UDC 004.85

# PROBLEMS OF DEEP MACHINE LEARNING

*Ziulkovskiy A.A.*

*Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus*

*Sokolova M.A. – lecturer of the department of foreign languages*

**Annotation**. This article discusses the main difficulties in the development and training of ML-models associated with the problem of generalization. The causes of these problems and the main difficulties caused by them are analyzed. The problem of general non-concretization of ML tasks and the vulnerability of neural networks to adversarial attacks is also considered.

**Keywords**. Machine learning, neural networks, generalization problem.

***Introduction.*** Today ML-models solve increasingly complex tasks, but how close are people to creating a real AI? There are a number of problems that hinder the use of neural networks and they are often associated with existing approaches to learning.

***Main part.*** One of the most common problems is the inability of ML-models to work correctly on a greater variety of examples than those encountered during training. Here we are talking not just about other examples (for example, test ones), but about other types of examples. Solving this problem is important not only for solving practical problems, but also in general for the further development of AI. Of course, all the difficulties are not limited to this; there are also difficulties with the interpretation of models, problems of bias and ethics, the resource intensity of training and others. But in this article, it is the problems of generalization that will be considered.

The task of machine learning is to write algorithms that automatically deduce general patterns from particular cases. This process is called generalization, or induction. It is often necessary to find the relationship between the source and target data in the form of some function.

Various data can be used to train the model: marked up (supervised learning), unreliably marked up (weakly-supervised learning) or completely unmarked (self-supervised learning). But in any case, we use a certain set of specifically taken examples. It is clear that the more material for training, the higher the quality of the resulting solution will be. There are even theorems that prove for various ML-algorithms the striving of the resulting solution to the ideal with an unlimited increase in the amount of training data and the size of the model (this property is called universal consistency) [1].

But in practice, it is not always a simple increase in the volume of examples that leads to a qualitative improvement in the operation of the algorithm. A number of difficulties arise in the work of the model.

### *1) Data leak*

A data leak is a situation when there is a certain feature that, during training, contained more information about the target variable than during subsequent application of the model in practice. Data leakage can occur in a variety of forms. For example:

In the task of diagnosing real and fake vacancies, almost all fake vacancies related to Europe. A model trained on this data may not consider any vacancies from other regions suspicious. You can remove a region from the features, but the job description also has characteristic features that depend on the region and the neural network can focus on them [2].

This problem cannot be solved simply by increasing the amount of training data without changing the approach to training. Here it can be argued that data leakage is not a problem of the ML-algorithms themselves, since the algorithm cannot know which feature needs to be taken into account and which one does not.

### *2) Shortcut learning*

For a long time, researchers have noticed that many modern neural networks, even the largest ones today, resemble "clever Hans” [3, 4]. That is, when trying to solve a problem, the answer is sought in a roundabout way. The neural network does not solve it the way we wanted. And this workaround may stop working at any time.

Shortcut learning is a phenomenon when models get the right answer using generally incorrect reasoning ("right for the wrong reasons"), which work well only for training data distribution. Since the training and test samples are usually taken from the same distribution, such models can give good accuracy during testing.

For example, one of the most modern neural networks for detecting YOLOv5 objects can be easily deceived by unusual details that are not related to the object itself, or it can take several objects for one.

Language models are also a good example. Despite the fact that the current neural networks are trained on hundreds of gigabytes of text and have billions of parameters, they are still very imperfect. For example, the BERT model mainly learns superficial, stereotypical associations of words with each other (word co-occurrence), and hardly understands the meaning of the text well. The same can be said about RoBERTa, ALBERT, mT5, etc [5].

Even if we talk about modern networks (GPT-3, Gopher, InstructGPT, Wu Dao, etc.), they really have no idea how the world works. The knowledge they have taken from the statistics of texts is very superficial and disconnected from the underlying reality.

In general, the opinion of many researchers agrees that at the moment language models-transformers are far from understanding the meaning of texts in the form in which a person understands them [6].

As you can see, almost any dataset has a limited variety and does not cover all situations in which the correct operation of the model is desirable. There may be "spurious correlations" in the data, which allow predicting the answer with good accuracy only on this sample without complex data analysis.

In the article on the problem of shortcut learning, the authors consider several levels of generalization that machine learning models can achieve [7]:

a) Uninformative features: The network uses features that do not allow you to effectively predict the answer even on a training sample.

b) Overfitting features: The network uses features that make it possible to effectively predict the response on the training sample, but not on the entire distribution from which this sample was obtained [8].

c) Shortcut features: The network uses features that allow you to effectively predict the response to the distribution of data from which the training (and, as a rule, test) sample is taken. The authors call the ability of the algorithm to work with good accuracy on a certain fixed distribution of data independent and identically distributed generalization [9].

d) Intended features: The network uses features that allow you to effectively predict the response in the general case. Such signs will work well outside of the training data distribution, when the "workarounds" are closed.

The situation when the distribution of data differs during training and application is called a distributional shift. The good performance of the model in conditions of data shift is called out-of-distribution generalization (OOD generalization), it is also known as domain generalization, its connection with overcoming the problem of shortcut learning is obvious [10].

Effective out-of-distribution generalization remains an open problem that needs to be solved if we want to further develop AI.

### *3) The problem of non-concretization in ML-problems*

This problem is formulated as follows: for each task, there may be an infinite number of different trained models with different weights and computational architectures that give approximately the same accuracy on the training sample or training data distribution from which the test sample is taken. This is a well-known and central problem of machine learning [11]. In the work of D'Amour et al., this problem is called the problem of non-specification of the ML-problem (underspecification of ML-pipeline) [12]. Its essence lies in the fact that when conditions change (going beyond the training distribution) these models can start working very differently and often unpredictably, and the above-mentioned problems of shortcut learning and data leaks greatly aggravate the situation.

***4) Adversarial attacks***

With the work of Szegedy et al., research began in the field of so-called adversarial attacks on neural networks [13]. The authors have shown that by applying insignificant noise to the image, neural networks can be made to make mistakes in the classification task, predicting a completely different class. Of course, not any noise is suitable for this, but a special pattern that can be found using optimization.

Even more unexpected was the fact that exactly the same noise pattern caused other networks trained with other hyperparameters and on other subsamples of training data to make mistakes on the same image.

In 2017, a very simple way was invented to "deceive" a convolutional neural network: it is enough to stick a sticker with a special pattern on an object, and the network will cease to classify it correctly [14].

The problems of adversarial attacks and insufficient out-of-distribution generalization in computer vision are probably related. Human vision is also susceptible to this type of attack, as optical illusions prove, but it is much more resistant to this.

***Conclusion.*** ML-models often rely on "roundabout" ways to get an answer caused by the lack of diversity of the training data distribution and the presence of parasitic correlations in it (this resembles a data leak). Such a model only simulates the solution of the problem, and therefore may stop working correctly if conditions change.

ML-models have no idea about the problem being solved and the constant increase in the amount of data for training does not make significant progress. To create more stable algorithms, new approaches to machine learning in general are needed.

Models showing the same average accuracy during testing can work significantly differently, which is especially evident outside the data distribution on which the models were trained and tested.

Computer vision systems (and not only) are subject to adversarial attacks, in which the introduction of various interferences or the addition of minor details can spoil the operation of the model.

## ***References***

*1. Strong Universal Consistency of Neural Network Classifiers / A. Farago // Institute of Electrical and Electronics Engineers [Electronic resource]. – 1993. – Mode of access: https://ieeexplore.ieee.org/document/748747 Date of access: 26.03.2022*

*2. Real / Fake Job Posting Prediction // The University of the Aegean [Electronic resource]. – 2020. – Mode of access: https://www.kaggle.com/shivamb/real-or-fake-fake-jobposting-prediction Date of access: 26.03.2022*

*3. Unmasking Clever Hans Predictors and Assessing What Machines Really Learn / S. Lapuschkin // Cornell University [Electronic resource]. – 2019. – Mode of access: https://arxiv.org/abs/1902.10178 Date of access: 26.03.2022*

*4. When Choosing Plausible Alternatives, Clever Hans can be Clever / P. Kavumba // Cornell University [Electronic resource]. – 2019. – Mode of access: https://arxiv.org/abs/1911.00225 Date of access: 26.03.2022*

*5. GPT и BERT // [Electronic resource]. – 2021. – Mode of access: http://www.generalized.ru/GPT_%D0%B8_BERT Date of access: 26.03.2022*

*6. OpenAI’s massive GPT-3 model is impressive, but size isn’t everything / K. Wiggers [Electronic resource]. – 2020. – Mode of access: https://venturebeat.com/2020/06/01/ai-machine-learning-openai-gpt-3-size-isnt-everything/ Date of access: 26.03.2022*

*7. Unmasking Clever Hans Predictors and Assessing What Machines Really Learn / R. Geirhos // Cornell University [Electronic resource]. – 2020. – Mode of access: https://arxiv.org/abs/2004.07780 Date of access: 26.03.2022*

*8. Интерпретация моделей и диагностика сдвига данных: LIME, SHAP и Shapley Flow [Electronic resource]. – 2022. – Mode of access: https://habr.com/ru/company/ods/blog/599573/ Date of access: 26.03.2022*

*9. Независимые одинаково распределённые случайные величины [Electronic resource]. – 2017. – Mode of access: https://en.wikipedia.org/wiki/Independent_and_identically_distributed_random_variables Date of access: 26.03.2022*

*10. Domain Generalization in Vision: A Survey / K. Zhou // Cornell University [Electronic resource]. – 2021. – Mode of access: https://arxiv.org/abs/2103.02503 Date of access: 26.03.2022*

*11. The Need for Biases in Learning Generalizations / T. Mitchell // Rutgers University [Electronic resource]. – 1980. – Mode of access: http://www.cs.cmu.edu/~tom/pubs/NeedForBias_1980.pdf Date of access: 26.03.2022*

*12. Underspecification Presents Challenges for Credibility in Modern Machine Learning / A. D’Amour // Cornell University [Electronic resource]. – 2020. – Mode of access: https://arxiv.org/abs/2011.03395 Date of access: 26.03.2022*

*13. ntriguing properties of neural networks / C. Szegedy // [Electronic resource]. – 2021. – Mode of access: http://www.generalized.ru/Intriguing_properties_of_neural_networks Date of access: 26.03.2022*

*14. Adversarial Patch / T. Brown // Cornell University [Electronic resource]. – 2017. – Mode of access: https://arxiv.org/abs/1712.09665 Date of access: 26.03.2022*