
RESULTS OF ARTIFICIAL INTELLIGENCE RESEARCH CENTERS

Trusted Artificial Intelligence: Challenges and Promising Solutions

D. Yu. Turdakov^{a,d,f,*}, Academician of the RAS A. I. Avetisyan^{a,d,e,f}, K. V. Arkhipenko^a, A. V. Antsiferova^a,
D. S. Vatolin^d, S. S. Volkov^a, A. V. Gasnikov^{a,e}, D. A. Devyatkin^{d,g}, M. D. Drobyshevsky^{a,e},
A. P. Kovalenko^a, M. I. Krivososov^a, N. V. Lukashevich^d, V. A. Malykh^e, S. I. Nikolenko^f,
I. V. Oseledets^{b,c}, A. I. Perminov^{a,d}, I. V. Sochenkov^{b,g}, M. M. Tikhomirov^d,
A. N. Fedotov^d, and M. Yu. Khachay^h

Received October 28, 2022; revised October 29, 2022; accepted November 1, 2022

Abstract—Wide applications of artificial intelligence technologies have led to new threats that cannot be effectively addressed using current tools for secure software development. To meet the challenge, the Research Center for Trusted Artificial Intelligence based on the Institute for Systems Analysis of the Russian Academy of Sciences was founded within the federal project “Artificial Intelligence” in 2021. Its objectives are the creation of scientific and technological basis for building trust in AI technologies. This paper discusses the risks and threats of applying artificial intelligence technologies, and describes the research directions and intermediate results produced at the Research Center for Trusted Artificial Intelligence.

Keywords: trusted artificial intelligence, attacks on machine learning, explainable artificial intelligence, trusted intelligent systems

DOI: 10.1134/S1064562422060205

1. INTRODUCTION

Modern intelligent systems use a variety of technologies consisting of artificial intelligence (AI) methods and algorithms, machine learning frameworks (e.g., TensorFlow, PyTorch), and infrastructure solutions for their support (cloud systems, specialized hardware systems, etc.). Building up trust in such systems is a long-term challenge, and a solution has been actively sought by the world community for several years. However, now there is no scientific and techno-

logical basis for the development of highly reliable trusted and simultaneously efficient systems using AI technologies (intelligent systems), including tools for searching for new types of vulnerabilities and counteracting new types of threats specific to these technologies.

A solution to this challenge is sought in two opposite directions. The first is the development of requirements for intelligent systems and the development of state standards [1, 2].

The second direction is the creation of a scientific and technological basis for supporting the emerging standards. Without creating tools ensuring the safety of functioning intelligent systems, the developed and prospective standards cannot be implemented in full. Work in this direction is carried out at the Research Center for Trusted Artificial Intelligence of the Institute for Systems Analysis of the Russian Academy of Sciences.

The program of the Center consists of the following key research directions (Fig. 1):

- the study of the structure of open machine learning (ML) frameworks, analysis of their source codes, and the creation of trusted ML frameworks;
- research and development of software tools for analyzing, detecting, and counteracting threats specific to AI technologies (AIT), including the following three sections:

^a *Ivannikov Institute for System Programming, Russian Academy of Sciences, Moscow, Russia*

^b *Skolkovo Institute of Science and Technology, Moscow, Russia*

^c *Artificial Intelligence Research Institute, Moscow, Russia*

^d *Lomonosov Moscow State University, Moscow, Russia*

^e *Moscow Institute of Physics and Technology (National Research University), Dolgoprudnyi, Moscow oblast, Russia*

^f *National Research University Higher School of Economics, Moscow, Russia*

^g *Institute for Systems Analysis, Federal Research Center “Computer Science and Control,” Russian Academy of Sciences, Moscow, Russia*

^h *Krasovskii Institute of Mathematics and Mechanics, Ural Branch, Russian Academy of Sciences, Yekaterinburg, Russia*

*e-mail: turdakov@ispras.ru

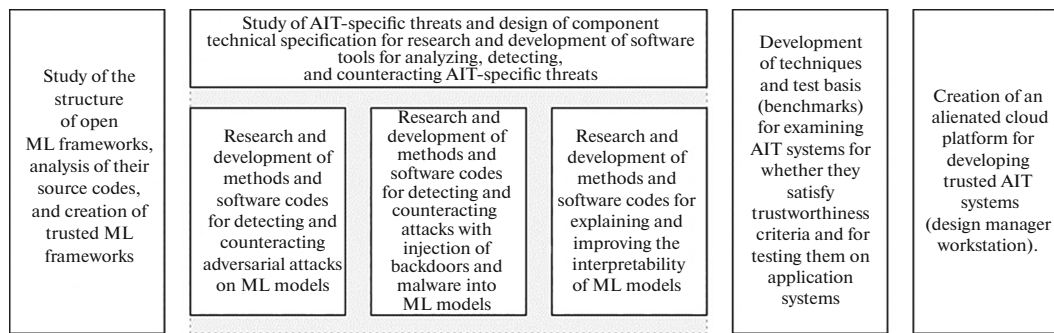


Fig. 1. Directions of study of the Center.

- o research and development of methods and software codes for detecting and counteracting adversarial attacks on ML models;

- o research and development of methods and software codes for detecting and counteracting attacks with injection of backdoors and malware into ML models;

- o research and development of methods and software codes for explaining and improving the interpretability of ML models;

- development of techniques and a test basis (benchmarks) for examining whether intelligent systems satisfy trustworthiness criteria and for testing them on application systems;

- creation of an alienated cloud platform for developing trusted AI systems.

The results of the current studies and developments in each direction are discussed in the next sections.

2. TRUSTED MACHINE LEARNING FRAMEWORKS

Libraries and ML frameworks are used in large numbers for productive development of intelligent systems. These software tools significantly facilitate the creation of application products. However, as any software code, they may contain bugs and undocumented features at the level of the source code or software dependencies. Such vulnerabilities can be used by a malefactor to attack the target system. Thus, overall trust in intelligent systems cannot be created without building trust in these key system components.

The development of trusted industrial software is regulated by the secure development lifecycle (SDL). An important aspect of SDL practices is that not only developed components, but also open source software involved have to be analyzed. The basic analysis methods used in SDL are static and dynamic analysis (fuzzing). With the help of these techniques, it is possible to detect critical defects in software and then eliminate them prior to product release.

The Center was tasked with developing trusted versions of popular frameworks, such as Tensorflow and

PyTorch. The attack surface of these frameworks was analyzed. A promising attack vector was chosen to be the model load component, since trained models can often be taken from an untrusted source. Additionally, for TensorFlow, fuzzing targets were recovered in the OSS-Fuzz project, which ensures continuous fuzzing for open source software [3].

The frameworks were fuzzed using the Sydr tool [4]. As a result, seven errors for PyTorch [5, 6] and one error for TensorFlow [7] were revealed. The source codes of the frameworks were also analyzed using the Svmc static analyzer [8]. As a result, 13 errors were found in PyTorch [9] and 8 errors, in TensorFlow [10].

The found errors were reported to the developer community, and suggestions on the correction of some of them were made. For further use, all corrections were collected in trusted framework versions based on PyTorch v1.11.0 and TensorFlow v2.8.2.

Note that the frameworks are still under development and new versions appear. Accordingly, constant checks of new code modifications and synchronization of trusted versions with the original repositories are required. To facilitate this continuous process, a hardware-software infrastructure combining SDL instruments with tools for their automated application has been developed for building trust in baseline ML frameworks.

3. THREATS SPECIFIC TO INTELLIGENT SYSTEMS AND COUNTERACTION METHODS

A central issue of the development of trusted intelligence systems is that mappings learned by ML models are unstable to variations in input data. Even a minor modification of the input, for example, adding visually unseen noise to an image, can significantly change the model prediction on the new object, although it is nearly indistinguishable from the original one both visually and in terms of similarity metrics. This phenomenon has given rise to so-called adversarial attacks. The relevance of methods for counteracting adversarial attacks on ML models was confirmed in experiments carried out at the Center

with models and datasets from scientific partners (Innopolis University, Lobachevsky University of Nizhny Novgorod). A successful white box attack was performed on a system of X-ray picture segmentation and on models for human age prediction from gene expression data based on decision tree boosting.

Approaches to the design of attack-robust models are divided into two classes. In the first, the model itself is modified (smoothed), which leads to higher empirical robustness. It is possible to obtain a theoretically guaranteed attack size for which model predictions do not change. Within the work of the Center of Trusted AI, a universal probabilistic approach to the design of models with certified robustness based on Chernoff–Cramér bounds was proposed for the first time by researchers from Skoltech [11]. By applying this approach, the probability of a model to fail can be formally estimated for an attack sampled from a certain distribution. The theoretical findings of [11] are supported by experimental results on different datasets.

In the second approach, the learning method changes. Studies in this direction concerning optimization problems under (adversarial) noises were conducted at the Center. Specifically, black-box optimization models in which attacks are modeled by small noise added to the objective function value were investigated [12]. A general scheme for reducing the convex gradient-free optimization problem to a convex smooth optimization one with a stochastic gradient oracle was proposed. For the first time, it was shown that a method that is optimal in terms of the number of oracle calls (the number of objective function evaluations) is also optimal in terms of the number of successive iterations and, most importantly, that this method is most sensitive to the inaccuracy in the function value, as suggested by evidence. Thus, an approach was proposed that is optimal in terms of all three criteria (namely, the number of oracle calls, maximum parallelism, and the maximum admissible level of noise). Additionally, difficulties associated with formalizing attacks on optimization problem parameters were overcome in such a way that the original optimization problem was eventually replaced by a saddle-point problem. Approaches to the solution of such optimization problems with proved optimality were proposed [13, 14].

Additionally, researchers of the Center tested an application approach for gaining robustness against adversarial attacks, namely, adversarial training of models, i.e., training on adversarially perturbed data, as well as data augmentation methods against adversarial attacks were also tested [15]. A new neural network architectures that are more robust against adversarial attacks with standard forms of perturbations were also developed [16].

4. IMPROVEMENT OF MODEL INTERPRETABILITY

Neural networks are widely used as powerful modeling tools and are included in most major commercial software products for data mining. Modeling, however, is only part of the data mining process. Additionally, the influence of input variables on model results has to be analyzed. Low-quality interpretation results produced by a model can be used in malicious attacks on the target system, for example, via data poisoning. Thus, building trust in intelligent systems requires the interpretability of application models.

As part of its work, the Center was tasked with improving the interpretability of results and detecting incorrect results produced by multilayered neural networks. A test bench for visualizing a fully connected neural network with sublinear activation functions for the binary classification problem in a two-dimensional feature space was developed. The test bench was used to analyze the geometric interpretation of a trained neural network model with each neuron assigned to a separating hyperplane in the original feature space. On this basis, several new methods for model analysis were proposed. For an input example, k nearest neighbors in the training set from the model's point of view are presented as an interpretation of the network solution. A method is proposed for constructing a decision tree imitating the work of the original neural network such that the leaf nodes of the tree correspond to the mentioned sectors or their unions. Statistical analysis in them can show, for example, whether the network solution in a given domain is reliable or a definite decision cannot be made. The proposed approaches can be used in fully connected neural networks with arbitrary numbers of neurons and layers. For convolution neural networks, a method for recovering interpolated images from points of feature space was proposed. Additionally, analysis methods for filters for determining a given class of images are under development.

Two computationally efficient methods for uncertainty estimation of multilayered neural networks based on the Transformer architecture were created in [17]. According to the first method, whether analyzed objects belong to certain classes is estimated using an ensemble of neural networks, each with masked individual weights of the output layer. The second method is based on estimating the distance between the latent embeddings of analyzed objects and the nearest objects from the training set. Experiments on text classification tasks and named entity recognition showed that the proposed methods significantly improve the reliability of misclassification detection as compared with more computationally intensive methods.

Additionally, an approach for obtaining semantically interpretable categories for embeddings based on WordNet semantic networks was examined. Semantic

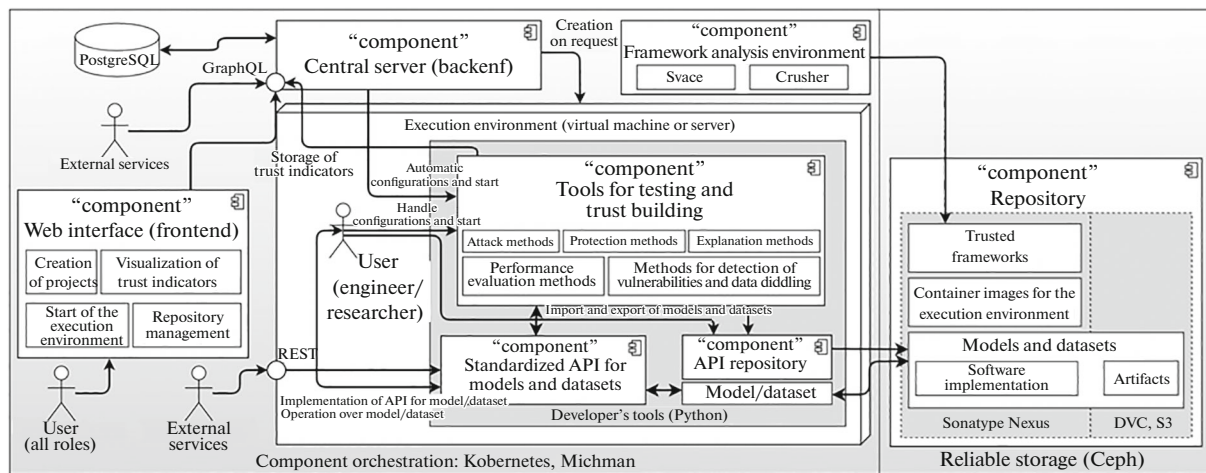


Fig. 2. High-level architecture of the Platform.

classes (superconcepts) combining sets of words were used as interpretable dimensions.

5. DEVELOPMENT OF TECHNIQUES AND A TEST BASIS (BENCHMARKS) FOR TRUSTWORTHINESS ASSESSMENT IN APPLICATION SYSTEMS

The presence or absence of a malefactor, scenarios of malefactor influence on life cycle processes in application intelligent systems, and the variety of real-world tasks and intelligent systems have to be carefully analyzed and systematized. Relying on such an analysis, we have developed trustworthiness criteria for intelligent systems, which we intend to test on actual intelligent systems in the future.

Note that trust in intelligent application systems cannot be considered separately from the efficiency of these systems. For example, the development of an ML model that is invulnerable, for example, to adversarial attacks is a trivial task if minimum requirements for the accuracy and recognition rate of this model are specified. Thus, an important task is to create a test basis consisting of datasets and models involving real-world tasks from various application areas and to develop techniques for examining whether intelligent systems and their components satisfy the trustworthiness and efficiency requirements.

Benchmarks have been developed for solving real-world tasks involving data of different types, such as text, images, video, graphs, and tables. As benchmarks, simple (baseline) models were used and state-of-the-art solutions were created, on which the efficiency of trust building methods can be demonstrated under actual conditions.

Specifically, a new procedure for creating artificial motion for training more stable neural network algorithms intended for video processing was developed. The proposed procedure was used to train a neural

network algorithm for semantic matting of videos with people [18]. The emphasis in the algorithm was placed on the temporal stability of results. By applying the proposed procedure, the size of the training set was increased significantly and the stability of the matting method to different types of motion was enhanced.

A novel single-pass method inspired by object detection algorithms from computer vision was developed for open information extraction from texts [19]. The approach uses an order-agnostic loss based on bipartite matching that forces unique predictions and a Transformer-based encoder-only architecture for sequence labeling. The proposed approach demonstrates superior or similar performance as compared with state-of-the-art models on standard benchmarks in terms of both quality metrics and inference time.

6. CLOUD PLATFORM FOR DEVELOPING TRUSTED INTELLIGENT SYSTEMS

Based on the results of conducted research, we have developed the concept of a cloud platform for developing intelligent systems. The platform will

- contain tools for analyzing datasets and ML models, including explanation of models, their attack robustness assessment, and performance evaluation;
- store trusted models, algorithms, and datasets for training models and carrying out experiments (benchmarks);
- provide life cycle components for developing secure software.

Modern ML methods require a large amount of computational resources at the stage of model training. At the other stages, these resources may be used inefficiently. The cloud computing model resolves this problem by providing access to computational resources on request. Accordingly, the Platform should be deployed in public and private clouds. The

performance of the Platform, together with a cloud architecture, will be demonstrated as applied to the Asperitas cloud platform and the Michman orchestrator [20], which deploys the necessary software components on request.

Another key requirement for the Platform is extensibility. As new software tools for analyzing and implementing attacks and defense appear constantly, it will be possible to add them to the Platform. Specifically, software interfaces (Fig. 2) and techniques for adding them are under development.

7. CONCLUSIONS

The Research Center for Trusted Artificial Intelligence has been developing a scientific-engineering basis for creating trusted intelligent systems. For this purpose, staff members and partners of the Center carried out works in the following directions: the creation of trusted machine learning frameworks, the study of AI-specific threats and counteraction methods, enhancement of the interpretability of machine learning models, development of techniques and benchmarks for trustworthiness evaluation, and combining application tools in a cloud platform for developing trusted intelligent systems. The research results were published in nine papers reported at A* conferences and in seven papers in journals from the first quartile (Q1). The main application results obtained by the staff of the Center in 2022 are the trusted versions of the TensorFlow and PyTorch frameworks, which have been turned over to the Center's industrial partners for trial operation.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

1. State Standard R 59276–2020. Artificial intelligence systems. Trust building methods. General principles.
2. State Standard R 59921–2021. Artificial intelligence systems in clinical medicine.
3. Pull request for new fuzzing targets for the TensorFlow framework. <https://github.com/google/oss-fuzz/pull/7704>. Accessed October 26, 2022.
4. A. Vishnyakov, A. Fedotov, D. Kuts, A. Novikov, D. Parygina, E. Kobrin, V. Logunova, P. Belecky, and S. Kurmangaleev, “Sydr: Cutting edge dynamic symbolic execution,” in *2020 Ivannikov ISPRAS Open Conference (ISPRAS)* (IEEE, 2020), pp. 46–54.
5. Pull request in PyTorch. <https://github.com/pytorch/pytorch/pull/79192>. Accessed October 26, 2022.
6. Pull request in PyTorch. <https://github.com/pytorch/pytorch/pull/84343>. Accessed October 26, 2022.
7. Pull request, Fix endless loop in TF. <https://github.com/tensorflow/tensorflow/pull/>. Accessed October 26, 2022.
8. V. P. Ivannikov, A. A. Belevantsev, A. E. Borodin, V. N. Ignatiev, D. M. Zhurikhin, and A. I. Avetisyan, “Static analyzer Sspace for finding defects in a source program code,” *Program. Comput. Software* **40** (5), 265–275 (2014).
9. <https://github.com/pytorch/pytorch/pull/85705>. Accessed October 26, 2022.
10. <https://github.com/tensorflow/tensorflow/pull/57892>. Accessed October 26, 2022.
11. M. Pautov, N. Tursynbek, M. Munkhoeva, N. Muravev, A. Petiushko, and I. Oseledets, “CC-Cert: A probabilistic approach to certify general robustness of neural networks,” *Proc. AAAI Conf. Artif. Intell.* **36** (7), 7975–7983 (2022).
12. A. Gasnikov, A. Novitskii, V. Novitskii, F. Abdukhakimov, D. Kamzolov, A. Beznosikov, M. Takáč, P. Dvurechensky, and B. Gu, “The power of first-order smooth optimization for black-box non-smooth problems,” *Proceedings of the 39th International Conference on Machine Learning* (2022).
13. D. Kovalev, A. Gasnikov, and P. Richtárik, “Accelerated primal-dual gradient method for smooth and convex-concave saddle-point problems with bilinear coupling,” *NeurIPS* (2022).
14. D. Kovalev and A. Gasnikov, “The first optimal algorithm for smooth and strongly-convex-strongly-concave minimax optimization,” *NeurIPS* (2022).
15. A. Chistyakova, M. Cherepnina, K. Arkhipenko, S. D. Kuznetsov, C. S. Oh, and S. Park, “Evaluation of interpretability methods for adversarial robustness on real-world datasets,” in *2021 Ivannikov Memorial Workshop (IVMEM)* (IEEE, 2021), pp. 6–10.
16. E. O. Kurdenkova, M. S. Cherepnina, A. S. Chistyakova, and K. V. Arkhipenko, “Influence of transformation on the success of adversarial attacks for image classifiers Clipped BagNet and ResNet,” *Ivannikov Workshop* (2022).
17. A. Vazhentsev, G. Kuzmin, A. Shelmanov, A. Tsvigun, E. Tsymbalov, K. Fedyanin, and L. Zhukov, “Uncertainty estimation of transformer predictions for misclassification detection,” *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Vol. 1: *Long Papers* (2022), pp. 8237–8252.
18. I. Molodetskikh, M. Erofeev, A. Moskalenko, and D. Vatolin, “Temporally coherent person matting trained on fake-motion dataset,” *Digital Signal Process.* **126**, 103521 (2022).
19. M. Vasilkovsky, A. Alekseev, V. Malykh, I. Shenbin, E. Tutubalina, D. Salikhov, M. Stepanov, A. Chertok, and S. Nikolenko, “DetIE: Multilingual open information extraction inspired by object detection,” in *Proceedings of the 36th AAAI Conference on Artificial Intelligence* (2022).
20. E. Aksenova, N. Lazarev, D. Badalyan, O. Borisenko, and R. Pastukhov, “Michman: An orchestrator to deploy distributed services in cloud environments,” in *2020 Ivannikov ISPRAS Open Conference (ISPRAS)* (IEEE, 2020), pp. 57–63.

Translated by I. Ruzanova