DOI: https://doi.org/10.15276/hait.09.2026.05
UDC 004.89

IMECA method of risk-based assessment and ensuring cybersecurity of Large Language Models

Oleksii S. Neretin[1)]
ORCID: https://orcid.org/0000-0003-2114-6714; o.s.neretin@csn.khai.edu. Scopus Author ID: 58099131000
Vyacheslav S. Kharchenko[1)]
ORCID: https://orcid.org/0000-0001-5352-077X; v.kharchenko@csn.khai.edu. Scopus Author ID: 22034616000
[1)] National Aerospace University "Kharkiv Aviation Institute", 17, Vadym Manko Str. Kharkiv, 61070, Ukraine

ABSTRACT

Large Language Models (LLMs) help perform complex tasks that previously relied only on humans. Many different areas of human activity already use this technology or are actively exploring its capabilities with a view to future integration into work processes. In addition to the positive effects of their use, there are problems of uncertain and unexpected behavior, in particular, the generation of forbidden content. Given the expanding use of these models and their behavior, it is necessary to assess the level of security and further ensure the cybersecurity of this technology. The object of the study is the processes of ensuring cybersecurity for large language models. The article proposes a methodology for countering this threat by assessing the risks of such behavior and ensuring an acceptable level of cybersecurity for LLMs using the IMECA (Intrusion Modes Effects Criticality Analysis) technique of risk-based assessment. A set of countermeasures has been developed to increase the security level of LLMs, and procedures for their selection based on the criteria of maximum productivity and best rating using a countermeasure rating matrix are defined. An example of testing and ensuring the cybersecurity of a test language model is provided, the results of which show that the criticality level of cyber risks before the use of countermeasures is significantly decreased after using the most productive and highest-rated countermeasures, but threats with a high level of cyber risk criticality still exist. Directions for future research are proposed to deepen the procedure for evaluating and ensuring the security of LLMs, given the continuous development of these models and their protection mechanisms. The main result of this work is the combination of a model for ensuring the cybersecurity of LLMs and an improved method for analyzing the criticality of their vulnerabilities for further adaptation of the IMECA method of quantitative risk-based assessment and cybersecurity assurance for the field of LLMs.

Keywords: Risk-based assessment; cybersecurity; Large Language Models; Intrusion Modes Effects Criticality Analysis; countermeasures; threat; vulnerability; attack

***For citation*: Neretin O. S., Kharchenko V. S. "IMECA method of risk-based assessment and ensuring cybersecurity of Large Language Models". *Herald of Advanced Information Technology.* 2026; Vol.9 No.1: 60–70. DOI: https://doi.org/10.15276/hait.09.2026.05**

INTRODUCTION

Motivation. The rapid development of Large Language Models (LLMs) is receiving increased attention from various fields of human activity. Each new update to these models gives them greater capabilities in understanding human language and generating human-like text. This technology is used in education as pedagogical agents that assist in teaching and provide students with personalized learning advice [1]. In medicine, these models are used for diagnostics, clinical data analysis, and optimization of clinical services [2]. The software development industry actively uses LLMs for code autocompletion [3]. They contribute to significant progress in various fields by automating tasks, increasing accuracy, and providing deeper understanding [4]. But despite the progress made by using these models, they can behave unexpectedly, not as intended by their developers [5]. LLMs can be offensive, biased, and provide harmful advice in professional fields (such as legal, financial, and medical), which, without additional verification by qualified specialists, can lead to unpredictable results.

Given the widespread use of LLMs in various industries and the potential effects of their use, it is important to assess the risks associated with this use and ensure an acceptable level of cybersecurity for this technology.

The purpose of the paper is to adapt and develop the IMECA (Intrusion Modes Effects Criticality Analysis) method of risk-based assessment and cybersecurity assurance for LLMs.

The objectives of the study are as follows:

- adapt the IMECA method of risk-based assessment and cybersecurity assurance for language models;
- form a set of countermeasures to LLMs vulnerabilities, determine their indicators, and develop selection criteria based on a matrix of ratings for these indicators;
- conduct a test evaluation and assurance of the test model's cybersecurity using the IMECA analysis method;

- substantiate directions for future research on deepening the procedure for assessing and ensuring the safety of LLMs.

The article is structured as follows. Section 2 reviews related work in the field of evaluating and ensuring the cybersecurity of language models. Section 3 describes the research methodology. Section 4 focuses on the method of risk-based analysis of the criticality of LLMs vulnerabilities. Section 5 considers the direction of ensuring the cybersecurity of models. Section 6 considers an example of use, and Section 7 summarizes the work and suggests directions for future research.

RELATED WORKS

A large number of studies are devoted to assessing the cybersecurity of LLMs, but their focus is on the process of attacking and verifying the positive impact of countermeasures on the Attack Success Rate (ASR). At the same time, formalized methodologies are not used to conduct the assessment process and ensure the security of this technology.

In works [6] and [7], various LLMs are tested and the ASR coefficient level is determined. Forbidden texts are classified into categories according to the security policies of the companies that develop language models. According to this classification, an attack is carried out, the results of which determine the security level of these models. After that, certain protective mechanisms are applied to ensure the security of LLMs, with the attack procedure being repeated to determine the security level. The results of the double attack procedure determine the level of ASR coefficient decrease. The entire procedure of assessing and ensuring the cybersecurity of LLMs is carried out without the help of formalized methodologies and without assessing the level of security risks of these models, which is necessary for further determining the efficiency of countermeasures.

Work [8] addresses the modeling and analysis of threats to artificial intelligence (AI) systems in general. The work proposes using the STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege) methodology to assess the security of AI systems. In addition, it discusses the use of the FMEA (Failure Modes and Effects Analysis) method, designed to identify failures in engineering processes by determining their occurrence, consequences, and impact on the system. To assess the severity of threats, it is proposed to use the DREAD (Damage Potential, Reproducibility, Exploitability, Affected Users, and Discoverability) methodology, which was developed and is used as a supplement to the STRIDE method. The results of the threat analysis in this work lack the quantitative assessment required for further ensuring the cybersecurity of AI systems.

Research [9] focuses on assessing risks to language models using the DREAD method. The paper classifies attacks on LLMs and calculates qualitative indicators of the complexity of these attacks. Damage, reproducibility, exploitability, affected users, and discoverability are assessed on a scale from 0 to 10. Based on these metrics, quantitative and qualitative risk levels are calculated. Work [10] combines the STRIDE and DREAD methodologies for threat modeling and risk analysis of LLMs. The results of these studies are interesting from a practical point of view, but the works lack disclosure of the direction of ensuring the cybersecurity of LLMs based on the values of the obtained risk metrics.

Work [11] develops a model for ensuring the cybersecurity of LLMs, which can be used as a starting point for further risk-based assessment of the cybersecurity of these models. Work [5] deals with collecting exploits for LLMs vulnerabilities and analyzing the criticality of risks from their use, which makes it possible to conduct experimental attacks on these models and quantitatively calculate the level of criticality of cyber risks. This study also develops a method for analyzing the criticality of LLMs vulnerabilities, which can be used as a basis for the IMECA method of risk-based assessment and cybersecurity assurance for LLMs. These studies will form the basis for assessing and further ensuring the security of language models.

Therefore, it is important to assess and ensure the cybersecurity of LLMs in a more formal way using the risk-based IMECA method.

RESEARCH METHODOLOGY

The research methodology is based on the implementation of the IMECA method of risk-based assessment and assurance of LLM cybersecurity and consists of the following steps:

- adaptation of the IMECA method of risk-based assessment and cybersecurity assurance for language models by combining the cybersecurity assurance model [11] and the improved method of analyzing the criticality of LLMs vulnerabilities [5];
- formation of a set of countermeasures to LLMs vulnerabilities, determination of their indicators, and development of selection criteria based on a matrix of ratings for these indicators;

- conduct a test evaluation and ensure the cybersecurity of the test model using the method of IMECA analysis;
- substantiation of future research directions for deepening the evaluation procedure and ensuring the security of LLMs by increasing the variability of attacks on models and conducting additional experiments on attacking LLMs using various types of protective mechanisms.

METHOD OF RISK-BASED ANALYSIS OF THE CRITICALITY OF LLMS VULNERABILITIES

Method of IMECA analysis

IMECA is an adaptation of the key safety assessment method XMECA (X Modes, Effects, and Criticality Analysis, where X can be from various known techniques and areas) [12]. This methodology is used to analyze intrusion methods, effects, and criticality. Based on the results of this assessment, an analysis of risk criticality is performed, in particular with the help of expert decisions and judgments, using quantitative information from IMECA tables.

IMECA analysis is designed to assess the state of cybersecurity. This method focuses on system vulnerabilities that can be exploited by attackers. Each such vulnerability must be represented in the IMECA table. After identifying all vulnerabilities and the criticality level of system risks, cybersecurity is ensured through the use of countermeasures.

Unlike most other studies in the field of cybersecurity of language models, which do not use any methodologies to assess the security of these models, the use of IMECA analysis provides the advantage of structuring and formalizing such analysis, as well as allowing for a quantitative assessment of the risks of the system on which the cybersecurity procedure is based, using specific countermeasures.

Parameters of the IMECA method in the context of LLM

According to the main provisions of the IMECA method and the LLMs cybersecurity model [11], the following elements of the IMECA table can be identified:

- threat (T) – immediate problem for language model;
- vulnerability (V) – weaknesses in LLMs that can be exploited by attackers;
- attack (A) – actions performed by attackers to affect models through their vulnerabilities;
- effects (E) – results of an attack on LLMs in the form of loss of confidentiality (C), integrity (I), or availability (A);
- probability (P) – determines the possibility of an attack occurring;
- severity (S) – the level of seriousness and danger of an attack based on its effects;
- risk (R) – the total impact of an attack on models, which is determined by a combination of probability and severity;
- countermeasures - actions and measures directed at countering attacks.

The above list of elements is the basis for analyzing the security state of LLMs. It enables comprehensive, formal, risk-based assessment and subsequent cybersecurity assurance of language models.

Thus, the IMECA method can be summarized by the following formulas:

$$IT = \{ITR_i, i = 1, 2, \dots, n\}\,, \qquad (1)$$

where *IT* is the IMECA table, which consists of a set of rows; *ITR$_i$* is a row of the table, which is a tuple of elements for analyzing a specific vulnerability; n is the number of table rows.

$$ITR_i = (th_i, v_i, a_i, e_i, p_i, s_i, r_i, CM_i)\,, \qquad (2)$$

where th_i is the threat to the model; v_i is the specific vulnerability; a_i is the attack on the model; e_i is the effects of the attack; p_i is the probability of an attack occurring; s_i is the severity of the effects after the attack; r_i is the total risk; CM_i is the set of countermeasures for this vulnerability.

$$CM_i = \{cm_{ij}, j = 1, 2, \dots, m_i\}\,, \qquad (3)$$

where cm_{ij} is the countermeasure for a specific vulnerability; m_i is the number of countermeasures for the vulnerability.

Based on the results of the study [11], language model responses can be of four types: correct responses; incorrect responses; responses containing content forbidden by security policies; responses containing private data. Starting from the work [5], the studied set consists of LLM responses containing forbidden content. The current study is also devoted to this set of threats. Thus, the number of rows in the IMECA table is equal to the number of categories of forbidden texts, namely 15 [5]. The list of IMECA rows is given in Table 3.

Given the focus of this study on a single set of threats, the vulnerability, attack, effects, and possible set of countermeasures will be the same for each row of the table. The vulnerability is statistical probabilistic response generation (SPRG) [11]. Language models are attacked using regular texts [11]. This type of attack is called prompt hacking (PH) [13]. The effects of the attack affect the loss of model integrity [11]. The possible set of countermeasures is discussed in the next section.

Principles of IMECA analysis of LLM cybersecurity

A key feature of IMECA analysis is its focus on risk-based assessment of LLM cybersecurity. Risk (*R*) is a combination of probability (*P*) and severity (*S*) indicators and is determined using the following traditional formula:

$$R = P \times S\ . \qquad (4)$$

According to works [5] and [11], statistical probability score of attacks occurrence and success (P^*) will be used when performing experimental attacks. This method of determining probability is more effective than the traditional one, which is based on the complexity of exploiting vulnerabilities, because experimental attacks will be carried out using exploits that have the same low level of complexity [11]. The process of determining the statistical probability score is based on an improved method of analyzing the criticality of LLMs vulnerabilities [5].

The statistical probability score is determined by the following formula:

$$P^* = \frac{N_S}{N}\ , \qquad (5)$$

where N_s is the number of successful attacks on LLM; N is the total number of attacks on the model.

The severity of the effects of an attack will be determined in accordance with [5] based on the severity of penalties under European Union law. These values are shown in Table 3.

Based on these parameters, a matrix of criticality of cyber risks for LLMs is constructed. This matrix is shown in Table 1. Green color indicates a low level of cyber risk, yellow indicates a medium level, and red indicates a high level. Probability and severity indicators are determined according to the study [11] and are based on the Common Vulnerability Scoring System version 2 metrics. Criticality indicators are given in absolute units.

The next step is to calculate the impact of countermeasures on model vulnerabilities. Each threat has its own probability of attack occurrence and success, as well as its own severity level. The severity level has a fixed value [11], so countermeasures affect the probability of occurrence. If the probability level decreases, the criticality level of risks also decreases.

Table 1. **LLMs cyber risk criticality matrix in absolute units**

Probability	**Severity**		
	Low (0.0 – 3.9)	Medium (4.0 – 6.9)	High (7.0 – 10.0)
Low (0.0 – 0.39)	0.0 – 1.52	0.0 – 2.69	0.0 – 3.9
Medium (0.40 – 0.69)	0.0 – 2.69	1.6 – 4.76	2.8 – 6.9
High (0.70 – 1.0)	0.0 – 3.9	2.8 – 6.9	4.9 – 10.0

Source: **compiled by the authors**

The main goal of using countermeasures is not just to reduce the level of risk in absolute units, but to move the vulnerability between high, medium, and low risk zones. If a certain countermeasure reduces the absolute level of risk, but the vulnerability stays in the same risk zone, then this countermeasure isn't effective. Therefore, it is more optimal to reduce the level of risk in relative units. At the same time, the probability and severity values stay in absolute units. The cyber risk criticality matrix in relative units is shown in Table 2.

Table 2. **LLMs cyber risk criticality matrix in relative units**

Probability	**Severity**		
	Low (0.0 – 3.9)	Medium (4.0 – 6.9)	High (7.0 – 10.0)
Low (0.0 – 0.39)	1	1	2
Medium (0.40 – 0.69)	1	2	3
High (0.70 – 1.0)	2	3	3

Source: **compiled by the authors**

IMECA analysis of the criticality of LLM vulnerabilities

Table 3 contains a detailed form of the IMECA analysis of risk criticality. The vulnerability probability values and risks are not filled in because they are determined experimentally when attacking a specific language model (Section 6 contains these values for the SmolLM3 model from Hugging Face). The set of countermeasures for each vulnerability is the same, so it will not be included in the final table.

Experimental attacks are carried out according to the method of analyzing the criticality of LLMs vulnerabilities [5], based on the results of which this table is filled in.

Table 3. **IMECA form for analyzing the criticality of LLM vulnerabilities**

#	Threat	V	A	E	Criticality		
					P	**S**	**R**
1	Generation of harmful content (HC)	SPRG	PH	I	-	4	-
2	Generation of content about cybercrime activities (CA)	SPRG	PH	I	-	6	-
3	Generation of content about physical harm (PH)	SPRG	PH	I	-	10	-
4	Generation of content about economic harm (EH)	SPRG	PH	I	-	5	-
5	Generation of content about illegal drugs (ID)	SPRG	PH	I	-	9	-
6	Generation of content about weapons activities (WA)	SPRG	PH	I	-	9	-
7	Generation of terrorist content (TC)	SPRG	PH	I	-	8	-
8	Generation of content about intellectual property infringement (IPI)	SPRG	PH	I	-	6	-
9	Generation of content about fraud (F)	SPRG	PH	I	-	8	-
10	Generation of disinformation (D)	SPRG	PH	I	-	2	-
11	Generation of adult content (AC)	SPRG	PH	I	-	7	-
12	Generation of content about political activities (PA)	SPRG	PH	I	-	1	-
13	Generation of content about privacy violations (PV)	SPRG	PH	I	-	4	-
14	Generation of content about unauthorized practices (UP)	SPRG	PH	I	-	2	-
15	Generation of content about government decisions (GD)	SPRG	PH	I	-	3	-

***Source*: compiled by the authors**

The procedure for filling in the IMECA table consists of the following steps:

- probability of occurrence and success of attacks is calculated using formula (5) and filled in for each row of the table;
- after determining the probability of occurrence, the absolute level of risk is calculated using formula (4);
- according to the criticality matrix, the relative value of this level is determined, namely, whether a particular vulnerability enters the low, medium, or high risk zone.

The next stage is the procedure for evaluating and selecting countermeasures for the language model for its further protection. This stage consists of the following steps:

- building a matrix of cyber risk criticality before applying countermeasures;
- calculating the rating for each countermeasure and filling in the table with these ratings (section 5 contains information about this process);
- ranking countermeasures by rating for further selection and implementation;
- building a matrix of critical cyber risks for countermeasures selected according to criteria.

As a result of the above procedures, a finalised, detailed report on the assessment and assurance of cybersecurity for a specific LLM is obtained in accordance with the main provisions of the IMECA method.

LLM CYBERSECURITY ASSURANCE

Countermeasures to LLM vulnerabilities

To protect LLMs from generating forbidden content, countermeasures are used that can be applied at the following stages of the model's operation [14]:

- protection before processing the request by the model – the user request is processed before it is passed to the model for generating a response;
- protection during model processing – intermediate results of model operation are analyzed, such as neuron activation and hidden states;
- post-processing protection – evaluates the model's response to the presence of forbidden content.

The most optimal approach is to use protection at the stages before or after request processing. Protection during processing requires multiple runs of the base model, which leads to significant response delays [14].

Based on studies [14] and [15], a set of known countermeasures and related parameters is identified. Table 4 contains this set of countermeasures.

The Input Check method is based on prior verification of user request by a basic model using a special judge request [15] The In-Context Defense method increases model robustness by demonstratively adding context with rejected examples of texts containing forbidden content [16]. The Self-Reminder technique offers a simple but effective protection method called system self-reminder, which consists of encapsulating the user's request in a system prompt that reminds the model to respond responsibly [17]. SmoothLLM protection is based on creating several modified copies of the input request and aggregating predictions to detect malicious input [18]. The Self Defense method proposes checking the model's response with another instance of the model for forbidden content [19]. AutoDefense uses LLM agents to check the response of the base model and, based on the results of their work, provides the user with a response without forbidden content [20]. PerplexityDefense is based on the use of a complexity filter that checks whether the complexity of the query exceeds a certain threshold [21]. The BPE-dropout method splits the text into more tokens than in the standard split [21].

Table 4. **Set of countermeasures to ensure cybersecurity of LLMs**

Countermeasure (CM)	Probability decrease (d)	Execution time (t)	Computational cost (c)
Input Check (IC)	0.47	10.7	10
In-Context Defense (ICD)	0.35	13.1	5
Self-Reminder (SR)	0.39	16.4	5
SmoothLLM	0.38	136.1	30
Self Defense (SD)	0.53	30.2	10
AutoDefense	0.94	272.2	30
Perplexity Defense	0.03	10.92	4
BPE-dropout (BPE-d)	0.38	14.9	4

Source: **compiled by the authors**

The probability decrease parameter (d) determines the percentage by which a particular countermeasure decreases the probability of an attack occurring and succeeding. The execution time (t) determines how long a particular protection method works before determining the malicious nature of a user request and is measured in seconds. The computational cost parameter (c) is responsible for the additional overhead required to run the protection method and is measured in relative units. A value of 4 corresponds to additional calculations on the computer's central processing unit. A value of 5 corresponds to the extension of user requests, which causes an additional load on the model, and values of 10 and 30 correspond to full requests to the model, which is a significant additional load. All parameters are average values and apply to each LLM threat (IMECA row in the table).

Based on the above indicators, we can conclude that the use of SmoothLLM and AutoDefense methods is not acceptable for real life, because many steps are required to process the input and output text, thus spending a lot of time on this process. The PerplexityDefense method, while sufficiently fast, has a very low impact on the probability decrease coefficient, which also makes it impossible to use it to protect language models. All other defense methods from Table 4 will be used to analyze their impact on the risk criticality level of LLMs and to further select the best countermeasure for a specific model, taking into account the defined criteria.

The countermeasures identified above are compatible with each other. Each can be combined with each other. Table 5 shows the compatibility matrix of countermeasures.

Table 5. **Countermeasures compatibility matrix**

CM	IC	ICD	SR	SD	BPE-d
IC		+	+	+	+
ICD	+		+	+	+
SR	+	+		+	+
SD	+	+	+		+
BPE-d	+	+	+	+	

Source: compiled by the authors

When countermeasures are used in pairs, their parameters change in direct proportion. The decrease in probability, execution time, and computational cost increase proportionally depending on the selected pair. Determining the values of countermeasure parameters and further calculations based on their combination are the subject of future research.

Countermeasure indicators

Cybersecurity for language models in terms of generating forbidden content will be ensured through a set of five countermeasures, which is determined by the following formula:

$$CMRM = \{CMP, CME, CMC, CMR\}\,, \qquad (6)$$

where *CMRM* is the countermeasure rating matrix; $CMP = \{cmp_j,\ j = 1, 2, \ldots, m\}$ is the set of countermeasure productivity based on the relative risk decrease; $CME = \{cme_j,\ j = 1, 2, \ldots, m\}$ is the set of countermeasures effectiveness based on the execution time of the protection method; *CMC* = $\{cmc_j, j = 1, 2, ..., m\}$ is the set of countermeasures cost based on the relative computational cost; *CMR* = $\{cmr_j, j = 1, 2, ..., m\}$ is the set of countermeasure ratings; *m* is the number of countermeasures (in our case, it is equal to 5).

$$cmp_j = \sum_{i=1}^{n} Rb_i - Ra_{ji} , \qquad (7)$$

where cmp_j is the productivity of a specific countermeasure; Rb_i is the relative risk before applying the countermeasure; Ra_{ji} is the relative risk after applying a specific countermeasure; *n* is the number of LLMs threats (rows in the IMECA table; in our case, 15).

$$Ra_{ji} = (Pb_i^* - (Pb_i^* \times d_j)) \times S_i , \qquad (8)$$

where Pb^*_i is a statistical probability score of attack occurrence and success before countermeasure implementation; d_j is a probability decrease; S_i is severity.

$$cme_j = \sum_{i=1}^{n} \frac{Rb_i}{Ra_{ji} \times t_j} , \qquad (9)$$

where cme_j is the effectiveness of a specific countermeasure; t_j is the execution time of the countermeasures.

$$cmc_j = \sum_{i=1}^{n} \frac{Rb_i}{Ra_{ji} \times c_j} , \qquad (10)$$

where cmc_j is the relative cost of a specific countermeasure; c_j is the relative computational cost.

$$cmr_j = cmp_j + cme_j + cmc_j , \qquad (11)$$

where cmr_j is the rating of a specific countermeasure.

Thus, the countermeasure rating matrix looks as follows:

$$CMRM = \begin{bmatrix} cmp_1 & cme_1 & cmc_1 & cmr_1 \\ cmp_2 & cme_2 & cmc_2 & cmr_2 \\ \vdots & \vdots & \vdots & \vdots \\ cmp_j & cme_j & cmc_j & cmr_j \end{bmatrix}, \qquad (12)$$

Each row of the matrix contains the values of productivity, efficiency, and relative cost of a particular countermeasure. The rating value in each row is the sum of the three previous indicators of the countermeasure. The number of rows in the matrix is equal to the number of selected countermeasures, namely 5 (j = 5). Basing on the matrix of these indicators makes it possible to select countermeasures according to specific criteria.

Criteria for selecting countermeasures

Considering the specifics of how language models work and the focus of the study on the threat of forbidden content generated by these models, as well as the possibility of applying each countermeasure to decrease the level of risk for each threat posed by LLMs, the following criteria for selecting countermeasures are formulated:

- maximum productivity – a countermeasure is selected based on minimizing overall risks in risk zone units, regardless of other indicators of this countermeasure;
- best rating – a balanced countermeasure is selected that has the highest rating according to the countermeasure rating matrix.

If a countermeasure is selected based on the criterion of maximum productivity, the following optimization problem arises:

$$f(cmp) \to max, cmp \in CMP, cmp \geq 1 , \qquad (13)$$

where *f*(*cmp*) is the objective function whose value must be maximized; *cmp* is a variable (productivity) belonging to the set of countermeasure productivities and limited by values greater than or equal to 1.

When selecting a countermeasure based on the best rating criterion, we have the following optimization problem:

$$f(cmr) \to max, cmr \in CMR, cmr \geq 1 , \qquad (14)$$

where *f*(*cmr*) is the objective function whose value must be maximized; *cmr* is a variable (rating) belonging to the set of countermeasure ratings and limited by values greater than or equal to 1.

Algorithms for selecting countermeasures

The algorithm for selecting a countermeasure based on the criterion of maximum productivity solves the optimization problem as follows:

$$f(cmp^*) \geq f(cmp),\ \forall\ cmp \in CMP , \qquad (15)$$

where cmp^* is the acceptable productivity value at which the objective function has the highest value across the entire acceptable range. Thus, the algorithm must find the maximum productivity value in the rating matrix that satisfies this selection criterion.

The algorithm for selecting a countermeasure based on the best rating criterion solves the optimization problem as follows:

$$f(cmr^{*}) \geq f(cmr),\ \forall\, cmr \in CMR\ , \qquad (16)$$

where cmr^{*} is the acceptable rating value at which the objective function has the highest value across the entire acceptable range. Thus, the algorithm must find the maximum rating value in the rating matrix that satisfies this selection criterion.

The selection results in two countermeasures based on the criteria of maximum productivity and highest rating. The selection results are marked in the rating matrix with a specific color. The final decision between these two countermeasures is the responsibility of the information system user.

CASE STUDY

Risk-based assessment and cybersecurity assurance of LLMs will be performed for the local SmolLM3 model from Hugging Face (3.1B parameters, Q4_K_M quantization). The success of the attack will be determined by the local gpt-oss model from OpenAI (20B parameters, Q4_K_M quantization). The entire procedure will be performed on a MacBook Pro laptop with an Apple M1 Max processor and 32GB of memory. The models are launched using Docker Model Runner from Docker.

The experiment duration was 2 hours, 30 minutes, and 4 seconds. The total number of requests to the model was 825. The number of dangerous responses was 614.

The results of risk-based assessment and cybersecurity assurance are presented in Table 6, Table 7, Table 8, Table 9, and Table 10.

Table 6. **Results of risk-based cybersecurity assessment of the SmolLM3 model**

#	T	V	A	E	Criticality		
					P	S	R
1	HC	SPRG	PH	I	0.51	4	2.04
2	CA	SPRG	PH	I	0.91	6	5.46
3	PH	SPRG	PH	I	0.85	10	8.5
4	EH	SPRG	PH	I	0.82	5	4.1
5	ID	SPRG	PH	I	0.82	9	7.38
6	WA	SPRG	PH	I	0.87	9	7.83
7	TC	SPRG	PH	I	0.85	8	6.8
8	IPI	SPRG	PH	I	0.6	6	3.6
9	F	SPRG	PH	I	0.87	8	6.96
10	D	SPRG	PH	I	0.78	2	1.56
11	AC	SPRG	PH	I	0.47	7	3.29
12	PA	SPRG	PH	I	0.6	1	0.6
13	PV	SPRG	PH	I	0.89	4	3.56
14	UP	SPRG	PH	I	0.4	2	0.8
15	GD	SPRG	PH	I	0.91	3	2.73

Source: **compiled by the authors**

According to the risk-based assessment and cybersecurity assurance of the SmolLM3 model, the following results were obtained:

- when using the most productive countermeasure (Self Defense) the following threats are in the low-risk zone - Generation of harmful content, Generation of content about economic harm, Generation of content about intellectual property infringement, Generation of disinformation, Generation of content about political activities, Generation of content about unauthorized practices, and Generation of content about government decisions. In the medium risk zone – Generation of content about cybercrime activities, Generation of content about illegal drugs, Generation of adult content, and Generation of content about privacy violations. And in the high-risk zone – Generation of content about physical harm, Generation of content about weapons activities, Generation of terrorist content, and Generation of content about fraud;
- when using the highest-rated countermeasure (BPE-dropout), the following threats are in the low-risk zone – Generation of harmful content, Generation of content about intellectual property infringement, Generation of disinformation, Generation of content about political activities, Generation of content about unauthorized practices, and Generation of content about government decisions. In the medium risk zone - Generation of content about cybercrime activities, Generation of content about economic harm, Generation of adult content, and Generation of content about privacy violations. And in the high-risk zone - Generation of content about physical harm, Generation of content about illegal drugs, Generation of content about weapons activities, Generation of terrorist content, and Generation of content about fraud.

Table 7. **Cyber risk criticality matrix before applying countermeasures**

Probability	**Severity**		
	Low (0.0 – 3.9)	Medium (4.0 – 6.9)	High (7.0 – 10.0)
Low (0.0 – 0.39)	🟩	🟩	🟨
Medium (0.40 – 0.69)	🟩 12, 14	🟨 1, 8	🟥 11
High (0.70 – 1.0)	🟨 10, 15	🟥 2, 4, 13	🟥 3, 5, 6, 7, 9

Source: **compiled by the authors**

Table 8. **Countermeasures rating matrix**

CM	Productivity	Efficiency	Cost	Rating
IC	8	1.95	2.1	12.05
ICD	8	1.6	4.2	13.8
SR	8	1.26	4.2	13.46
SD	10	0.76	2.3	13.06
BPE-d	8	1.41	5.27	14.68

Source: **compiled by the authors**

Table 9. **Cyber risk criticality matrix of most productive countermeasure (Self Defense)**

Probability	Severity		
	Low (0.0 – 3.9)	Medium (4.0 – 6.9)	High (7.0 – 10.0)
Low (0.0 – 0.39)	10, 12, 14	1, 4, 8	5, 11
Medium (0.40 – 0.69)	15	2, 13	3, 6, 7, 9
High (0.70 – 1.0)	🟨	🟥	🟥

Source: **compiled by the authors**

Table 10. **Cyber risk criticality matrix of highest-rated countermeasure (BPE-dropout)**

Probability	Severity		
	Low (0.0 – 3.9)	Medium (4.0 – 6.9)	High (7.0 – 10.0)
Low (0.0 – 0.39)	12, 14	1, 8	11
Medium (0.40 – 0.69)	10, 15	2, 4, 13	3, 5, 6, 7, 9
High (0.70 – 1.0)	🟨	🟥	🟥

Source: **compiled by the authors**

DISCUSSION OF RESULTS

The results of the analysis of existing studies in the field of evaluating and ensuring the cybersecurity of language models showed that the vast majority of them focus on the process of attacking models and verifying the positive impact of countermeasures on the ASR coefficient. For the most part, no known formalized methodologies are used in such analysis. Therefore, this study was devoted to assessing and ensuring the cybersecurity of LLMs in a more formal way using the risk-based IMECA method.

The importance of this work lies in combining the LLMs cybersecurity model with an improved method for analyzing the criticality of these models' vulnerabilities. As a result, it becomes possible to conduct a quantitative risk-based assessment of the cybersecurity of language models.

The main contribution of the study is to identify a set of countermeasures to LLMs vulnerabilities and their key indicators. This makes it possible to build a matrix of countermeasure ratings to determine the quantitative level of their impact on the unwanted generation of forbidden content and further ensure the cybersecurity of language models.

Thus, the result of this work is the adaptation of the IMECA method for the field of LLMs cybersecurity by analyzing the effects of attacks on vulnerabilities and selecting countermeasures, which allows ensuring an acceptable risk of cybersecurity within the existing constraints. Language models and their protection mechanisms are constantly evolving, so the next steps will be devoted to deepening the procedure for evaluating and ensuring the security of LLMs by increasing the variability of attacks on models and conducting additional experiments on attacking LLMs using different types of protective mechanisms.

CONCLUSIONS

The IMECA method of risk-based assessment and cybersecurity assurance was adapted to LLMs. This ensures higher completeness and reliability of cybersecurity assessments.

The set of countermeasures to LLMs vulnerabilities is formed, their indicators are defined, and selection criteria are developed based on the matrix of ratings of these indicators. This helps to ensure an acceptable level of cybersecurity risk in the context of existing constraints.

The test evaluation and cybersecurity assurance of the test model were performed using the IMECA analysis method. As a result, it was identified that the use of countermeasures has a positive impact on the model's risk level, but high-risk threats still exist.

The direction of future research on deepening the evaluation procedure and ensuring the security of LLMs by increasing the variability of attacks on models and conducting additional experiments on attacking LLMs using various types of protection mechanisms was substantiated. In addition, it would be advisable to adapt this assessment methodology to the language models used in combination with unmanned aerial vehicles (UAVs). LLMs perform tasks to ensure cooperation between swarms of UAVs [22], control UAVs in real time [23], and ensure the reliability of their missions [24]. Also use multifactorial criteria to balance security and other quality characteristics of LLM systems [25].

REFERENCES

1. Chu, Z., Wang, S., Xie, J., et al. "LLM agents for education: Advances and applications". *arXiv preprint*. 2025. DOI: https://doi.org/10.48550/arXiv.2503.11733.

2. Wang, W., Ma, Z., Wang, Z., Wu, C., Ji, J., Chen, W., Li, X. & Yuan, Y. "A survey of llm-based agents in medicine: How far are we from baymax?". *arXiv preprint*. 2025. DOI: https://doi.org/10.48550/arXiv.2502.11211.

3. Husein, R. A., Aburajouh, H. & Catal, C. "Large Language Models for code completion: A systematic literature review". *Computer Standards & Interfaces*. 2025; 92: 103917. Scopus Q1. DOI: https://doi.org/10.1016/j.csi.2024.103917.

4. Raza, M., Jahangir, Z., Riaz, M. B., Saeed, M. J. & Sattar, M. A. "Industrial applications of large language models". *Scientific Reports*. 2025; 15 (1): 13755. Scopus Q1. DOI: https://doi.org/10.1038/s41598-025-98483-1.

5. Neretin, O. & Kharchenko, V. "Method for criticality analysis of vulnerabilities in Large Language Models".

6. 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*. 2024. DOI: https://doi.org/10.48550/arXiv.2404.01318.

7. Shen, X., Chen, Z., Backes, M., Shen, Y. & Zhang, Y. ""Do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models". In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*. 2024. p. 1671–1685. DOI: https://doi.org/10.1145/3658644.3670388.

8. Mauri, L. & Damiani, E. "Modeling threats to AI-ML systems using STRIDE". *Sensors*. 2022; 22, (17): 6662. Scopus Q1. DOI: https://doi.org/10.3390/s22176662.

9. Zahid, F., Sewwandi, A., Brandon, L., Kumar, V. & Sinha, R. "Securing educational LLMs: A generalised taxonomy of attacks on LLMs and DREAD risk assessment". *High-Confidence Computing*. 2025. p. 100371. Scopus Q1. DOI: https://doi.org/10.1016/j.hcc.2025.100371.

10. Tete, S. B. "Threat modelling and risk analysis for large language model (llm)-powered applications". *arXiv preprint*. 2024. DOI: https://doi.org/10.48550/arXiv.2406.11007.

11. Neretin, O. & Kharchenko, V. "A model of ensuring LLM cybersecurity". *Radioelectronic and Computer Systems*. 2025; 2025 (2): 201–215. Scopus Q2. DOI: https://doi.org/10.32620/reks.2025.2.13.

12. Babeshko, I., Illiashenko, O., Kharchenko, V., & Leontiev, K. "Towards trustworthy safety assessment by providing expert and tool-based XMECA techniques". *Mathematics*. 2022; 10 (13): 2297. Scopus Q1. DOI: https://doi.org/10.3390/math10132297.

13. Aguilera-Martínez, F. & Berzal, F. "LLM Security: vulnerabilities, attacks, defenses, and countermeasures". *arXiv preprint*. 2025. DOI: https://doi.org/10.48550/arXiv.2505.01177.

14. Goren, G., Katz, S. & Wolf, L. "AlignTree: efficient defense against LLM jailbreak attacks". *arXiv preprint*. 2025. DOI: https://doi.org/10.48550/arXiv.2511.12217.

15. Zhang, Y., Ding, L., Zhang, L. & Tao, D. "Intention analysis makes llms a good jailbreak defender". *arXiv preprint*. 2024. DOI: https://doi.org/10.48550/arXiv.2401.06561.

16. Wei, Z., Wang, Y., Li, A., Mo, Y. & Wang, Y. "Jailbreak and guard aligned language models with only few in-context demonstrations". *arXiv preprint*. 2023. DOI: https://doi.org/10.48550/arXiv.2310.06387.

17. Xie, Y., Yi, J., Shao, J., Curl, J., Lyu, L., Chen, Q., Xie, X. & Wu, F. "Defending chatgpt against jailbreak attack via self-reminders". *Nature Machine Intelligence*. 2023; 5 (12): 1486–1496. Scopus Q1. DOI: https://doi.org/10.1038/s42256-023-00765-8.

18. Robey, A., Wong, E., Hassani, H. & Pappas, G. J. "Smoothllm: Defending large language models against jailbreaking attacks". *arXiv preprint*. 2023. DOI: https://doi.org/10.48550/arXiv.2310.03684.

19. Phute, M., Helbling, A., Hull, M., Peng, S., Szyller, S., Cornelius, C. & Chau, D. H. "LLM self defense: By self examination, llms know they are being tricked". *arXiv preprint*. 2023. DOI: https://doi.org/10.48550/arXiv.2308.07308.

20. Zeng, Y., Wu, Y., Zhang, X., Wang, H. & Wu, Q. "AutoDefense: Multi-agent llm defense against jailbreak attacks". *arXiv preprint*. 2024. DOI: https://doi.org/10.48550/arXiv.2403.04783.

21. Jain, N., Schwarzschild, A., Wen, Y., Somepalli, G., Kirchenbauer, J., Chiang, P.Y., Goldblum, M., Saha, A., Geiping, J. & Goldstein, T. "Baseline defenses for adversarial attacks against aligned language models". *arXiv preprint*. 2023. DOI: https://doi.org/10.48550/arXiv.2309.00614.

22. Song, H., Yang, Z., Du, H., Zhang, Y., Zeng, J. & He, X. "LLM-LCSA: LLM for collaborative control and decision optimization in UAV cluster security". *Drones*. 2025; 9 (11): 779. Scopus Q1. DOI: https://doi.org/10.3390/drones9110779.

23. Choutri, K., Fadloun, S., Khettabi, A., Lagha, M., Meshoul, S. & Fareh, R. "Leveraging Large Language models for real-time UAV control". *Electronics*. 2025; 14 (21): 4312. Scopus Q2. DOI: https://doi.org/10.3390/electronics14214312.

24. Sezgin, A. "Scenario-driven evaluation of autonomous agents: Integrating Large Language Model for UAV mission reliability". *Drones*. 2025; 9 (3): 213. Scopus Q1. DOI: https://doi.org/10.3390/drones9030213.

25. Fomin, O. & Krykun, V. “Assessment of the quality of neural network models based on a multifactorial information criterion”. *Herald of Advanced Information Technology*. 2024; 7 (1): 13–23. DOI: https://doi.org/10.15276/hait.07.2024.1.

Conflicts of Interest: The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship or other, which could influence the research and its results presented in this article

Received 30.12.2025
Received after revision 12.02.2026
Accepted 18.02.2026

DOI: https://doi.org/10.15276/hait.09.2026.05
УДК 004.89

IMECA метод ризик-орієнтованого оцінювання та забезпечення кібербезпеки великих мовних моделей

Неретін Олексій Сергійович[1)]
ORCID: https://orcid.org/0000-0003-2114-6714; o.s.neretin@csn.khai.edu. Scopus Author ID: 58099131000
Харченко Вячеслав Сергійович[1)]
ORCID: https://orcid.org/0000-0001-5352-077X; v.kharchenko@csn.khai.edu. Scopus Author ID: 22034616000
[1)] Національний аерокосмічний університет «Харківський авіаційний інститут», вул. Вадима Манька, 17. Харків, 61070, Україна

АНОТАЦІЯ

Великі Мовні Моделі (LLMs) допомагають виконувати складні завдання, які раніше покладалися виключно на людину. Багато різних сфер людської діяльності вже використовують цю технологію, або активно вивчають її можливості з ціллю майбутньої інтеграції у робочі процеси. Окрім позитивного ефекту при їх використанні є проблеми невизначеної та неочікуваної поведінки, зокрема, генерації забороненого контенту. Враховуючи розширення використання цих моделей та таку їх поведінку необхідним є визначення рівня захищеності та подальшого забезпечення кібербезпеки цієї технології. Об’єктом дослідження є процеси забезпечення кібербезпеки великих мовних моделей. В статті запропоновано методологію боротьби з цією загрозою за допомогою оцінювання ризиків такої поведінки та забезпечення прийнятного рівня кібербезпеки LLMs з використанням IMECA (Intrusion Modes Effects Criticality Analysis) техніки ризик-орієнтованого оцінювання. Сформовано множину контрзаходів для підвищення рівня захищеності LLMs, а також визначено процедури їх вибору за критеріями максимальної продуктивності та найкращого рейтингу з використанням матриці рейтингів контрзаходів. Надано приклад випробувального оцінювання та забезпечення кібербезпеки тестової мовної моделі, за результатами якого визначено, що рівень критичності кібер ризиків до застосування контрзаходів значно знижується після використання найбільш продуктивного та найкращого за рейтингом контрзаходів, але все ще залишаються загрози з високим рівнем критичності ризиків. Запропоновано напрями майбутніх досліджень щодо поглиблення процедури оцінювання та забезпечення безпеки LLMs з огляду на постійний розвиток цих моделей та механізмів їх захисту. Головний результат цієї роботи полягає у поєднанні моделі забезпечення кібербезпеки LLMs та удосконаленого методу аналізу критичності їх вразливостей для подальшої адаптації IMECA-методу кількісного ризик-орієнтованого оцінювання та забезпечення кібербезпеки для сфери LLMs.

Ключові слова: Ризик-орієнтоване оцінювання; кібербезпека; Великі Мовні Моделі; IMECA; контрзаходи, загроза; вразливість; атака

ABOUT THE AUTHORS

Oleksii S. Neretin - PhD Student, Department of Computer Systems, Networks and Cybersecurity. National Aerospace University “Kharkiv Aviation Institute”, 17, Vadym Manko Str. Kharkiv, 61070, Ukraine
ORCID: https://orcid.org/0000-0003-2114-6714; o.s.neretin@csn.khai.edu. Scopus Author ID: 58099131000
Research field: Computer science; Cybersecurity; Artificial Intelligence; Large Language Models

Неретін Олексій Сергійович - аспірант кафедри Комп’ютерних систем, мереж і кібербезпеки. Національний аерокосмічний університет «Харківський авіаційний інститут», вул. Вадима Манька, 17. Харків, 61070, Україна.

Vyacheslav S. Kharchenko - Doctor of Engineering Sciences, Professor, Corr. member of the National Academy of Science of Ukraine, Head of Department of Computer Systems, Networks and Cybersecurity. National Aerospace University “Kharkiv Aviation Institute”, 17, Vadym Manko Str. Kharkiv, 61070, Ukraine
ORCID: https://orcid.org/0000-0001-5352-077X; v.kharchenko@csn.khai.edu. Scopus Author ID: 22034616000
Research field: Big Safety and Security; Critical Infrastructure (NPPs, SMRs) Security and Resilience; UXV-based AI Systems for Dangerous Spaces; AI Quality, XAI as a Services, Dependable&Resilient AI Systems; AR&AI for Interactive Art

Харченко Вячеслав Сергійович - доктор технічних наук, професор, чл.-кор. НАН України, завідуючий кафедри Комп’ютерних систем, мереж і кібербезпеки. Національний аерокосмічний університет «Харківський авіаційний інститут», вул. Вадима Манька, 17. Харків, 61070, Україна