

УДК: 004.896

DOI: 10.30987/2658-6436-2020-2-17-23

В.В. Лаптев, О.М. Гергет, В.В. Данилов

## ИССЛЕДОВАНИЕ ГЕНЕРАТИВНО-СОСТЯЗАТЕЛЬНЫХ СЕТЕЙ ДЛЯ СИНТЕЗА НОВЫХ МЕДИЦИНСКИХ ДАННЫХ

*В работе рассматривается создание генеративно-сопоставительной модели нейронной сети (GAN – Generative Adversarial Nets), для синтеза новых медицинских данных. GAN состоит из двух моделей, обучаемых одновременно: генеративная модель (G – Generator), фиксирующая распределение данных, и дискриминирующая модель (D – Discriminator), оценивающая вероятность того, что выборка получена из обучающих данных, а не от генератора G. Для создания G применяется собственная архитектура нейронной сети, основанная на сверточных слоях, с применением экспериментальных методов глубокого обучения на основе Tensor Flow Addons. Для создания дискриминатора D применяется подход Transfer Learning (TL). Процедура обучения состоит в максимизации вероятности того, что D допустит ошибку. Эксперименты показывают, что предложенная архитектура GAN, полностью справляется с поставленной задачей – синтез новых медицинских данных.*

**Ключевые слова:** генеративно-сопоставительная сеть, сверточная нейронная сеть, генератор, предобученная модель, дискриминатор, вероятность, обучение.

V.V. Laptev, O.M. Gerget, V.V. Danilov

## RESEARCH OF GENERATIVE ADVERSARIAL NETWORKS FOR THE SYNTHESIS OF NEW MEDICAL DATA

*The paper considers the development of a Generative Adversarial Network (GAN) for the synthesis of new medical data. The developed GAN consists of two models trained simultaneously: a generative model (G - Generator), estimating the distribution of data, and a discriminating model (D - Discriminator), which estimates the probability that the sample is obtained from the training data, and not from generator G. To create G, we used own neural network architecture based on convolutional layers using experimental functions of Tensor Flow Addons. To create discriminator D, we used a Transfer Learning (TL) approach. The training procedure is to maximize the likelihood that discriminator D will make a mistake. Experiments show that the proposed GAN architecture completely copes with the task of synthesizing of new medical data.*

**Keywords:** generative adversarial network, convolutional neural network, generator, pre-trained model, discriminator, probability, training.

### Введение

В настоящее время генеративные модели, основанные на нейронных сетях глубокого обучения, используются для решения различных задач: генерация рукописных цифр [1], генерация лиц [2], написания стихов [3] и др.

В современной медицинской практике, в том числе и в области кардиологии, продолжает сохраняться устойчивый рост использования методов автоматической обработки графических данных. Наиболее популярными являются алгоритмы обработки анатомических структур на основе данных МРТ и КТ. Однако в ряде случаев использование представленных модальностей невозможно, так как, например, основным ограничением компьютерной томографии (КТ) является отсутствие режима реального времени. Для решения данной проблемы необходимо разработать алгоритм трекинга и визуализации данных на основе интеллектуального анализа. GAN позволит синтезировать уникальные данные для обучения модели трекинга, подобно [4].

Популярные генеративные модели, как ограниченная машина Больцмана и ее множество вариантов [5, 6, 7], были успешно использованы в ограниченных условиях, такие

как послойная предварительная подготовка и некоторые прикладные задачи. Но общее развитие генеративных моделей как самостоятельных инструментов в значительной степени отстают, из-за, возникающих проблем, при оценке максимального правдоподобия. В этой статье мы предлагаем собственную архитектуру G, и приводим сравнение различных transferlearning моделей D для синтеза медицинских данных.

### Исследование

**2.1. Генеративно–сопоставительные сети.** GAN состоят из двух нейронных сетей: генератора и дискриминатора. Генератор семплирует случайные числа из некоторого распределения  $P(Z)$ , входного шума, например  $N(0, 1)$  и генерирует из них объекты  $X_p = G(Z; \theta_g)$ , а дискриминатор, получая на вход семплы от оригинальной выборки  $X_s$  и генератора  $X_p$ , учится предсказывать природу семпла (откуда он пришел), выдавая на выходе скаляр  $D(X; \theta_d)$ . На рисунке 1 представлена классическая структура GAN.

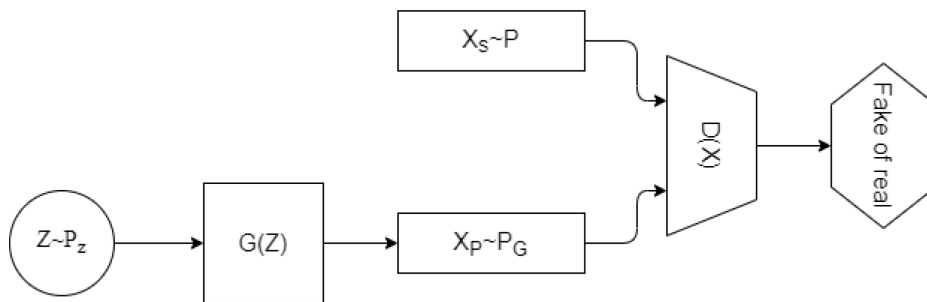


Рис. 1. Классическая архитектура GAN

Процесс обучения GAN можно представить в виде следующего алгоритма (генератор и дискриминатор обучаются отдельно, но в рамках одной сети):

Шаг 1. Задаем произвольный начальный  $G(Z)$ .

Шаг 2. Обучаем дискриминатор, параметры  $\theta_d$  обновляются в сторону уменьшения бинарной кросс-энтропии (BCE сокр. от Binary Cross Entropy):

$$BCE = -\frac{1}{N} \sum_{i=0}^N y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i) \quad (1)$$

$$\theta_d = \theta_d - \nabla_{\theta_d} (\log(D(X_s)) + \log(1 - D(G(Z)))) \quad (2)$$

Шаг 3. Обучение генератора, обновляем параметры генератора  $\theta_g$  в сторону увеличения логарифма вероятности, а дискриминатору присвоить сгенерированному объекту лейбл реального:

$$\theta_g = \theta_g + \nabla_{\theta_g} (\log(1 - D(G(Z)))) \quad (3)$$

Задачу, что решает GAN можно сформулировать следующим образом:

$$\min_G \max_D E_{X \sim P} [\log(D(X))] + E_{Z \sim P_Z} [\log(1 - D(G(Z)))] \quad (4)$$

При заданном генераторе оптимальный дискриминатор выдает вероятность:

$$D(X) = \frac{P(X)}{P_g(X) + P(X)} \quad (5)$$

В своей работе Ian J. Goodfellow [8] показывает, что при достаточной мощности обеих сетей у данной задачи есть оптимум, в котором генератор научится генерировать распределение  $P_g(X)$ , совпадающее с  $P(X)$ , а дискриминатор на любое  $X$  выдает вероятность 0,5. На рисунке 2 приведена иллюстрация процесса тренировки порождающих состязательных сетей GAN из [4]. Где, черная точечная кривая – настоящее распределение  $P(X)$ , зеленая – распределение генератора  $P_g(X)$ , синяя – распределение вероятности  $D(X; \theta_d)$  дискриминатора предсказать принадлежность объекта. В результате многократного повторения шагов а, б, с  $P_g(X)$  совпало с  $P(X)$  и дискриминатор не способен отличить одно от другого.

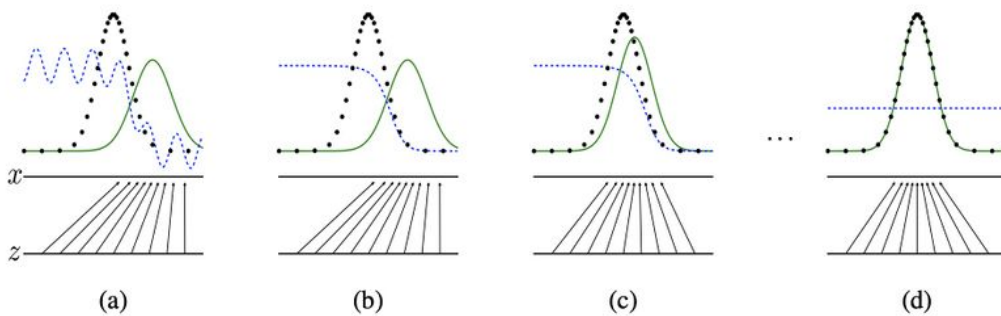


Рис. 2. Иллюстрация процесса тренировки GAN

**2.2. Генератор.** Для решения поставленной задачи было принято реализовать собственную глубокую архитектуру сети генератора (рисунок 3), базирующуюся на обучаемых сверточных и деконволюционных слоях с применением параллельной свертки и последующей конкатенации для усиления признаков. Стоит отметить, что глубокий сверточный генератор имеет 6 уровней повышения размерности и общее количество весов составило 17,074,528. Описание структуры уровня повышения (1 уровень,  $l = 6$ ) приведено в таблице 1.

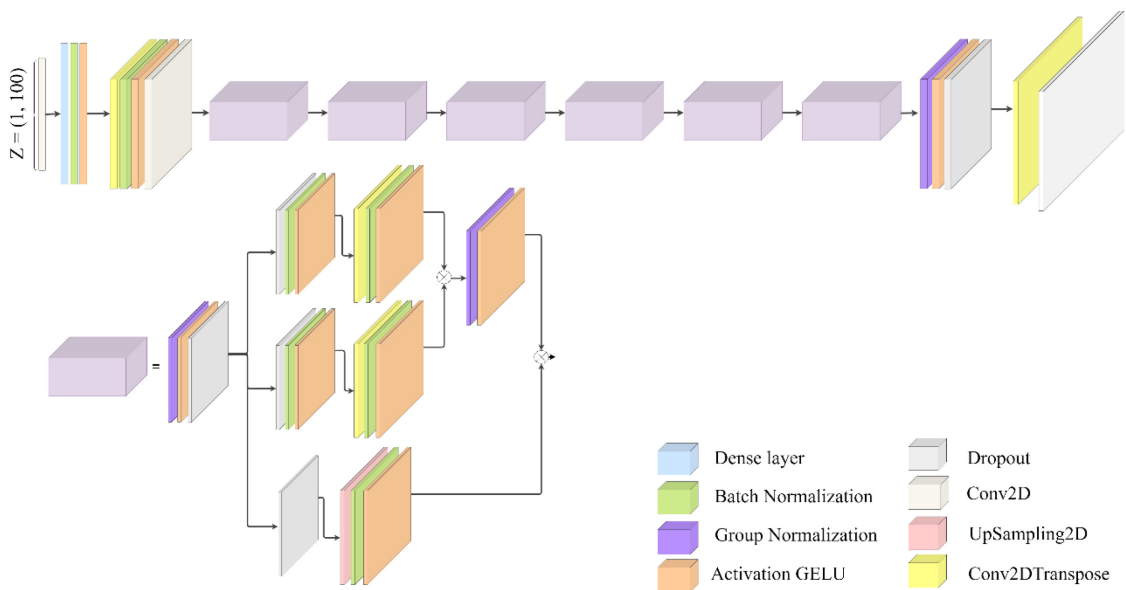


Рисунок 3. Архитектура генератора

Таблица 1. Описание слоев архитектуры генератора G

Name_layer	Kernel_size	Pool_size	Padding	filters	Strides
Conv2d	(3, 3)	None	same	512 (2**1*8)	None
Up_Sampling2d	None	(2, 2)	None	None	None
Conv2d_1	(1, 1)	None	same	1024 (2**(1+1)*8)	None
Conv2dTranspose_1	(3, 3)	None	same	512 (2**1*8)	(2, 2)
Conv2d_2	(1, 1)	None	same	1024 (2**(1+1)*8)	None
Conv2dTranspose_2	(1, 1)	None	same	512 (2**1*8)	(2, 2)

**2.3. Дискриминатор.** В качестве дискриминатора было принято решение использовать модель transferlearning, обученные для нахождения вектора признаков (featurevector) изображений Imagenet. Рассмотренные модели приведены в таблице 2. Проведенный анализ показал, что модель Xception имеет наилучшее соотношение Accuracy / Paramets, в связи с этим она будет использована для дальнейшего решения задачи. Общее количество весов модели представлено в таблице 3.

Таблица 1. Таблица сравнения моделей transferlearnong

Model	Size	Top-1 Accuracy	Top-5 Accuracy	Parameters
ResNet50	98 MB	0.749	0.921	25 636 712
ResNet101	171 MB	0.764	0.928	44 707 176
ResNet50V2	98 MB	0.760	0.930	25 613800
InceptionV3	92 MB	0.779	0.937	23 851784
MobileNet	16 MB	0.704	0.895	4 253864
MobileNetV2	14 MB	0.713	0.901	3 538984
NASNetMobile	23 MB	0.744	0.919	5 326716
Xception	88 MB	0.790	0.945	22 910480

Таблица 3. GAN Model Weights

	Generator	Discriminator	GAN
Total params	17 074528	20 867624	37 942152
Trainable params	17 016608	2048	17 018 656
Non-trainableparams	57920	20 865576	20 923 496

**2.4. Результаты.** Результаты функций потерь отражены на рисунках 4 – 5. Исходя из графиков видно, что модель стремится свести ошибку к усредненному значению. Так каждый скачек потери генератора, сказывается уменьшением ошибки дискриминатора и наоборот. Процесс обучения апробирован на медицинских данных, полученные результаты, представленный в таблице 4. В строке а) – примеры изображений, генерируемых сетью GAN после 1-ой, 100-ой и 200-ой эпохи обучения соответственно. Можно заметить, что результат становится лучше в процессе обучения. В строке б) представлены результаты 200-ой, 250-ой и 300-ой эпохи соответственно, тенденция по улучшению в строке а) сохраняется. В строке в), отображающей результат после 300-ой, 350-ой и 400-ой эпохи соответственно

изображение, генерируемые сетью, продолжают улучшаться (информативность растет). В строке г), результаты 400-ой, 500-ой и 600-ой эпохи соответственно. По данным результатам видно, что изображения достигли высокой точности к концу обучения, а значит сеть обучена и готова к использованию.

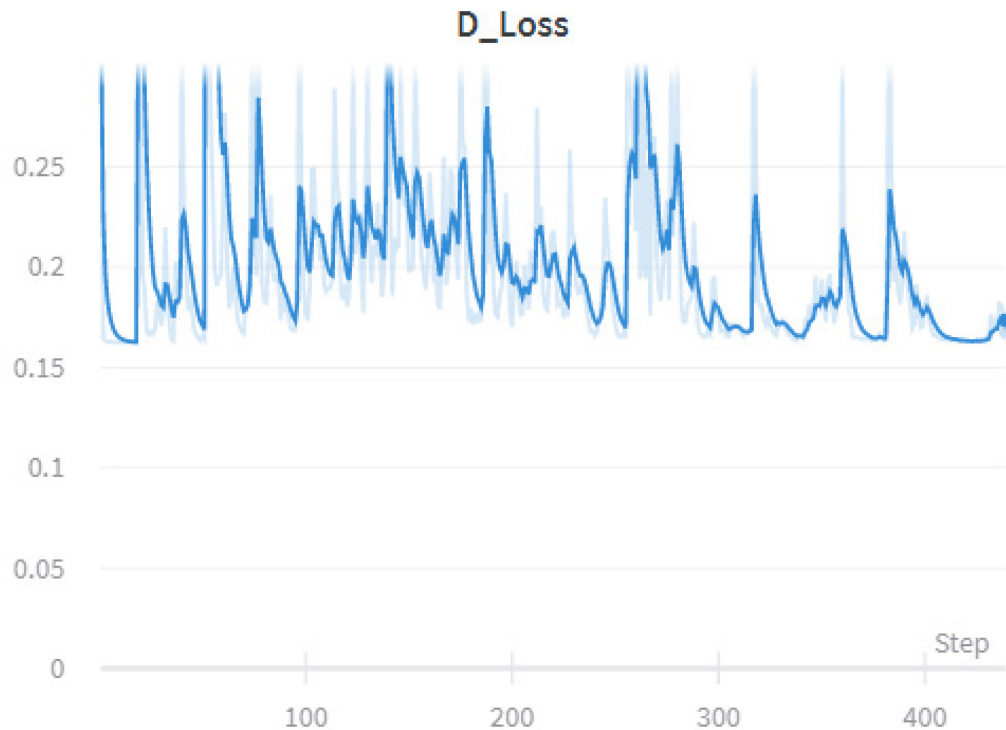


Рис. 4. Динамика кросс-энтропии дискриминатора

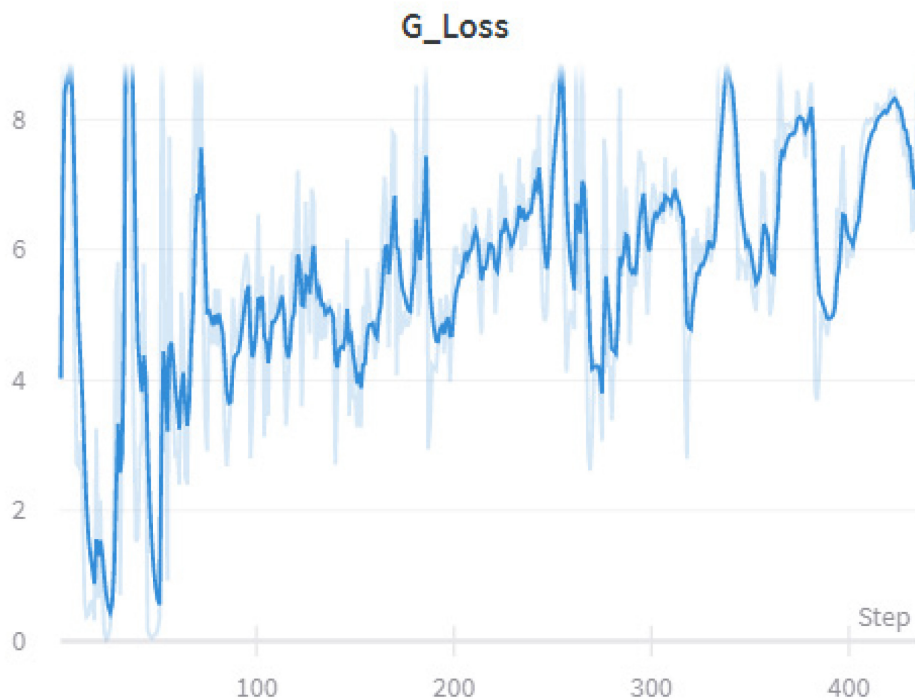
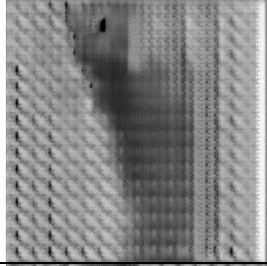
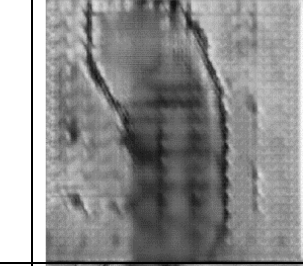
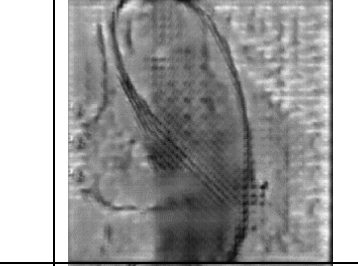

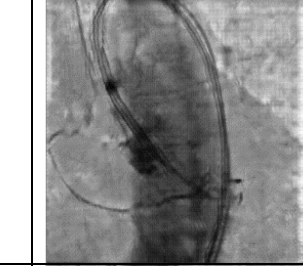
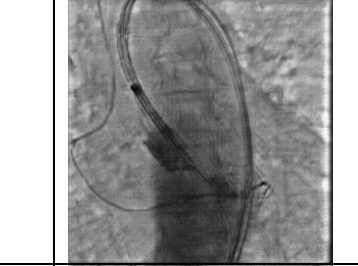

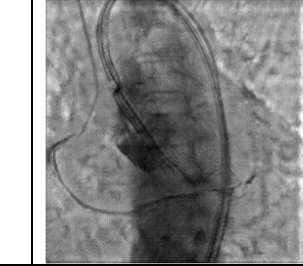
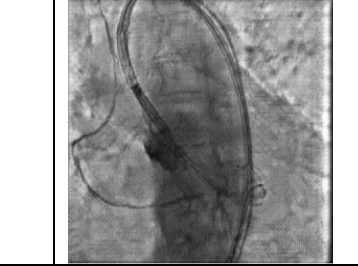
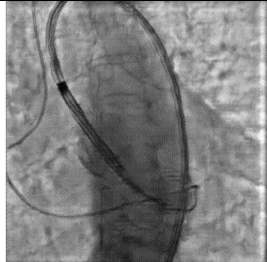
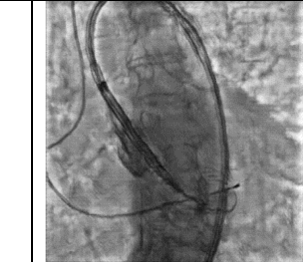
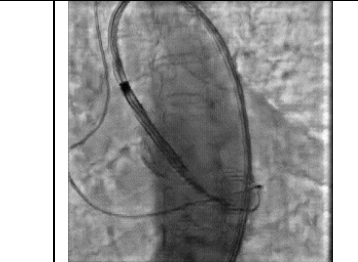


Рис. 5. Динамика кросс-энтропии генератора

Таблица 4. – Визуализация процесса обучения

а) визуализация результата с 1 по 200 эпоху			
б) визуализация результата с 200 по 300 эпоху			
в) визуализация результата с 300 по 400 эпоху			
г) визуализация результата с 400 по 500 эпоху			

### Заключение

В результате исследования была разработана собственная архитектура глубокого сверточного генератора, а также скомпилирована модель дискриминатора для GAN, на основе которой произведена генерация медицинских данных. Данный метод использован в качестве расширения исходной выборки с последующим использованием ее в задаче трекинга ключевых точек анатомических структур и медицинских инструментов.

Данная работа выполнена в рамках РФФИ 19-07-00351\19 и государственного задания «Наука» №FFSWW-2020-0014.

#### Список литературы:

1. Francois, C. Introducing Variational Autoencoders (in Prose and Code), [Электронный ресурс] – URL: <https://blog.keras.io/building-autoencoders-in-keras.html>, last accessed 2019/11/20.
2. Kateryna, K. Facial Surface and Texture Synthesis via GAN), [Электронный ресурс] – URL: <https://neurohive.io/en/computer-vision/facial-surface-and-texture-synthesis-via-gan>, last accessed 2020/01/09.
3. Anastasyev, D. Как научить свою нейросеть

#### References:

1. Francois Chollet, Introducing Variational Autoencoders (in Prose and Code), [Elektron. resurs] – URL: <https://blog.keras.io/building-autoencoders-in-keras.html>, last accessed 2019/11/20.
2. Kateryna Koidan, Facial Surface and Texture Synthesis via GAN), [Elektron. resurs] – URL: <https://neurohive.io/en/computer-vision/facial-surface-and-texture-synthesis-via-gan>, last accessed 2020/01/09.
3. Dan Anastasyev, Automatically Generating Verse

- генерировать стихи, [Электронный ресурс] – URL: <https://habr.com/ru/post/334046>, last accessed 2020/04/02.
4. Gerget, O. M. Bionic models for identification of biological systems // Journal of Physics: Conference Series, 2017. – Vol. 803. - Pp. 68 – 73.
5. Bengio, Y. Better Mixing via Deep Representations / Y. Bengio, G. Mesnil, Y. Dauphin, S. Rifai // arXiv:1207.4404 [cs], July 2012. arXiv: 1207.4404.
6. Hinton, G. E. Reducing the dimensionality of data with neural networks / G. E. Hinton, R. R. Science, 313(5786), 2006 - Pp. 504–507.
7. Salakhutdinov, R. Deep Boltzmann machines / R. Salakhutdinov, G. E. Hinton // In International Conference on Artificial Intelligence and Statistics, 2009 - Pp. 448–455.
8. Goodfellow, I. Generative Adversarial Nets / I. Goodfellow, J. Pouget-Abadie, M. Mirza // Advances in Neural Information Processing Systems 27 / Ed. by Z. Ghahramani, M. Welling, C. Cortes et al. — Curran Associates, Inc, 2014. - Pp. 2672–2680 [Электронный ресурс] – URL:<http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>.
4. Gerget, O. M. Bionic models for identification of biological systems // Journal of Physics: Conference Series, 2017. – Vol. 803. - Pp. 68 – 73.
5. Bengio, Y. Better Mixing via Deep Representations / Y. Bengio, G. Mesnil, Y. Dauphin, S. Rifai // arXiv:1207.4404 [cs], July 2012. arXiv: 1207.4404.
6. Hinton, G. E. Reducing the dimensionality of data with neural networks / G. E. Hinton, R. R. Science, 313(5786), 2006 - Pp. 504–507.
7. Salakhutdinov, R. Deep Boltzmann machines. / R. Salakhutdinov, G. E. Hinton // In International Conference on Artificial Intelligence and Statistics, 2009 - Pp. 448–455.
8. Goodfellow, I. Generative Adversarial Nets / I. Goodfellow, J. Pouget-Abadie, M. Mirza // Advances in Neural Information Processing Systems 27 / Ed. by Z. Ghahramani, M. Welling, C. Cortes et al. — Curran Associates, Inc, 2014. - Pp. 2672–2680 [Elektronnyj resurs] – URL:<http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>.

*Статья поступила в редколлегию 28.04.2020.*

*Рецензент:*

*д-р. техн. наук, доцент,*

*Брянский государственный технический университет*

*Захарова А.А.*

*Статья принята к публикации 06.05.2020.*

**Сведения об авторах:**

**Лаптев Владислав Витальевич**

Студент магистратуры Томский Политехнический  
Университет (Томск, Россия)  
E-mail: [vv139@tpu.ru](mailto:vv139@tpu.ru)

**Данилов Вячеслав Владимирович**

Инженер-исследователь лаборатории дизайна  
медицинских изделий, Томский Политехнический  
Университет (Томск, Россия)  
E-mail: [viacheslav.v.danilov@gmail.com](mailto:viacheslav.v.danilov@gmail.com)

**Гергет Ольга Михайловна**

д.т.н., профессор, Томский Политехнический  
Университет (Томск, Россия)  
E-mail: [olgagerget@mail.ru](mailto:olgagerget@mail.ru)

**Information about authors:**

**Laptev Vladislav Vitalievich**

Graduate student Tomsk Polytechnic University (Tomsk,  
Russia)  
E-mail: [vv139@tpu.ru](mailto:vv139@tpu.ru)

**Danilov Vyacheslav Vladimirovich**

Research Engineer of Medical Devices Design  
Laboratory, Tomsk Polytechnic University (Tomsk,  
Russia)  
E-mail: [viacheslav.v.danilov@gmail.com](mailto:viacheslav.v.danilov@gmail.com)

**Gerget Olga Mihailovna**

Full Professor, Professor, Tomsk Polytechnic University  
(Tomsk, Russia)  
E-mail: [olgagerget@mail.ru](mailto:olgagerget@mail.ru)