

Научная статья

УДК 004.056.5

<https://doi.org/10.37493/2307-910X.2026.2.5>

Разработка программного средства защиты генерации данных для машинного обучения

Алексей Витальевич Вечканов^{1*}, Александр Сергеевич Гиш²^{1,2} Северо-Кавказский федеральный университет (д. 1, ул. Пушкина, Ставрополь, 355017, Россия)¹ lexa184z@yandex.ru; <https://orcid.org/0009-0004-0811-7075>² alexandrgish@gmail.com; <https://orcid.org/0009-0000-5790-8675>

*Автор, ответственный за переписку

Аннотация. Введение. В статье рассматривается задача разработки программного средства защиты генерации данных для машинного обучения. Актуальность темы обусловлена тем, что синтетические данные всё чаще применяются для обучения и тестирования моделей, однако процесс их генерации может сопровождаться рисками раскрытия конфиденциальной информации, запоминания исходных записей генеративной моделью и нарушения требований к управлению данными. **Материалы и методы.** Для проведения исследования использованы теоретический анализ публикаций по синтетическим данным, дифференциальной приватности, управлению рисками искусственного интеллекта и защите данных, а также проектный подход к построению программного средства на основе модулей классификации признаков, приватизации, генерации, аудита и принятия решения о публикации. **Результаты и обсуждение.** Описана структура программного средства SynthGuard-ML, предназначенного для безопасной генерации данных для машинного обучения. Представлены уровни контроля защиты данных, общая схема работы средства, требования к защите данных и алгоритмы обработки исходного набора, формирования защищённого представления, генерации синтетических данных и проверки результата. **Заключение.** Полученные результаты показывают, что предложенное программное средство позволяет организовать защищённый конвейер генерации данных, при котором синтетический набор публикуется только после проверки приватности, полезности и соответствия заданной политике безопасности.

Ключевые слова: синтетические данные, машинное обучение, защита данных, дифференциальная приватность, генерация данных, информационная безопасность, аудит данных, программное средство

Для цитирования: Вечканов А. В., Гиш А. С. Разработка программного средства защиты генерации данных для машинного обучения // Современная наука и инновации. 2026. № 2. С. 61–68. <https://doi.org/10.37493/2307-910X.2026.2.5>

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Статья поступила в редакцию 01.04.2026;
одобрена после рецензирования 21.05.2026;
принята к публикации 21.06.2026.

Research article

Development of a software tool for protecting data generation for machine learning

Aleksey A. Vechkanov^{1*}, Alexander S. Gish²^{1,2} North Caucasus Federal University (1, Pushkin St., Stavropol, 355017, Russia)¹ lexa184z@yandex.ru; <https://orcid.org/0009-0004-0811-7075>² alexandrgish@gmail.com; <https://orcid.org/0009-0000-5790-8675>

*Corresponding author

© Вечканов А. В., Гиш А. С., 2026

Abstract. Introduction. The article considers the development of a software tool for protecting data generation for machine learning. The relevance of the topic is determined by the increasing use of synthetic data for training and testing machine learning models, while the generation process may involve risks of confidential information disclosure, memorization of source records by generative models, and violation of data governance requirements. **Materials and methods.** The study employed a theoretical analysis of publications on synthetic data, differential privacy, artificial intelligence risk management, and data protection, as well as a design-oriented approach to building a software tool based on modules for feature classification, privacy transformation, generation, auditing, and release control. **Results and discussion.** The structure of the SynthGuard-ML software tool for secure data generation is described. The paper presents the levels of data protection control, a general scheme of the tool operation, data protection requirements, and algorithms for processing the source dataset, forming a protected representation, generating synthetic data, and verifying the result. **Conclusion.** The obtained results demonstrate that the proposed software tool makes it possible to organize a protected data generation pipeline in which a synthetic dataset is released only after privacy, utility, and security policy compliance checks.

Keywords: synthetic data, machine learning, data protection, differential privacy, data generation, information security, data audit, software tool

For citation: Vechkanov AV, Gish AS. Development of a software tool for protecting data. *Modern Science and Innovations*. 2026;(2):61-68. <https://doi.org/10.37493/2307-910X.2026.2.5>

Conflict of Interest: the authors declare no conflicts of interest.

The article was submitted 01.04.2026;
approved after reviewing 21.05.2026;
accepted for publication 21.06.2026.

Introduction. The development of machine learning is largely determined by the availability of high-quality data. The more complex the model, the higher the requirements for the volume, diversity, and representativeness of the training sample. However, real-world data often contains personal, medical, financial, commercial, and other sensitive information, so its free use and transfer between organizations are limited. In such circumstances, synthetic data, which is created algorithmically and mimics the statistical properties of the original datasets, is becoming increasingly important [1].

Synthetic data reduces model developers' dependence on proprietary information sources. They can be used for testing software systems, training classifiers, modeling rare events, and expanding small samples. However, the synthetic nature of the data does not in itself guarantee complete security. A generative model can remember individual records, reproduce rare combinations of features, or preserve dependencies that allow indirect reconstruction of information about the original objects [2, 3].

This problem is particularly significant when using generative models in areas with increased data protection requirements. These areas include healthcare, banking, government information systems, insurance, and industrial analytics. In these areas, it is important not only to create a useful synthetic dataset but also to prove that its publication does not lead to the disclosure of sensitive information. Modern approaches to AI risk management also require consideration of transparency, traceability, data quality, and model lifecycle management [6, 7].

Materials and methods. The study employed a theoretical analysis of publications on synthetic data, differential privacy, artificial intelligence risk management, and data protection, as well as a design-oriented approach to building a software tool based on modules for feature classification, privacy transformation, generation, auditing, and release control.

Results and discussion. This paper examines the development of SynthGuard-ML, a software tool designed to protect the process of generating machine learning data. Unlike a conventional generator, this tool not only creates a synthetic dataset but also pre-assesses the source data, applies protective transformations, audits the result, and decides whether it can be published. This approach allows data generation to be viewed as a controlled and monitored information security process.

Basics control protection data generation / Basics of data generation protection control. Data generation security controls are a set of methods aimed at preventing the disclosure of

sensitive information during the creation of synthetic datasets. Unlike conventional data preprocessing, security controls must cover the entire generation lifecycle: from loading the source dataset to unloading the synthetic result. This is because leaks can occur not only in the source file, but also at the level of the trained model, generation parameters, API requests, or the final dataset [4].

In the proposed software, the original dataset is not transmitted directly to the user. Instead, it enters a secure processing loop, where feature classification is performed and the sensitivity level of each field is determined. A security policy is then generated, defining acceptable privacy parameters, the quality of the synthetic dataset, and the publication conditions. The data is then transformed into a secure representation, which is used to train a generative model. The resulting synthetic dataset undergoes an audit, including verification of the risk of record ownership, similarity to the original objects, and usefulness for subsequent machine learning tasks [3, 5]. The general flowchart of the data generation security software is shown in Figure 1:

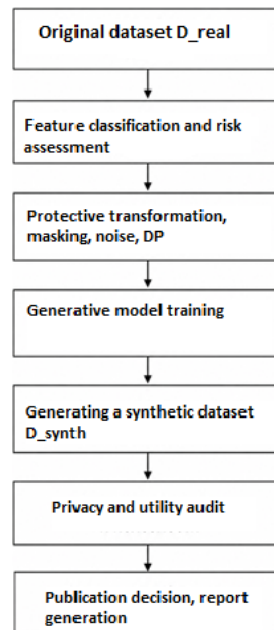


Figure 1. General diagram of the operation of the data generation protection software

SynthGuard-ML is based on a modular model. The central element is not the generative model itself, but the control pipeline, which determines which data can be used, what transformations need to be applied, and whether the result can be delivered to the user. This approach is consistent with the modern understanding of synthetic data as a technology that requires not only generation but also subsequent risk verification [1, 2]. We consider the main levels of data protection control for generating data in Table 1:

Table 1. Generation data protection control levels

Level of control	Object of control	Potential risk	Method of protection	Check result
Incoming inspection	Original dataset	Availability of direct identifiers	Identification and removal of sensitive fields	A list of protected characteristics has been compiled
Statistical control	Distribution of features	Saving rare combinations	Category aggregation and outlier suppression	Reduced risk of reidentification
Model control	Generative model	Remembering training notes	Regularization and differential privacy	The impact of a single entry is limited
Attacking control	Synthetic set	Attack of Belonging	Checking membership inference	The risk of disclosure of educational objects has been assessed
Utility control	Synthetic data	Loss of educational value	Comparison of statistics and ML metrics	Data suitability confirmed
Publication control	The final result	Unloading an unsafe set	Automatic policy decision	Is publication permitted or prohibited?

As the table shows, protecting data generation isn't limited to removing identifiers. In practice, it's necessary to evaluate not only the original features but also the properties of the synthetic output. If the synthetic dataset is too close to the original, it may be useful for training models but pose a privacy risk. If protection is overly strong, the data becomes more secure but may lose its practical value. Therefore, the software must maintain a balance between privacy and usefulness.

An example of how the tool works is to transform the original D_{real} dataset into a protected representation $D_{\text{protected}}$, and then into a synthetic D_{synth} dataset. A sensitivity level is determined for each feature, after which a protection policy P is generated, including an acceptable risk level, privacy parameters, and a minimum acceptable utility. The resulting dataset is approved for publication only if the privacy risk does not exceed the specified threshold, and the utility remains sufficient for solving machine learning problems.

Data protection requirements. The effectiveness of a data generation security software solution is determined by how fully it addresses the requirements for confidentiality, integrity, manageability, and auditability of the process. First and foremost, the solution must prevent direct user access to the source data. Actual data must be processed only within a secure environment, and all operations with it must be logged. This reduces the risk of unauthorized copying of source files and ensures subsequent auditability of actions.

An important requirement is the automatic detection of sensitive features. In tabular datasets, such features may include last name, first name, address, telephone number, date of birth, diagnosis, account number, geolocation, and other information that can be used to identify a specific subject. However, it's not only direct identifiers that pose a risk. Indirect features, such as age, region, profession, and rare diagnoses, when used together, can also lead to duplicate identification. Therefore, the software must consider both individual fields and their combinations [3].

The next requirement is support for protective transformations. These include masking, pseudonymization, aggregation, rare value suppression, statistical noise addition, and differential privacy. Protective methods must be configured through a security policy, as the same level of protection cannot be applied equally to medical, financial, educational, and technical data. For more sensitive datasets, stricter privacy parameters must be used.

Protecting the generative model is particularly important. If the model is trained on a closed dataset, it can become a source of leakage in its own right. In particular, when overfitting, the generator can reproduce individual records or create synthetic objects that are almost identical to real ones. Therefore, the software must include training constraints, overfitting controls, and verification of the result using attack scenarios. One such scenario is a membership attack, which tests whether it is possible to determine whether a specific record was included in the training set [5].

Equally important is the requirement for access control. System users must have distinct roles. The administrator manages policies and settings, the researcher initiates generation and obtains authorized results, and the auditor analyzes reports and logs. This separation reduces the likelihood of erroneous or unauthorized data publication. Furthermore, the tool must limit the number of API requests, as an excessive number of synthetic samples can increase the risk of statistical analysis and indirect reconstruction of the original data.

Reporting requirements are also mandatory. Each synthetic dataset must be accompanied by a report outlining the generation parameters, protection methods applied, privacy check results, utility assessment, and the final decision on publication. Such a report makes the generation process reproducible and verifiable. This is particularly important in the context of modern approaches to artificial intelligence risk management, which emphasize the need for documentation, transparency, and control throughout the lifecycle of AI systems [6, 7, 9].

Therefore, a software solution for protecting data generation must ensure not only the technical processing of the dataset but also the organization of a controlled process. Its purpose is to combine the generative model, privatization methods, auditing, logging, and access control into a single system suitable for practical use.

Algorithm of the software operation. To implement the discussed principles in practice, the paper proposes the SynthGuard-ML software tool. Its operation is based on the sequential execution of data analysis, protection, generation, and verification operations. The input is the original dataset D_{real} and the protection policy P . The output is the synthetic dataset D_{synth} or a rejection of publication if the result does not meet security requirements.

The first stage involves analyzing the source dataset. The system determines the data structure, feature types, the presence of direct and indirect identifiers, and rare combinations of values. This procedure is necessary to avoid passing information to the generator that could lead to the reconstruction of the original objects. The initial analysis procedure is presented in Algorithm 1.

Algorithm 1 Initial analysis of the original dataset

Require: Initial set D_{real}

Ensure: Description of meta features, risk score

```

1: meta  $\leftarrow$  empty feature description
2: risk_score  $\leftarrow$  0
3: Get the feature list from  $D_{\text{real}}$ 
4: for all feature  $\in D_{\text{real}}$  do
5: Determine the feature type
6: Check for direct identifiers
7: Check the participation of the feature in rare combinations
8: Save feature information in meta
9: Update the risk_score value
10: end for
11: Display "Primary data analysis completed"
12: return meta, risk_score
  
```

This algorithm generates a description of the dataset and identifies which features require mandatory protection. If the risk value is high, the system must apply more stringent transformations. Importantly, this decision is made before training the generative model, meaning potentially dangerous features are not fed into the generator without prior processing.

After the initial analysis, a secure data representation is generated. At this stage, direct identifiers are removed or replaced, numerical features can be given controlled noise, rare categories are merged, and sensitive fields are generalized. The corresponding procedure is presented in Algorithm 2.

Algorithm 2 Formation of a secure data representation

Require: Source set D_{real} , feature description meta, security policy P

Ensure: Protected set $D_{\text{protected}}$

```

1:  $D_{\text{protected}} \leftarrow$  copy of  $D_{\text{real}}$ 
2: for all feature  $\in$  meta do
3: If feature is a direct identifier, remove or mask feature
4: If feature is a sensitive feature, apply generalization
5: If a feature contains rare values, combine them into a larger category
6: If the policy requires adding noise, apply a noise transform
7: end for
8: Apply differential privacy parameters from  $P$ 
9: Display "Protected data representation generated"
10: return  $D_{\text{protected}}$ 
  
```

The result of the second algorithm is a $D_{\text{protected}}$ dataset, which preserves the key statistical properties of the original data but reduces the risk of disclosing sensitive information. This representation is used to train a generative model. Depending on the data type, the generator can use a tabular data model, a variational autoencoder, or a diffusion model. This article discusses the

software at the general architecture level, so a specific model can be replaced without changing the protection logic.

The next step is generating a synthetic set. The system must respect security policy constraints and prevent the creation of records that are too close to the original objects. The generation procedure is presented in Algorithm 3.

Algorithm 3 Generating a synthetic dataset

Require: Protected set $D_{\text{protected}}$, protection policy P , required volume n

Ensure: Synthetic set D_{synth}

```

1: Initialize the generative model  $G$ 
2: Train  $G$  on the set  $D_{\text{protected}}$ 
3:  $D_{\text{synth}} \leftarrow$  empty dataset
4: while size  $D_{\text{synth}} < n$  do
5:   Generate a new  $x_{\text{synth}}$  entry
6:   Check for absence of direct identifiers in  $x_{\text{synth}}$ 
7:   Check the distance  $x_{\text{synth}}$  to the nearest  $D_{\text{protected}}$  records
8:   If the entry passes the test, add  $x_{\text{synth}}$  to  $D_{\text{synth}}$ 
9: end while
10: Print "Synthetic dataset generated"
11: return  $D_{\text{synth}}$ 
  
```

After generation, synthetic data should not be immediately transmitted to the user. An audit is required to determine the security and usefulness of the resulting dataset. Privacy testing may include assessing the similarity of synthetic and real records, a membership attack, checking for rare combinations, and analyzing the possibility of linking to external sources. Utility testing can be performed by comparing statistical distributions and training a test model on the synthetic dataset [2, 5]. The corresponding procedure is presented in Algorithm 4.

Algorithm 4 Audit of a synthetic set and decision making

Require: Source set D_{real} , synthetic set D_{synth} , protection policy P

Ensure: Decision decision, report report

```

1: report  $\leftarrow$  empty report
2: privacy_risk  $\leftarrow$  rate risk privacy  $D_{\text{synth}}$ 
3: utility_score  $\leftarrow$  evaluate the utility of  $D_{\text{synth}}$ 
4: mia_score  $\leftarrow$  perform an affiliation attack check
5: Save privacy_risk, utility_score, and mia_score to report
6: If privacy_risk  $\leq P.\text{risk\_max}$  and utility_score  $\geq P.\text{utility\_min}$ , then
7:   decision  $\leftarrow$  "Publication permitted"
8: Otherwise
9:   Decision  $\leftarrow$  "Publication prohibited"
10: end if
11: Save report to external file
12: Display "Synthetic data audit completed"
13: return decision, report
  
```

The presented algorithm completes the software's workflow. If the synthetic dataset meets the policy requirements, it can be transmitted to the user via the API or stored in secure storage. If at least one of the criteria is not met, publication is blocked. In this case, the user receives a report indicating the reason for the rejection, such as excessive privacy risk, insufficient usefulness, or the detection of records too close to the original data.

The practical value of the proposed approach lies in its ability to make the data generation process manageable. The user receives not just a synthetic dataset, but a result verified according to specified criteria. The administrator can modify the security policy, the auditor can control reports, and the researcher can use approved datasets to train machine learning models.

However, the proposed software tool has limitations. The quality of the synthetic data depends on the quality of the source dataset and the chosen generative model. If the source data contains biases, the synthetic dataset may also reproduce them. Furthermore, overly strict privacy parameters can reduce the usefulness of the data for training models. Therefore, when using SynthGuard-ML in practice, it is necessary to configure a security policy based on the subject area, data sensitivity, and acceptable risk level.

Conclusion. This paper examines the development of a software tool for protecting data generation for machine learning. It demonstrates that synthetic data can be used as a tool for reducing dependence on closed source datasets; however, its use requires mandatory controls for privacy, usefulness, and compliance with information security requirements. A generative model can not only create new records but also preserve unwanted dependencies, rare combinations, and fragments of the original data, so the generation process must be considered a protected entity.

The proposed software tool, SynthGuard-ML, is a modular pipeline that includes analysis of the source dataset, formation of a secure representation, training of a generative model, creation of synthetic data, auditing of the result, and making a publication decision. The paper describes the security control levels, requirements for the tool, and its operating algorithms. Particular attention is paid to the fact that publication of the synthetic dataset should only be performed after assessing the privacy risk and confirming its sufficient usefulness for machine learning tasks.

The practical significance of the proposed solution lies in its applicability to organizations that work with sensitive data but require training sets for model development and testing. SynthGuard-ML improves the manageability of data generation, ensures traceability of actions, and reduces the risk of confidential information disclosure. Further development of this solution may include support for federated learning, trusted execution environments, automatic bias estimation, and integration with corporate access management systems.

Список источников

1. Jordon J., Szpruch L., Houssiau F., Bottarelli M., Cherubin G., Maple C., Cohen S. N., Weller A. Synthetic Data – what, why and how? ArXiv preprint. 2022. arXiv:2205.03257. [Электронный ресурс]. URL: <https://arxiv.org/abs/2205.03257> (дата обращения: 01.04.2026).
2. Houssiau F., Cohen S. N., Szpruch L., Daniel O., Lawrence M. G., Mitra R., Wilde H., Mole C. A Framework for Auditable Synthetic Data Generation. ArXiv preprint. 2022. arXiv:2211.11540. [Электронный ресурс]. URL: <https://arxiv.org/abs/2211.11540> (дата обращения: 01.04.2026).
3. Giomi M., Boenisch F., Wehmeyer C., Tasnádi B. A Unified Framework for Quantifying Privacy Risk in Synthetic Data. ArXiv preprint. 2022. arXiv:2211.10459. [Электронный ресурс]. URL: <https://arxiv.org/abs/2211.10459> (дата обращения: 01.04.2026).
4. Hassan C., Salomone R., Mengersen K. Deep Generative Models, Synthetic Tabular Data, and Differential Privacy: An Overview and Synthesis. ArXiv preprint. 2023. arXiv:2307.15424. [Электронный ресурс]. URL: <https://arxiv.org/abs/2307.15424> (дата обращения: 01.04.2026).
5. van Breugel B., Sun H., Qian Z., van der Schaar M. Membership Inference Attacks against Synthetic Data through Overfitting Detection. ArXiv preprint. 2023. arXiv:2302.12580. [Электронный ресурс]. URL: <https://arxiv.org/abs/2302.12580> (дата обращения: 01.04.2026).
6. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework (AI RMF 1.0). 2023. [Электронный ресурс]. URL: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=958388 (дата обращения: 01.04.2026).
7. ISO/IEC 42001:2023. Information technology – Artificial intelligence – Management system. 2023. [Электронный ресурс]. URL: <https://www.iso.org/standard/42001> (дата обращения: 01.04.2026).
8. Regulation (EU) 2024/1689 of the European Parliament and of the Council laying down harmonised rules on artificial intelligence. Artificial Intelligence Act. 2024. [Электронный ресурс]. URL: <https://eur-lex.europa.eu/eli/reg/2024/1689/oj> (дата обращения: 01.04.2026).
9. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile. NIST AI 600-1. 2024. [Электронный ресурс]. URL: <https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf> (дата обращения: 01.04.2026).
10. OWASP Foundation. OWASP Top 10 for Large Language Model Applications. Version 2025. 2025. [Электронный ресурс]. URL: <https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/> (дата обращения: 01.04.2026).

References

1. Jordon J, Szpruch L, Houssiau F, Bottarelli M, Cherubin G, Maple C, Cohen SN, Weller A. Synthetic Data – what, why and how? ArXiv preprint. 2022. arXiv:2205.03257. Available from: <https://arxiv.org/abs/2205.03257> [Accessed 1 Apr 2026].
2. Houssiau F, Cohen SN, Szpruch L, Daniel O, Lawrence MG, Mitra R, Wilde H, Mole C. A Framework for Auditable Synthetic Data Generation. ArXiv preprint. 2022. arXiv:2211.11540. Available from: <https://arxiv.org/abs/2211.11540> [Accessed 1 Apr 2026].
3. Giomi M, Boenisch F, Wehmeyer C, Tasnádi B. A Unified Framework for Quantifying Privacy Risk in Synthetic Data. ArXiv preprint. 2022. arXiv:2211.10459. Available from: <https://arxiv.org/abs/2211.10459> [Accessed 1 Apr 2026].
4. Hassan C, Salomone R, Mengersen K. Deep Generative Models, Synthetic Tabular Data, and Differential Privacy: An Overview and Synthesis. ArXiv preprint. 2023. arXiv:2307.15424. Available from: <https://arxiv.org/abs/2307.15424> [Accessed 1 Apr 2026].
5. van Breugel B, Sun H, Qian Z, van der Schaar M. Membership Inference Attacks against Synthetic Data through Overfitting Detection. ArXiv preprint. 2023. arXiv:2302.12580. Available from: <https://arxiv.org/abs/2302.12580> [Accessed 1 Apr 2026].
6. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework (AI RMF 1.0). 2023. Available from: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=958388 [Accessed 1 Apr 2026].
7. ISO/IEC 42001:2023. Information technology – Artificial intelligence – Management system. 2023. Available from: <https://www.iso.org/standard/42001> [Accessed 1 Apr 2026].
8. Regulation (EU) 2024/1689 of the European Parliament and of the Council laying down harmonised rules on artificial intelligence. Artificial Intelligence Act. 2024. Available from: <https://eur-lex.europa.eu/eli/reg/2024/1689/oj> [Accessed 1 Apr 2026].
9. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile. NIST AI 600-1. 2024. Available from: <https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf> [Accessed 1 Apr 2026].
10. OWASP Foundation. OWASP Top 10 for Large Language Model Applications. Version 2025. 2025. Available from: <https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/> [Accessed 1 Apr 2026].

Информация об авторах

Алексей Витальевич Вечканов – студент, Северо-Кавказский федеральный университет, Ставрополь.
Александр Сергеевич Гиш – студент, Северо-Кавказский федеральный университет, Ставрополь.

Вклад авторов: все авторы внесли равный вклад в подготовку публикации.

Information about the authors

Aleksey V. Vechkanov – student, North-Caucasus Federal University, Stavropol.
Alexander S. Gish – student, North-Caucasus Federal University, Stavropol.

Author contributions: the authors contributed equally to this article.