## ***ПРОБЛЕМЫ ТЕХНИЧЕСКОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ***

УДК 004.934

# ОБНАРУЖЕНИЕ ФИЗИЧЕСКИХ АТАК ПОВТОРНОГО ВОСПРОИЗВЕДЕНИЯ РЕЧИ С ПОМОЩЬЮ ЛЕГКОЙ СВЕРТОЧНОЙ СЕТИ С ГРАФОВЫМ ВНИМАНИЕМ

**Белослюдов Александр Сергеевич, Лепендин Андрей Александрович, Филин Яков Александрович**

Алтайский государственный университет, г. Барнаул
e-mail: andrey.lependin@gmail.com

# DETECTION OF PHYSICAL SPEECH SPOOFING ATTACKS USING LIGHT CONVOLUTION NEURAL NETWORK WITH GRAPH ATTENTION LAYER

**Beloslyudov Aleksandr S., Lependin Andrey A., Filin Jacob A.**

Altai State University, Barnaul

*Аннотация:* В статье предложена модель, основанная на модификации сверточной нейронной сети LCNN за счет применения слоев графового внимания, способная эффективно обнаруживать физические атаки на данные речи. Показана актуальность и значимость проблемы обнаружения речевых подделок в контексте повышенного интереса к голосовым технологиям и угрозе безопасности, связанной с возможностью подделки или изменения аудиоданных. Проведена реализация предложенного подхода на языке Python с использованием библиотеки PyTorch. Обучение и тестирование модели осуществлено на данных из набора ASVspoof 2019. Проведен выбор числа «голов» в слое графового внимания. Выбранная версия нейросетевой модели сопоставлена по метрикам точности и эквивалентной ошибки EER с базовой моделью, в качестве которой выступала LCNN-сеть. Продемонстрировано превосходство модифицированного подхода, предложенного в данной работе, как по качеству распознавания поддельных голосовых сообщений, так и по числу параметров модели.

*Ключевые слова:* атака презентации, атака повторным воспроизведением речи,

*Abstract:* In this paper a model based on modification of the LCNN convolutional neural network through the use of graph attention layers was proposed. It is capable of effectively detecting physical attacks on speech data. The relevance and significance of the problem of detecting speech spoofing was shown in the context of increased interest in voice technologies and the security threat associated with the possibility of forging or changing audio data. The proposed approach was implemented in Python using the PyTorch library. The model was trained and tested using data from the ASVspoof 2019 set. The number of “heads” in the graph attention layer was selected. The selected version of the neural network model was compared in terms of accuracy and equivalent error EER with the base model, which was the LCNN network. The superiority of the modified approach proposed in this work has been demonstrated, both in terms of the quality of recognition of spoofed speech and in the number of model parameters.

*Keywords:* spoofing attack, replay speech attack, deep learning, light convolution network, graph attention.

глубокое обучение, легкая сверточная сеть, графовое внимание.

*Для цитирования: Белослюдов А.С., Лепендин А.А., Филин Я.А. Обнаружение физических атак повторного воспроизведения речи с помощью легкой сверточной сети с графовым вниманием // Проблемы правовой и технической защиты информации. 2023. №11. С. 8-15.*

*For citation: Beloslyudov A.S., Lependin A.A., Filin J.A. Detection of physical speech spoofing attacks using light convolution neural network with graph attention layer // Legal and Technical Problems Information Protection. 2023. No. 11. P. 8-15.*

*Введение.* С ростом количества обрабатываемых и передаваемых данных в современном мире защита от несанкционированного доступа становится все более важной задачей. Для ее решения все чаще используются различные методы проверки личности пользователя. В настоящее время одним из самых распространенных и простых способов верификации является использование биометрических систем. Такие системы позволяют идентифицировать пользователя по его уникальным биологическим характеристикам, таким как отпечатки пальцев, сетчатке глаза или голосу. Современные технологии голосовой биометрии позволяют создавать надежные системы идентификации пользователей. Одним из главных их преимуществ является удобство использования, так как обычно для проверки не требуется никакого специального дополнительного оборудования. Голосовая биометрия может использоваться совместно с различными устройствами и средствами связи, что делает ее универсальной и гибкой. Она широко используется в различных сферах, таких как криминалистика, интернет-банкинг, системы контроля доступа и многих других.

Технологии голосовой биометрии, однако, имеют ряд серьезных проблем, связанных с атаками представления (так называемыми спуфинг-атаками), когда вместо речи настоящего диктора системе предоставляется подделка [1]. Можно выделить несколько основных классов голосовых подделок: имитация голоса злоумышленником, синтез речи целевого пользователя, преобразование речи злоумышленника, а также повторное воспроизведение записи подлинного голоса. Если первые три класса атак (выделяемых в группу так называемых «логических атак») требуют специальных навыков и применения специализированных алгоритмов, то последний (также называемый «физической атакой») легко может быть реализован с помощью простых программных и аппаратных средств аудиозаписи. Наиболее важным отличием логических атак от физических с точки зрения их выявления, является то, что в них злоумышленник имеет полный контроль за фоновым шумом и реверберацией в моделируемом им виртуальным помещении, где «записывается» поддельный сигнал.

Несмотря на то, что о спуфинг-атаках известно давно, разработка действенных методов борьбы с ними началась только в 2010-х годах. С 2015 года стали проводиться открытые конкурсы на разработку новых методов обнаружения речевого спуфинга ASVspoof [2]. Именно в рамках этих конкурсов были собраны представительные наборы речевых образцов поддельных голосовых сигналов, начали разрабатываться алгоритмы их обнаружения, основанные на современных методах машинного обучения. Разработка новых методов обнаружения голосового спуфинга все еще остается крайне актуальной задачей в силу того, что качество работы предлагаемых методов недостаточно высоко, а в условиях моделирования/записи подделок в зашумленных условиях оно существенно деградирует даже в лучших решениях.

За последние несколько лет для решения широкого круга задач, связанных с

обработкой сложно устроенных многокомпонентных данных, стали использоваться так называемые графовые нейронные сети [3], позволяющие эффективно выявлять связи между отдельными компонентами. Записи речи в частотно-временном представлении как раз представляют собой хороший пример подобных сложных данных.

В данной работе предложен новый метод обнаружения физических атак представления, основанный на применении глубокой нейросетевой модели, сочетающей в сочетании с графовым слоем с самовниманием.

*Предложенный метод обнаружения физической атаки.* На рисунке 1 представлена архитектура разработанной нейронной сети. Она состояла из экстрактора признаков, построенного на основе нейронной сети Light Convolution Neural Network (LCNN) [4] и слоя с графовым вниманием (Graph Attention, GAT) [5]. На вход нейронной сети подавались результаты Q-константного преобразования [6] исходного сигнала в виде тензора размером 84×200×1, который проходит через экстрактор признаков LCNN. Основной составляющей архитектуры LCNN являлись сверточные слои, который выделяли признаки все более высокого уровня из сигнала, а также MFM-слои [7], которые попарно сравнивали формируемые карты признаков и отбирали из них наиболее информативные. Также в LCNN присутствовали типичные для сверточных сетей слои подвыборки [8], которые уменьшали размерности формируемых признаков, и преобразования нормализации [9], которые были необходимы для приведения к нормальному распределению сигналов в сети. Последние были нужны для ускорения процесса обучения и повышения качества работы сверточной сети.

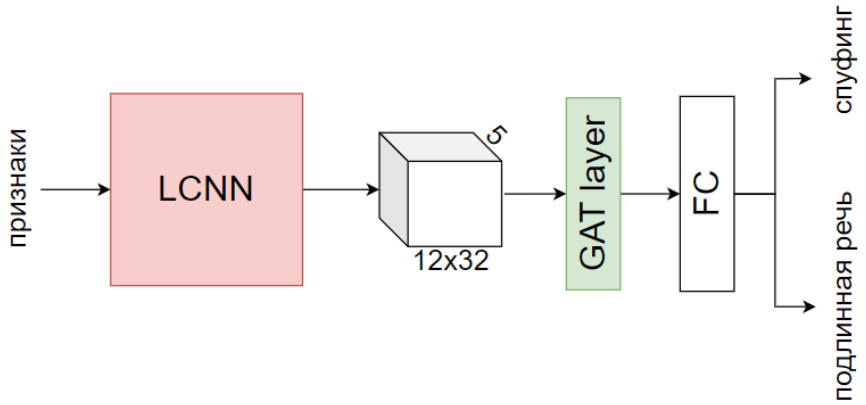


Рисунок 1. Архитектура предложенной нейронной сети для обнаружения презентационных/спуфинг атак

На выходе LCNN-экстрактора формировался тензор признаков аудиосигнала размером 5×12×32. При подаче его на GAT-слой он преобразовывался в набор из 5 векторов длины 384, которые соответствовали состояниям 5 вершин полносвязного графа. В слое GAT использовался механизм самовнимания с варьируемым числом так называемых голов преобразования для вычисления перевзвешенных векторов представления сигнала. После усреднения вычисленных в GAT-слое новых векторов вершин графа, они подавались на полносвязный слой, осуществлявший непосредственно классификацию аудиосигналов.

*Q-константное преобразование.* В качестве векторов признаков использовались частотные разложения с постоянным значением добротности Q (Q-константные, CQT-разложения) [6]. В

отличие от Фурье-преобразований, для них характерно степенное распределение частот полосовых фильтров для выделения компонент сигнала. Вводилось понятие добротности фильтра, определяемое как отношение центральной частоты фильтра к ее полосе пропускания (ширина окна):

$$Q = fk/\delta f, \quad (1)$$

где $f_k$ – центральная частота полосового фильтра, $\delta f$ – так называемая ширина фильтра. Такое преобразование подразумевало константное значение Q-фактора во всех частотных полосах. Q-константное преобразование для дискретного сигнала определялось следующим образом:

$$X(k, n) = \sum_{j=n-N_k/2}^{n+N_k/2} x(j)\alpha_k^*(j - n + N_k/2), \quad (2)$$

где $k$ = 0, 1, 2, ..., $K$-1 - индекс частотной полосы, $N_k$ – длина фрейма сигнала (в отсчетах), $\alpha_k^*(n)$ – комплексно-сопряженное к величине:

$$\alpha_k = (n/CN_k)\exp\left(i(2\pi n f_k/f_s + \Phi_k)\right), \quad (3)$$

где $f_k$ – центральная частота $k$-й частотной полосы, $f_s$ – частота дискретизации сигнала, $\Phi_k$ – фазовый сдвиг, C – масштабирующий коэффициент. Последний вычислялся как

$$C = \sum_{j=-N_k/2}^{N_k/2} w\left(\frac{l+N_k/2}{N_k}\right), \quad (4)$$

где $w(t)$ – используемая оконная функция. Центральные частоты выделяемых спектральных полос удовлетворяли следующему соотношению:

$$f_k = f_1 2^{(k-1)/B}, \quad (5)$$

где $f_1$ – центральная частота нижней частотной полосы, $B$ – число полос в расчете на одну октаву (удвоение частоты). Значение Q-фактора (1) определялось как

$$Q = \frac{f_k}{f_{k+1}-f_k} = (2^{\frac{1}{B}} - 1)^{-1}. \quad (6)$$

CQT-преобразование за счет использования специфичной структуры выделяемых частотных полос (5) обеспечивало высокое временное разрешение по времени для области высоких частот и высокое разрешение по частоте для областей нижних частот.

*Слой графового внимания.* Основная идея, лежащая в основе слоев с графовым самовниманием состояла в том, чтобы представить обрабатываемые данные в виде графа с некоторыми состояниями вершин и вычислить векторные представления состояний каждой вершины, «обращая внимание» на состояние связанных с ней вершин. Входными данными для таких слоев являлись узлы графа с их признаковым описанием: $h = \{h_1, h_2, h_3, \dots, h_N\}$, где $h_i \in \mathbb{R}^F$, где N – количество узлов графа, F – количество признаков для узла. Далее в слое создавался набор новых вершин с признаками следующего уровня: $h' = \{h'_1, h'_2, h'_3, \dots, h'_N\}$, $h_i \in \mathbb{R}^{F'}$, которые являлись результатом работы слоя.

Первым шагом в данных вычислениях выступало получение коэффициентов внимания:

$$e_{ij} = \text{LeakyReLU}(a^T[Wh_i||Wh_j]), \quad (7)$$

где W – обучаемая матрица весов, α – функция внимания, $h_i$, $h_j$ – состояния смежных вершины графа, LeakyReLU – нелинейная функция активации [10], || – операция конкатенации векторов. После этого полученный ненормированный коэффициент внимания между вершинами i и j нормировался за счет использования функции softmax:

$$a_{ij} = \text{softmax}(e_{ij}) = \\ = \exp(e_{ij})/\sum_{k\in N_i}\exp(e_{ik}) \quad (8)$$

Полученная нормированная оценка внимания показывала, как узел j влияет на узел i и имела размерность $2F'$. Вычисление векторов признаков вершин выходного графа выполнялось следующим образом:

$$h'_i = \text{LeakyReLU}(\sum_{j\in N_i} a_{ij}Wh_j). \quad (9)$$

Преобразования (7-9) обобщались на случай так называемого мультивнимания. В этом случае для каждой вершины вычислялось несколько представлений $h'_i$. Каждое преобразование для вычисления подобного представления называлось «головой» внимания. Для вычисления

составного представления вершины графа использовалось объединение всех вычисленных K головами представлений:

$$h_i' = ||_{k=1}^{K} \text{LeakyReLU}(\sum_{j \in N_i} a_{ij}^k W^k h_j). \quad (10)$$

*Описание набора данных.* Для обучения и тестирования реализуемой архитектуры использовался набор данных, который применялся в конкурсе методов обнаружения поддельных голосовых сообщений ASVspoof 2019 [2]. Подмножество примеров для физического доступа включало в себя 3 не пересекающихся набора данных: для обучения, валидации и тестирования. Набор данных для обучения включает в себя 54000 записи подлинной и сфальсифицированной речи, данные для валидации содержат в себе 27000 различных записей, а данные для тестирования состоят из 80000 записей различного вида.

Условия формирования поддельных образцов речи заключались в следующем. Площадь помещения, где осуществлялась перезапись, составляла 7,5 м$^2$, с высотой потолка 2,7 м. Высота расположения приёмника/источника была зафиксирована на уровне 1,1 м. Предполагалось, что говорящий говорит в направлении микрофона. Сама атака повторного воспроизведения организовывалась путём скрытой записи положительной попытки доступа и последующего воспроизведения этой записи. В трех отдельных группах экспериментов менялось расстояние от динамика, воспроизводящего подделку до микрофона системы верификации. Записи делались в одной из трёх зон: до 0,7 м, от 0,7 до 1,4м и свыше 1,4 м. Также записи разделялись по качеству динамика, воспроизводящего поддельную речь (три категории – низкого, среднего и высокого качества).

*Детали реализации предложенной нейросетевой модели.* В таблице 1 приведена архитектура оригинальной сети LCNN [4]. Ее можно разделить на две части: экстрактор признаков и блок принятия решения о классе обрабатываемого аудиосигнала. Последний устроен следующим образом: из тензора размера 5×12×32 вытягивается «плоское» представление в виде вектора длины 1920, далее применяется модифицированный двухслойный перцептрон, сжимающий представление до вектора размера 80, модифицированный за счет добавления промежуточного MFM-преобразования [7] между полносвязными слоями (обозначены как FC в таблице 1).

Можно заметить, что наибольшее число настраиваемых параметров сети сосредоточено как раз в первом полносвязном слое FC1. Его замена на «легкое» преобразование может существенно уменьшить нейростетевую модель в целом.

В таблице 2 приведена модифицированная архитектура сети, соответствующая рисунку 1. Блок экстракции признаков LCNN-сети сохранен полностью. Однако извлечение признаков осуществляется за счет трех преобразований: самовнимания на 5-вершинном графе, усреднении векторов-состояний вершин графа и малом полносвязном слое FC_1 для непосредственной классификации. Число параметров в слое графового самовнимания не превосходило 37 тысяч (случай 3 голов преобразования).

Обучение сети, описанной в таблице 2 проходило в течение 22 эпох с размером batch_size = 256 и скоростью обучения learning_rate=10-5. В качестве оптимизатора применялся алгоритм Adam. Функция потерь представляла собой разрежённую категориальную кросс-энтропию. В GAT-слое использовался механизм внимания с несколькими головами (от 1 до 3) для снижения дисперсии. В качестве функции нелинейности для вычисления оценок внимания использовалась функция LeakyReLU с углом наклона alpha = 0,2.

Основными метриками качества, контролируемыми при обучении, были точность (Accuracy), определяемая как частота совпадения предсказанных меток с истинными, и эквивалентная ошибка $\text{EER} = P_{fa}(\bar{\theta}) = P_{miss}(\bar{\theta})$, вычисляемая для такого порога принятия решения $\bar{\theta}$, когда ошибка ложного пропуска подделки $P_{fa}$

равнялась ошибке ложного недопуска подлинного сигнала $P_{miss}$. Критерием остановки был выбран показатель потерь на валидационной подвыборке: остановка обучения происходила тогда, когда точность не уменьшалась в течении последовательных 20 эпох.

Таблица 1. Нейронная сеть Light CNN (согласно [4])

| Блок сети | Тип слоя | Размер / Шаг фильтра | Размер выхода | Число параметров |
|---|---|---|---|---|
| LCNN-экстрактор признаков | Conv_1 | 5×5/1×1 | (84, 200, 64) | 1664 |
| | MFM_1 | - | (84, 200, 31) | 0 |
| | MaxPool_1 | 2×2/2×2 | (42, 100, 32) | 0 |
| | Conv_2 | 1×1/1×1 | (42, 100, 64) | 2112 |
| | MFM_2 | - | (42, 100, 32) | 0 |
| | BatchNorm_1 | - | (42, 100, 32) | 128 |
| | Conv_3 | 3×3/1×1 | (42, 100, 96) | 27744 |
| | MFM_3 | - | (42, 100, 48) | 0 |
| | MaxPool_2 | 2×2/2×2 | (21, 50, 48) | 0 |
| | BatchNorm_2 | - | (21, 50, 48) | 192 |
| | Conv_4 | 1×1/1×1 | (21, 50, 96) | 4704 |
| | MFM_4 | - | (21, 50, 48) | 0 |
| | BatchNorm_3 | - | (21, 50, 48) | 192 |
| | Conv_5 | 3×3/1×1 | (21, 50, 128) | 55424 |
| | MFM_5 | - | (21, 50, 64) | 0 |
| | MaxPool_3 | 2×2/2×2 | (10, 25, 64) | 0 |
| | Conv_6 | 1×1/1×1 | (10, 25, 128) | 8320 |
| | MFM_6 | - | (10, 25, 64) | 0 |
| | BatchNorm_4 | - | (10, 25, 64) | 256 |
| | Conv_7 | 3×3/3×3 | (10, 25, 64) | 36928 |
| | MFM_7 | - | (10, 25, 32) | 0 |
| | BatchNorm_5 | - | (10, 25, 32) | 128 |
| | Conv_8 | 1×1/1×1 | (10, 25, 64) | 2112 |
| | MFM_8 | - | (10, 25, 32) | 0 |
| | BatchNorm_6 | - | (10, 25, 32) | 128 |
| | Conv_9 | 3×3/3×3 | (10, 25, 64) | 18496 |
| | MFM_9 | - | (10, 25, 32) | 0 |
| | MaxPool_4 | 2×2/2×2 | (5, 12, 32) | 0 |
| Слои принятия решения о классе сигнала | Flatten | | (1, 1920) | 0 |
| | FC_1 | | (1, 160) | 307360 |
| | MFM_10 | - | (1, 80) | 0 |
| | BatchNorm_7 | - | (1, 80) | 320 |
| | Dropout | | (1, 80) | 0 |
| | FC_2 | - | (1, 2) | 162 |
| Всего параметров | | | | 466370 |

Таблица 2. Модифицированная нейронная сеть LCNN + GAT

| Блок сети | Тип слоя | Размер / Шаг фильтра | Размер выхода | Число параметров |
|---|---|---|---|---|
| LCNN-экстрактор признаков | Те же, что и в таблице 1 | | (5, 12, 32) | 158528 |
| Слои принятия решения о классе сигнала | GAT layer | - | (5, 32) | ≤ 37249 |
| | AvgPool | - | (1, 32) | |
| | FC_1 | 66 | (1, 2) | 66 |
| Всего параметров | | | | ≤ 195843 |

*Результаты и обсуждение.* Вначале была проведена серия экспериментов по выбору числа голов внимания в слое принятия решения. Результаты этих экспериментов приведены в таблице 3. Видно, что рост числа голов приводил к ожидаемому повышению качества работы (уменьшению ошибки EER и росту точности). Дальнейшее увеличение числа голов не проводилось, так как видно, что относительный рост качества замедлялся, а время проведения численных экспериментов и доступные вычислительные мощности были ограничены.

Таблица 3. Выбор оптимального числа голов внимания в GAT и сравнение с LCNN-сетью

| Номер эксперимента | Нейросетевая модель | Accuracy, % | EER, % | Число параметров, ×103 |
|---|---|---|---|---|
| - | LCNN | 95,09 | 13,80 | 466,4 |
| 1 | LCNN+GAT-1 голова | 92,71 | 11,78 | 171,2 |
| 2 | LCNN+GAT-2 головы | 93,90 | 8,91 | 183,1 |
| 3 | LCNN+GAT-3 головы | 93,68 | 9,20 | 196,6 |

Для сравнения в таблице 3 приведены результаты оценок качества для модели LCNN [4], обученной и протестированной на тех же данных. Видно, что предложенная замена классификатора на легкий слой графового внимания привело к повышению качества распознавания подделок. Одновременно существенно (более чем на 50%) уменьшилось число обучаемых параметров, что позволило существенно уменьшить время обучения моделей.

Далее были проведены эксперименты по сравнению качества работы (в виде ошибки EER) для базовой модели LCNN и предложенной LCNN+GAT с тремя головами внимания. В таблице 4 приведены оценки EER на тестовых подмножествах, соответствовавших различным расстояниям от микрофона до динамика и различному качеству динамика. В скобках указаны относительные изменения EER при переходе к новой модели. Видно, что наилучший прирост качества был обеспечен для ситуации близкого расстояния и низкого качества динамика. Это говорит об относительной устойчивости предложенного метода к искажениям, вносимым в регистрируемый микрофоном сигнал.

Таблица 4. Сравнение качества работы LCNN и LCNN+GAT на разных типах атак

| Тип атаки | EER, % | |
|---|---|---|
| | LCNN | LCNN+GAT-3 головы |
| Расстояние от источника до микрофона | | |
| < 0,7 м | 4,53 | 2,80 (-38 %) |
| 0,7-1,4 м | 4,90 | 2,46 (-50 %) |
| > 1,4 м | 11,24 | 9,75 (-13 %) |
| Качество динамика | | |
| Низкое | 6,57 | 3,61 (-45 %) |
| Среднее | 7,28 | 4,27 (-41 %) |
| Высокое | 12,82 | 9,36 (-27 %) |

*Заключение.* В данной работе предложен новый метод распознавания физических голосовых подделок типа повторного воспроизведения речи. Он основан на двухэтапном преобразовании входного аудиосигнала, состоящем из кодирования его частотного разложения с помощью легкой сверточной сети и последующего применения механизма графового внимания. Предложенный метод продемонстрировал высокое качество работы при меньшем числе параметров нейросетевой модели по сравнению с альтернативным подходом, ограничивавшимся применением сверточной сети.

*Исследование выполнено за счет гранта Российского научного фонда № 22–21–00199, https://rscf.ru/project/22-21-00199/*

## Библиографический список


1. *Wu Z., Evans N., Kinnunen T., Yamagishi J., Alegre F., Li H. Spoofing and countermeasures for speaker verification: A survey. // Speech Communication. 2015. Т. 66. С. 130–153.*
2. *Nautsch A., Wang X., Evans N., Kinnunen T. H., Vestman V., Todisco M., Delgado H., Sahidullah Md., Yamagishi J., Lee K.A. ASVspoof 2019: spoofing countermeasures for the detection of synthesized, converted and replayed speech // IEEE Transactions on Biometrics, Behavior, and Identity Science. 2021. № 3. С. 252–265.*
3. *Zhou J., Cui G., Zhang Z. Graph Neural Networks: A Review of Methods and Applications // AI Open. 2020. № 1(1). С. 57-81.*
4. *Lavrentyeva G., Novoselov S., Malykh E., Kozlov A., Kudashev O., Shchemelinin V. Audio replay attack detection with deep learning frameworks // Proc. Interspeech 2017, Stockholm, Sweden, 20-24 августа 2017. С. 82-86.*
5. *Petar V., Preixens G.C., Paga A. C., Romero A., Lio P., Bengio Y. Graph attention networks // ICLR 2018, Vancouver, Canada, 30 апреля-3 мая 2018. 12 с.*
6. *Todisco M., Delgado H., Enavs N. A New Feature for Automatic Speaker Verification Anti-Spoofing: Constant Q Cepstral Coefficients // Proc. The Speaker and Language Recognition Workshop (Odyssey 2016), Bilbao, Spain, 21-24 июня 2016. С. 283–290.*
7. *Wu X., He R., Sun Z., Tan T. A Light CNN for Deep Face Representation with Noisy Labels // IEEE Transactions on Information Forensics and Security. 2018. № 11(13). С. 2884–2896.*
8. *LeCun Y., Bottou L., Bengio Y., Haffner P. Gradient-Based Learning Applied to Document Recognition // Proceedings of the IEEE. 1998. № 11(86). С. 2278–2324.*
9. *Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // arxiv.org: сайт. URL: https://arxiv.org/abs/1502.03167/ (дата обращения: 15.10.2023).*
10. *Xu B., Wang N., Chen T., Li M. Empirical Evaluation of Rectified Activations in Convolutional Network // arxiv.org: сайт. URL: https://arxiv.org/abs/1505.00853/ (дата обращения: 15.10.2023).*