BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en Vol. 30 No. 1. 2025

UDC 004.89:004.056.5
DOI: 10.62660/bcstu/1.2025.48

Utilising large language models for automated real-time cyber threat analysis

Denys Kovalchuk*
Postgraduate Student
International Humanitarian University
65009, 33 Fontanska Doroha Str., Odesa, Ukraine
https://orcid.org/0009-0003-2302-8698

Abstract. In the contemporary cybersecurity landscape, where the rapid growth in the quantity and complexity of threats has undermined the effectiveness of traditional rule- and signature-based detection methods, an urgent need has emerged for automated cyber threat analysis systems employing large language models. The objective of this study was to investigate the capabilities of large language models for automated cyber threat analysis, risk assessment, and improving incident response efficiency in corporate environments. To achieve this goal, machine learning and natural language processing techniques were employed, particularly the adaptation of large language models for threat classification, risk-level evaluation, and anomaly detection. A system was developed to analyse incoming and outgoing email communications, which during testing automatically identified phishing attacks and social engineering techniques, assigned risk scores to messages, and quarantined those exceeding a predefined threshold (e.g., 0.8) for further inspection. The system analysed a dataset of 100,000 emails, of which 70% were legitimate communications and 30% were phishing attacks. Additionally, real-time analysis of data streams from corporate logs and external sources enabled the detection of potential cyber incidents with an accuracy of up to 94%, while reducing the false-positive rate to 6.5%. The obtained results confirmed the efficacy of large language models, which achieved a threat classification accuracy of up to 97% with an F1-score of 95% and reduced incident response times by 30-40%. These findings can be leveraged by other researchers to refine phishing detection techniques, reduce false positives in corporate security systems, and integrate machine learning models with diverse data sources, including SIEM systems and external cybersecurity resources

Keywords: natural language processing; machine learning for security; phishing attack detection; anomaly detection; deep learning in cybersecurity; neural networks for security; cyber threat intelligence

INTRODUCTION

The advancement of digital technologies and the increasing volume of transmitted data have led to a rise in the sophistication of cyber threats, complicating their timely detection and mitigation. Cyberattacks have become increasingly dynamic, employing modern evasion techniques to bypass traditional defence mechanisms, such as signature analysis and static detection rules, and are increasingly targeting critical systems. Conventional threat analysis approaches, reliant on signature-based detection or static rule sets, demonstrate limited effectiveness against novel and modified attacks. Modern threats – such as phishing campaigns, social engineering, and malware-based attacks – often evade standard security tools due to their inability to analyse contextual cues and adapt to emerging threats in real time.

For instance, S. Jamal & H. Wimmer (2023) emphasised in their research that applying transformer-based models for phishing detection in corporate email communications significantly improves

Article's History: Received: 11.09.2024; Revised: 22.02.2025; Accepted: 17.03.2025.

Suggested Citation:
Kovalchuk, D. (2024). Utilising large language models for automated real-time cyber threat analysis. *Bulletin of Cherkasy State Technological University*, 30(1), 48-58. doi: 10.62660/bcstu/1.2025.48.

*Corresponding author

classification accuracy and reduces the number of missed attacks. O. Cherqi *et al*. (2023) proposed a methodology combining generative adversarial networks (GANs) and contrastive learning to better identify complex and rare attack types undetected by traditional methods. Consequently, there is a pressing need for intelligent cyber threat analysis methods capable of rapidly and accurately identifying suspicious activity across diverse data sources. One promising solution lies in the application of large language models (LLMs), which possess advanced capabilities in natural language processing (NLP), textual data analysis, and uncovering hidden patterns in large datasets. The use of LLMs in cybersecurity opens new avenues for automating threat analysis processes, including monitoring corporate communications for phishing attempts, analysing web content for potentially malicious resources, and aggregating threat intelligence from open and closed sources.

In the work of K. Singh *et al*. (2022), the potential of NLP techniques combined with machine learning algorithms was explored for detecting vulnerabilities in documentation and source code. The study found that integrating these methods not only identified obvious errors but also successfully flagged potential threats arising from inaccuracies in textual descriptions. The results confirmed the efficacy of this approach in threat analysis, enhancing the security of corporate information systems. A systematic review by Y. Gholami (2024) examined the prospects of leveraging LLMs for detecting latent threat patterns. It was noted that LLMs' capacity for deep contextual analysis enabled the identification of both explicit malicious indicators and subtle patterns often overlooked by traditional signature-based methods. Additionally, the analysis of heterogeneous data sources facilitated the early detection of emerging cyber threats.

The study by L. Rybalchenko & S. Ohrimenco (2024) evaluated the applicability of LLMs and other machine learning algorithms in large enterprises. It highlighted that the effectiveness of baseline models was constrained by insufficient adaptation to specific corporate cybersecurity requirements. The findings underscored the necessity for further optimisation, domain-specific data integration, and transfer learning techniques to reduce false positives and improve threat detection accuracy. In the work of O.P. Podvysotska & S.O. Nosok (2024), machine learning algorithms were applied to network traffic analysis. The study demonstrated that combining structural and semantic analysis enabled timely diagnosis of potential threats and provided actionable response recommendations. The results indicated the feasibility of integrating such methods into security information and event management (SIEM) systems to enhance their efficacy.

In the study by V.B. Mokin & M.G. Pradivliannyi (2024), innovative approaches to developing self-learning cybersecurity systems were examined through the integration of machine learning, data mining, and the "Artificial Intelligence of Things" (AIoT) concept. It was established that combining AIoT with LLMs enabled the timely detection of novel cyberattack types and enhanced system resilience against evolving adversarial tactics. However, the study also emphasised that achieving maximum efficacy required addressing challenges such as energy consumption, computational resource scalability, and the development of additional safeguards against adversarial attacks.

A. Partyka *et al*. (2024) focused on analysing cybercrime, particularly ransomware attacks, using AI models. The study demonstrated that applying these technologies enabled deep data analysis for early detection of ransomware attack signatures. The findings indicated that integrating such approaches reduced incident response times, automated decision-making processes, and significantly improved the security management of critical infrastructures. The objective of this study was to investigate the capabilities of modern LLMs in performing automated real-time cyber threat analysis and assess their practical efficacy when integrated into corporate security systems.

MATERIALS AND METHODS

The study employed a simulated large-scale organisational infrastructure with a typical corporate network architecture. Testing was conducted in the AWS Cloud, and pre-trained LLMs were accessed via REST API. To ensure representativeness, all data channels (email communications, network logs, user behaviour data) were configured to replicate the workload of a large enterprise with approximately 2,500 workstations, including 2,000 concurrently active users and 500 additional technical and service accounts. This approach provided a realistic corporate environment for effective cyber threat analysis.

To test the hypothesis that LLMs can detect phishing emails, a dataset of 100,000 emails was compiled. Approximately 70% comprised legitimate business communications, while the remaining 30% contained various phishing scenarios. This distribution simulated real-world conditions, where genuine threats are outnumbered by legitimate traffic. Raw data were cleansed of duplicates, technical headers, and mislabelled entries, then normalised into a unified text structure. To verify email categorisation as "malicious" or "benign", samples were generated using Python scripts and open-source LLMs. Phishing examples were divided into two categories: (1) simple variants (credential harvesting via spoofed links); and (2) advanced social engineering scenarios (executive impersonation, fake newsletters, etc.). The dataset was split into training and test sets, maintaining a similar ratio of benign to malicious samples.

Multiple threat identification approaches were employed to evaluate their responsiveness to novel and modified attacks. Signature-based analysis, classical machine learning, and LLMs were selected as

they best reflect contemporary trends in corporate security systems. The first approach considered was signature-based detection, which relied on predefined patterns and keywords. This method required minimal computational resources and ensured rapid execution; however, it proved vulnerable to novel formulations and evasion techniques. The second approach involved classical machine learning algorithms (logistic regression and support vector machines), which provided better generalisation capabilities compared to signature-based methods. These algorithms operated on a set of vectorised features derived from real emails containing both phishing examples and legitimate communications. While this approach required a well-labelled dataset, it enabled the detection of previously unknown threats by identifying patterns within textual data. The third toolset comprised LLMs, notably GPT-2, GPT-3, and a specially fine-tuned version of GPT-4. These models were employed for in-depth contextual analysis of emails, owing to their ability to detect even subtle deviations in style, vocabulary, and hidden indicators of social engineering. To enhance the relevance and accuracy of threat detection within a specific environment, LLMs were additionally supplemented with internal corporate data (real email correspondences, security policies).

For model generalisability assessment, 70% of the dataset (legitimate emails) served as training data, while 30% (malicious emails) formed the test set. Training samples maintained the original dataset's phishing-to-benign ratio. Signature-based methods used manually defined keywords, classical ML models trained on vectorised features, and LLMs underwent fine-tuning on internal corporate data to capture enterprise-specific linguistic patterns. Post-training, each algorithm was evaluated on unseen test data to avoid overfitting and objectively assess detection accuracy. Performance metrics included precision, recall, and F1-score, ensuring a balance between threat detection (minimising missed attacks) and false-positive reduction (avoiding misclassification of benign emails). Computations were performed on an Intel Xeon server with AWS acceleration for efficient large-scale data processing.

Concurrently, models were tested in network activity monitoring. Data sources included firewall, proxy, and authentication server logs aggregated by SIEM systems (Splunk, Elastic Stack, etc.), totalling 10 million records. A subset contained explicit attack indicators (SQL injection, brute force, port scanning, suspicious URLs), while the remainder constituted legitimate traffic. Evaluated approaches included: Rule-based SIEM correlation detectors; Random Forest; XGBoost ensembles; Domain-adapted LLMs (GPT-4). As network logs required detection of both known and novel threats, a risk-scoring mechanism was implemented. Each record was pre-processed to extract textual and metadata features (request type, access attempts, anomalous URIs, etc.). Signature-based methods matched known exploit patterns, classical ML models evaluated feature vectors, and LLMs analysed textual log fields for unusual phrases, escalation markers, or anomalous user behaviour. To maintain detector relevance against emerging attack vectors, periodic model retraining on recent incidents was implemented. Adversarial robustness was ensured via data sanitisation (removing hidden characters, distortions) and structural anomaly checks. This approach enhanced resilience against sophisticated multi-vector attacks where traditional signatures or attacker adaptations would otherwise evade detection.

RESULTS

The hypothesis regarding the capability of LLMs to detect phishing emails was tested by constructing a dataset of 100,000 emails, where 70% comprised legitimate business correspondence and 30% contained phishing attack scenarios. This approach simulated real-world conditions, where the volume of genuine threats is significantly lower than the flow of legitimate communications. Using risk assessment mechanisms, these models assigned threat levels to incoming messages and, upon exceeding a critical threshold, redirected them to quarantine for further analysis by security experts. Additionally, LLMs facilitated effective monitoring of cyber threat intelligence gathered from diverse sources, including corporate event logs, cybersecurity blogs, forums, and dark web resources. This methodology enabled real-time threat analysis, substantially improving incident response efficiency.

A critical function of LLMs was their ability to process and analyse corporate system log files, allowing for automated detection of anomalies in network activity and timely alerts to relevant security teams. E. Joshua *et al*. (2025) described the AIDTIS system, which integrates LLMs with threat intelligence sources (such as the dark web and cybersecurity blogs), providing a more comprehensive approach to analysing malicious activity and accelerating decision-making processes for blocking or isolating adversarial actions.

The presented results summarised experimental evaluations conducted to assess the potential of LLMs in automated real-time cyber threat analysis. The study focused on applying LLMs to detect phishing emails in corporate environments, assess social engineering risks, analyse anomalies in network logs, and integrate third-party threat intelligence sources (including cybersecurity blogs, forums, and dark web resources). The research aimed to determine whether models such as GPT or Bidirectional Encoder Representations from Transformers (BERT) met expectations regarding improved incident detection accuracy and accelerated response processes in enterprise settings.

The escalation of attacks targeting large enterprises, government institutions, and critical infrastructure underscored the necessity of real-time automated cyber threat analysis. Leveraging LLMs for pattern recognition in textual data, log files, email communications,

and external threat intelligence sources enabled a deeper level of threat analysis compared to traditional methods. The proposed methodology allowed for automated identification of phishing campaigns through textual analysis, detecting signs of social engineering, and evaluating individual message risk levels. Research findings confirmed that LLMs significantly reduced false-positive rates in threat detection systems due to their ability to contextualise messages and identify hidden lexico-semantic patterns characteristic of phishing and social engineering attacks.

The dataset included both trivial scenarios (e.g., password theft via spoofed links) and sophisticated social engineering tactics (e.g., impersonation of internal communications from executives, fake corporate newsletters). All emails were labelled by experts and divided into training and test sets. Multiple approaches were evaluated: rule-based detection and signatures, classical machine learning methods (logistic regression, Support Vector Machine), and LLMs (GPT-2, GPT-3, and a fine-tuned variant of GPT-4). The study revealed that even pre-trained LLMs outperformed traditional methods in classifying malicious emails. A notable advantage was their ability to detect modified phishing techniques, where attackers altered message structure, lexicon, or format to evade signature-based detection. In practice, this meant that GPT-derived models identified lexical and syntactic anomalies at a deeper level, analysing overall context rather than relying solely on isolated keywords. Testing demonstrated that LLM-based email screening achieved significantly higher accuracy, particularly against spear-phishing campaigns tailored to specific targets.

To illustrate the performance disparity between methods, key metrics were examined: accuracy, recall, precision, and F1-score. Results were aggregated in a comparative table (Table 1). Detection thresholds were standardised across models to prevent bias from varying configuration parameters. The table also documented false-positive and false-negative ratios, as both metrics were deemed critical from a security standpoint: excessive false alarms overload Security Operations Centre (SOC) analysts, while missed attacks pose direct organisational risks.

Table 1. Comparison of accuracy and F1-score for different phishing email detection methods

Method	Accuracy	Recall	Precision	F1-score
Rules/Signatures	0.84	0.76	0.82	0.79
Logistic regression	0.88	0.81	0.85	0.83
Support vector machine	0.9	0.84	0.87	0.85
GPT-2 (fine-tuned)	0.93	0.89	0.91	0.9
GPT-3 (fine-tuned)	0.95	0.92	0.93	0.92
Integrated LLM (GPT-4)	0.97	0.94	0.96	0.95

Source: compiled by the author

Notably, the GPT-4 variant achieved the highest accuracy (0.97) while maintaining an F1-score of 0.95, demonstrating a balanced trade-off between detecting genuine phishing threats (high recall) and correctly filtering benign emails (high precision). This performance was attributed to GPT-4's expanded parameter count and domain-specific fine-tuning on corporate datasets. Crucially, despite its complexity, GPT-4 adapted dynamically to evolving social engineering tactics, detecting novel patterns absent from initial signature databases.

Subsequent experiments evaluated LLM-driven "risk scoring" mechanisms for incoming emails. Each message was analysed for lexico-semantic features (keyword frequency, contextual usage), embedded links, attachments, and stylistic consistency with typical corporate correspondence. If a message's risk score exceeded a dynamic threshold (calibrated per environment), it was automatically quarantined for SOC review. This approach proved particularly effective against sophisticated attacks where legacy systems relied solely on known malicious link signatures or rigid detection rules. A model such as GPT-4 was capable of responding even to concealed and non-standard formulations, as well as detecting atypical use of communicative genre – for instance, the simulation of a scenario in which a department head addressed an employee in an uncharacteristic tone or issued a request for confidential information that would not align with normal organisational practices.

Further analysis examined LLM applications in network activity monitoring and log anomaly detection. A typical SIEM component aggregated logs from firewalls, authentication servers, email gateways, and proxies. The experiment involved a sample of 10 million log entries (user authentications, connection attempts, database queries, module errors). Experts labelled a subset as routine/"noise" and identified genuine cyberattacks (SQL injections, port scanning, low-intensity DDoS, brute-force attempts). Evaluated methods included: (1) rule-based SIEM correlation, (2) Random Forest, (3) XGBoost ensemble, and (4) integrated LLM (GPT-4 with domain adaptation). Key metrics were: mean processing time per log entry (ms), false-positive rate, and true-positive detection rate (Table 2).

Table 2. Comparison of approaches to anomaly detection in corporate logs

Approach	Average processing time (ms)	False positives (%)	Detected real threats (%)
Correlation Rules (Baseline SIEM)	0.5	15.2	83
Random Forest	1.7	12.8	88.5
XGBoost	2	10.2	90.1
Integrated LLM (GPT-4, fine-tuned)	2.5	6.5	94.3

Source: compiled by the author

During the analysis of these results, it was observed that the rule "the more complex the model, the longer the processing time" was indeed confirmed: GPT-4 required an average of 2.5 ms per log line, whereas the baseline SIEM processed data in 0.5 ms. However, a significantly lower number of false positives was detected (6.5% compared to 15.2% for correlation rules), along with a higher percentage of identified threats (94.3% vs. 83%). Consequently, there was a clear benefit in security analysts receiving fewer "empty" alerts while being more likely to capture genuine incidents. The reduction in manual verification volume, in turn, shortened the overall response time to sophisticated cyberattacks and diverted fewer human resources. Additional stress tests (10 million logs) demonstrated that the integrated LLM maintained stable efficiency without significant degradation in throughput or accuracy.

Parallel testing examined how LLM behaved when integrated with heterogeneous data sources outside the corporate environment, such as specialised blogs, forums, Cyber Threat Intelligence services, and darknet resources. It was observed that GPT-derived models quickly "picked up" emerging trends when certain vulnerabilities or exploits were frequently mentioned in forum posts or blogs. Such contextual information helped prioritise network events associated with potential exploit usage. This reduced the risk of missing attacks during the period before vendors released official security updates. As a result, Zero-Day detection became more agile, as LLMs could evaluate even indirect references to novel attacker techniques.

Additionally, analysis of embedded file content (e.g., Microsoft Office documents with macros) or links enabled the system to identify potentially malicious activity faster than conventional signature-based antivirus checks. In simulated attack scenarios where, malicious components were loaded in stages, traditional antivirus solutions often failed to detect threats until the final file or script was activated. In contrast, the LLM, by analysing message context and attachment characteristics, flagged high-risk indicators at the email ingestion stage. The ratio of misclassified "safe" emails with attachments decreased by approximately 13-15% compared to traditional methods, significantly lowering the likelihood of successful exploitation.

Furthermore, the hypothesis regarding improved incident detection and response metrics was confirmed. SOC workflows accelerated due to automated preliminary analysis, which filtered out "noisy" alerts and isolated incidents requiring expert scrutiny. The language model not only signalled potential threats but also provided concise explanations for their risk (e.g., alignment with new exploit intelligence, atypical user behaviour, or semantically suspicious text). Analysts thus received a preliminary "dossier" for each incident, enabling faster decision-making.

However, LLM integration also presented challenges. Continuous fine-tuning was necessary due to the rapidly evolving threat landscape and adversarial evasion techniques. Outdated models risked misclassifying novel attack patterns absent from their original training data. Protecting the model against adversarial inputs – where attackers deliberately manipulate text to deceive the LLM – was also critical. While GPT-4 exhibited resilience to minor lexical or syntactic alterations, carefully crafted adversarial constructs could still bypass detection. Thus, robust adversarial testing and secure training frameworks were recommended.

A key finding emphasised the benefits of integrating LLMs with other cybersecurity subsystems (firewalls, IPS/IDS, Data Loss Prevention modules). Such interoperability created a holistic data chain, correlating anomalous network activity, suspicious emails, and external threat intelligence. This "multi-layered defence" positioned the LLM as a unifying layer, transforming heterogeneous data (logs, forum texts, messages) into actionable signals, thereby reducing blind spots and improving threat correlation. Technical implementation required substantial computational resources: AWS Cloud services were utilised, with LLM operations conducted via Rest API. Though deployment costs and complexity increased, the resultant risk reduction, cost savings from mitigated incidents, and reputational gains justified the investment. Smaller enterprises could leverage cloud-based LLM services to minimise local infrastructure demands.

Overall, high-capacity LLMs like GPT-4 proved exceptionally valuable for analysing phishing emails and logs, enhancing both security and automation. Accuracy exceeded 0.95, with false positives reduced by several percentage points compared to classical methods – significantly improving SOC efficiency. Contextual analysis (semantic patterns, external threat feeds, corporate communication nuances) also made detection more adaptable, reducing reliance on static signatures or rules. This resilience was particularly effective against evolving phishing tactics and network anomalies.

Notably, incident response times improved. During testing, LLM-based pre-filtering assigned "priority levels" and explanatory context to logs/alerts, allowing SOC analysts to focus on critical incidents without sifting through false alarms. Decision-making cycles shortened by 30-40% compared to baseline signature/correlation-rule approaches. Nevertheless, safeguards against LLM-specific attacks (e.g., input manipulation or adversarial text) remained essential. Experiments documented cases where attackers obfuscated malicious intent via complex symbols or metadata, though GPT-4's contextual processing demonstrated superior robustness. Future work should integrate secure training methods and regular input audits to detect anomalous patterns.

Overall, the results confirmed the validity of applying LLMs in cybersecurity systems for automated real-time threat analysis. Improvements were observed in the detection of sophisticated phishing attacks, a significant reduction in false positive rates, and accelerated response times to potential incidents. The integration of LLMs with other components (e.g., traditional SIEM solutions, firewalls, Data Loss Prevention tools) provided an additional synergistic effect: information from heterogeneous sources was correlated at the semantic analysis level, facilitating the timely detection of non-obvious indicators of cyberattack preparation or deployment. NLP and deep learning technologies enabled the formation of a more comprehensive threat landscape, substantially enhanced the efficacy of Threat Intelligence practices, and paved the way for implementing a fully-fledged Threat Hunting approach.

The practical conclusion drawn from the obtained results emphasised the importance of regular model updates and rigorous testing to ensure sustained relevance in the rapidly evolving cybersecurity environment. It was recommended to track new LLM versions and explore their adaptability to specific sectoral requirements (industry, financial institutions, public sector, etc.). Specialised fine-tuning and periodic audits helped maintain high threat detection accuracy and prevent the accumulation of "blind spots". This approach allowed for more effective utilisation of NLP capabilities in detecting evolving and increasingly sophisticated variants of cyberattacks.

Thus, empirical evidence demonstrated that integrating LLMs into corporate defence ecosystems proved promising, enabling reduced data analysis time, improved threat classification accuracy, and a significant decrease in false positive alerts. Implementing such solutions required adequate computational resource preparation, involvement of competent machine learning specialists, and meticulous work with large datasets. At the same time, the resulting time and resource savings, alongside reduced likelihood of successful breaches, offset the implementation costs and complexity.

DISCUSSION

The study confirmed that adapting LLMs for real-time automated cyber threat analysis significantly improved phishing attack detection accuracy and accelerated incident response. Compared to predefined rule-based approaches, GPT-type models enabled contextual message analysis, detection of hidden social engineering patterns, and faster adaptation to novel threats as attackers altered their lexicon or format. These observations aligned with the work of J.T. Santoso *et al*. (2024), which highlighted the ability of transformer architectures to enhance response efficacy against complex attacks through self-attention mechanisms. The study achieved qualitative metrics even with relatively small datasets, while under substantially larger-scale conditions, the approach demonstrated robustness in more realistic scenarios.

The results also underscored the importance of multi-source analysis, where LLMs simultaneously processed corporate email content, network logs, and external threat intelligence (cybersecurity blogs, forums, darknet data). In such scenarios, the model better recognised potentially malicious activity by correlating indirect threat indicators across diverse data formats. A similar principle was outlined in O. Ogundairo & P. Broklyn (2024), where NLP algorithms were enriched with event log metadata, enabling more precise security incident classification and prioritisation. Comparative analysis showed that LLM deployment markedly reduced threat detection windows and strengthened contextual analysis.

The phishing email dataset used in the experiment included varying complexity levels: from classic credential harvesting to advanced social engineering variants mimicking inter-departmental corporate correspondence. GPT models successfully identified even spoofed executive communications, which often evaded simpler detectors. This trend corroborated the findings of P. Damola & A. Miracle (2024), who demonstrated LLMs' capability to autonomously recognise non-standard intrusion indicators and serve as a foundation for real-time suspicious activity blocking. S.M. Taghavi & F. Feyzi (2024) emphasised LLMs' effectiveness in rapid code and documentation review for detecting malicious patterns.

Substantial improvements in precision and F1-scores were achieved through domain-specific pre-training or fine-tuning. Analogous to the "SecureBERT" concept in E. Aghaei *et al*. (2022), GPT-4's competency increased when provided with representative corporate examples of real correspondence, security policies, and phishing patterns. Accuracy metrics (up to 0.97) surpassed traditional machine learning algorithms by 7-9% and signature-based approaches by up to 13%. Compared to P. Ranade *et al*. (2021), where generative transformers created synthetic phishing emails for training, the current study utilised significantly larger volumes of real samples, enhancing the model's ability to discern even subtle lexical deviations.

Concurrently, particular attention was required to protect LLMs from adversarial actions, as stressed by S. Liu *et al.* (2024). The research documented cases where attackers deliberately embedded hidden characters or distorted email structures to confuse models. While GPT-4 exhibited greater resilience to lexical obfuscation, data filters and regular red-team audits remained necessary to maintain system relevance. This approach aligned with warnings from M. Alabi & F. Ademola (2024), who analysed multiple attack vectors – from data poisoning to generative adversarial examples – and advocated for comprehensive detection mechanisms.

Prompt engineering also proved critical. GPT-4 required meticulous query calibration to ensure correct interpretation of threat patterns and their contextual relevance. This finding intersected with S. Parmar & H. Patel (2024) results, where well-formulated prompts substantially improved response accuracy. Early identification of phishing indicators mimicking internal corporate style emerged as a key success factor. M. Kuppam (2024) noted that application telemetry analysis helped detect attack-related behavioural anomalies.

Network log analysis – particularly SMTP headers, HTTP requests, authentication attempts, and other traffic elements – also showed promise. LLMs processed vast log volumes, correlated user actions with known malicious scanning/brute-force patterns, and assigned elevated risk scores where rule-based detection failed. This resonated with C. Avci *et al.* (2024) conclusions on self-attention mechanisms for complex threat detection. GPT-4 successfully identified subtle behavioural chains where multiple seemingly insignificant anomalies collectively indicated multi-stage attack preparation. N.G. Ambekar & S. Thokchom (2024) highlighted how unsupervised learning (e.g., variational autoencoders) could be augmented with LLMs for Zero-Day threat detection.

Comparative analysis with prior research revealed specific LLM limitations. Firstly, computational infrastructure demands grew substantially: GPT models (especially GPT-4) were resource-intensive, and near-real-time analysis required hardware scaling. However, cost-benefit analyses showed this was offset by reduced false positives and faster incident resolution. P. Gunda & T.R. Komati (2024) emphasised transformer flexibility and scalability, enabling deployment across sectors processing large volumes of complex textual data.

Secondly, questions arose concerning interpretability. Although GPT-4 could generally provide a concise commentary on why a particular email or log entry appeared suspicious, the decision-making mechanism remained challenging to explain in detail at the level of individual model layers. This aspect was also highlighted by F. Olaoye & A. Egon (2024), who proposed methods to enhance the transparency of artificial intelligence decision-making. During the research, a "general risk description" was employed, which nonetheless facilitated the work of SOC analysts by identifying atypical elements, though it did not always reveal the transformer's complex underlying logic.

In line with the recommendations of T. McIntosh *et al.* (2023), who utilised GPT-4 for formulating governance, risk management, and compliance policies, it was logical to apply LLMs not only for email detection but also for generating internal guidelines or "best practices" for responding to emerging threats. Within this study, the model also successfully drafted template service notifications when rapid employee alerts were required regarding new phishing campaigns. The examination of corporate system logs operating in near real-time confirmed the hypothesis regarding the advantage of LLMs in cases where attacks lacked classical signatures. For instance, attackers could fragment a malicious "chain" into a series of minor actions or periodically modify query parameters. GPT-4, owing to its in-depth semantic and contextual analysis, detected indicators of malicious activity that were difficult to formalise using SIEM correlation rules alone. A similar concept was observed in Z. Ding *et al.* (2023), where the hybridisation of BERT with Graph Convolutional Networks facilitated the detection of "hidden" patterns in code or metadata. In the conducted experiment, GPT-4 – without employing graph-based extensions – also demonstrated that contextual analysis could be key to filtering complex or obfuscated attack scenarios.

Considering approaches to prolonged attacks, LLMs such as GPT-4 also assisted in analysing compromise chains. L. Li & W. Chen (2024), for example, described how persistent threats could be better tracked when mechanisms for understanding interrelated events were in place. In this study, GPT-4, integrated with SIEM, effectively served as an additional layer that extracted potential anomalies from textual log descriptions and helped identify the "ground zero" of a compromise. This alignment with other studies reinforced the conclusion that LLMs significantly expanded the capabilities of traditional anomaly detection tools, particularly when discerning the holistic "narrative" of a threat.

Meanwhile, additional opportunities emerged in the field of prompt engineering. If "conversational" instructions were properly formulated, GPT-4 could provide in-depth message analysis or suggest initial steps for blocking malicious activity. A similar principle was employed by P. He *et al.* (2025) in multi-agent LLM-based systems, where "Red Team" attacks tested whether inter-agent communication channels could be corrupted. Although a multi-component architecture was not deployed, the practice of "red teaming" allowed for the detection of deliberate textual distortions, confirming the methodology's versatility.

R. Molleti *et al.* (2024) emphasised that multi-layered analysis involving LLM agents could substantially reduce response times to sophisticated cyberattacks by automating the processing of incoming data streams. With their assistance, atypical social engineering techniques or phishing variations were detected more

swiftly, as GPT-4 considered a broader context than mere alignment with a specific ATT&CK category. The presence of such "intelligent" real-time text analysis accelerated attack phase identification and helped prioritise incident handling.

However, when discussing risks, it is worth mentioning scenarios where adversaries could manipulate message content to "train" the model on false examples or poison input data. K. Alfarsi *et al*. (2024) addressed the topic of few-shot learning, where the model "learned" from a small number of samples, and highlighted the risk of rapid skewing if these samples were fabricated. During this study, the training dataset was periodically updated, but stringent quality checks were enforced on new phishing samples to prevent attackers from degrading system accuracy.

A separate area of interest was the use of LLMs for analysing external packages and dependencies when companies integrated third-party software into their systems. Analogous to W. Guo *et al*. (2024), who examined suspicious elements in npm or PyPI metadata, it was possible to process library descriptions, vulnerability reports, and even internal corporate documentation. Initial steps were taken in the test environment, and GPT-4 successfully flagged potential dangers in cases referencing "suspicious" repositories. The topic of extended data collection from dark web forums was discussed by K.S. Sangher *et al*. (2023), who developed a proactive cyber threat intelligence system and underscored the potential to detect mentions of novel attack schemes or planned data leaks. In this setup, GPT-4 received signals from both internal logs and specialised repositories dedicated to emerging evasion techniques.

The positive impact of LLMs also stemmed from reduced response times, as a greater number of "false" alerts were filtered out at the preliminary stage. For large organisations, this allowed SOC analysts to focus more quickly on serious incidents. Compared to older approaches relying solely on signatures or simple machine learning classifiers, GPT-4 demonstrated superior capability in distinguishing atypical vocabulary and word combinations. N.E. Chaabene *et al*. (2021), who researched anomaly detection in social networks, also demonstrated that multi-dimensional (including textual) analysis enabled earlier detection of suspicious activities, even if specific patterns had not yet been added to signature lists. Furthermore, LLMs were found to assist in formulating internal threat response policies. Finally, it is worth noting adjacent LLM use cases. J. Wang (2024) warned of the weaknesses in BERT-derived models and the risks of unauthorised access to their internal representations. Although GPT-4 had a different architecture, the risks of confidential parameter leakage and training data corruption remained relevant. This necessitated comprehensive safeguards: access controls, encrypted channels, network segmentation, and regular audits.

Thus, the obtained results confirmed the significant advantages of integrating LLMs into threat analysis and incident response systems: a detection accuracy of up to 0.97 for malicious emails, a substantial reduction in false positives, and accelerated response times. However, it remained critical to implement continuous retraining, protect the model from adversarial skewing, and monitor the quality of new samples. The synergy between GPT-4 and existing technologies (SIEM, antiviruses, multi-layered defence systems) amplified the effect, enabling timely mitigation of complex, previously undetected multi-stage attacks.

These findings correlated with the conclusions of Z. Ding *et al*. (2023) on deep contextual code analysis, L. Li & W. Chen (2024) on prolonged threat tracking, and K. Alfarsi *et al*. (2024) on few-shot approaches in malware detection. These works collectively affirmed the potential of transformer architectures in countering cyber threats while stressing that LLM deployment must be accompanied by systematic security measures. Ultimately, the study validated the relevance of LLMs in real-world corporate environments, demonstrated the feasibility of achieving high accuracy and rapid response times, and underscored the need for further refinement of methods to protect the models themselves from targeted attacks.

CONCLUSIONS

The findings of the study demonstrated that the integration of LLMs into cybersecurity systems significantly enhances the efficacy of real-time threat analysis. The primary objective of this work was to investigate the applicability of LLMs for the automated detection of cyberattacks based on the analysis of emails, network logs, and external data sources. The obtained results indicate that LLMs are capable of effectively recognising both known and novel threat variants, enabling deep contextual analysis of textual data.

A comparative analysis of traditional methods – signature-based approaches and classical machine learning algorithms (logistic regression, Support Vector Machine) – against LLM-based solutions revealed that models leveraging GPT-2, GPT-3, and particularly a customised version of GPT-4 exhibit superior classification accuracy and a lower false-positive rate. Due to their ability to analyse both structural and lexico-semantic features of messages, LLMs can detect even concealed patterns of phishing attacks and social engineering, substantially reducing the risk of overlooking critical incidents.

A key advantage of LLM deployment is their capacity for continuous fine-tuning in response to evolving adversarial tactics. Automated analysis, integrating data from corporate monitoring systems, email, and external resources (blogs, forums, the darknet), enables rapid response to emerging threats and minimises incident response times. This approach facilitates the establishment of a multi-layered defence framework, wherein

data from diverse sources are correlated to construct a comprehensive threat landscape.

However, alongside these benefits, the study also identified certain challenges. Among them is the necessity for persistent model updates and adaptations to prevent successful adversarial attacks aimed at deceiving LLMs through adversarial examples. Another critical aspect is ensuring sufficient computational resources for real-time model operation, which demands additional investment in supporting infrastructure. The results substantiate the viability of integrating LLMs into cybersecurity systems to enhance automation, reduce false positives, and accelerate cyber incident response processes.

The study established that traditional signature-based methods demonstrated a classification accuracy of 0.84 with an F1-score of 0.79, whereas classical machine learning algorithms – namely logistic regression and Support Vector Machine – yielded metrics of 0.88 (F1 = 0.83) and 0.9 (F1 = 0.85), respectively. GPT-2 and GPT-3 models, fine-tuned on corporate data, achieved accuracies of 0.93 (F1 = 0.9) and 0.95 (F1 = 0.92), while a specially adapted GPT-4 variant attained the highest performance – 0.97 accuracy with an F1-score of 0.95.

In network log analysis, a baseline SIEM-based detector processed data in 0.5 ms with a 15.2% false-positive rate and an 83% true-threat detection rate. The application of a Random Forest model and an XGBoost ensemble improved these metrics (1.7 ms/12.8%/88.5% and 2 ms/10.2%/90.1%, respectively). However, an integrated, fine-tuned GPT-4-based LLM achieved processing in 2.5 ms, reducing false positives to 6.5% while detecting genuine threats in 94.3% of cases. Future research should focus on strengthening LLM resilience against adversarial manipulation, refining adaptation mechanisms for sector-specific threats, and expanding toolkits for more flexible and granular model customisation in response to the evolving conditions of the modern cyber landscape.

ACKNOWLEDGEMENTS

None.

FUNDING

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Aghaei, E., Niu, X., Shadid, W., & Al-Shaer, E. (2022). SecureBERT: A domain-specific language model for cybersecurity. In F. Li, K. Liang, Z. Lin & S.K. Katsikas (Eds.), *Security and privacy in communication networks* (pp. 39-56). Cham: Springer. doi: 10.1007/978-3-031-25538-0_3.

[2] Alabi, M., & Ademola, F. (2024). *Adversarial robustness and defense mechanisms in machine learning*. Retrieved from https://www.researchgate.net/publication/383250866_Adversarial_Robustness_and_Defense_Mechanisms_in_Machine_Learning.

[3] Alfarsi, K., Rasheed, S., & Ahmad, I. (2024). Malware classification using few-shot learning approach. *Information*, 15(11), article number 722. doi: 10.3390/info15110722.

[4] Ambekar, N.G., & Thokchom, S. (2024). UL-VAE: An unsupervised learning approach for zero-day malware detection using variational autoencoder. In *Proceedings of the international conference on computational intelligence and network systems* (pp. 1-7). Dubai: IEEE. doi: 10.1109/CINS63881.2024.10864450.

[5] Avci, C., Tekinerdogan, B., & Catal, C. (2024). Design tactics for tailoring transformer architectures to cybersecurity challenges. *Cluster Computing*, 27, 9587-9613. doi: 10.1007/s10586-024-04355-0.

[6] Chaabene, N.E., Bouzeghoub, A., Guetari, R., & Ghezala, H.B. (2021). Deep learning methods for anomalies detection in social networks using multidimensional networks and multimodal data: A survey. *Multimedia Systems*, 28, 2133-2143. doi: 10.1007/s00530-020-00731-z.

[7] Cherqi, O., Moukafih, Y., Ghogho, M., & Benbrahim, H. (2023). Enhancing cyber threat identification in open-source intelligence feeds through an improved semi-supervised generative adversarial learning approach with contrastive learning. *IEEE Access*, 11, 84440-84452. doi: 10.1109/ACCESS.2023.3299604.

[8] Damola, P., & Miracle, A. (2024). LLM-based security automation: Revolutionizing threat detection and incident response. *Cybersecurity & Cybercrime*, 6(7), 9-15.

[9] Ding, Z., Xu, H., Guo, Y., Yan, L., Cui, L., & Hao, Z. (2023). Mal-Bert-GCN: Malware detection by combining Bert and GCN. In *Proceedings of the international conference on trust, security and privacy in computing and communications* (pp. 175-183). Wuhan: IEEE. doi: 10.1109/TrustCom56396.2022.00034.

[10] Gholami, Y. (2024). Large Language Models (LLMs) for cybersecurity: A systematic review. *World Journal of Advanced Engineering Technology and Sciences*, 13(1), 57-69. doi: 10.30574/wjaets.2024.13.1.0395.

[11] Gunda, P., & Komati, T.R. (2024). Integrating self-attention mechanisms for contextually relevant information in product management. *International Journal of Computational and Experimental Science and Engineering*, 10(4), 1361-1371. doi: 10.22399/ijcesen.651.

[12] Guo, W., Liu, C., Wang, L., Wu, J., Xu, Z., Huang, C., Fang, Y., & Liu, Y. (2024). PackageIntel: Leveraging large language models for automated intelligence extraction in package ecosystems. *ArXiv*. doi: 10.48550/arXiv.2409.15049.

[13] He, P., Lin, Y., Dong, S., Xu, H., Xing, Y., & Liu, H. (2025). Red-teaming LLM multi-agent systems via communication attacks. *ArXiv*. doi: 10.48550/arXiv.2502.14847.
[14] Jamal, S., & Wimmer, H. (2023). An improved transformer-based model for detecting phishing, spam, and ham: A large language model approach. *Research Square*. doi: 10.21203/rs.3.rs-3608294/v1.
[15] Joshua, E., Do, J., & Patel, R. (2025). AI-driven threat intelligence system (AIDTIS): Leveraging large language models for automated threat research and detection development. *International Journal of Science and Research Archive*, 14(3), 270-285. doi: 10.30574/ijsra.2025.14.3.0339.
[16] Kuppam, M. (2024). Turning data telemetry into insights using application performance monitoring solutions. *FeedForward*, 3(2), 23-32.
[17] Li, L., & Chen, W. (2024). ConGraph: Advanced persistent threat detection method based on provenance graph combined with process context in cyber-physical system environment. *Electronics*, 13(5), article number 945. doi: 10.3390/electronics13050945.
[18] Liu, S., Chen, J., Ruan, S., Su, H., & Yin, Z. (2024). Exploring the robustness of decision-level through adversarial attacks on LLM-based embodied models. In J. Cai, M. Kankanhalli, B. Prabhakaran & S. Boll (Eds.), *Proceedings of the 32nd ACM international conference on multimedia* (pp. 8120-8128). New York: Association for Computing Machinery. doi: 10.1145/3664647.3680616
[19] McIntosh, T., Liu, T., Susnjak, T., Alavizadeh, H., Ng, A., Nowrozy, R., & Watters, P. (2023). Harnessing GPT-4 for generation of cybersecurity GRC policies: A focus on ransomware attack mitigation. *Computers & Security*, 134, article number 103424. doi: 10.1016/j.cose.2023.103424.
[20] Mokin, V.B., & Pradivliannyi, M.G. (2024). *Machine learning, intelligent data analysis and artificial intelligence of things*. Vinnytsia: Vinnytsia National Technical University.
[21] Molleti, R., Goje, V., Luthra, P., & Raghavan, P. (2024). Automated threat detection and response using LLM agents. *World Journal of Advanced Research and Reviews*, 24(2), 79-90. doi: 10.30574/wjarr.2024.24.2.3329.
[22] Ogundairo, O., & Broklyn, P. (2024). *Natural language processing for cybersecurity incident analysis*. Retrieved from https://easychair.org/publications/preprint/zXlC.
[23] Olaoye, F., & Egon, A. (2024). *Explainable AI for security decision making*. Retrieved from https://easychair.org/publications/preprint/TtJd.
[24] Parmar, S., & Patel, H. (2024). *Prompt engineering for large language model*. doi: 10.13140/RG.2.2.11549.93923.
[25] Partyka, A., Harasymchuk, O., Nyemkova, E., Sovyn, Y., & Dudykevych, V. (2024). Development of a method for investigating cybercrimes by type of ransomware using artificial intelligence models in the critical infrastructure Information security management system. *Social Development and Security*, 14(2), 52-63. doi: 10.33445/sds.2024.14.2.6.
[26] Podvysotska, O.P., & Nosok, S.O. (2024). Applying machine learning algorithms to detect network traffic anomalies. In *All-Ukrainian scientific and practical conference of students, postgraduates and young scientists* (pp. 288-290). Kyiv: National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".
[27] Ranade, P., Piplai, A., Mittal, S., & Joshi, A. (2021). Generating fake cyber threat intelligence using transformer-based models. In *Proceedings of the international joint conference on neural networks* (pp. 1-9). Shenzhen: IEEE. doi: 10.1109/IJCNN52387.2021.9534192.
[28] Rybalchenko, L., & Ohrimenco, S. (2024). The impact of cybersecurity and crime on national security. *Philosophy, Economics and Law Review*, 4(2), 62-72. doi: 10.31733/2786-491X-2024-2-62.
[29] Sangher, K.S., Singh, A., Pandey, H.M., & Kumar, V. (2023). Towards safe cyber practices: Developing a proactive cyber-threat intelligence system for dark web forum content by identifying cybercrimes. *Information*, 14(6), article number 349. doi: 10.3390/info14060349.
[30] Santoso, J.T., Hartono, B., Silalahi, F.D., & Muthohir, M. (2024). Transformers in cybersecurity: Advancing threat detection and response through machine learning architectures. *Journal of Technology Informatics and Engineering*, 3(3), 382-396. doi: 10.51903/jtie.v3i3.211.
[31] Singh, K., Grover, S.S., & Kumar, R.K. (2022). Cyber security vulnerability detection using natural language processing. In *World AI IoT congress* (pp. 174-178). Seattle: IEEE. doi: 10.1109/AIIoT54504.2022.9817336.
[32] Taghavi, S.M., & Feyzi, F. (2024). Using large language models to better detect and handle software vulnerabilities and cyber security threats. *Research Square*. doi: 10.21203/rs.3.rs-4387414/v1.
[33] Wang, J. (2024). *Exploring vulnerabilities in BERT models*. doi: 10.20944/preprints202407.0204.v2.

Використання великих мовних моделей для автоматизованого аналізу кіберзагроз у режимі реального часу

Денис Ковальчук

Аспірант
Міжнародний гуманітарний університет
65009, вул. Фонтанська дорога, 33, м. Одеса, Україна
https://orcid.org/0009-0003-2302-8698

Анотація. У сучасному ландшафті кібербезпеки, де стрімке зростання кількості та складності загроз позначилося на ефективності традиційних методів виявлення, базованих на правилах та сигнатурах, було встановлено нагальну потребу у впровадженні автоматизованих систем аналізу кіберзагроз із застосуванням великих мовних моделей. Метою роботи було дослідити можливості великих мовних моделей для автоматизованого аналізу кіберзагроз, оцінки ризиків та підвищення ефективності реагування на інциденти в корпоративному середовищі. Для досягнення поставленої мети використовувалися методи машинного навчання та обробки природної мови, зокрема адаптація великих мовних моделей для класифікації загроз, оцінки рівня ризику та виявлення аномалій. Було розроблено систему аналізу вхідних та вихідних повідомлень електронної пошти, яка під час тестування автоматично ідентифікувала фішингові атаки та техніки соціальної інженерії, присвоювала повідомленням ризиковий бал і при перевищенні порогового значення (наприклад, 0.8) направляла їх у карантин для подальшої перевірки. Система аналізувала датасет із 100 000 електронних листів, з яких 70 % становили безпечні повідомлення, а 30 % – фішингові атаки. Крім того, здійснювався аналіз потоків даних із корпоративних логів та зовнішніх джерел, що дозволяло виявити потенційні кіберінциденти з точністю до 94 % та знизити відсоток хибнопозитивних спрацьовувань до 6,5 %. Отримані результати підтвердили ефективність застосування великих мовних моделей, які забезпечували точність класифікації загроз до 97 % із F1-мірою до 95 % і скорочували час реагування на інциденти на 30-40 %. Отримані результати можуть бути використані іншими дослідниками для покращення методик виявлення фішингових атак, зниження кількості помилкових спрацьовувань у корпоративних системах безпеки та інтеграції моделей машинного навчання з різними джерелами даних, включаючи SIEM-системи та зовнішні ресурси з кібербезпеки

Ключові слова: обробка природної мови; машинне навчання для безпеки; виявлення фішингових атак; виявлення аномалій; глибоке навчання у кібербезпеці; нейронні мережі для безпеки; розвідка кіберзагроз