]. Uncertainty intervals, Prophet Documentation, Meta Open Source, https://facebook.github.io/prophet/docs/uncertainty_intervals.html

[16]. S. L. Wickramasuriya, G. Athanasopoulos, R. J. Hyndman, Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization, *Journal of the American Statistical Association*, Vol. 114, Issue 526, 2019, pp. 804-819.

[17]. P. J. Rousseeuw, C. Croux, Alternatives to the median absolute deviation, *Journal of the American Statistical Association*, Vol. 88, Issue 424, 1993, pp. 1273-1283.

[18]. V. Vovk, A. Gammerman, G. Shafer, Algorithmic Learning in a Random World, *Springer*, New York, 2005.

[19]. A. C. Cameron, P. K. Trivedi, Regression Analysis of Count Data (2nd ed.), *Cambridge University Press*, Cambridge, 2013.

(022)

A Multi-Tier Framework for Adversarial-Resilient Detection of Unwanted Business Communication at Scale

Kamya Batra
Meta Platforms, Inc., 250 Howard Street, San Francisco, CA 94105, USA
Tel.: +1 747 273 9109
E-mail: kamyab@meta.com

Summary: Business messaging platforms process billions of enterprise-to-consumer interactions daily, creating an abuse surface on which single-layer detection is forced into an untenable trade-off between latency and analytical depth. This paper describes a multi-tier framework for detecting unwanted business communication, decomposed across three adversarial timescales: real-time behavioral scoring of individual messages, near-line sequence analysis of session trajectories, and offline graph mining for coordinated activity. A drift-aware retraining pipeline and an adversarial acceptance gate govern maintenance and promotion to production. The algorithms employed are standard; the contribution is architectural, namely the orchestration of detection across timescales, motivated by the structural limitation that a single-stage decision cannot observe behavior occurring after it. We further specify an evaluation design for this setting, in which ground truth is drawn independently of the system under test and detection performance is reported against a paired over-enforcement counter-metric. The framework is deployed in production; per-tier evaluation is ongoing.

Keywords: Big data analytics, Unwanted business communication, Adversarial machine learning, Graph-based abuse detection, Concept drift.

1. Introduction

Business messaging platforms now process billions of enterprise-to-consumer interactions daily, and with that scale comes an attack surface that conventional anti-abuse systems were never designed for. Unwanted business communication (UBC) – spam, scams, impersonation, and gray-area promotional outreach – builds on a longer line of work on messaging spam detection [8, 9], but is not a static classification problem. In practice, three challenges dominate: first, abusive messages represent a small fraction of total traffic, but even a modest false-negative rate translates to millions of harmful interactions per day. Second, adversarial pressure induces concept drift: enforcement actions change attacker behavior, so a model that performs well at launch degrades as the input distribution shifts [1, 4]. Third, and most underappreciated, the cost of a false positive is not symmetric with a false negative: incorrectly blocking a legitimate business damages platform trust and creates downstream revenue and reputational harm that is difficult to reverse.

This paper proposes a multi-tier detection framework that addresses these challenges through a combination of real-time behavioral scoring, near-line sequence modeling, and offline graph analysis, augmented by a drift-aware retraining pipeline for sustained adversarial resilience. The individual algorithms employed are standard; our contribution is architectural: the multi-timescale orchestration of real-time, near-line, and offline detection tiers under a shared drift-aware maintenance loop, together with the adversarial acceptance gate that governs promotion to production. The framework described is deployed in production on a large-scale business messaging platform; per-tier evaluation of the composition is ongoing work.

2. Proposed Framework

Most production integrity systems rely on a single classification layer, which forces an impossible tradeoff between latency and depth. Our framework instead decomposes the problem into three tiers, each targeting a different adversarial timescale (Fig. 1); the feature categories used at each tier are summarised in Table 1.

Table 1. Feature categories by detection tier.

Tier	Feature Category	Examples
Real-Time	Behavioral	Send velocity, reply-rate asymmetry, session patterns
Real-Time	Graph (local)	Fan-out ratio, recipient cluster overlap
Real-Time	Content	Template similarity hash, keyword density
Near-Line	Sequential	Message-session embeddings, ramp-up trajectories
Offline	Graph (global)	Community structure, bipartite anomaly scores

Single-stage gating is effective at the point of decision: in prior work on actor-level trust gating, replacing heuristic rules with a model-based gate materially reduced the proportion of malicious

accounts passing the gate while reducing legitimate accounts substantially less. Such a gate is, however, a one-time decision – it cannot by construction detect an account that is approved on good behavior and escalates afterwards. That limitation motivates decomposition across timescales rather than additional depth at a single layer.

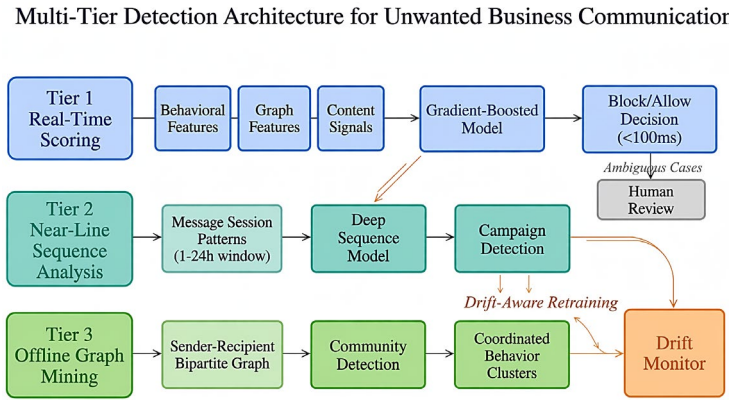


Fig. 1. Multi-tier detection architecture. Tier 1 (real-time) handles per-message scoring under a latency budget; Tier 2 (near-line) captures slow-burn campaigns via sequence models; Tier 3 (offline) surfaces coordinated networks via graph mining. All tiers feed into the drift-aware retraining pipeline.

Tier 1 – Real-Time Scoring. A gradient-boosted ensemble (XGBoost) scores each message under a real-time serving budget. Features combine behavioral signals (send velocity, reply-rate asymmetry, timing variance), local graph features (fan-out ratio, 2-hop recipient overlap with abuse clusters), which follow web-scale graph feature practice [2], and content fingerprints (locality-sensitive hashing), an approach applied to online spam detection by Ali et al. [8]. We deliberately exclude raw text features – adversaries mutate content faster than behavioral patterns. Threshold tuning uses asymmetric loss weights that penalise false positives substantially more than false negatives, following the cost-sensitive learning formulation of Elkan [6].

Tier 2 – Near-Line Sequence Analysis. A GRU-based sequence model, chosen for materially lower inference cost than transformer alternatives at comparable accuracy, processes session patterns over 24-hour windows. The key insight: sophisticated accounts stay below Tier 1 thresholds individually, accruing a substantial history of benign messages before escalating – a reputation-farming pattern [3]. The sequence model learns the trajectory shape, not just the snapshot.

Tier 3 – Offline Graph Mining. A nightly batch constructs a large-scale sender-recipient bipartite graph and applies Louvain community detection [7] for coordinated inauthentic behavior, extending graph-plus-metadata approaches to collective abuse detection [3] and complementing recent work on automated detection of coordinated inauthentic behavior [5]. Graph structure proves more durable than content signals – adversaries mutate text easily but reorganizing operational networks is costly. This tier surfaces account clusters that are individually unremarkable but structurally anomalous.

3. Adversarial Resilience

The most expensive lesson in production integrity work is that a model that performs well at launch will degrade within weeks if not actively maintained. Our drift-aware retraining pipeline follows the concept-drift framing of Lu et al. [4] and monitors Population Stability Index (PSI) across the most influential features (ranked by SHAP importance) between training and live serving distributions. When PSI exceeds 0.2 – a conventional drift threshold [4] – for any monitored feature, targeted retraining is triggered on the affected feature subspace, refitting only the drifted components rather than the full model, which completes substantially faster than a full retrain. This shortens the detection-rate degradation window from weeks, under periodic retraining, to days (Fig. 2).

Before any model reaches production, it must survive a red-team evaluation suite, in the spirit of adversarial-example stress-testing [1]: a suite of adversarial scenarios including reputation-farming sequences, template mutation chains, coordinated burst patterns, and cross-tier evasion attempts. Models that pass standard precision-recall gates but fail a threshold number of adversarial scenarios are rejected – a gate that, in our experience, catches a non-trivial share of otherwise-passing candidates.

4. Precision-Recall Tradeoff and Human-in-the-Loop

A false positive is not the mirror image of a false negative – blocking a legitimate business triggers escalations, revenue loss, and trust erosion. We address this through cost-sensitive learning [6] with empirically calibrated class weights and a

human-in-the-loop escalation path for cases within a tunable confidence band. In practice, a small and tunable share of daily volume routes to human review.

This asymmetry is measurable rather than hypothetical. In a randomised online experiment on the actor-level gate described in Section 2, accounts were allocated to control and treatment on a rolling daily basis to hold the assignment ratio stable as the population grew, and labelled by subsequent enforcement within a fixed forward window, with robustness checked across window lengths. Evaluation reported a matched pair of metrics rather than a single detection metric: the model-based gate reduced malicious accounts passing the gate by approximately 18 % (CI 11–24 %) against approximately 5 % (CI 4–6 %) for non-malicious accounts, the majority of which proved inactive. The principle carried into the present framework is that any enforcement component must be assessed against a paired counter-metric, since a detection metric alone cannot distinguish improved detection from increased over-enforcement.

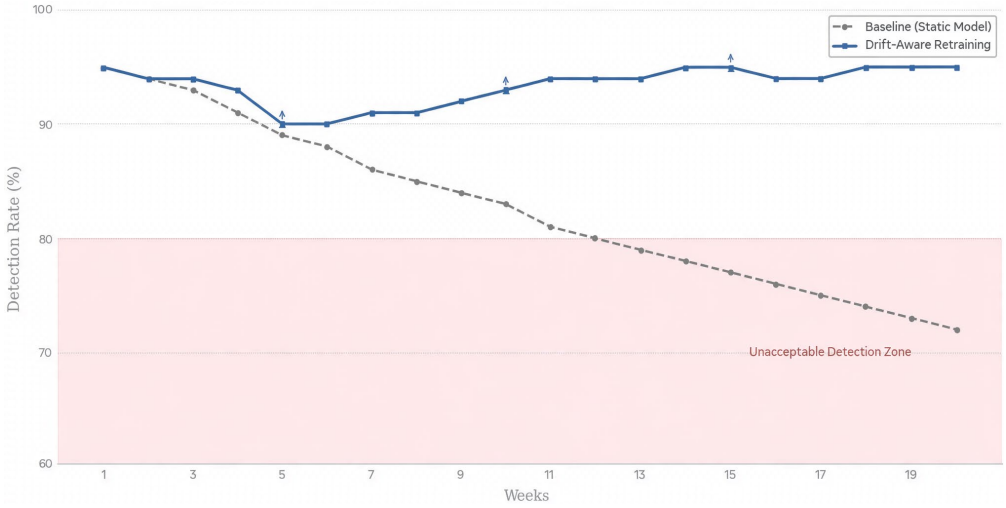


Fig. 2. Conceptual schematic of the mechanism described in Section 3: without targeted retraining, detection degrades between retraining cycles as the input distribution shifts; with it, degradation is bounded by the detection-to-retraining interval rather than by the retraining schedule. The figure is illustrative and depicts expected qualitative behavior only; it is not measured data and no experimental values are implied.

5. Evaluation Approach

Unit and weighting. Decisions are evaluated at the level of business accounts rather than individual messages: abuse is identified from content within business–user threads, but labels, decisions and enforcement all attach to the account. Metrics are weighted by messages or unique business–user threads rather than counting accounts equally, so that reported performance reflects harm averted rather than decisions taken – an account reaching many thousands of users is not equivalent to one reaching ten.

Dataset and partitioning. Evaluation uses a sample of business accounts over a defined observation window, in which abusive accounts are a small minority of the population. Assignment to comparison groups is performed at account level, with accounts allocated to control and treatment at the start of an evaluation period and newly eligible accounts allocated randomly on a rolling basis thereafter, so that the assignment ratio is held constant as the population grows. Message-level assignment is avoided, since the same account appearing in both groups would confound the comparison. Absolute volumes and class proportions are withheld.

Ground truth. Detection performance cannot be assessed against the output of the system under test; doing so measures only what the system already catches. Account-level labels are therefore drawn from an independent basis – a classifier separate from the detection pipeline, calibrated against a human-labelled reference set, with its own precision, recall and confidence intervals reported continuously. Because these labels do not depend on the enforcement decision, they support both directions of evaluation: of abusive accounts present, the share actioned; and of accounts actioned, the share not in fact abusive.

Baseline and ablation. The comparison baseline is the single-layer classifier the framework replaces. Tiers are then enabled cumulatively – Tier 1 alone, Tier 1 with Tier 2, and all three – with incremental recall measured at a fixed false-positive budget, so that no tier's apparent contribution is confounded by a more permissive operating point. Drift stability is assessed by tracking detection rate over time with and without the retraining loop of Section 3, against the same independent labels.

Uncertainty. Because abusive accounts are rare, required sample size and test duration are established by null-comparison (A/A) simulation before any

online evaluation rather than inferred afterwards, and intervals are reported alongside point estimates.

Scope. The experiment reported in Section 4 evaluates a single actor-level stage. Per-tier ablation of the three-tier composition, following the design above, is the subject of ongoing work.

6. Conclusions

Integrity at scale is not a model problem – it is a systems problem. No single classifier, however sophisticated, can simultaneously handle the latency demands of real-time decisions, the depth required for campaign-level detection, and the structural analysis needed for coordinated behavior. The multi-tier decomposition, combined with continuous adversarial stress-testing and drift-aware maintenance, provides a framework that degrades gracefully rather than failing silently – which, in production, is the difference that matters.

References

[1]. I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, *in Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

[2]. R. Ying, R. He, K. Chen, P. Eksombatchai, et al., Graph convolutional neural networks for web-scale recommender systems, in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2018, pp. 974-983.

[3]. S. Rayana, L. Akoglu, Collective opinion spam detection: Bridging review networks and metadata, in *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2015, pp. 985-994.

[4]. J. Lu, A. Liu, F. Dong, F. Gu, et al., Learning under concept drift: A review, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 31, Issue 12, 2019, pp. 2346-2363.

[5]. T. Kumarage, J. Garber, et al., Harnessing LLMs for automating detection of coordinated inauthentic behavior, in *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, Vol. 38, 2024, pp. 22175-22184.

[6]. C. Elkan, The foundations of cost-sensitive learning, in *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI)*, 2001, pp. 973-978.

[7]. V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, *Journal of Statistical Mechanics: Theory and Experiment*, Vol. 2008, Issue 10, 2008, P10008.

[8]. S.-H.-A. Ali, S. Ozawa, J. Nakazato, T. Ban, et al., An online malicious spam email detection system using resource allocating network with locality sensitive hashing, *Journal of Intelligent Learning Systems and Applications*, Vol. 7, Issue 2, 2015, pp. 39-53.

[9]. G. Istrati, A. Madhusoodanapanicker, A method and device for spam SMS detection, Patent WO2018011264A1, *World Intellectual Property Organization (WIPO)*, 2018.

(018)

Enhanced Privacy and Fidelity Evaluation of LLM-Generated Cybersecurity Tabular Data

G. Giacomello and O. Preka
Center for Computational Social Science, DSPS, University of Bologna,
Strada Maggiore 45, Bologna, Italy.
E-mail: giampiero.giacomello@unibo.it

Summary: The development of data-driven cybersecurity policies is severely hampered by data scarcity due to privacy concerns, under-reporting, and reputational risks. To overcome this critical bottleneck, this study explores synthetic tabular data generation using Large Language Models (LLMs). Leveraging the GReaT framework, we fine-tuned a tabular-specific LLM (tabularisai/Qwen3-0.3B-distil) on a corpus of 3,388 reported cyber incidents and benchmarked its performance against a generic LLM (unsloth/Llama 3.2 1B). The generated data was rigorously evaluated for privacy and fidelity using metrics such as Distance to Closest Record (DCR) and PRDC. Results demonstrate that while the generic model exhibited severe memorization of the training data and poor structural alignment, the targeted tabular model achieves high structural and statistical fidelity to the original dataset without compromising privacy. Ultimately, this comparative analysis highlights that targeted LLM selection is critical, providing a viable solution for cybersecurity data augmentation.

Keywords: Synthetic data, Cybersecurity, Tabular data generation.

1. Research Problem

The development of data-driven cybersecurity policies and defensive mechanisms is critically hampered by a persistent, multifaceted challenge: data scarcity. In an epoch characterised by increasingly frequent and sophisticated cyber threats, the existence of datasets for training intrusion detection systems, analysing malware behaviour, and modelling threat landscapes has assumed a crucial role.

Nevertheless, compressive data availability is extremely scarce. Key factors include the small fraction of cyber-attacks being detected (and even less reported). in addition to existing datasets suffering from incompleteness and inconsistency. This raises significant concerns regarding their reliability and validity.

The data scarcity issue is not accidental but systemic, stemming from several factors. First, the very nature of cyber-attacks implies that malicious actors actively conceal their identity and methods used, thus making data detection and collection inherently difficult. Second, targeted subjects tend not to share incident data due to fears of exposing unpatched vulnerabilities, incurring substantial reputational damage, and ultimately creating competitive disadvantages. This hesitation is further compounded by a complex web of legal and regulatory implications surrounding data privacy and breach notifications. Finally, technology vendors and software companies are often constrained by market pressures, making them hesitant to reveal product weaknesses that could be inferred from detailed attack data. These constraints create a critical bottleneck that impedes the research and development of data-driven cybersecurity solutions.

2. Methodological Approach: Synthetic Data Generation with LLMs

To address these significant challenges, we explore the potential of synthetic data generation using Large Language Models (LLMs) as a novel solution. Synthetic data, defined as algorithmically generated information that resembles the statistical properties and structural characteristics of a source dataset, offers a powerful paradigm for overcoming data access barriers. Its utility has been demonstrated across various domains for tasks such as data anonymization to protect privacy in sensitive sectors like healthcare, data augmentation to enrich sparse datasets, and bias mitigation in machine learning models.

Trained on vast quantities of text, code, and other data types, LLMs have shown advanced generative capabilities and versatile problem-solving skills that extend well beyond traditional Natural Language Processing [1]. Their capabilities, such as in-context learning and instruction following, can be leveraged via sophisticated prompt engineering techniques [2]. This has opened up opportunities for new LLMs applications, including the generation of tabular data.

While the application of LLMs to generate structured tabular data was in its early stage until recently, more recent advancements have shown promising results in areas such as imputing missing values [3], augmenting sparse data [4], and rebalancing imbalanced classes [5]. These developments lay the foundational for our current investigation into synthetic data generation [6].

3. Extension over Previous Work

This work significantly extends our previous study [7], which established the initial proof-of-concept for

this approach. While our previous study established the initial proof-of-concept for applying the GReaT framework to cybersecurity data, the current work significantly expands upon this foundation in four key areas. First, regarding the generative model, we transition from GReaT implementation with generic unsloth/Llama 3.2 1B model [8] to fine-tuning a specifically pretrained tabular LLM (tabularisai/Qwen3-0.3B-distil) [9], demonstrating that domain- and structure-aligned models yield superior results. Second, we introduce a more rigorous split and evaluation protocol: compared to the previous study, the test set size was increased to a 70/30 train-test ratio to ensure robust evaluation against unseen data, as test data are of crucial importance in evaluating data quality. Third, we have substantially expanded our fidelity metrics beyond basic statistical similarity to include Maximum Mean Discrepancy (MMD), Classifier Two-Sample Test (C2ST) [10], and PRDC (Precision, Recall, Density, Coverage) metrics, providing a multidimensional view of data quality. Finally, and most crucially, we introduce a dedicated privacy risk evaluation using Distance to Closest Record (DCR) [11] distributions to explicitly test for memorization

4. Methodology and Experimental Design

Our methodology leverages state-of-the-art LLM-based frameworks for tabular data generation. Recognising the inherent challenges of tabular data - which includes heterogeneous feature types (e.g., categorical, numerical), class imbalance across categories [5], and complex context-based interconnections [12] - we begin with a key preprocessing step: text-based serialization. This process converts structured tabular data into a linear textual representation that can be processed by an LLM, translating the table's schema and values into a coherent sequence while preserving semantic knowledge from column names.

Current state-of-the-art methods for tabular data generation with large language models generally follow the same approach, though with some variations. **GReaT** [13] fine-tunes a pretrained GPT-2 model on the original data and employs random permutations of feature order to reduce dependence on column sequences. For more complex scenarios, **ReaLTabFormer** [14] is noted for its enhanced performance on relational databases. Other prominent methods include **TAPTAP** [15], which pretrains GPT-2 on a vast collection of public tables to build foundational knowledge, and **Tabula** [16], which accelerates training by starting from a randomly initialized model.

Our experimental work builds on our previous study, which focused on the implementation of the **GReaT** approach. GReaT was chosen primarily due to its advantages over other models, consisting mainly of a straightforward implementation without significant data processing that might result in data alteration. We apply the GReaT method to a collection of publicly available reported cyber-attacks, with the final dataset containing 3388 cases. We focus particularly on the technical characteristics (e.g., source name, target ID and target name), and free-text descriptions.

An LLM pretrained specifically for tabular data, the **tabularisai/Qwen3-0.3B-distil** model [14], has been fine-tuned on training data. This data was obtained by previously splitting our dataset using a 70/30 train-test ratio.

The evaluation of our data generation approach relies on two fundamental attributes of synthetic data compared to the original dataset: (1) privacy and (2) data quality (encompassing both fidelity and diversity). To assess these characteristics, we compare our chosen tabularisai/Qwen3-0.3B-distil model against the unsloth/Llama 3.2 1B model used in our previous study. These metrics are designed to measure how closely the generated data resembles unseen records from the original dataset without simply memorizing the training data.

To assess privacy risk, we employ the DCR metric, which calculates the distance between each generated data point and its nearest neighbor in the training set. Figure 1 compares the DCR distributions of the baseline (Train vs. Test) against the synthetic data (Synthetic vs. Train) for both models. The results reveal a stark contrast between the two architectures. For the tabularisai/Qwen3-0.3B-distil model (Fig. 1.a), the distributions are nearly identical, indicating that the synthetic data remains as distant from the training data as the real test set does. This confirms low privacy risk and no systematic replication. Conversely, the unsloth/Llama 3.2 1B model (Fig. 1.b) exhibits a massive spike near zero, indicating severe memorization of the training samples and a privacy risk, as the generated data is merely reproducing the original dataset rather than generating novel samples.

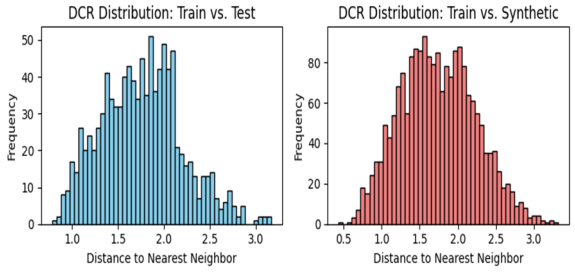


Fig. 1.a DCR distributions, referring to the tabularisai/Qwen3-0.3B-distil model, computed with standardized numerical features and L2-normalized LM embeddings, using Euclidean (cosine-equivalent) distance. "Test vs Train" shows the DCR distance between test and training data set, while "Synthetic vs Train" shows the distance between synthetic data and test data. The nearly identical DCR distributions indicate no systematic replication of training samples. Privacy risk appears low, with only isolated minor closeness that remains within acceptable bounds.

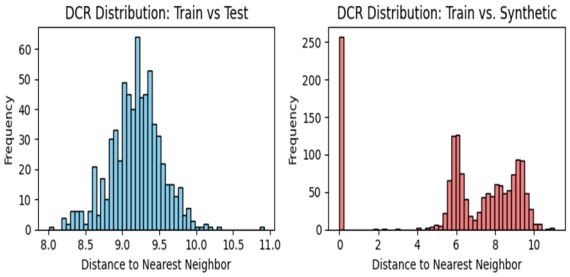


Fig. 1.b DCR distributions, referring to the Unloth/Llama 3.2 1B model. Unlike the previous model, the DCR distributions are highly dissimilar. The "Train vs. Synthetic" plot exhibits a massive spike near zero, indicating systematic replication and severe memorization of the training samples. Privacy risk for this model appears high, as the synthetic data is not generating novel samples but rather reproducing the original data.

Synthetic data fidelity is evaluated through various indicators. First, we analyze the DCR distributions (Fig. 2) comparing the baseline (Train vs. Test) to the synthetic data (Synthetic vs. Test). The results reveal a clear performance gap between the two models**.** For the tabularisai/Qwen3-0.3B-distil model (Fig. 2.a), the distributions are nearly identical in shape and spread. This suggests high fidelity, as the synthetic generator successfully produces data points that mirror the complexity and density of the unseen test set. Conversely, the unsloth/Llama 3.2 1B model (Fig. 2.b) exhibits notable discrepancies. While it produces data points with a lower minimum DCR (indicating some points are remarkably close to the test data), the overall distribution is skewed, with a distinct secondary concentration at lower distances that is absent in the baseline. This indicates that while the Llama model can generate realistic individual examples, it shows a weaker alignment with the overall structural spread of the real data.

Considered together, these figures reinforce confidence in the high quality of the synthetic data, not only with regard to the fidelity and diversity of the original data, but also with regard to privacy. Furthermore, Table 1 provides a comprehensive summary of the metrics comparing the synthetic data quality generated by both models. These indicators demonstrate that the Qwen-based model produces superior synthetic data compared to the Llama-based model. Specifically, the Qwen model achieves a strong balance of fidelity and diversity, closely mirroring the real data distribution and avoiding the severe overfitting observed in the Llama model. Ultimately, the Qwen model's synthetic data are highly faithful and suitable for augmentation, whereas the Llama model suffers from reduced structural diversity and memorization of the training set.

While the Classifier Two-Sample Test (C2ST) shows similar scores for both models (~0.70, slightly above the ~0.50 real-to-real baseline), indicating they are not trivially separable, though some systematic differences remain, the divergence becomes clear in the distance metrics. The first model exhibits a much lower Maximum Mean Discrepancy (MMD) of 0.6% compared to 3.39% for the second model.

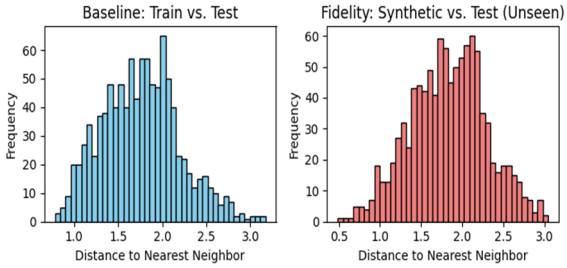


Fig. 2.a DCR distributions, referring to the tabularisai/Qwen3-0.3B-distil model computed with standardized numerical features and L2-normalized LM embeddings, using Euclidean (cosine-equivalent) distance. "Test vs Train" shows the DCR distance between test and training data set, while "Synthetic vs Test" shows the distance between test and synthetic data. The visual comparison between the two distributions indicates nearly identical DCR distributions in shape and spread, despite slight differences concentrated in restricted intervals.

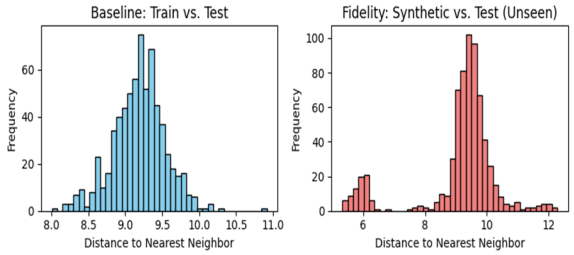


Fig. 2.b DCR distributions, referring to the unsloth/Llama model. The visual comparison between the two distributions reveals notable discrepancies in shape and spread. While both distributions share a peak around a distance of 9.0 to 9.5, the synthetic data exhibits a wider overall spread and a distinct secondary concentration of samples at lower distances (approximately 6.0 to 8.0) that is not present in the baseline.

Furthermore, the "Distance to real manifold" metrics reveal a critical difference in behavior. For the first model (blue), the synthetic data lies close to the real manifold (mean 2.00 vs. baseline 1.78; 95th percentile 2.80 vs. 2.70), with 8.2% of samples falling beyond the 95th percentile—evidence of modest, controlled edge expansion. Conversely, the second model (red) generates synthetic data with a mean distance of 6.2, which is closer to the training data than the real test set's baseline of 9.19. This, combined with only 2.52% of samples falling beyond the 95th percentile, strongly suggests the second model is memorizing the training data rather than generating novel samples **(aligning with the massive spike at zero in its DCR distribution)**.

Table. 1. Comparison of synthetic data quality generated by two models (**tabularisai-Qwen3-0.3B-distil vs unsloth/Llama 3.2 1B).** The synthetic data generated by the first model **(tabularisai/Qwen3-0.3B-distil)** demonstrate superior performance and higher fidelity to the real held-out distribution compared to the second model**(unsloth/Llama)**.

Metric		LLM used			
		Tabularis ai-Qwen3-0.3B-distil / unsloth/Llama 3.2 1B		Tabularis ai-Qwen3-0.3B-distil / unsloth/Llama 3.2 1B	
		Real-to-real distance baseline		Synthetic-to-real distances	
Classifier Two-Sample Test (C2ST)		0.516	0.502	0.7	0.7021
Maximum Mean Discrepancy (MMD)				**0.6%**	**3.39%**
Distance to real manifold	**Mean**	**1.78**	**9.19**	**2.00**	**6.2**
	95th perc.	**2.70**	**9.89**	**2.80**	**9.5**
	Fraction of point beyond the 95th percentile			**0.082**	**0.025**
Nearest-Neighbour Consistency Metrics (PRDC metrics)	Precision	0.98	0.97	0.93	0.94
	Recall	**0.98**	**0.98**	**0.96**	**0.85**
	Density	1.001	1.06	0.76	0.984
	Coverage	0.968	0.98	0.90	0.93

Finally, the PRDC metrics confirm this disparity. The first model (blue) shows excellent precision (0.93) and recall (0.96), meaning it accurately represents the real data's distribution. The second model (red) achieves slightly higher precision (0.94) and density (0.984) but suffers a notable drop in recall (0.85). **This indicates that while the second model's synthetic samples are realistic, they fail to cover the full breadth of the real data's local neighborhoods.**

5. Conclusion

This work demonstrates a viable and efficient approach to generating synthetic tabular data of high quality in the cybersecurity domain with an LLM-based approach adapted from GReaT and fine-tuned on a categorical-heavy corpus of nearly 3400 incidents.

Results show that using the GReaT approach combined with the **tabularisai/Qwen3-0.3B-distil LLM** model ensures high fidelity to key statistical and structural properties of real cybersecurity datasets, while maintaining strict privacy guarantees. As evidenced by (DCR) distributions and manifold metrics, the **tabularisai**/Qwen3-0.3B-distil model avoids systematic replication of training samples, whereas generic LLMs (i.e., unsloth/Llama 3.2 1B) exhibit severe memorization and, as a result, high privacy risks. Furthermore, the Qwen-based model generates more robust and diverse synthetic data compared to these generic models, maintaining strong PRDC metrics (precision 0.93, recall 0.96). If edge coverage is a priority, it can also safely expand the data manifold, placing a controlled fraction of synthetic points (8.2%) slightly outside the original real-value boundaries without compromising privacy. Beyond confirming the viability of synthetic data generation established in our prior research, these findings demonstrate a significant advancement in data quality, diversity, and privacy preservation.

Ultimately, this suggests that using appropriate, targeted LLMs aligned with the underlying original data is of crucial importance in the cybersecurity domain. While LLMs can trivially memorize and leak training data if poorly matched to the task, selecting a model that effectively generalizes the original distribution allows for safe, high-fidelity data augmentation.

References

[1]. T. Khot, H. Trivedi, M. Finlayson, Y. Fu, et al., Decomposed prompting: A modular approach for solving complex tasks, in *Proceedings of the 11th International Conference on Learning Representations (ICLR 2023)*, OpenReview.net, 2023, Paper ID _nGgzQjzaRy.

[2]. P. Liu, W. Yuan, J. Fu, Z. Jiang, et al., Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, *ACM Computing Surveys*, Vol. 55, Issue 9, 2023, Article 195, 35 p.

[3]. A. Jolicoeur-Martineau, K. Fatras, T. Kachman, Generating and imputing tabular data via diffusion and flow-based gradient-boosted trees, in *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS 2024)*, *Proceedings of Machine Learning Research*, Vol. 238, *PMLR*, 2024, pp. 1288–1296.

[4]. S. Onishi, S. Meguro, Rethinking data augmentation for tabular data in deep learning, arXiv preprint, arXiv:2305.10308, 2023.

[5]. R. Sauber-Cole, T. M. Khoshgoftaar, The use of generative adversarial networks to alleviate class imbalance in tabular data: A survey, *Journal of Big Data*, Vol. 9, 2022, Article 98.

[6]. X. Fang, W. Xu, F. A. Tan, J. Zhang, et al., Large language models (LLMs) on tabular data: Prediction, generation, and understanding – A survey, arXiv preprint, arXiv:2402.17944, 2024.

[7]. G. Giacomello, O. Preka, Developing synthetic data for cybersecurity policies, in *Proceedings of the First International Conference on Big Data Analytics & Applications (BDAA' 2025)*, Innsbruck, Austria, 25–27 November, 2025, pp. 24–26.

[8]. Unsloth AI, Llama-3.2-1B-Instruct-GGUF, Large Language Model, Hugging Face, Revision 86dea7b, 2025. Available at: https://huggingface.co/unsloth/Llama-3.2-1B-Instruct-GGUF (accessed 5 September 2025).

[9]. Tabularis AI, Qwen3-0.3B-distil, Large Language Model, Hugging Face, Revision d778cac, 2025. Available at: https://huggingface.co/tabularisai/Qwen3-0.3B-distil (accessed 5 July 2026).

[10]. D. Lopez-Paz, M. Oquab, Revisiting classifier two-sample tests, in *Proceedings of the 5th International Conference on Learning Representations (ICLR 2017)*, OpenReview.net, 2017, Paper ID SJkXfE5xx.

[11]. N. Park, M. Mohammadi, K. Gorde, S. Jajodia, et al., Data synthesis based on generative adversarial networks, *Proceedings of the VLDB Endowment*, Vol. 11, Issue 10, 2018, pp. 1071–1083.

[12]. T. Liu, Z. Qian, J. Berrevoets, M. van der Schaar, GOGGLE: Generative modelling for tabular data by learning relational structure, in *Proceedings of the 11th International Conference on Learning Representations (ICLR 2023)*, OpenReview.net, 2023, Paper ID fPVRcJqspu.

[13]. V. Borisov, K. Seßler, T. Leemann, M. Pawelczyk, et al., Language models are realistic tabular data generators, in *Proceedings of the 11th International Conference on Learning Representations (ICLR 2023)*, OpenReview.net, 2023, Paper ID cEyqmQNOeI.

[14]. A. V. Solatorio, O. Dupriez, REaLTabFormer: Generating realistic relational and tabular data using transformers, arXiv preprint, arXiv:2302.02041, 2023.

[15]. T. Zhang, S. Wang, S. Yan, J. Li, et al., Generative table pre-training empowers models for tabular prediction, in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*, *Association for Computational Linguistics*, Singapore, 2023, pp. 14836–14854.

[16]. Z. Zhao, R. Birke, L. Y. Chen, TabuLa: Harnessing language models for tabular data synthesis, in *Proceedings of the 29th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2025)*, *Springer Nature*, Singapore, 2025, pp. 247–259.

ISBN 978-84-09-93454-6

