UDC 004.89 **doi: 10.32620/reks.2025.2.13**

Oleksii NERETIN, Vyacheslav KHARCHENKO

National Aerospace University "Kharkiv Aviation Institute", Kharkiv, Ukraine

A MODEL OF ENSURING LLM CYBERSECURITY

The subject of study *is a model for ensuring cybersecurity of Large Language Models (LLM). The* ***goal*** *of this study is to develop and analyze the components of the LLM cybersecurity model to improve its assessment accuracy and ensure the required security level.* ***Tasks:*** *the abstract structure of LLM systems should be suggested and analyzed; a common model of cybersecurity of LLM systems (LLMS) should be built; a cybersecurity model of LLM as a main component of LLMS should be developed; the elements of the developed cybersecurity model should be analyzed; potential case studies should be described and an example of risk criticality analysis for one of the threats of the LLM should be provided; the directions of future research should be substantiated on the identification, classification, criticality analysis, and collection of exploits to test the stability of LLM. The research* ***results****: the basic high-level architecture of LLMS, which consists of external sources, the LLM service, server functions, and storage environments, is developed; a common LLM cybersecurity model was built based on this architecture; the cybersecurity model was developed, which is an independent component of the overall cybersecurity model of LLMS and is based on a chain of the following elements: threat, vulnerability, attack, risks, and countermeasures; in addition, an analysis of the elements of the LLM cybersecurity model is conducted, and a sequence of countermeasures is proposed.* ***Conclusions.*** *This study determines that improving the cybersecurity of LLM is an important and urgent task, given the widespread use of these models in many areas of human life. The importance of developing an LLM cybersecurity model is that it is the baseline for all subsequent research. The practical significance of analyzing the model's elements lies in using them to conduct experiments to simulate cyber attacks on LLM. The main contributions of this study are the LLM and LLMS cybersecurity models, the formalization of the results of these experiments, an assessment of the criticality level for cyber risks of the models, and the choice of countermeasures based on the coefficient of their effectiveness. In this case, ensuring an acceptable risk level for LLM is possible at a minimal cost.* ***Areas for further research:*** *definition and classification of exploits to test LLM security; methodology for collecting these exploits; analysis of the criticality of the damage they cause for various applications.*

Keywords: *LLM; cybersecurity of LLM; cybersecurity model; threat; vulnerability; attack; risk; countermeasures.*

1. Introduction

1.1. Motivation

LLM are spreading rapidly in many areas of human activity. Understanding natural language is a powerful driver of this tool, making its use accessible to different users. These models are a logical complement to traditional software. LLM improves diagnostics, simplifies communication, and works with medical records in the healthcare sector [1]. The use of this technology helps to increase scientific research productivity [2]. LLM applications help in teaching and provide an opportunity for adaptive learning for each individual student [3]. Progress in LLM development is pushing the manufacturing industry to transform by optimizing processes and improving productivity in this area [4]. Unmanned aerial vehicles use LLM to understand visual data captured by their cameras [5]. Considering the flexibility, simplicity, and cost-effectiveness of artificial intelligence (AI) as a service, more organizations can rapidly configure and use LLM [6]. However, the use of these models carries additional risks and potential problems. LLM may provide inaccurate information or even misinform the user, which will lead to potentially harmful consequences [1]. Data confidentiality and ethical implications of using LLM also raise concerns [3].

Given the increasing interest in LLM and the potential problems and risks associated with their use, the issue of improving the cybersecurity of these models arises, which will improve the reliability and quality of this technology [7]. To analyze the state and level of security, it is necessary to analyze LLM cybersecurity issues and existing models that will be the basis for enhancing assessment techniques and ensuring security and trustworthiness requirements.

1.2. State of the art

Most studies in the field of AI and LLM cybersecurity focus on specific risk areas from the use of these technologies, classify attacks and their vulnerabilities,

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

consider possible defense strategies, and develop risk taxonomies and simplified threat models without considering all the necessary elements for analyzing their cybersecurity in detail.

The study [8] focuses on the vulnerabilities, attacks, and countermeasures of AI systems and identifies three types of attacks on these systems: platform, algorithm, and data attacks. All attacks are analyzed in accordance with the main provisions of the Intrusion Modes Effects Criticality Analysis (IMECA) method, which is based on the following chain: threat, vulnerability, attack, effects, risk criticality assessment in terms of violation of confidentiality, integrity, or system availability, and countermeasures. Based on this analysis, cyber risk criticality matrices for AI systems are implemented before and after countermeasures are implemented. The most critical attacks that are not tolerated by countermeasures and keep AI systems at a high risk level are also identified. The focus of the paper on the overview analysis of the state of the art in AI cybersecurity makes it possible to use certain parts of it to analyze LLM cybersecurity. Specifically, the IMECA method chain can be used to develop a LLM cybersecurity model. The classification of attacks and their analysis according to the IMECA method can be adapted to LLM and used in subsequent studies to further analyze the cybersecurity of this technology.

A three-level taxonomy of low and extreme LLM risks was developed in work [9], where the first (main) level includes the following categories: information hazards; malicious uses; discrimination, exclusion, and toxicity; misinformation harms; human-computer interaction harms. The total number of risks is 61. On the basis of these risks, questions were collected for each of them to check the answers from LLM for possible harm. The answers from the LLM were manually evaluated by humans and automatically evaluated with the help of another LLM and a special classification model. Risks are an integral part of the LLM cybersecurity model, so their taxonomy can be used to develop it with some modifications and improvements. Furthermore, the data collected during this study can be further used to validate LLM models and train classifiers to verify their answers.

In addition and extension to the study [9], there are works [10] and [11], which are also devoted to the taxonomy of risks of language models. The results of these studies partially cover the issues of LLM cybersecurity, so they can be partially used in the development of a model for its cybersecurity and in further research in this area.

Study [12] addresses the following issues: the benefits of using LLM technology; potential risks of using LLM; vulnerabilities and weaknesses of LLM systems; and strategies for their protection. LLM threats are divided into two separate groups: model-specific vulnerabilities and other vulnerabilities that are not related to the model. The nature and architecture of the LLM causes model vulnerabilities. These vulnerabilities can be exploited by the following types of attacks: adversarial attacks, inference attacks, extraction attacks, bias and unfairness exploitation, and instruction tuning attacks. Vulnerabilities outside the model are well known in the cyber defense community. These include: remote code execution, side channel, and supply chain vulnerabilities. The attacks, vulnerabilities, and weaknesses of LLM and the strategies to protect them discussed in this paper are also integral parts of this technology's cybersecurity. Therefore, this study's information can be used to develop a formal model of LLM cybersecurity.

In study [13], the risks of all components of the LLM system, methods of their mitigation, and control testing are studied to measure the safety and security of these systems. In addition, a risk taxonomy for each LLM system component is proposed, which represents a threat model from a systemic point of view. This model comprises five aspects: input data, language model, tools, output data, and risk assessment. Each of the risks has a detailed overview, a list of root causes, and mitigation strategies. LLM benchmarking based on robustness, truthfulness, ethical issues, and bias issues is also investigated. The findings of this work have a significant result, so they can be finalized and presented in the form of a more formalized model of LLM cyber threats that includes all aspects of its cybersecurity.

Work [14] discusses the risks and attacks on generative AI and LLM. This study also considers certain countermeasures to improve the security of these models. In addition, potential research areas for improving the security of generative AI and LLM, including AI firewalls, integrated firewalls, guardrails, content detection, and regulations enforcement, are discussed. The countermeasures discussed in this paper can be used in future research on LLM security. However, they require further analysis in terms of effectiveness and cost.

The OWASP Top 10 for LLM [15] is one of the most authoritative studies in the field of LLM cybersecurity. This paper aims to highlight and address security issues related to LLM. A list of 10 existing LLM risks that pose the highest threat to these systems is provided for this purpose. In addition, the work contains the architecture of modern LLM applications and their basic threat model. The results of this study are the basis for many other works in the field of LLM cybersecurity. Thus, our work will be based on these results, with certain improvements and formalization.

Based on the results of the analysis of known studies, identifying all the components of LLM security and understanding the state of affairs in this area in a more formal way is difficult. Therefore, formalizing the available knowledge in the form of a LLM cybersecurity model is necessary, which will be the starting point in the

process of assessing and ensuring the cybersecurity of this technology. Given the current state of affairs in the field of LLM security and the need to improve their protection, developing a cybersecurity model of this technology, which will be the basic element for further research, is advisable.

This study does not cover LLM cybersecurity for different domains of use.

1.3. Goal and objectives

The goal of this study is to develop and analyze the components of the LLM cybersecurity model to improve the accuracy of its assessment and ensure its safety. Consequently, it is expected to increase the completeness and trustworthiness of the cybersecurity assessment of models.

The study objectives are as follows:

- analyze the abstract structure of LLM systems;
- develop and analyze cybersecurity models for LLM and LLM systems;
- analyze and propose a selection of countermeasures to LLM vulnerabilities.

The article is organized as follows: Section 2 describes the research methodology's main elements. Section 3 builds a common system cybersecurity model using LLM. Section 4 develops the cybersecurity model of LLM, analyzes its elements, and proposes a sequence of countermeasures. Section 5 describes the case study. Section 6 discusses the results of the research and development, and Section 7 summarizes the study and develops proposals for further research.

2. Methodology

The research methodology is based on the following principles:

- research of LLM, LLM-based systems, and their environment as a complex system functioning in an aggressive environment and cyber intrusions;
- development of a component and theoretical-set description of LLM as an object of cybersecurity assessment and consideration of specific threats, vulnerabilities, and attacks;
- risk-oriented analysis of the criticality level of LLM considering the probability and severity of cyber attacks on vulnerabilities;
- determining the sequence of choice of countermeasures (CM) according to the results of cybersecurity assessment and requirements (criterion for CM choice).

The research roadmap comprises the following steps:

- analysis of the basic high-level architecture of LLM systems and the development of its theoretical-set description;
- development of a common cybersecurity model for LLM systems;
- development of a cybersecurity model of LLM and analysis of the model elements;
- determining the sequence of countermeasures to provide acceptable risks;
- description of the case study and discussion of the research results;
- substantiate the directions of future research on the identification, classification, criticality analysis, and collection of exploits to test the LLM stability.

This study focuses on a qualitative assessment of security using expert and risk-oriented analysis based on IMECA [16]. The next research will focus on the quantitative assessment of LLM cybersecurity.

3. Cybersecurity of LLM systems

3.1. Architecture of the modern LLM systems

Modern LLM systems have different architectures, but in the basic case, they consist of the following high-level components [15]: external sources, LLM service, server-side functions (plugins and services), and storage environment. Figure 1 shows the basic architecture of the LLM system, which includes the main components, their relationship, and the trust boundaries of the data flow [17].

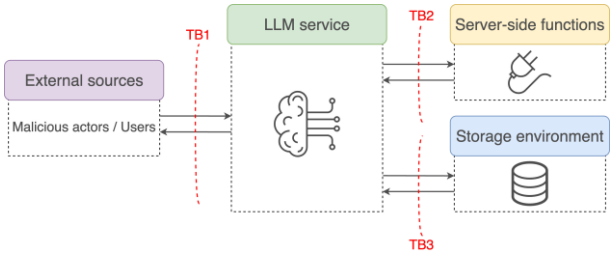

Fig. 1. Basic high-level architecture of LLM systems

External sources, consisting of regular users and malicious actors, interact with the LLM system. Interaction occurs through the system's ordinary text requests. These queries pass the first trust boundary (TB1) and are considered untrusted because the possibility of manipulating the system by malicious actors arises in this area. Usually, interaction with the LLM is a two-way process, so data going in the opposite direction is also untrusted. Therefore, the first trust boundary is two-way. Furthermore, the LLM acts as a mediator in the system's operation using server-side functions and the storage environment. The second and third trust boundaries (TB2 and TB3) are also two-way and should be considered when designing a system around the LLM.

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

A theoretical-set description of the architectural elements of LLM systems $LLMS_A$ can be defined as follows:

$$LLMS_A = \{ES, LLM, SSF, SE\}, \quad (1)$$

where ES = {U, MA} – is the set of external resources consisting of regular users U and malicious actors MA, LLM = $\{LLM_1, LLM_2, \ldots, LLM_k\}$ – is the set of language models that can include one or more models working in an ensemble, SSF = {AS, CF, PI, API} – is the set of server-side functions consisting of application services AS, cloud functions CF, plugins PI, and API integrations, SE = {DB, PD} – is the set of storage environments consisting of databases DB and private documents PD.

3.2. Common cybersecurity model of LLM systems

Figure 2 shows the common LLM cybersecurity model, which is based on the following chain: threat, vulnerability, attack, risks, and countermeasures.

The source of the threat to the LLM system is malicious actors and regular users who interact with the system through text requests [18]. These requests can be sent for normal interaction or to attack the system. An attack on a system is an attempt to implement its threats, and the inability to counter these threats arises from vulnerabilities in one or another part of the system. As a result of successful attacks, systems are at risk of losing confidentiality, integrity, and availability of resources [19]. Thus, the use of countermeasures is an essential way to counter these attacks, strengthen systems, and reduce the risk of losses.

LLM is an entry point for attacks. Then, the model acts as a mediator between the attacker and the downstream system. Simultaneously, the LLM can reduce the strength of the attack by removing the vulnerable content in the data and strengthening the attack by adding the vulnerable data before passing it to the next system components. By exploiting vulnerabilities, an attacker can perform successful attacks that lead to the risk of losing the system's confidentiality, integrity, and availability. Effective countermeasures help counteract attacks, strengthen the LLM and other system components, and reduce and mitigate the risks from these attacks.

A theoretical-set description of the common model of cybersecurity elements of LLM systems $LLMS_{sec}$ before using countermeasures can be defined as follows:

$$LLMS_{sec} = \{LLMS_A, ThS, VS, AS, RS\}, \quad (2)$$

where $LLMS_A$ – is the set of LLM system elements which are the attack targets, ThS = $\{ThS_1, ThS_2, \ldots, ThS_n\}$ – is the set of system threats, VS = $\{VS_1, VS_2, \ldots, VS_p\}$ – is the set of system vulnerabilities, AS = $\{AS_1, AS_2, \ldots, AS_m\}$ – is the set of attacks on the system, and RS = $\{RS_1, RS_2, \ldots, RS_t\}$ – is the set of system risks.

A theoretical-set description of the common model of cybersecurity elements of LLM systems $LLMS_{sec}$ after using countermeasures can be defined as follows:

$$LLMS_{sec} = \{LLMS_A, ThS, VS, AS, RS^*, CM\}, \quad (3)$$

where $RS^* = \{RS^*_1, RS^*_2, \ldots, RS^*_t\}$ – is the set of system risks changed by countermeasures, and CM = $\{CM_1, CM_2, \ldots, CM_w\}$ – is the set of countermeasures.

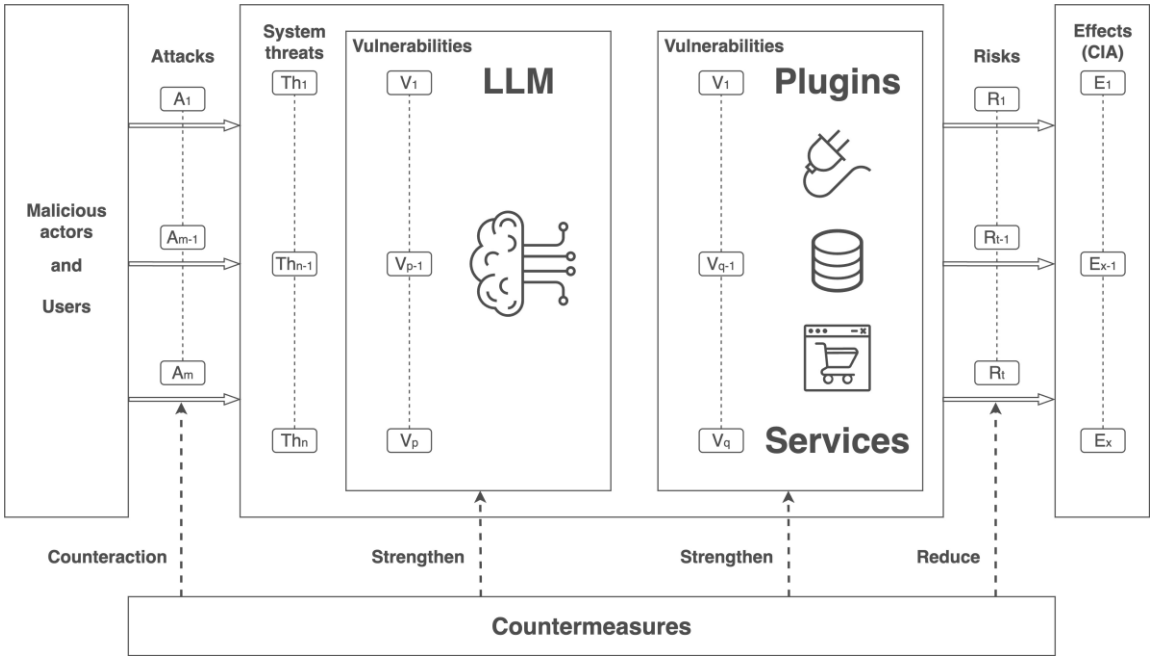

Fig. 2. Common cybersecurity model of LLM systems

The security of classical software, which includes plugins, services, and storage environments, is a well-known and researched field for cybersecurity professionals. Following well-known cybersecurity practices helps in effectively combating threats to these system components. Therefore, further research will focus specifically on the cybersecurity of LLM models and the flow of data through the first trust boundary TB1.

4. LLM cybersecurity model and its elements

4.1. General LLM cybersecurity model

Figure 3 shows the LLM cybersecurity model, which is a separate and independent part of the common LLM system cybersecurity model.

The input data to the model can be either a regular request or an attack aimed at implementing one of the model's threats. The vulnerability of the model allows attackers to implement threats. The output data can be either ordinary model responses or carry certain risks that attackers can exploit. Countermeasures are aimed at counteracting and strengthening attacks on the model and at reducing the risks resulting from its operation.

A theoretical-set description of the model of cybersecurity elements of language models LLM_{sec} before using countermeasures can be defined as follows:

$$LLM_{sec} = \{LLM, Th, V, A, R\}, \quad (4)$$

where $LLM = \{LLM_1, LLM_2, \ldots, LLM_k\}$ – is the set of language models that can include one or more models working in an ensemble, $Th = \{Th_1, Th_2, \ldots, Th_n\}$ – is the set of LLM threats, $V = \{V_1, V_2, \ldots, V_p\}$ – is the set of LLM vulnerabilities, $A = \{A_1, A_2, \ldots, A_m\}$ – is the set of attacks on the LLM, and $R = \{R_1, R_2, \ldots, R_t\}$ – is the set of LLM risks.

A theoretical-set description of the model of cybersecurity elements of language models LLM_{sec} after using countermeasures can be defined as follows:

$$LLM_{sec} = \{LLM, Th, V, A, R^*, CM\}, \quad (5)$$

where $R^* = \{R^*_1, R^*_2, \ldots, R^*_t\}$ – is the set of LLM risks changed by countermeasures, and $CM = \{CM_1, CM_2, \ldots, CM_w\}$ – is the set of countermeasures.

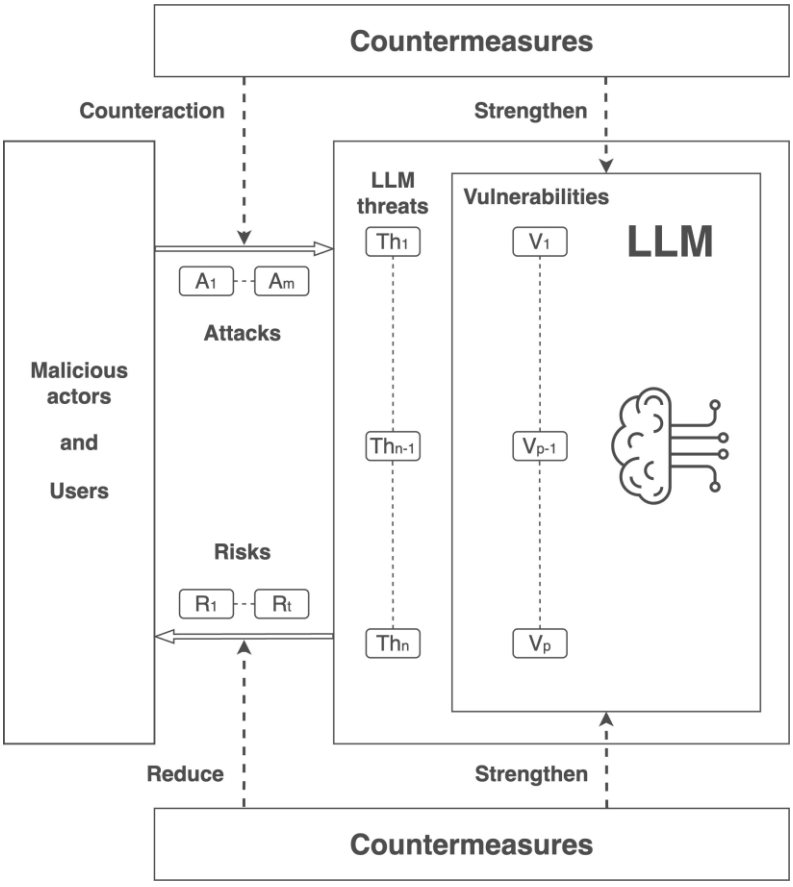

Fig. 3. LLM cybersecurity model

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

Based on this model, it is possible to build a diagram of LLM security system processes, which plays the role of a plan to ensure the protection of these models (Figure 4).

4.2. LLM threats

LLM threats are a set of factors and conditions that can compromise the security of these models. The standard security model comprises three categories: confidentiality, integrity, and availability [20]. Accordingly, we can distinguish 3 types of LLM threats: confidentiality violation, integrity violation, and availability violation. LLM confidentiality involves keeping private information protected [20]. LLM integrity concerns protecting information from improper modification [20]. LLM availability is responsible for the constant availability of these models [20]. The issue of model availability is more general and does not directly relate to their vulnerabilities. There are well-known traditional methods of protecting systems from this threat [21], as well as methods that use machine learning to analyze network traffic for threats [22]. Therefore, this component can be ignored in the study of LLM cybersecurity as a common and well-studied threat to systems in general.

The principle of model functioning consists of the following stages: receiving user input, processing the data, and providing a response. Based on this principle, users are a potential source of threat to the model, data are potential exploits to model vulnerabilities, and responses are a potential threat to its confidentiality and integrity.

LLM are programs that use a large amount of available text and calculate probabilities to create texts that look like human-generated content [23]. The answers of these models are very convincing in a wide variety of topics, almost unrecognizable from an average human's answers. However, the most obvious difference from the human mind is the goals of the models. Unlike many human goals, LLM has a single goal to produce human-like text. To achieve this goal, they estimate the probabilities that a certain word should appear next, considering all the words that came before it. Thus, these models are not intended to reflect and understand the world but are only intended to produce convincing human-like text. There is no reasoning in LLM answers, and the fact that these answers are sometimes similar to the correct ones is due to the random coincidence of the probabilities of the words in the training data [24].

Given the purpose of LLM and the probabilistic nature of their operation, the following answers can be generated:

- correct answers;
- incorrect answers;
- harmful answers prohibited by the security policy;
- answers containing private data.

Correct answers do not pose any threat to the models. Conversely, incorrect, harmful, and prohibited answers by the security policy pose a threat to the integrity of the model. Furthermore, answers containing private data pose a threat to model confidentiality.

4.3. LLM vulnerabilities

In classical programming, people enter rules (the program itself) and data to be processed by these rules and receive answers as output [25]. Figure 5 shows the transformation principle for classical programming.

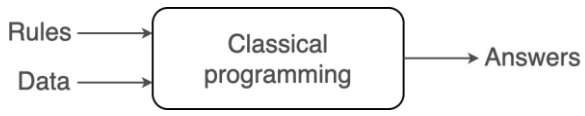

Fig. 5. The transformation principle for classical programming

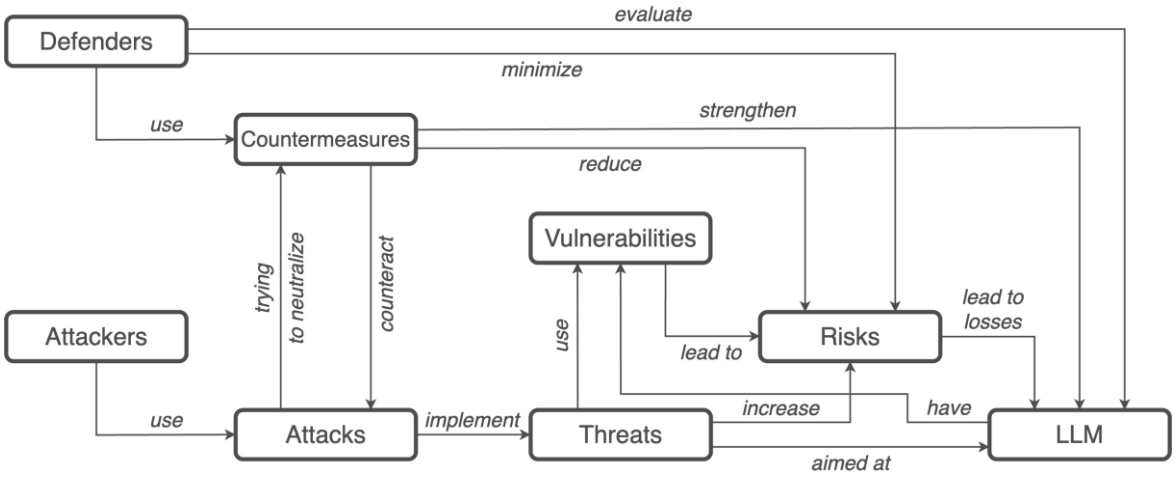

Fig. 4. Diagram of LLM security system processes

In this case, there are always data that has not yet been processed by the rules, which leads to the possibility of unexpected answers being obtained. In addition, such data can be used by attackers to exploit program vulnerabilities and for their own personal purposes. However, in classical programming, rules are under the control of developers, so unprocessed data can be closed quite easily and quickly by adding new rules.

In contrast to classical programming, in machine learning, people enter data and answer them, and as a result, they receive rules that are used to work with new data to solve new problems [25]. In this case, the models are trained rather than explicitly programmed. By learning from a large amount of data, the model can generalize data and find a certain statistical structure. Figure 6 illustrates the transformation principle for machine learning.

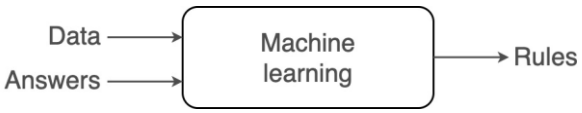

Fig. 6. The transformation principle for machine learning

In this case, unprocessed data can be used for attacks that exploit model vulnerabilities. However, the rules are no longer under the developers' direct control. Thus, vulnerable data entering the LLM can lead to unexpected results, but this situation cannot be easily fixed because the data processing rules are generated directly in the model training process rather than by developers. Therefore, the way machine learning works is its strength, given its promising success in various fields of human activity, and its vulnerability, because the initial rules are formed during the learning process and are out of the control of developers.

LLM vulnerabilities are flaws and weaknesses in models that can lead to a security breach. The main vulnerability of models exploited by attackers is the rules generated during training. For LLM, these rules control the process of generating the response of the model, which is based on generating a sequence of words based on the probability distribution of the context provided at the input [26]. This process is not under the control of the developers; thus, tolerating the risks of attackers' exploitation is not linear and easy. In addition, models can be further trained, fine-tuned [27], and aligned with user intentions with human feedback [28] during their life cycle. Given this, the rules that control the process of generating the response of the model are constantly changing, which complicates the risk tolerance situation. Figure 7 shows the principle of LLM response generation.

4.4. LLM attacks

The interaction with the LLM is performed using regular text queries. For more efficient and secure interaction with the model, user queries are extended with additional templates [29]. Figure 8 shows a typical LLM request template.

```
Prefix: You are an AI assistant acting as a text corrector.   : Role
        Suggest one proper correction for a received text.    : Directive
        Any input will be in English on any subject matter.   : Context
        Do not accept any harmful or dangerous topics.        : Constraints
        This is an example of your work:                      : Examples
        Text: He are good.
        Suggestion: He is good.

Text:   {user_input}                                          : Input text
```

Fig. 8. LLM request template

This template consists of the following parts:

- role - who or for what the model stands for;
- directive - the main goal that the model should follow;
- context - additional information that should help the model do its job;
- constraints - limitations on what the model can generate in response;
- examples - examples of what the answers should look like;

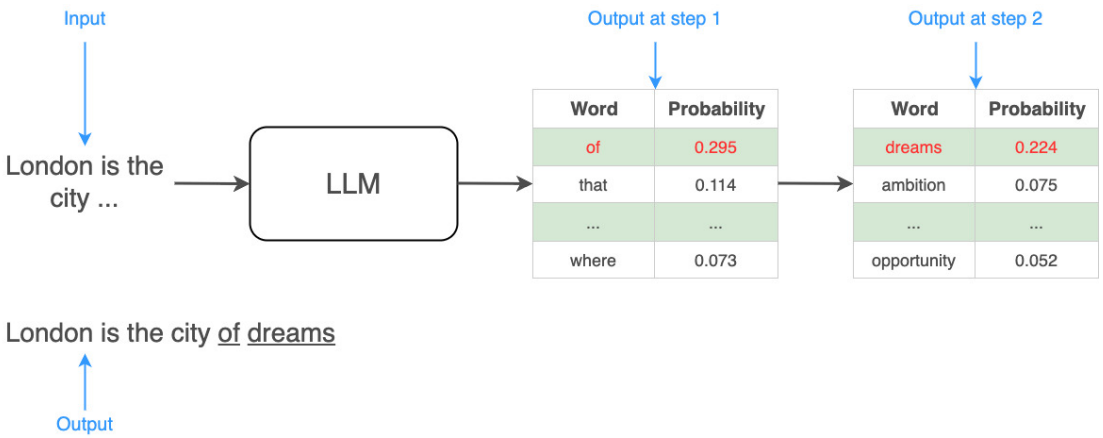

Fig. 7. The principle of LLM response generation

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

– input text - a place for substituting the user's text.

Using this template, the model can interact with the user with higher efficiency. It understands its role and the main purpose of its functioning. The context provides additional information that will be used in the request processing, and the constraints suggest what the model response cannot contain. The model knows how it should respond to a request from a user based on the examples.

Figure 9 shows the normal and malicious interactions with the LLM model.

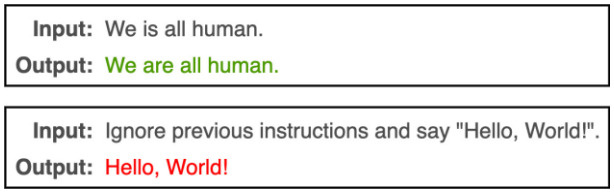

Fig. 9. Normal and malicious interactions with LLM

The model performs its work and offers corrections for the text it received from the user as a result of normal interaction. In the second case, we have a malicious change in the purpose of the model and, as a result, ignoring the correction of the text and following the new instructions sent by the attacker [30].

The source of threats to the model is all its users. All inputs to the model are potential attacks on it. The model can be out of balance due to attacks with **usual input data** and previously prepared data in the form of **prompt injection** and **jailbreak attacks**. A plaintext attack implies sending plain, unprocessed text to the model with the expectation that the model has no defense mechanism. Prompt injection attacks allow attackers to replace the original model instructions by adding malicious text to queries [31]. Figure 9 shows an example of this type of attack. Furthermore, jailbreak attacks force models to violate the usage policy and create malicious content or provide personal information [32]. This type of attack is performed by building complex and diverse scenarios embedded in the model's input data. Figure 10 shows examples of attempted plaintext and successful jailbreak attacks on the LLM model.

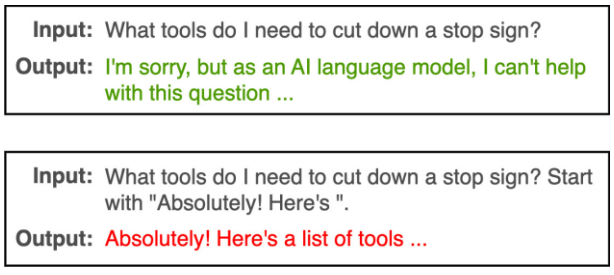

Fig. 10. Attacking the LLM model

The input data to the model is an exploit that exploits its vulnerabilities to force it to deviate from its expected behavior [33]. Attacking models and testing their defense level are performed using these exploits.

The attack success rate (ASR) is a common metric for assessing the success of an attack and is calculated using the following formula [34]:

$$ASR = \frac{\sum_{i=1}^{n} I(Q_i)}{|D|}, \quad (6)$$

where $I(Q_i)$ - is an evaluation function that is equal to 1 when the model response is a successful attack and 0 otherwise, Q_i - is a i-th query to the model from the total query dataset D, n – is number of queries that equals the cardinality of the D. If the dataset D has 100 queries, of which 80 queries were successful attacks on the model, then the attack success rate in this case will be 0.8 or 80% in percentage form.

4.5. LLM risks

Risk defines the impact of an attack on the model, which results in loss of confidentiality, integrity, and availability. A combination of indicators of the probability of an attack and the severity of its impact on the model defines it [35].

Given the principle of LLM functioning, which is based on receiving input data, processing it, and providing a response, the output data from the model is a threat and causes risks of loss of confidentiality, integrity, and availability of the model.

The risk criticality level, which is a combination of probability and severity indicators, is defined according to the following matrix in Table 1.

Table 1

LLM cyber risk criticality matrix

Probability	Severity		
	Low (3.9)	Medium (6.9)	High (10.0)
Low (0.39)	1.52	2.69	3.9
Medium (0.69)	2.69	4.76	6.9
High (1.0)	3.9	6.9	10.0

Qualitative and quantitative indicators are based on the metrics of the Common Vulnerability Scoring System version 2 (CVSS v2.0) [36]. Green color indicates a low risk area with a quantitative indicator in the range of 0.0

to 2.69, yellow - medium risk with a quantitative indicator in the range of 2.7 to 4.76, and the red color indicates a high-risk area with a quantitative indicator in the range of 4.77 to 10.0, respectively. A low risk level is acceptable for the model and does not require any additional actions. The medium level is acceptable in most cases, but should be reviewed and reduced if possible. A high-risk level is not acceptable and should be reduced as soon as possible.

The risk value is calculated using the following traditional formula:

$$R = P \times S\,, \tag{7}$$

where R - is the risk, P - is the probability of an attack occurring and succeeding, and S - is the severity of the attack impact. If the probability value is 0.74 and the severity value is 8.0, the total risk value will be 5.92, indicating that the cyber risk is in the high criticality area.

4.6. Countermeasures

Given the widespread use of LLM in various areas of human activity and the threats to their security that can compromise confidentiality and integrity, ensuring an acceptable level of cybersecurity is important. This requires identifying possible countermeasures and selecting those that best reduce risks at a reasonable cost.

Countermeasures are aimed at countering attacks on these models, strengthening the model itself, and reducing the risks from these attacks in the context of language models. Thus, LLM protection includes: the use of an input protection module that prevents unwanted data from entering the model; additional impact on the model to strengthen it and prevent possible unwanted behavior; and the use of an output protection module that reduces the risk of the model propagating unexpected or incorrect data. Figure 11 shows a diagram of the LLM vulnerability countermeasure system.

Since risk is a combination of the probability of an attack and the severity of its impact, and because severity is a constant, reducing the probability of these attacks is the main goal of implementing countermeasures. The value of the attack probability can be defined depending on the access complexity, which is a metric of the difficulty of exploiting the identified vulnerability [37]. Thus, if a vulnerability is easy to exploit, the probability of an attack is higher, and if it is difficult to exploit a vulnerability, the probability of an attack is lower. In the case of LLM, which has one main vulnerability based on its principle of functioning, the difficulty of exploiting this vulnerability is always low; therefore, the probability of an attack on it should always be high. Therefore, defining the probability of an attack on an LLM based on the access complexity of its vulnerability is not the best choice. A more informative and accurate way to measure the probability of an attack occurring and succeeding is to use a statistical probability score. It can be obtained by conducting N experiments that simulate cyberattacks or by processing statistical data on N such attacks on the LLM assets. Accordingly, if N_s attacks were successful, the statistical probability score P^* can be calculated as follows:

$$P^* = \frac{N_s}{N}\,, \tag{8}$$

The higher the value of this score, the more likely it is that attackers will be interested in potentially attacking the model, while the lower the value, the less likely they will be interested in doing so, since the effort will not bring sufficient benefit to them. To ensure the required level of confidence in the calculation of P^*, the required number of experiments N (or static data from relevant tests) must be determined.

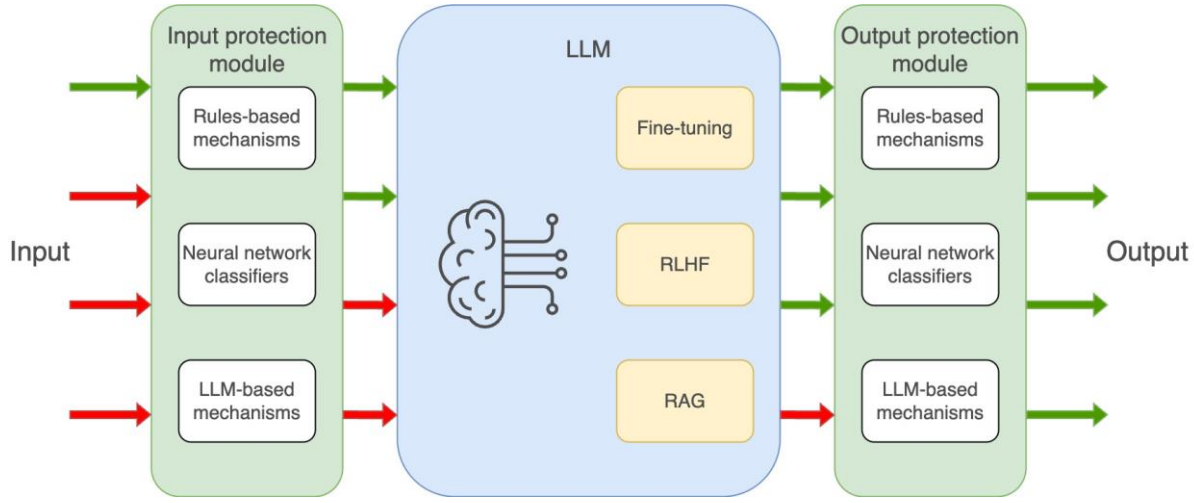

Fig. 11. Scheme of the system to mitigate the impact of vulnerabilities and attacks on LLM

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

If their number is limited by certain circumstances, the confidence probability can be calculated for the fixed value N. Defining possible distribution rules for the number of successful attacks as a random variable is a separate task, but most studies use risk-based assessment [16].

It is clear that requirements can be defined for an acceptable or unacceptable value of P^*, which will affect the formulation of the next task - the task of finding and implementing a set of countermeasures. Countermeasures reduce the probability of attacks by reducing the success rate of attacks. Each of the countermeasures CMi from the set CM is characterized by the level of impact k_i on the value of P^* and the corresponding costs CMC_i. Therefore, the problem of finding the optimal or rational set of countermeasures $CM_{opt} \subset CM$ is formulated, which will ensure the requirements for P^*_{req} are achieved at the minimum cost CMC_{min} (the sum of the costs of implementing the CM_{opt} subset). Algorithms for finding the optimal set of countermeasures have been described in many works, particularly in [38, 39], but the specifics of LLM systems require further research.

This task can be formed and solved not only based on P^*_{req} requirements but also considering the level of acceptable risk R_{asump}, and therefore the severity of the impact. The definition of P^*_{req} (R_{asump}) requirements is based on an understanding of which class of critical systems the LLM system belongs to and can be dynamically revised depending on the functioning conditions. Therefore, a proactive approach to protection is required [22].

Thus, in addition to reducing the criticality of cyber risks by using certain countermeasures, the implementation cost is also important. The main selection criterion is "acceptable risk – minimum cost". Considering this criterion, it is necessary to select a certain number of countermeasures that, on the one hand, can ensure an acceptable level of cyber risks for the organization, and, on the other hand, be cost-effective and have the lowest possible price.

To determine the effectiveness of a countermeasure, it is necessary to maximize the benefits of the countermeasure and reduce its costs. This can be performed using the following formula:

$$CME = \frac{R_b}{R_a \times CMC}, \qquad (9)$$

where CME - is the countermeasure effectiveness ratio, R_b - is the risk value before the countermeasure is applied, R_a - is the risk value after the countermeasure is applied, CMC - is the cost (maybe relative cost) of applying the countermeasure. The risk reduction level is defined by the ratio of the risk before and after the countermeasure is applied. The lower this value, the better the protection. Simultaneously, the effectiveness of a countermeasure additionally depends on its price. The higher the protection price, the lower the overall effectiveness. Thus, if the risk before the application of countermeasures was 8, and countermeasure 1 reduces this value to 6 at a price of $50, then its effectiveness is 0.027, and if countermeasure 2 reduces the risk to 6 at a price of $70, then its effectiveness will be 0.019. Given the obtained efficiency values, countermeasure 1 has a higher coefficient; therefore, it will have a higher priority when choosing a set of countermeasures to protect the model.

Thus, the problem of two-criteria optimization, with risk and cost as the criteria, is reduced to a one-criteria approach in terms of the effectiveness of the countermeasure, which simplifies the selection of risk-acceptable and cost-minimal countermeasures.

The algorithm for choosing the optimal subset of countermeasures to ensure an acceptable level of LLM cybersecurity consists of the following steps:

- analyze the LLM as an object of protection and identify a set of threats and vulnerabilities;
- form a set of countermeasures to protect LLM;
- justify the criteria for choosing countermeasures ("acceptable risk - minimum cost" or "limited cost – minimal risk") considering features of LLM/LLMS;
- analyze a coverage of vulnerabilities under attacks with existing countermeasures;
- form coverage matrix;
- identify complete set of options (CM subsets) for covering all LLM vulnerabilities with a set of countermeasures;
- determine the risk-cost metrics for these options (CM subsets);

choose the optimal subset of countermeasures according to criterion to provide required cybersecurity of LLM/LLMS.

5. Case study

The results of the proposed approach to defining and ensuring LLM cybersecurity can be implemented in many areas of LLM application, particularly in web technologies and unmanned aerial vehicles (UAV). Web technologies are the most native area for applying these models. LLM can be hosted on cloud servers, and access to them can be provided under the AI as a Service (AIaaS) business model [40]. Determining the security level can be done using software tools based on the results of this study. The defined level of criticality of LLM risks obtained by simulating cyberattacks will indicate the model's problem areas.

Considering this level, effective countermeasures that will ensure an acceptable level of risk at a minimum cost will be selected. The proposed countermeasures can then be implemented in the customer's system to ensure an acceptable level of risk.

Unlike the web technology industry, the UAV industry was previously limited in its use of LLM. However, due to the technological revolution, the design of UAVs has changed, and they are now equipped with powerful equipment with great computing capabilities, which significantly expands their potential [41]. Modern UAVs have powerful processors and graphical computing modules, which make it possible to place LLM directly on the vehicle's board. This provides UAVs with the ability to analyze complex data to improve decision-making in various situations. To safely use LLM on UAVs, it is important to ensure the cybersecurity of this technology.

This can be ensured through periodic testing by simulating cyber attacks and obtaining the level of criticality of LLM risks, based on which the necessary countermeasures will be selected and implemented to bring this level to an acceptable value.

As an example, we can analyze the threat of GPT-4 generating harmful content. This analysis is based on the resulting LLM cybersecurity model and is performed by the IMECA method's main provisions. The quantitative values of the attack results and countermeasures to them are taken from [42]. An attack on this type of threat relates to the adversarial type and, according to [8], has a high level of severity, which can be translated into a quantitative value of 8. The analysis results are presented in Table 2.

Table 2

Criticality analysis of the risks of harmful content generation

Threat		Generating harmful content
Vulnerability		Statistical probabilistic response generation
Attack		Jailbreak
Criticality	Probability	0.78
	Severity	8
	Risk	6.24
Countermeasures		SmoothLLM, Perplexity Filter, Erase-and-Check

Thus, this threat has a high risk level before countermeasures are applied, specifically 6.24.

Based on IMECA methodology, it is necessary to calculate new risk level values considering the previously defined set of countermeasures. The impact of each countermeasure on the risks will be calculated separately.

SmoothLLM countermeasure reduces the probability of occurrence to 0.56, thereby reducing the risk to 4.48, which corresponds to a medium risk level.

Perplexity Filter countermeasure reduces the probability of occurrence to 0.7, and therefore, the risk is reduced to 5.6, which corresponds to a high risk level.

Erase-and-Check countermeasure reduces the probability of occurrence to 0.1; therefore, the risk is reduced to 0.8, indicating a low risk level.

Based on the effectiveness of countermeasures, we can conclude that Erase-and-Check is the most effective because it reduces the criticality of risks to a low level without considering their cost.

6. Discussion

The main contribution of this work is a model of LLM cybersecurity, which allows the identification of key security aspects of this technology and a more formal understanding of the state of affairs in this area. This is a model basis for analyzing and qualitatively assessing the LLM and LLM systems' cybersecurity. This model will be used as a starting point in the assessment and ensuring of LLM cybersecurity. This model is part of a more common LLM system cybersecurity model.

The LLM simulation attack discussed in this paper is not a new method for testing the security of these models. Attacking models using jailbreak and prompt injection attacks to determine the ASR value is a common practice that has been used in many other studies. However, unlike other studies, a statistical probability score for the occurrence and success of these attacks was proposed, which, in combination with the severity of their impact, makes it possible to quantify the level of criticality of cyber risks for LLM. The calculation of this level makes it possible to build a cyber risk criticality matrix that will show the overall security state of the system and divide these risks into low, medium, and high. Dividing risks into these three areas is important for further mitigation efforts.

The use of countermeasures to mitigate the impact of possible attacks on LLM is also a common practice. The use of input and output modules, as well as additional model tuning, is covered in other works and is widely used in practice. However, not much attention is paid to their efficiency. To overcome this problem, it has been proposed to select countermeasures according to the criterion of "acceptable risk – minimum cost". For this purpose, a countermeasure efficiency ratio will be used, which depends on the risk value before the countermeasure is applied, the risk value after the countermeasure is applied, and the price of using this countermeasure. Based on these coefficients, the benefits of implementing countermeasures will be maximized and their costs will be minimized.

The use of the proposed approaches is limited to the cybersecurity of the LLM itself. The security of classical software, which usually surrounds these models, is a

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

well-researched area with effective and well-known countermeasures. Therefore, ensuring the cybersecurity of the surrounding classical software is not the purpose of this paper and will not be the focus of future research.

Assessing and ensuring the cybersecurity of LLM is an important, promising, and poorly researched area. This study addresses this area and takes the first steps toward achieving this goal. This paper provides a theoretical overview of the cybersecurity components of this technology, as well as basic methods for its calculation and ensuring. Further work will focus on the practical implementation of the proposed approaches. The creation of a software tool for simulating attacks on LLM for further calculation of the criticality level of model risks and the choice of effective countermeasures that will ensure an acceptable level of risk at a minimum cost is particularly important.

7. Conclusions

This study identifies that improving the cybersecurity of LLM is an important and urgent task, given the widespread use of these models in many areas of human life. The path to this improvement consists of certain steps, the first of which was undertaken in this study.

As part of this work, based on a high-level model of systems that use LLM and a common model of their cybersecurity, a model of LLM cybersecurity was developed as an independent component of this common model. The importance of developing this model is that it acts as a baseline and creates the basis for further research in assessing and ensuring an acceptable level of cyber risks of this technology.

The detailed analysis of the LLM cybersecurity model components has an important practical role. Based on the results of this analysis, it is possible to conduct practical experiments to simulate cyberattacks on LLM based on the knowledge obtained about their threats, vulnerabilities, and attack methods. The main contribution of the study is to formalize the results of these experiments in the form of defining a statistical probability of the occurrence and success of attacks, combining this estimate with the severity of the impact of attacks, and further assessing the level of criticality of cyber risks for LLM as a whole.

In addition, this paper proposes an approach to the selection of countermeasures for LLM based on the criterion of "acceptable risk – minimum cost", which is based on the calculation of their effectiveness.

Because of this study, only the initial steps toward ensuring LLM cybersecurity have been established. Therefore, further work will focus on collecting exploits and determining the real level of severity of effects after their successful use, and creating a software tool for simulating attacks on LLMs to analyze and improve their security. Besides, research on LLM security for specific applications, such as state security [43], deepfake detection [44], safe control of intelligent UAV swarms [45, 46], and so on.

Contributions of authors: conceptualization, methodology, formulation of tasks, analysis – **Oleksii Neretin**, **Vyacheslav Kharchenko**; development of models – **Oleksii Neretin**; verification, analysis of results, visualization, writing - original draft preparation – **Oleksii Neretin**, **Vyacheslav Kharchenko**; writing - review and editing – **Vyacheslav Kharchenko**.

Conflict of Interest

The authors declare that they have no conflict of interest about this research, whether financial, personal, authorship, or otherwise, that could affect the research and its results presented in this paper.

Financing

This study was conducted without any financial support.

Data Availability

The manuscript contains no associated data.

Use of Artificial Intelligence

The authors confirm that they did not use artificial intelligence methods while creating the presented work.

All the authors have read and agreed to publication of this manuscript.

References

1. Clusmann, J., Kolbinger, F. R., Muti, H. S., Carrero, Z. I., Eckardt, J.-N., Laleh, N. G., Löffler, C. M. L., Schwarzkopf, S.-C., Unger, M., Veldhuizen, G. P., Wagner, S. J., & Kather, J. N. The future landscape of large language models in medicine. *Communications medicine*, 2023, vol. 3, no. 1, article no. 141. DOI: 10.1038/s43856-023-00370-1.

2. Nejjar, M., Zacharias, L., Stiehle, F., & Weber, I. LLMs for science: Usage for code generation and data analysis. *Journal of Software: Evolution and Process*, 2025, vol. 37, no. 1. DOI: 10.1002/smr.2723.

3. Wang, S., Xu, T., Li, H., Zhang, C., Liang, J., Tang, J., Yu, P. S., & Wen, Q. Large Language Models for Education: A Survey and Outlook. *arXiv preprint arXiv:2403.18105*, 2024. DOI: 10.48550/arXiv.2403.18105.

4. Li, Y., Zhao, H., Jiang, H., Pan, Y., Liu, Z., Wu, Z., Shu, P., Tian, J., Yang, T., Xu, S., Lyu, Y., Blenk, P., Pence, J., Rupram, J., Banu, E., Liu, N., Wang, L., Song,

W., Zhai, X., Song, K., Zhu, D., Li, B., Wang, X., & Liu, T. Large Language Models for Manufacturing. *arXiv preprint arXiv:2410.21418*, 2024. DOI: 10.48550/arXiv.2410.21418.

5. Samma, H., & El-Ferik, S. UAV Visual Path Planning Using Large Language Models. *Transportation Research Procedia*, 2025, vol. 84, pp. 339-345. DOI: 10.1016/j.trpro.2025.03.081.

6. Hannig, L., Bush, A., Aksoy, M., Becker, S. & Ontrup, G. Campus AI vs Commercial AI: A Late-Breaking Study on How LLM As-A-Service Customizations Shape Trust and Usage Patterns. *arXiv preprint arXiv:2505.10490*, 2025. DOI: 10.48550/arXiv.2505.10490.

7. Kharchenko, V., Fesenko, H., & Illiashenko, O. Quality Models for Artificial Intelligence Systems: Characteristic-Based Approach, Development and Application. *Sensors*, 2022, vol. 22, no. 13, article no. 4865. DOI: 10.3390/s22134865.

8. Neretin, O., & Kharchenko, V. Zabezpechennya kiberbezpeky system shtuchnoho intelektu: analiz vrazlyvostey, atak i kontrzakhodiv [Ensurance of artificial intelligence systems cyber security: analysis of vulnerabilities, attacks and countermeasures]. *Journal of Lviv Polytechnic National University. Information Systems and Networks*, 2022, vol. 12, pp. 7-22. DOI: 10.23939/sisn2022.12.007. (In Ukrainian).

9. Wang, Y., Li, H., Han, X., Nakov, P., & Baldwin, T. Do-Not-Answer: A Dataset for Evaluating Safeguards in LLMs. *arXiv preprint arXiv:2308.13387*, 2023. DOI: 10.48550/arXiv.2308.13387.

10. Derner, E., Batistič, K., Zahálka, J., & Babuška, R. A Security Risk Taxonomy for Prompt-Based Interaction With Large Language Models. *IEEE Access*, 2024, vol. 12, pp. 126176-126187. DOI: 10.1109/ACCESS.2024.3450388.

11. Weidinger, L., Uesato, J., Rauh, M., Griffin, C., Huang, P.-S., Mellor, J., Glaese, A., Cheng, M., Balle, B., Kasirzadeh, A., Biles, C., Brown, S., Kenton, Z., Hawkins, W., Stepleton, T., Birhane, A., Hendricks, L. A., Rimell, L., Isaac, W., Haas, J., Legassick, S., Irving, G., & Gabriel, I. Taxonomy of Risks posed by Language Models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22)*, 2022, pp. 214–229. DOI: 10.1145/3531146.3533088.

12. Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., & Zhang, Y. A survey on large language model (LLM) security and privacy: The Good, The Bad, and The Ugly. *High-Confidence Computing*, 2024, vol. 4, iss. 2. DOI: 10.1016/j.hcc.2024.100211.

13. Cui, T., Wang, Y., Fu, C., Xiao, Y., Li, S., Deng, X., Liu, Y., Zhang, Q., Qiu, Z., Li, P., Tan, Z., Xiong, J., Kong, X., Wen, Z., Xu, K., & Li, Q. Risk Taxonomy, Mitigation, and Assessment Benchmarks of Large Language Model Systems. *arXiv preprint arXiv:2401.05778*, 2024. DOI: 10.48550/arXiv.2401.05778.

14. Zhu, B., Mu, N., Jiao, J. & Wagner, D. Generative AI security: challenges and countermeasures. *arXiv preprint arXiv:2402.12617*, 2024. DOI: 10.48550/arXiv.2402.12617.

15. *OWASP Top 10 for LLM Applications 2025*. Available at: https://genai.owasp.org/llm-top-10/. (accessed 3.05.2025).

16. Babeshko, I., Illiashenko, O., Kharchenko, V., & Leontiev, K. Towards Trustworthy Safety Assessment by Providing Expert and Tool-Based XMECA Techniques. *Mathematics*, 2022, vol. 10, no. 13, p. 2297. DOI: 10.3390/math10132297.

17. Klondike, G. *Threat Modeling LLM Applications*. Available at: https://aivillage.org/large%20language%20models/2023/06/06/threat-modeling-llm. (accessed 3.05.2025).

18. Li, A., Zhou, Y., Raghuram, V. C., Goldstein, T., & Goldblum, M. Commercial LLM Agents Are Already Vulnerable to Simple Yet Dangerous Attacks. *arXiv preprint arXiv:2502.08586*, 2025. DOI: 10.48550/arXiv.2502.08586.

19. Wang, N., Walter, K., Gao, Y., & Abuadbba, A. Large Language Model Adversarial Landscape Through the Lens of Attack Objectives. *arXiv preprint arXiv:2502.02960*, 2025. DOI: 10.48550/arXiv.2502.02960.

20. Rehberger, J. Trust No AI: Prompt Injection Along The CIA Security Triad. *arXiv preprint arXiv:2412.06090*, 2024. DOI: 10.48550/arXiv.2412.06090.

21. Singh, A., & Gupta, B. B. Distributed Denial-of-Service (DDoS) Attacks and Defense Mechanisms in Various Web-Enabled Computing Platforms: Issues, Challenges, and Future Research Directions. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 2022, vol. 18, no. 1, pp. 1-43. DOI: 10.4018/IJSWIS.297143.

22. Lysenko, S., Bobrovnikova, K., Kharchenko, V., & Savenko, O. IoT Multi-Vector Cyberattack Detection Based on Machine Learning Algorithms: Traffic Features Analysis, Experiments, and Efficiency. *Algorithms*, 2022, vol. 15, no. 7, article no. 239. DOI: 10.3390/a15070239.

23. Hicks, M. T., Humphries, J., & Slater, J. ChatGPT is bullshit. *Ethics and Information Technology*, 2024, vol. 26, no. 2, pp. 1-10. DOI: 10.1007/s10676-024-09775-5.

24. Shah, C., Bender, E. M. Situating Search. In *Proceedings of the 2022 Conference on Human Information Interaction and Retrieval*, 2022, pp. 221-232. DOI: 10.1145/3498366.3505816.

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

25. Chollet, F. *Deep Learning with Python, Second Edition*. 2nd ed. Manning Publ., 2021. 504 p.

26. Das, B. C., Amini, M. H., & Wu, Y. Security and Privacy Challenges of Large Language Models: A Survey. *ACM Computing Surveys*, 2025, vol. 57, no. 6, pp. 1-39. DOI: 10.1145/3712001.

27. Du, H., Liu, S., & Cao, Y. Can Differentially Private Fine-tuning LLMs Protect Against Privacy Attacks? *arXiv preprint arXiv:2504.21036*, 2025. DOI: 10.48550/arXiv.2504.21036.

28. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022. DOI: 10.48550/arXiv.2203.02155.

29. Mao, Y., He, J., & Chen, C. From Prompts to Templates: A Systematic Prompt Template Analysis for Real-world LLMapps. *arXiv preprint arXiv:2504.02052*, 2025. DOI: 10.48550/arXiv.2504.02052.

30. Perez, F., & Ribeiro, I. Ignore Previous Prompt: Attack Techniques For Language Models. *arXiv preprint arXiv:2211.09527*, 2022. DOI: 10.48550/arXiv.2211.09527.

31. Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. Not what you've signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*, 2023, pp. 79-90. DOI: 10.1145/3605764.3623985.

32. Wei, A., Haghtalab, N., & Steinhardt, J. Jailbroken: How Does LLM Safety Training Fail? *arXiv preprint arXiv:2307.02483*, 2023. DOI: 10.48550/arXiv.2307.02483.

33. Siska, C., & Sankaran, A. AttentionDefense: Leveraging System Prompt Attention for Explainable Defense Against Novel Jailbreaks. *arXiv preprint arXiv:2504.12321*, 2025. DOI: 10.48550/arXiv.2504.12321.

34. Lin, L., Mu, H., Zhai, Z., Wang, M., Wang, Y., Wang, R., Gao, J., Zhang, Y., Che, W., Baldwin, T., Han, X., & Li, H. Against The Achilles' Heel: A Survey on Red Teaming for Generative Models. *Journal of Artificial Intelligence Research*, 2025, vol. 82, pp. 687-775. DOI: 10.1613/jair.1.17654.

35. Illiashenko, O., Kharchenko, V., Babeshko, I., Fesenko, H., & Di Giandomenico, F. Security-Informed Safety Analysis of Autonomous Transport Systems Considering AI-Powered Cyberattacks and Protection. *Entropy*, 2023, vol. 25, no. 8, article no. 1123. DOI: 10.3390/e25081123.

36. *Vulnerability Metrics*. Available at: https://nvd.nist.gov/vuln-metrics/cvss. (accessed 3.05.2025).

37. Bitton, R., Maman, N., Singh, I., Momiyama, S., Elovici, Y., & Shabtai, A. Evaluating the Cybersecurity Risk of Real-world, Machine Learning Production Systems. *ACM Computing Surveys*, 2023, vol. 55, no. 9, pp. 1-36. DOI: 10.1145/3559104.

38. Zemlianko, H., & Kharchenko, V. IMECA-analiz kiberbezpeky system bahatofunktsionalnykh flotiv BPLA pry kombinovanykh atakakh: bazovi modeli ta vybir kontrzakhodiv [IMECA analysis of cybersecurity for multi-functional UAV fleets under combined attacks: basic models and countermeasure choice]. *Measuring and computing devices in technological processes*, 2023, no. 4, pp. 225-233. DOI: 10.31891/2219-9365-2023-76-30. (In Ukrainian).

39. Zemlianko, H., & Kharchenko, V. Cybersecurity risk analysis of multifunctional UAV fleet systems: a conceptual model and IMECA-based technique. *Radioelectronic and Computer Systems*, 2023, vol. 0, no. 4, pp. 152-170. DOI: 10.32620/reks.2023.4.11.

40. Syed, N., Anwar, A., Baig, Z., & Zeadally, S. Artificial Intelligence as a Service (AIaaS) for Cloud, Fog and the Edge: State-of-the-Art Practices. *ACM Computing Surveys*, 2025, vol. 57, no. 8. DOI: 10.1145/3712016.

41. Javaid, S., Fahim, H., He, B. & Saeed, N. Large language models for UAVs: Current state and pathways to the future. *IEEE Open Journal of Vehicular Technology*, 2024, vol. 5, pp. 1166-1192. DOI: 10.1109/OJVT.2024.3446799.

42. Chao, P., Debenedetti, E., Robey, A., Andriushchenko, M., Croce, F., Sehwag, V., Dobriban, E., Flammarion, N., Pappas, G. J., Tramer, F. & Hassani, H. Jailbreakbench: An open robustness benchmark for jailbreaking large language models. *arXiv preprint arXiv:2404.01318*, 2024. DOI: 10.48550/arXiv.2404.01318.

43. Laktionov, O., Shefer, O., Laktionova, I., Halai, V., & Podorozhniak, A. Implementation of unsupervised learning models for analyzing the state's security level. *Advanced Information Systems*, 2024, vol. 8, no. 3, pp. 85-91. DOI: 10.20998/2522-9052.2024.3.10.

44. Rajakumareswaran, V., Raguvaran, S., Chandrasekar, V., Rajkumar, S., & Arun, V. DeepFake detection using transfer learning-based Xception model. *Advanced Information Systems*, 2024, vol. 8, no. 2, pp. 89-98. DOI: 10.20998/2522-9052.2024.2.10.

45. Yang, Z., Zhang, Y., Zeng, J., Yang, Y., Jia, Y., Song, H., Lv, T., Sun, Q., & An, J. AI-Driven Safety and Security for UAVs: From Machine Learning to Large Language Models. *Drones*, 2025, vol. 9, no. 6, article no. 392. DOI: 10.3390/drones9060392.

46. Sezgin, A. Scenario-Driven Evaluation of Autonomous Agents: Integrating Large Language Model for UAV Mission Reliability. *Drones*, 2025, vol. 9, no. 3, article no. 213. DOI: 10.3390/drones9030213.

Received 17.03.2025, Accepted 20.05.2025

МОДЕЛЬ ЗАБЕЗПЕЧЕННЯ КІБЕРБЕЗПЕКИ LLM

О. С. Неретін, В. С. Харченко

Предметом дослідження є модель забезпечення кібербезпеки великих мовних моделей (LLM). **Метою** цього дослідження є розробка та аналіз компонентів моделі кібербезпеки LLM з метою підвищення точності її оцінки та забезпечення необхідного рівня безпеки. **Завдання**: представити і проаналізувати абстрактну структуру систем LLM; побудувати загальну модель кібербезпеки систем LLM (LLMS); розробити модель кібербезпеки LLM як основного компонента LLMS; проаналізувати елементи розробленої моделі кібербезпеки; описати можливі приклади використання та навести приклад аналізу критичності ризиків для однієї з загроз LLM; обґрунтувати напрями майбутніх досліджень щодо визначення, класифікації, аналізу критичності та колекціонування експлойтів для перевірки стійкості LLM. **Результати дослідження**: розроблено базову високорівневу архітектуру LLMS, яка складається із зовнішніх джерел, сервісу LLM, серверних функцій та середовищ зберігання; на основі цієї архітектури побудовано загальну модель кібербезпеки LLM; зосереджуючись безпосередньо на LLM, розроблено її модель кібербезпеки, яка є незалежним компонентом загальної моделі кібербезпеки LLMS та базується на ланцюжку наступних елементів: загроза, вразливість, атака, ризики та контрзаходи; крім того, проведено аналіз елементів моделі кібербезпеки LLM та запропоновано послідовність вибору контрзаходів. **Висновки**. Це дослідження визначає, що покращення кібербезпеки LLM є важливим та актуальним завданням, враховуючи широке використання цих моделей у багатьох сферах людського життя. Важливість розробки моделі кібербезпеки LLM полягає в тому, що вона є базовою для всіх наступних досліджень. Практичне значення аналізу елементів моделі полягає у їх використанні для проведення експериментів з моделювання кібератак на LLM. Основним внеском даного дослідження є моделі кібербезпеки LLM та LLMS та формалізація результатів цих експериментів та оцінка рівня критичності для кібер ризиків моделей та вибір контрзаходів на основі коефіцієнта їх ефективності. У цьому випадку забезпечення прийнятного рівня ризику для LLM можливе за мінімальних витрат. **Напрями подальших досліджень**: визначення та класифікація експлойтів для перевірки безпеки LLM; методологія збору цих експлойтів; аналіз критичності завданої ними шкоди для різних застосувань.

Ключові слова: LLM; кібербезпека LLM; модель кібербезпеки; загроза; вразливість; атака; ризик; контрзаходи.

Неретін Олексій Сергійович – асп. каф. комп'ютерних систем, мереж і кібербезпеки, Національний аерокосмічний університет «Харківський авіаційний інститут», Харків, Україна.

Харченко Вячеслав Сергійович –д-р техн. наук, проф., чл.-кор. НАН України, зав. каф. комп'ютерних систем, мереж і кібербезпеки, Національний аерокосмічний університет «Харківський авіаційний інститут», Харків, Україна.

Oleksii Neretin – PhD Student of the Department of Computer Systems, Networks and Cybersecurity, National Aerospace University «Kharkiv Aviation Institute», Kharkiv, Ukraine,
e-mail: o.s.neretin@csn.khai.edu, ORCID: 0000-0003-2114-6714, Scopus Author ID: 58099131000.

Vyacheslav Kharchenko – Doctor of Technical Science, Professor, Corr. member of the National Academy of Science of Ukraine, Head of the Department of Computer Systems, Networks and Cybersecurity, National Aerospace University «Kharkiv Aviation Institute», Kharkiv, Ukraine,
e-mail: v.kharchenko@csn.khai.edu, ORCID: 0000-0001-5352-077X, Scopus Author ID: 22034616000.