DOI: 10.15514/ISPRAS-2024-36(3)-6

SLAP – простая линейная атака на персептрон

А.И. Перминов, ORCID: 0000-0001-8047-0114 <*perminov@ispras.ru*>

Институт системного программирования им. В.П. Иванникова РАН,
Россия, 109004, г. Москва, ул. А. Солженицына, д. 25.

Аннотация. В статье представлен новый подход атаки на нейронные сети на основе перцептрона с кусочно-линейными функциями активации с использованием базовой линейной алгебры. Атака формулируется как система линейных уравнений и неравенств и демонстрирует упрощённый и эффективный в вычислительном отношении подход к созданию разнообразных наборов состязательных примеров. Алгоритмы предлагаемой атаки реализованы в коде, доступном в репозитории с открытым исходным кодом. В исследовании подчёркивается серьёзная проблема, которую представляет предлагаемая методология атаки для современной защиты нейронных сетей, подчеркивая острую необходимость в инновационных стратегиях защиты. Благодаря всестороннему изучению состязательных уязвимостей это исследование способствует повышению состязательной устойчивости машинного обучения, открывая путь для разработки более надежных и заслуживающих доверия систем искусственного интеллекта в реальных приложениях.

Ключевые слова: нейронные сети; состязательная атака; линейная алгебра; доверенный искусственный интеллект.

Для цитирования: Перминов А.И. SLAP – простая линейная атака на персептрон. Труды ИСП РАН, том 36, вып. 3, 2024 г., стр. 83–92. DOI: 10.15514/ISPRAS–2024–36(3)–6.

SLAP – Simple Linear Attack for Perceptron

A.I. Perminov ORCID: 0000-0001-8047-0114 <*perminov@ispras.ru*>

Ivannikov Institute for System Programming of the Russian Academy of Sciences,
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.

Abstract. This article introduces a new approach to tricking perceptron based neural networks with piecewise linear activation functions using basic linear algebra. By formulating the attack as a system of linear equations and inequalities, it demonstrates a streamlined and computationally efficient approach to generating diverse sets of adversarial examples. The algorithms for the proposed attack have been implemented in code, that accessible in the open-source repository. The study highlights the formidable challenge posed by the proposed attack methodology for contemporary neural network defenses, emphasizing the pressing need for innovative defense strategies. Through a comprehensive exploration of adversarial vulnerabilities, this research contributes to the advancement of adversarial robustness in machine learning, paving the way for the development of more reliable and trustworthy artificial intelligence systems in real-world applications.

Keywords: neural networks; adversarial attack; linear algebra; trustworthy artificial intelligence.

For citation: Perminov A.I. SLAP – Simple Linear Attack for Perceptron. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 3, 2024. pp. 83-92 (in Russian). DOI: 10.15514/ISPRAS-2024-36(3)-6.

1. Введение

В постоянно развивающемся мире машинного обучения и искусственного интеллекта нейронные сети служат важной основой, расширяющей широкий спектр приложений. Их широкое распространение связано с их способностью моделировать сложные взаимосвязи в данных и делать точные прогнозы. Однако за своей кажущейся надёжностью нейронные сети обладают уязвимостью, которая часто остаётся незамеченной: они подвержены состязательным атакам.

Состязательные атаки на нейронные сети используют присущие им уязвимости в границах принятия решений, что приводит к ошибочным прогнозам с потенциально серьёзными последствиями. Эти атаки, часто незаметные для людей-наблюдателей, манипулируют входными данными, вызывая неправильную классификацию или давая ошибочный прогноз. В данной статье основной акцент сделан на перцептроне с кусочно-линейными функциями активации (ReLU, Leaky ReLU и Abs), которые используются в подавляющем большинстве моделей других архитектур. Алгоритмы предлагаемой атаки реализованы в коде и доступны в репозитории с открытым исходным кодом [1].

В то время как традиционные состязательные атаки [2] обычно используют методы итеративной оптимизации для создания возмущений, нацеленных на конкретные экземпляры, предлагаемый метод отклоняется от этой парадигмы. Используя принципы линейной алгебры, он формулирует задачу генерации состязательных примеров в виде системы алгебраических уравнений, предлагая новый подход, способный исчерпывающе исследовать пространство решений. В отличие от итеративных методов, которые создают один состязательный пример, предлагаемый подход может генерировать полный набор примеров атак, подчёркивая многогранные уязвимости, присущие моделям перцептрона.

Предлагаемая методология атаки предлагает значительное преимущество с точки зрения теоретической простоты и простоты исполнения. Используя принципы линейной алгебры для решения систем алгебраических уравнений, процесс атаки становится вычислительно эффективным. В отличие от итеративных алгоритмов, обычно используемых в состязательных атаках, которые могут потребовать обширных вычислительных ресурсов и итераций, этот подход обеспечивает простые средства генерации состязательных примеров. Эта простота не только облегчает практическую реализацию атаки, но и повышает ее масштабируемость, позволяя исследовать уязвимости в различных архитектурах нейронных сетей.

Более того, универсальность атаки выходит за рамки только перцептронов и охватывает свёрточные нейронные сети (CNN), которые по своей архитектуре тоже являются перцептронами, но с разреженными и общими матрицами весов. Таким образом, предложенная методология атаки может быть легко перенесена на CNN для генерации состязательных примеров. Такая адаптивность подчеркивает широкую применимость и эффективность предлагаемого подхода при исследовании уязвимостей различных архитектур нейронных сетей.

Более того, скрытый характер атаки создаёт серьёзную проблему для обнаружения состязательных примеров во время развёртывания модели. В типичных сценариях обслуживания пользователи не имеют доступа к внутренним параметрам модели, что делает традиционные механизмы обнаружения неэффективными по сравнению с предлагаемой методологией. Эта неизбежная трудность в распознавании моделей атак подчёркивает острую необходимость в надёжных стратегиях противоборствующей защиты, выходящих за рамки традиционных методологий обнаружения. Поскольку состязательные атаки продолжают развиваться и усложняться, решение проблемы их обнаружения становится первостепенным для обеспечения надежности и достоверности развернутых моделей нейронных сетей в реальных приложениях.

2. Связанные работы

В последние годы состязательным атакам на нейронные сети уделяется значительное внимание, что привело к увеличению количества исследований, направленных на понимание и смягчение их воздействия. Градиентные алгоритмы оптимизации, такие как Fast Gradient Sign Method (FGSM) [3], представленный Goodfellow et al. и алгоритм Projected Gradient Descent (PGD) [4], предложенный Madry et al. стали выдающимися методологиями создания состязательных искажений. Эти методы используют градиент функции потерь по входным данным для итеративного искажения входных выборок, тем самым генерируя состязательные примеры. Несмотря на свою эффективность, эти методы часто влекут за собой трудоёмкие итерационные процессы и могут создавать единичные состязательные примеры, ограничивая их масштабируемость и разнообразие.

В отличие от подходов, основанных на градиенте, в недавних исследованиях изучались альтернативные методологии создания состязательных примеров. Например, Croce and Hein [5] представили метод, основанный на линейном программировании, для генерации состязательных возмущений, демонстрирующий конкурентоспособную производительность по сравнению с методами, основанными на градиенте. Точно так же Kolter and Wong [6] предложили подход выпуклой релаксации для создания состязательных примеров, используя методы выпуклой оптимизации. Эти методы отличаются от традиционных методов итеративной оптимизации, предоставляя рациональные и эффективные в вычислительном отношении альтернативы для создания состязательных возмущений.

В этом контексте предлагаемая методология представляет новый подход к состязательным атакам на нейронные сети, используя принципы линейной алгебры и формулируя атаку как систему линейных неравенств. Отход от итеративных методов оптимизации обеспечивает простоту и вычислительную эффективность, позволяя генерировать полные наборы состязательных примеров в различных архитектурах нейронных сетей. Преодолевая ограничения методов, основанных на градиенте и сохраняя при этом вычислительную гибкость, разработанный подход вносит свой вклад в более широкий спектр исследований состязательных атак, предлагая новые идеи и возможности для изучения уязвимостей нейронных сетей.

3. Предлагаемая атака

Пусть имеется обученный перцептрон $c(x)$ с кусочно-линейными функциями активации (например, ReLU) для классификации изображений (для других доменов дальнейшие рассуждения также будут корректными) и два изображения – x_t и x_a.

- x_t - целевое изображение.
- x_a - изображение, которое будет атаковано.
- Пусть n - входной размер вектора изображения.
- Пусть m $(m < n)$ - количество выходов (классов) модели.
- Пусть $y_t = c(x_t)$ – результат применения перцептрона к целевому изображению.

Основная цель атаки – найти входной пример x, такой, что $y = c(x)$ будет точно соответствовать значению y_t или иметь тот же класс, что и y_t (менее строгая цель), и при этом x будет максимально похож на x_a и максимально непохож на x_t (рис. 1). Более формально это можно записать следующим образом (система (1)):

$$\begin{cases} \|x_a - x\| \rightarrow min \\ \|x_t - x\| > 0 \\ c(x) = c(x_t) \text{ или } argmax\ c(x) = argmax\ c(x_t) \end{cases} \quad (1)$$

Кроме того, для повышения скрытности целесообразно наложить ограничения на допустимый диапазон входных значений переменной x, обозначаемый как

$[x_{min}, x_{max}]$, если это возможно. В контексте изображений разумно предположить, что все значения пикселей попадают в интервал от 0 до 1.

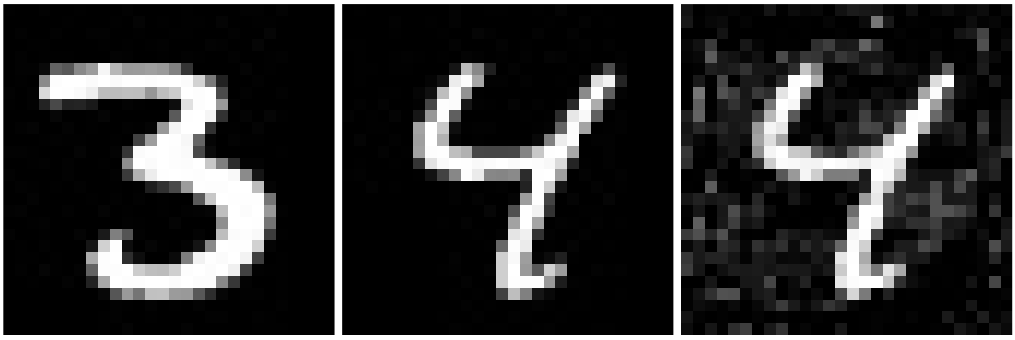

Рис. 1. x_t — целевое изображение (слева), x_a — атакуемое изображение (в центре) и x — атакующее изображение (справа)
Fig. 1. x_t - target image (left), x_a - image for attack (center) and x - attacked image (right)

3.1 Атака на однослойный перцептрон

Рассмотрим сценарий атаки, нацеленный на модель, состоящую из единственного слоя, характеризующегося весовой матрицей W с размерностью $m \times n$ и вектором смещения b с размерностью $m \times 1$, слой моделирует функцию $y = Wx + b$. Функцию активации можно игнорировать или выбирать произвольно, учитывая, что основной целью атаки является получение значений выхода слоя. Рассмотрим два сценария: один, в котором диапазон входных значений можно игнорировать, и другой, в котором его необходимо учитывать.

3.1.1 Атака, игнорирующая ограничения на значения x

Этот сценарий представляет собой наиболее простую формулировку проблемы, решение которой влечёт за собой применение фундаментальных понятий линейной алгебры (рис. 2):

- выберем подматрицу W_1 размера $m \times m$ (первые или случайные столбцы).
- оставшуюся подматрицу размера $m \times (n - m)$ назовём W_2.
- аналогично разделим x_a на x_{a_1} и x_{a_2}.
- первые m значений атаки получим как $x^* = W_1^{-1} \cdot \left(b^T - W_2 \cdot x_{a_2}\right)$.
- атакующее изображение получается путём конкатенации x^* и x_{a_2}.

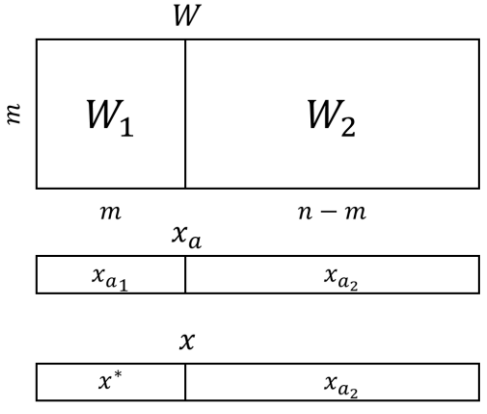

Рис. 2. Схема матричной атаки
Fig. 2. Diagram of matrix attack

Рис. 3. Пример применения матричной атаки на наборе данных CIFAR10 [7] с выбором случайных столбцов, диапазон значений x составляет [-1055, 926]
Fig. 3. Example of matrix attack application on the CIFAR10 [7] dataset with selecting random columns, x range is [-1055, 926]

Минусы:

- Атака применима только в том случае, если имеется прямой доступ к тензорам (впрочем, почти все состязательные атаки представляют собой алгоритмы белого ящика).
- Не воспроизводится при сохранении x (кроме случаев сохранения вектора как есть, что может быть выгодно, поскольку визуальный осмотр не выявит заметных дефектов изображения).

Плюсы:

- Практически всегда применима (за редким исключением, поскольку вероятность того, что W_1 необратима, близка к нулю).
- Минимально заметна для человека (поскольку действует исключительно в пределах тензора).

Пример атаки показан на рис. 3. Изменение изображения атаки кажется почти незаметным. Однако если попытаться повторно отправить то же изображение в сеть, атака не удастся. Это происходит потому, что при сохранении изображения диапазон пикселей сжимается до 0-255, что усиливает скрытность атаки, но также ограничивает ее применимость.

3.1.2 Атака, учитывающая ограничения на значения x

Описанный выше подход становится непрактичным при необходимости учитывать ограничения на диапазон входных значений для x. Чтобы решить эту проблему, предлагается использовать решатель задачи квадратичного программирования (QP) [8] в постановке, показанной в системе (2).

$$QP_{task}: \begin{cases} \frac{1}{2}x^T Px + q^T x \to min \\ Ax = b \\ Gx \leq h \\ l_b \leq x \leq r_b \end{cases} \quad (2)$$

Возьмём единичную матрицы E в качестве матрицы P, $-x_a$ в качестве q, W и $y_t - b$ в качестве A и b соответственно и положим $G = 0$ и $h = 0$. В результате этих замен задача будет переформулирована в соответствии с системой (3).

$$QP_{attack}: \begin{cases} \frac{1}{2} x^T P x + x_a^T x \to \min \\ Wx = y_t - b \\ x_{min} \le x \le x_{max} \end{cases} \quad (3)$$

Если решение этой проблемы существует, выходные данные перцептрона для результирующего вектора x будут совпадать со значения модели на x_t, сохраняя при этом диапазон входных значений. К недостаткам этого метода можно отнести его не 100% применимость и необходимость относительно обширной серии изменений (рис. 4) изображения для атаки. Однако главным преимуществом этого подхода является устойчивость к сохранению тензора в виде изображения.

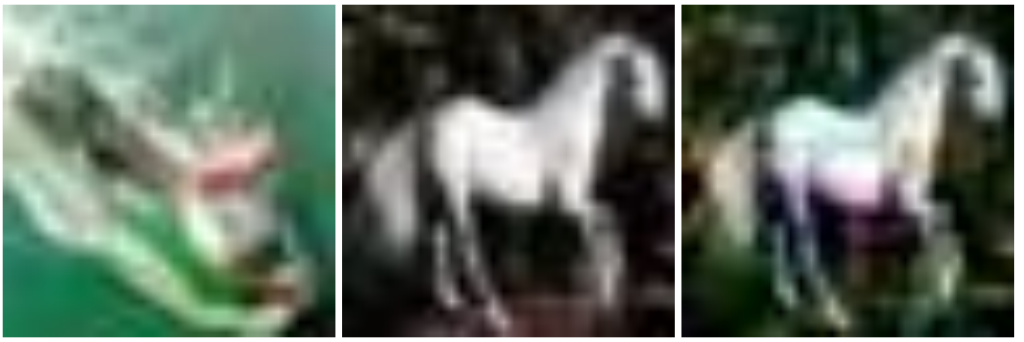

Рис. 4. Пример применения атаки QP на наборе данных CIFAR10 [7] — лошадь справа имеет тот же выход перцептрона, что и корабль.
Fig. 4. Example of QP attack application on the CIFAR10 [7] dataset - the horse on the right has the same perceptron output as the ship

Чтобы повысить вероятность существования решения, можно использовать менее строгую цель атаки. Вместо того, чтобы требовать точное совпадение значений перцептрона, возможным подходом является стремление к приблизительному сходству. Для достижения этой цели необходимо использовать два неравенства, которые легко сформулировать с помощью матриц G и h (система (4)):

$$\begin{cases} Wx + b \ge y_t - \epsilon \\ Wx + b \le y_t + \epsilon \end{cases} \to \begin{cases} -Wx \le b - y_t + \epsilon \\ Wx \le -b + y_t + \epsilon' \end{cases} \quad (4)$$

где ϵ - небольшое положительное число.

3.1.3 Атака на многослойный персептрон

При использовании перцептрона, состоящего из двух и более слоёв, возникает проблема нелинейности из-за функций активации на выходах этих слоёв (за исключением последнего). При соблюдении условий, изложенных ранее, используем факт кусочной линейности выбранных функций активации. Без ограничения общности предположим, что эти функции активации имеют единственную точку сочленения в нуле (верно для ReLU, Leaky ReLU и Abs). Следовательно, зафиксировав знаки слагаемых, предшествующих функции активации, можно эффективно устранить нелинейность.

Табл.1. Значения функций активации для разных знаков x
Table 1. Values of activation functions for different signs of x

Функция активации	$f(x)$ если $x < 0$	$f(x)$ если $x \ge 0$
ReLU ($max(0, x)$)	0	x
Leaky ReLU ($max(ax, x)$)	ax	x
Abs ($\|x\|$)	$-x$	x

Как видно из табл. 1, выход остаётся линейной функцией независимо от знака x. Это означает, что последовательность из двух слоев может быть преобразована в один слой при условии учёта знаковых ограничений. Проиллюстрируем это на примере трёхслойного перцептрона (уравнение (5)):

$$y = W_3 \cdot f_2(W_2 \cdot f_1(W_1 x + b_1) + b_2) + b_3 \tag{5}$$

Предполагая, что все функции активации являются Abs и все значения первого слоя неотрицательны, первый слой можно раскрыть в соответствии с системой (6).

$$\begin{cases} W_1 x + b_1 \geq 0 \\ y = W_3 \cdot f_2(W_2 W_1 x + W_2 b_1 + b_2) + b_3 \end{cases} \tag{6}$$

Предположим, что все значения второго слоя неположительны (поэтому требуется раскрытие с отрицательным знаком). В этом случае раскрытие второго слоя можно сформулировать в соответствии с системой (7):

$$\begin{cases} W_1 x + b_1 \geq 0 \\ W_2 W_1 x + W_2 b_1 + b_2 \leq 0 \\ y = -(W_3 W_2 W_1 x + W_3 W_2 b_1 + W_3 b_2) + b_3 \end{cases} \tag{7}$$

Введём дополнительные переменные: $W_{21} = W_2 W_1$, $b_{21} = W_2 b_1 + b_2$, $W_{321} = -W_3 W_2 W_1$ и $b_{321} = b_3 - W_3 W_2 b_1 - W_3 b_2$. Тогда система неравенств подвергнется следующему преобразованию (система (8)):

$$\begin{cases} W_1 x + b_1 \geq 0 \\ W_{21} x + b_{21} \leq 0 \\ y = W_{321} x + b_{321} \end{cases} \tag{8}$$

Следовательно, все нелинейности будут устранены и это позволит решить задачу с использованием того же QP решателя.

Возникает принципиальный вопрос: как выбирать знаки для раскрытия нелинейностей? В худшем случае необходимым перебрать все возможные комбинации знаков. Однако, учитывая размеры слоёв, такой подход крайне непрактичен. Альтернативная стратегия предполагает использование тех же знаков, которые образуются в процессе прохождения целевого или атакуемого изображений (или их комбинация с некоторым шумом) по сети. В первом случае решение гарантировано, хотя и похоже на x_t, а не на x_a, что фактически приводит к неудачной атаке. Оценка вероятности существования решения в последнем случае выходит за рамки данного исследования. Результат примера применения атаки показан на рис. 5.

Рис. 5. Пример многослойной перцептронной атаки на набор данных Cat vs Dog [9]: целевое изображение (слева), изображение для атаки (в центре) и атакованное изображение (справа).
Fig. 5. Example of a multilayer perceptron attack on Cat vs Dog dataset [9]: target image (left), image for attack (center) and attacked image (right)

3.1.4 Бонус: генерация произвольного входа с тем же выводом, что и y_t

Входная размерность слоя обычно значительно превосходит выходное измерение. Линейный оператор, составляющий один слой перцептрона, выполняет сжимающее преобразование, в результате чего получается почти бесконечное количество потенциальных входных данных для данного выходного сигнала. В сфере доверенного ИИ исследователи стремятся использовать результаты работы сети для оценки уверенности модели в её прогнозах. Однако, учитывая вышеупомянутые обстоятельства, достижение этой цели редко возможно без реализации дополнительных проверок.

Получение произвольных изображений с выходными данными, идентичными y_t, возможно путём замены q любым случайным вектором и/или заменой P случайной диагональной матрицей, содержащей значения в пределах входного диапазона. Результат такой атаки изображен на рис. 6. Для каждого последующего изображения, отображаемого справа от исходного, выходные данные перцептрона точно совпадают с выходными данными целевого изображения.

Следует отметить, что аналитические решатели способны вывести набор уравнений, а не полагаться на численный решатель QP. Этот подход позволяет генерировать почти бесконечный массив образов атак путём фиксации свободных переменных и вычисления зависимых переменных на основе полученного аналитического решения.

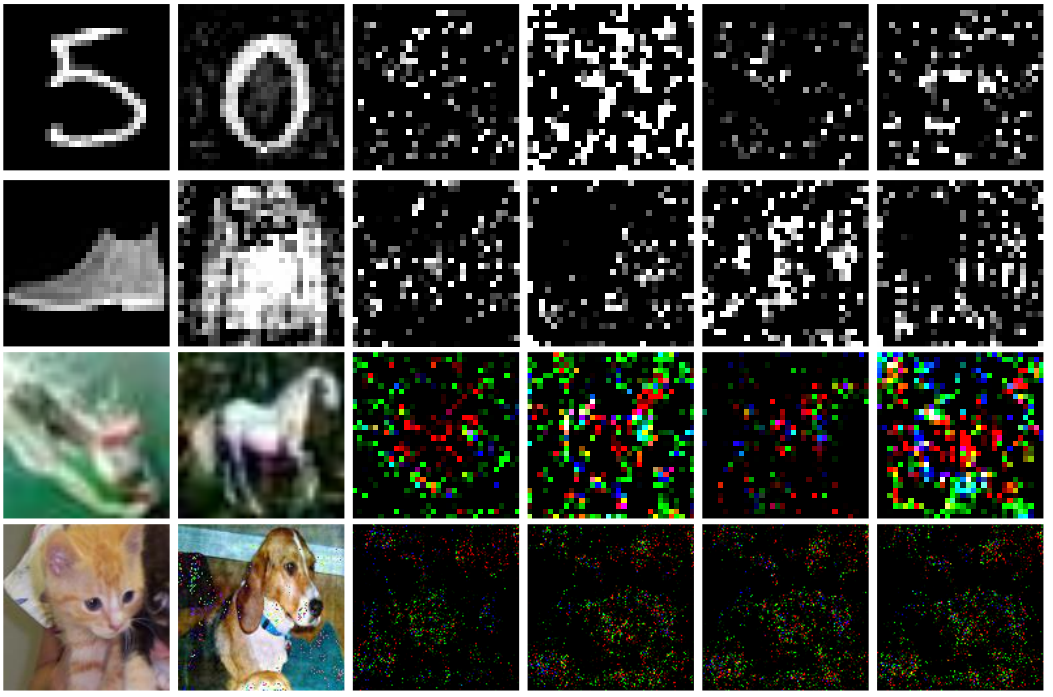

Рис. 6. Пример целевой (второе изображение) и случайной (изображения 3–6) атак — все изображения имеют одинаковый выход перцептрона.
Fig. 6. Example of target (second image) and random (images 3-6) attacks - all images have the same output of perceptron

4. Эксперименты

Зачастую алгоритмы атак проходят тестирование на установленных моделях, таких как ResNet [10], AlexNet [11], EfficientNet [12] и других. Однако общепризнанных широко используемых полносвязных сетей нет. Поэтому для оценки предложенного алгоритма были обучены различные перцептроны на традиционных наборах данных, а именно MNIST [13] и

CIFAR10 [7]. Обученные модели и их соответствующая точность представлены в первом столбце табл. 2. Все модели обучались в течение 40 эпох с размером пакета 32, оптимизатором Adam и скоростью обучения 0.0004.

Атака была протестирована в двух вариантах. В первом, целью было скопировать значения изображения из другого класса. Во втором же целью было получить на выходе персептрона значения, соответствующие другому классу.

Алгоритм атаки работает следующим образом:

- тестовая часть набора данных была разделена на пакеты;
- для каждого изображения в пакете случайным образом выбиралось изображение из того же пакета, но с другим классом;
- атакующее изображение было создано так, чтобы оно было похоже на тестовое изображение, но имело характеристики случайно выбранного изображения.

Табл.2. Результаты оценки атаки
Table 2. Attack evaluation results

Модель	**Датасет**	**Accuracy**	**Атака (цель - значения)**		**Атака (цель - класс)**	
			$\|\|x - x^*\|\|_\infty$	**Accuracy**	$\|\|x - x^*\|\|_\infty$	**Accuracy**
784-10	MNIST	0.9288	0.019	0.003	0.019	0.002
784-10-10		0.9326	0.021	0.007	0.022	0.001
784-100-10		0.9805	0.052	0.009	0.051	0.005
784-1000-10		0.9849	0.091	0.012	0.092	0.009
784-160-80-40-20-10		0.9792	0.117	0.000	0.114	0.000
3072-10	CIFAR10	0.3989	0.027	0.014	0.024	0.011
3072-100-10		0.4853	0.054	0.032	0.055	0.018
3072-1000-10		0.5236	0.095	0.041	0.096	0.023
3072-320-160-80-40-10		0.5353	0.121	0.049	0.119	0.037

Результаты экспериментов представлены в табл. 2. Столбцы $\|x - x^*\|_\infty$ показывают среднюю норму разницы между исходными примерами и построенными на их основе атакуемыми изображениями. Согласно таблице, предложенный алгоритм эффективно создаёт атакующие примеры, очень похожие на исходные изображения. В глубоких моделях видно, что различия между исходным и атакующим изображениями более выражены, что объясняется более сложным ландшафтом, образованным большой системой неравенств.

5. Заключение

В данном исследовании представлена новая методология атаки, использующая принципы линейной алгебры для использования уязвимостей, присущих нейронным сетям на основе перцептрона. Используя упрощенный подход, основанный на решении систем линейных неравенств, эта методология существенно отличается от традиционных итеративных методов, предоставляя упрощенные, но эффективные средства генерации атакующих примеров. Предложенная атака применима не только к перцептронным сетям, но и к свёрточным нейронным сетям, что подчеркивает её универсальность и эффективность в различных архитектурах нейронных сетей. Предложенные алгоритмы атаки реализованы в коде, доступном с открытом репозитории [1].

Список литературы / References

[1]. Репозиторий SLAEAttack (ссылка недоступна при слепом просмотре).
[2]. Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., & Mukhopadhyay, D. (2021). A survey on adversarial attacks and defences. CAAI Transactions on Intelligence Technology, 6(1), 25-45.

[3]. Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.
[4]. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.
[5]. Croce, F., & Hein, M. (2019). Sparse and imperceivable adversarial attacks. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 4724-4732).
[6]. Wong, E., & Kolter, Z. (2018, July). Provable defenses against adversarial examples via the convex outer adversarial polytope. In International conference on machine learning (pp. 5286-5295). PMLR.
[7]. Cifar10 dataset, https://www.cs.toronto.edu/ kriz/cifar.html. Last accessed 12 Mar 2024
[8]. QP solvers, https://pypi.org/project/qpsolvers/. Last accessed 12 Mar 2024
[9]. Cats and Dogs Dataset, https://www.microsoft.com/enus/ download/details.aspx?id=54765. Last accessed 12 Mar 2024
[10]. Targ, S., Almeida, D., & Lyman, K. (2016). Resnet in resnet: Generalizing residual architectures. arXiv preprint arXiv:1603.08029.
[11]. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25.
[12]. Tan, M., & Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning (pp. 6105-6114). PMLR.
[13]. MNIST dataset, https://www.kaggle.com/datasets/hojjatk/mnist-dataset. Last accessed 12 Mar 2024

Информация об авторах / Information about authors

Андрей Игоревич ПЕРМИНОВ является аспирантом, стажером-исследователем. Его научные интересы включают цифровую обработку сигналов, нейросетевую обработку данных, разработку доверенных моделей и алгоритмов машинного обучения и создание искусственных данных.

Andrey Igorevich PERMINOV is a postgraduate student, researcher. His research interests include digital signal processing, neural network data processing, development of trusted models and machine learning algorithms and creation of artificial data.