**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

|  |  |
| --- | --- |
|  | **ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ**  **Кафедра**  **«Криптология и кибербезопасность»** |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

ОТЧЕТ

О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

ИССЛЕДОВАНИЕ СПОСОБОВ СОВЕРШЕНИЯ ADVERSARIAL-АТАК НА НЕЙРОННЫЕ СЕТИ.

|  |  |  |
| --- | --- | --- |
| Исполнитель:  студент гр. Б16-505 | \_\_\_\_\_\_\_\_\_\_\_ (подпись, дата) | Свинин В.П. |
| Научный руководитель: | \_\_\_\_\_\_\_\_\_\_\_ (подпись, дата) | Бурлаков М.Е. |
| Зам. зав. каф. № 42: | \_\_\_\_\_\_\_\_\_\_\_ (подпись, дата) | Когос К.Г. |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва – 2019**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Реферат.**

**Содержание.**

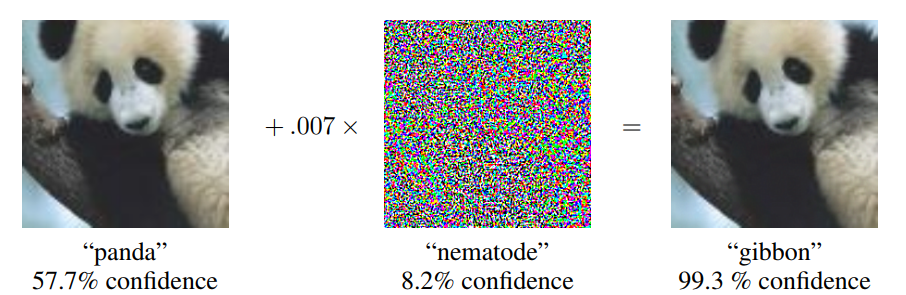
**Определения, обозначения и сокращения.**

В настоящем отчете применяются следующие термины с соответствующими определениями, обозначениями и сокращениями, используемые для описания adversarial-атак на системы глубокого обучения, решающие задачи компьютерного зрения. В следствие отсутствия устоявшейся терминологии, некоторые термины не были переведены и используются в изначальном виде. Пояснение к ним дано в скобках.

* Adversarial-атака – от англ. «adversary» – противник, соперник.
* Adversarial-пример/изображение – намеренно искаженная (например, путём добавления шума) версия исходного изображения, используемая для обмана алгоритма машинного обучения.
* Adversarial-искажение – шум, добавляемый к исходному изображению для того, чтобы сделать из него adversarial-пример.
* Adversarial-обучение – добавление adversarial-примеров к обучающей выборке с целью повышения устойчивости модели adversarial-атакам.
* Black-box атака – атака на выбранную модель, совершаемая без знания её структуры. Иногда допускается, что атакующий обладает ограниченными знаниями о структуре модели (например, о её процессе обучения или архитектуре), но не знает точные значения её параметров. В этом случае используется термин *gray-box* или *semi-black-box* атака.
* Детектор – позволяет определить, является ли изображение adversarial-примером.
* Fooling ratio (доля успешных обманов) – доля изображений, на которых модель меняет результат предсказания после добавления к ним adversarial-искажений.
* Одношаговые методы – методы создания adversarial-искажений, предполагающие однократное вычисление, например, вычисление градиента функции потерь модели. Напротив – итеративные методы, предполагающие совершение множества последовательных однотипных операций и являющиеся, как правило, вычислительно более сложными.
* Квазинезаметные искажения – искажения, практически не изменяющие изображение для человека.
* Rectifier (очиститель) – позволяет модифицировать adversarial-пример, восстанавливая исходное предсказание модели.
* Targeted-атаки (направленные атаки) – используются для того, чтобы модель неверно выбрала целевой класс (от англ. «target» - цель). Их противоположность – non-targeted-атаки (ненаправленные атаки), цель которых – заставать модель сделать любое неверное предсказание.
* Модель угроз – описывает типы потенциальных атак, в соответствии с моделью их совершения (например, black-box/white-box атака, направленная/ненаправленная атака).
* Transferability (переносимость) – отражает возможность использования adversarial-примеров на моделях, которые не были использованы при его создании.
* Универсальное искажение – искажение, имеющее высокую вероятность обмануть модель на любом выбранном изображении. Следует отметить, что универсальность – это свойство независимости искажения от класса изображения, в то время как переносимость – независимость искажения от используемой модели.
* White-box-атака предполагает, что у атакующего имеются полные сведения о модели, включая её параметры, архитектуру, метод обучения и обучающую выборку.

**1. Введение.**

Глубокое обучение заметно продвинуло машинное обучение в задачах компьютерного зрения (таких как классификация, распознавание и сегментация), задачах распознавания и генерации речи, машинного перевода и др. Последние годы заметно возросла доля систем, полагающихся на глубокое обучение, например, беспилотные автомобили, использующие системы глубокого обучения для навигации на дороге [12], [182]; дроны и роботы [157], [180]; банки использующие системы распознавания лиц в банкоматах [183]; системы распознавания вирусов, основанные на поиске аномалий в поведении с помощью глубокого обучения [34]; системы распознавания голоса [7]. Очевидно, что системы, полагающиеся на глубокое обучение, особенно в задачах компьютерного зрения, постепенно занимают всё более важную роль в повседневной жизни. Множество примеров использования глубокого обучения поднимает важные вопросы, связанные с безопасностью и защищенностью этих систем. Недавние исследования показывают, что системы глубокого обучения могут быть уязвимыми к специально сгенерированным входным данным, называемыми adversarial-примерами, остающимся для человека внешне неотличимым от исходных. Использование таких примеров может заставить классификатор (нейронную сеть) поменять своё мнение об исходном примере, при этом внушая высокую уверенность в неправильном решении. Более того, один и тот же adversarial-пример способен обмануть множество не связанных между собой cетей. Пример такого adversarial-изображения, нацеленного на обман классификатора, приведён на рис. 1.



В данном отчете приведено описание и классификация основных подходов создания adversarial-примеров и заданы основные условия и ограничения, определяющие возможность совершения с их помощью adversarial-атак. В заключении приведена оценка применимости существующих атак в реальных условиях.

**2. Классификация adversarial-атак.**

Классификацию подходов, применяемых для создания adversarial-примеров, можно провести исследуя их различия в трёх направлениях: модель угроз, adversarial-искажение и атакуемая модель/набор данных.

**2.1 Модель угроз.**

В соответствии с работой Papernot и др. [arXiv:1611.03814], все рассматриваемые adversarial-атаки объединены следующим набором свойств:

* Модель может быть атакована только на этапе тестирования, при этом атакующих способен изменять входные данные только после завершения обучения модели. Атакующий может иметь некоторую информацию о модели (такую как архитектура или параметры), но не способен изменять саму модель. Атаки на процесс обучения представляют собой отдельную тему для исследований и не будут рассмотрены в данной работе. [2) 38-42 - пара ссылок на работы?]
* В работе рассматриваются атаки на модели, построенные на основе нейронных сетей.
* Целью атакующего является нарушение целостности модели, выраженное в виде падения значения целевой метрики (доли успехов, F-меры и др.).

Далее, в зависимости от условий совершения, adversarial-атаки могут быть охарактеризованы типом вызываемой ошибки, наличием у атакующего информации о модели, направленностью атаки и количеством вычислительных шагов для создания adversarial-примера.

2.1.1 Тип вызываемой ошибки.

* Атаки, вызывающие ошибки первого рода (атакующий создаёт adversarial-пример, вызывающий ложной срабатывание – false positive). Например, в задаче распознавания вредоносных программ, ложным срабатыванием будет принятие обычной программы за зловредную. В задаче классификации за ложное срабатывание принимается случай, когда искаженное изображение неправильно классифицируется человеком (или не несёт для него смысла), при этом не меняя предсказание модели (см. рис. 2).
* Атаки, вызывающие ошибки второго рода (атакующий создаёт adversarial-пример, вызывающий пропуск события – false negative). При распознавании вредоносных программ ошибки этого вида приведут к пропуску вредоносной программы, а при классификации, adversarial-пример, легко классифицируемый человеком, не будет распознан моделью (или вызовет высокую уверенность в ложном классе).

  
Рис. 2. Пример ошибки первого рода. Нейронная сеть, обученная на ImageNet, показывает высокую уверенность (более 99.6%) при классификации adversarial-изображений. [83/2]

2.1.2 Наличие информации о модели у атакующего.

В зависимости от полноты информации об атакуемой модели, adversarial-атаки можно разделить на *white-box* и *black-box*.

* В случае white-box атаки, атакующему известна вся информация о модели: её параметры и архитектура, процесс обучения, гиперпараметры и использованная обучающая выборка. Большинство атак этого типа для генерации adversarial-примеров напрямую используют градиенты параметров модели.
* В случае block-box-атаки, атакующий выступает в роли простого пользователя, которому известна лишь реакция модели на входные данные (метка класса или его вероятность). Допустим вариант gray-box атаки, в которой у атакующего имеется ограниченная информация о модели (например, гиперпараметры, использовавшиеся в процессе обучения). Этот тип не выделяется в отдельный класс, т.к. дополнительной информации недостаточно для перехода к white-box атаке, и может быть классифицирован как black-box атака с дополнительными сведениями о модели.

2.1.3 Направленность атаки.

* Targeted атака (направленная атака) стремится заставить модель выбрать целевой класс, даже если он является неправильным. Например, если нейронная сеть, обученная на MNIST, на изображении из тестового набора выдаёт результат «0» и требуется изменить его на «7», то применяется *targeted* атака.
* Non-targeted атака (ненаправленная атака) стремится добиться любого неправильного результата (в случае бинарной классификации, атаки этих типов совпадают). Например, при совершении non-targeted атаки на систему распознавания лиц, атакующему стремится быть идентифицированным как любой другой человек. Ненаправленные атаки осуществляются проще, чем направленные, т.к. у атакующего имеется больше свободы в поиске adversarial-примеров. Обычно, ненаправленные атаки осуществляются одним из двух способов:   
  1) путём создания нескольких adversarial-примеров (с помощью направленной атаки) и выбором одного из них (например, наиболее похожего на исходное изображение);   
  2) путём минимизации вероятности, приписанной классификатором корректному классу.

2.1.4 Зависимость от количества шагов вычислений.

* Одношаговые (one-step) атаки предполагают однократное вычисление при поиске adversarial-примера, например, вычисление градиента функции потерь модели (FGSM).
* Итеративные методы атак предполагают многократное последовательное изменение исходного примера до превращения его в adversarial-пример. Часто атаки этого типа являются вычислительно более сложными, но приводят к лучшим результатам. Для некоторых задач (например, для обучения с подкреплением) атаки этого типа могут быть неприменимы в связи с трудностью вычисления adversarial-искажений.

**2.2 Adversarial-искажение.**

Поиск малых искажений, незаметных для человека, но заметно влияющих на предсказание модели является основной задачей adversarial-атак. Adversarial-искажения можно классифицировать, ориентируясь в трёх направлениях: универсальность искажения, способ его ограничения и используемая при этом норма.

* В зависимости от универсальности созданного искажения, атаки можно разделить на: 1) *индивидуальные*, создающие различные искажения для различных исходных примеров; 2) *универсальные*, создающие искажение, которое может быть успешно применено ко всем исходным примерам вне зависимости от их класса. Следует заметить, что в условиях реальных систем использование универсальных adversarial-искажений заметно упрощает задачу атакующего.
* Adversarial-искажения могут являться как результатом, так и ограничением оптимизационной проблемы. При оптимизации самого искажения атакующий стремиться уменьшить его величину так, чтобы оно было незаметно для человека, в то время как использование искажения в качестве ограничения просто требует от искажения быть достаточно малым.
* Оценку величины искажения можно производить, основываясь на его -норме:  
  Наиболее распространенными нормами являются , и :  
  – -норма равняется количеству пикселей, измененных в исходном изображении при создании adversarial-примера.  
  – -норма равняется евклидову расстоянию между исходным изображением и adversarial-примером.  
  – -норма равняется модулю максимального значения изменения пикселей исходного изображения.

**2.3 Атакуемая модель/набор данных.**

Исследователи оценивают эффективность adversarial-атак основываясь на разных атакуемых моделях и наборах данных, используемых для их обучения, что может привести к появлению неоднозначности. Поэтому при оценке эффективности необходимо учитывать:

* Использованный набор данных. MNIST, CIFAR-10 и ImageNet являются наиболее популярными наборами данных для оценки эффективности adversarial-атак, хотя в последнее время, из-за недостаточного объема и сложности, MNIST и CIFAR-10 становятся всё менее популярными. Для правильной оценки adversarial-атак требуется хороший набор данных.
* Атакуемая модель. Наиболее популярными архитектурами моделей для проведения атак являются LeNet, VGG, AlexNet, GoogLeNet, CaffeNet и ResNet. [ссылки]

**3. Известные adversarial-атаки.**

В этом разделе рассмотрены наиболее известные методы атак на нейронные сети; их эффективность оценивается на стандартных наборах данных (MNIST, CIFAR-10 или ImageNet) в типичных задачах компьютерного зрения. Следует заметить, что подходы, рассмотренные в данном разделе, составляют основу для атак в реальных условиях и каждый из них имеет потенциал повлиять на развитие глубокого обучения в практических задачах.  
 Обзор в данном разделе ведётся, в основном, в хронологическом порядке, к основным подходам и главным направлениям атак даётся краткое техническое описание, в то время как остальные подходы описаны в общих чертах, подробности читатель может найти в приложенных отсылках к исходным работам.

В части 3.1 рассмотрены атаки на наиболее распространённую задачу компьютерного зрения – классификацию/распознавание. [описать что где]

**3.1 Атаки на классификаторы изображений.**

3.1.1 Box-constrained L-BFGS (L-BFGS).

Впервые существование adversarial-искажений продемонстрировал Szegedy и др. в [22]. Пусть исходное изображение представлено в виде вектора , индекс «c» подчеркивает, что изображение «чистое» («clear»). Чтобы найти искажение , прибавление которого к исходному изображению изменит результат классификации, визуально не изменяя изображение, было предложено найти решение следующей задачи:

Авторы предложили найти нетривиальное решение (1), в котором предсказание классификатора отличается от исходного класса изображения. В этом случае нахождение становится нетривиальной проблемой, и для нахождения приблизительного решения авторы предлагают использовать алгоритм L-BFGS: находится минимальное для которого выполняется условие:

В случаях, когда функция потерь классификатора представляет собой выпуклую функцию, решение (2) даёт единственное решение. Но, в случае глубоких нейронных сетей, это, чаще всего, не так.  
 Для получения adversarial-примера, вычисленное искажение просто прибавляется к исходному изображению. Как показано на рис. (2), полученные adversarial-изображения визуально неотличимы от исходных изображений. Также было замечено, что искажения, вычисленные для нейронной сети могут работать и для других сетей.

Во время публикации работы считалось, что нейронная сеть выделяет признаки, определяющие пространство, где воспринимаемые между изображениями различия приблизительно равняются евклидовому расстоянию между ними. Эти противоречивые результаты пробудили интерес исследователей к adversarial-атакам.

3.1.2 Fast Gradient Sign Method (FGSM).

Szegedy и др. в [22] было замечено, что устойчивость нейронных сетей к adversarial-примерам может быть повышена за счёт adversarial-обучения. Для того, чтобы повысить эффективность adversarial-обучения, Goodfellow и др. в [23] бы разработан метод быстрого расчёта adversarial-искажений:

градиент функции стоимости в точке, определяемой параметрами модели ,

по отношению к исходному изображению ,скаляр, ограничивающих размер искажения.

Adversarial-примеры, полученные с помощью FGSM, используют «линейность» нейронных сетей в многомерном пространстве, хотя в то время такие модели считались сильно нелинейными. Goodfellow и др. в [23] выдвинули гипотезу, что дизайн современных нейронных сетей, намеренно поощряющий линейное поведение для вычислительной простоты (ReLU), делает их уязвимыми к простому поиску подобных искажений. Эта идея получила название «гипотезы линейности» [linearity hypothesis] и поддерживается существованием FGSM похода.  
 Kurakin и др. [80] отмечают, что adversarial-примеры, полученные с помощью FGSM для изображений из ImageNet, достигают успешного обмана сетей (top-1 error rate) в 63-69% случаев при Также авторы предложили «одношаговый направленный FGSM метод», в котором вместо использования в (3) исходного класса , используется вычисляется градиент по отношению к классу , имеющему наименьшую вероятность среди предсказанных сетью для изображения [не совсем направленная?]

Полученное таким образом искажение вычитается из исходного изображения, чтобы сделать из него adversarial-изображение. Для нейронной сети, использующей кросс-энтропию как функцию потерь, такая операция максимизирует вероятность выбора в качестве класса adversarial-примера. Предполагается, что использование случайного класса в качестве также может привести к успешному обману сети, хотя это может привести к менее интересным результатам, например к классификации одной породы собак в качестве другой. Авторы также показали, что использование adversarial-обучения повышает устойчивость сети к атакам с помощью FGSM и его вариантов.

FGSM искажает изображение, увеличивая значение функции потерь для правильного класса. Функция позволяет максимизировать величину искажения, в то время как ограничивает его равномерную норму (-норму) [проверить мат. термин]. Miyato и др. [103] предложил похожий метод вычисления adversarial-искажений:

В приведенном выражении, градиент нормализуется делением на -норму. Kurakin и др. [80] назвали этот способ «FGL2» методом и предложили использовать для нормализации равномерную норму, назвав получившуюся технику «FGL∞». Все эти методы рассматриваются как «одношаговые» методы вычисления adversarial-искажений.

3.1.3 Basic & Least-Likely-Class Iterative Methods (BIM & ILCM).

Одношаговые методы искажают изображения, делая один большой шаг в направлении, увеличивающим значение функции потерь классификатора. Интуитивное продолжение этой идеи – итеративное совершение множества маленьких шагов, сопровождающееся корректировкой направления движения на каждом шаге.  
 Простой итеративный метод (BIM) [35] предполагает именно это:

где – искаженное изображение на -ой итерации, - ограничивает значения пикселей изображения на , – определяет размер шага (обычно, ).

Алгоритм начинается с и совершает количество итераций, определяемое выражением . Было замечено [55], что BIM является эквивалентом PGM (Projected Gradient Descent) – стандартному методу оптимизации выпуклых функций.

По аналогии с модификацией FGSM до его «направленной» версии, Kurakin и др. [35] предложили расширить BIM до Iterative Least-likely Class Method (ILCM). В этом случае, в качестве класса в (5) предлагается использовать класс (с наименьшей вероятностью, данной классификатором). Было показано, что adversarial-примеры, полученные с помощью ILCM, сильно влияют на точность классификации сетей с архитектурой Inception v3, даже для малых значений .

3.1.4 Jacobian-based Saliency Map Attack (JSMA).

Наиболее частый подход к генерации квазинезаметных adversarial-изменений – это ограниение их или -норм. Papernot и др. [60] предложили способ проведения атак, путём ограничения -норм adversarial-искажения. На практике, это означает что такие искажения модифицируют только несколько пикселей исходного изображения. Основой их алгоритма является поочередная модификация пикселей исходного изображения и отслеживание изменений в результирующей классификации. Изменения отслеживаются путём построения карты градиентов выходных слоёв сети – в ней для каждого пикселя исходного изображения задаётся значение – большее значение соответствуют большей вероятности изменения предсказанного класса при изменении данного пикселя. После этого, алгоритм выбирает пиксель с наибольшей вероятностью обмануть сеть и изменяет его. Этот процесс повторяется до тех пор, пока не достигается максимальное количество измененных пикселей или пока не удастся обмануть классификатор.

3.1.5 One Pixel Attack

Крайним случаем adversarial-атаки можно считать изменение единственного пикселя исходного изображения. Su и др. [68] провели атаку на три разные сети с успехом на 70.97% тестовых изображений, изменяя значения всего лишь одного пикселя. Было также показано, что средняя уверенность классификатора на ложных примерах равнялась 97.47%. Примеры adversarial-изображений из [68] см. на (рис. 3).  
 Adversarial-примеры были получены используя метод дифференциальной эволюции [148]: были выбраны 400 векторов, содержащих xy-координаты и RGB значения случайных пикселей исходного изображения. Затем, значения векторов были случайным образом изменены, создавая следующее поколение, при этом потомок соревнуется со своим родителем, используя в качестве целевого критерия вероятность, полученную от классификатора. После нескольких итераций, для изменения изображения выбирается последний оставшийся потомок.

Даже с такой простой эволюционной стратегией Su и др. [68] удалось провести успешные атаки на нейронные сети. Следует заметить, что дифференциальная эволюция позволяет генерировать adversarial-примеры без доступа к параметрам и градиентам сети. Единственная необходимая информация – это вероятности, полученные от классификатора.

3.1.6 Carlini and Wagner Attacks (C&W).

При появлении методов защиты простив adversarial-изменений, Carlini и Wagner предложили набор из 3 атак, делающих изменений квазинезаметными путём ограничения их и -норм. Ими было показано, что данные атаки способны преодолеть механизм защиты, называемый defensive distillation [38][166]. Более того, было показано, что adversarial-примеры, полученные для незащищенных этим механизмом сетей, могут быть перенесены на защищенные, что допускает совершение black-box-атак.

В то время, когда наиболее частым методом совершения black-box-атак является использование переносимости adversarial-примеров, Chen и др. [41] предложили «Zeroth Order Optimization» атаку, заключающуюся в приблизительном вычислении градиентов атакуемой модели.

3.1.7 DeepFool.

Moosavi-Dezfooli и др. [72] предложили вычислять квазинезаметные искажения итеративным способом. Их алгоритм заключается в линеаризации функции потерь и вычисления минимального искажения, необходимого для изменения классификации при данной линеаризации. На каждом шаге алгоритма функция потерь вычисляется при добавлении полученного на прошлом шаге искажения. Так происходит до тех пор, пока классификатор не станет выдавать ошибочный результат (таким образом DeepFool можно отнести к ненаправленным атакам). Авторы [72] показывают что полученные adversarial- искажения имеют меньшие нормы, чем полученные с помощью FGSM и имеющие тот же процент успешных обманов.

3.1.8 Universal Adversarial Perturbations.

В то время, как FGSM, ILCM, DeepFool и др. методы нацелены на вычисление adversarial-изменений, обманывающих классификатор на единственном изображении, «универсальные» adversarial-искажения, вычисленные Moosavi-Dezfooli и др. [16], с высокой вероятностью подходят для любого изображения. Как показано на (рис. 1) эти искажения могут быть квазинезаметными для человека. Для их формального определения предположим, что исходные «чистые» изображения получены из распределения Ω. Тогда изменение можно назвать «универсальным», если оно удовлетворяет следующему ограничению:

где – вероятность, – процент успешных обманов, -норма,

– константа, определяющая величину искажения.

Строго говоря, искажения, удовлетворяющие (6) должны называться   
«-универсальным», подчёркивая сильную зависимость от названных параметров. Как бы то ни было, в литературе их обычно называют просто «универсальными».

Авторы [16] вычислили универсальные adversarial-искажения, ограничивая их и   
-норму, и показали, что искажения в 4% от нормы исходного изображения достигают значительного процента успешных обманов (80%) на лучших классификаторах. Их метод похож на метод, используемый в DeepFool – последовательного сдвига изображения к границе класса, определенной классификатором. Но в этом случае «все» изображения из тренировочной выборки последовательно сдвигаются к своим границам, накапливая суммарное искажение, ограничиваемое по -норме величиной .

Алгоритм, предложенный в [16] вычисляет adversarial-искажения, нацеленные на единственную модель. Тем не менее, было показано, что эти искажения также распространяются и на другие сети, особенно имеющие сходную архитектуру. В этом отношении, авторы [16] называют искажения «вдвойне универсальными». Более того, ими было показано, что высокого процента успешных обманов (более 50%) можно достичь на тренировочной выборке, состоящей всего лишь из 2000 изображений.

Другие методы получения универсальных искажений были предложены   
Хрульковым и др. [190] (позволяет получать хорошие результаты, используя маленькое количество изображений), Mopuri и др. [135] (получили универсальные искажения, независимые от данных).

3.1.9 UPSET и ANGRI.

Эти две black-box-атаки были предложены Sarkar и др. [146].  
UPSET (Universal Perturbations for Steering to Exact Targets): для n классов алгоритм вычисляет n независимых от изображения искажений, соответствующих направленному изменению классификации для любого изображения. UPSET основан на генерирующей сети , принимающей на вход выбранный класс и создающей направленное adversarial-искажение . Полностью, UPSET-сеть решает следующую оптимизационную проблему:

где значения пикселей изображения нормализованы в промежутке , а s – скаляр.

Значения пикселей искаженного изображения также ограничиваются интервалом .

ANGRI похожим образом вычисляет индивидуальные для каждого изображения искажения. Было показано, что оба алгоритма достигают высоких показателей на выборках из MNIST [10] и CIFAR-10 [152].

3.1.10 Houdini.

Cisse и др. [131] предложили алгоритм для обмана основанных на градиентном обучении подходов, подходящий для различных функций потерь. Типичные adversarial-атаки предполагают наличие дифференцируемой функции потерь, что иногда может не соблюдаться. Например, функция потерь в задаче распознавании речи может быть основана на доле правильно разобранных слов, что не предполагает прямого использования градиента. Houdini используется для создания adversarial-примеров для таких задач. Кроме удачного создания adversarial-примеров для классификации, Houdini был использован для совершения атаки на систему распознавания речи [151]. Также авторами была продемонстрирована переносимость атак на распознавание речи, путём совершения black-box-атаки на Google Voice.

3.1.11 Adversarial Transformation Networks (ATNs).

Baluja и Fisher [42] обучили нейронные сеть создавать adversarial-примеры для набора других нейронных сетей. Adversarial-примеры, созданные этими сетями, вычисляются путём минимизации составной функции потерь: первая часть обеспечивает квазинезаметность adversarial-искажений, а вторая направлена на изменение классификации результирующего изображения.

Hayex и Danezis [47] также использовали нейронную сеть для создания adversarial-примеров для black-box-атак. В представленных результатах, adversarial-примеры, созданные сетью остаются визуально неотличимыми от исходных изображений, но уменьшают точность классификации с 99.4% до 0.77% на MNIST [10] и с 91.4% до 6.8% на CIFAR-10 [152].

3.1.12 Другие атаки.

Описанные выше adversarial-атаки являются либо самыми популярными в недавней литературе, либо представителями основных направлений исследований. Ниже представлено краткое описание других методов совершения adversarial-атак.

Sabour и др. [26] описали вероятность создания adversarial-примеров путём изменения начальных слоёв нейронной сети. Авторы показали, что возможно изменить внутреннее представление adversarial-изображения, повторяющее представление изображений различных классов. Papernot и др. в [109] изучили переносимость adversarial-атак. Narodytska и Kasiviswanathan в [54] представили способ совершения black-box-атаки, путём изменения нескольких пикселей изображения. Liu и др. в [31] представили «эпсилон-окрестность»-white-box-атаку, успешность которой по отношению к защищенным через defensive distillation сети достигла 100%. Oh и др. в [133] применили теорию игр для разработки стратегии противодействия существующим способам защиты от adversarial-атак. Mpouri и др. в [135] разработали метод вычисления универсальных искажений. Hosseini и др. в [98] ввели понятие «семантические-adversarial-примеры» - изображения, которые обозначают одинаковые объекты для человека, но разные для классификатора. Kanbak и др. [73] предложили ManiFool – алгоритм, измеряющий устойчивость нейронной сети к геометрически искаженным изображениям. Dong и др. [170] предложил итеративный метод совершения adversarial-black-box-атак. Недавно, Carlini и Wagner [59] показали, что десять методов защиты от adversarial-атак могут быть преодолены, используя новые атаки, использующие измененные функции потерь. Rozsa и др. [94] предложили метод создания нескольких различных adversarial-примеров для исходного изображения. Следует упомянуть, что adversarial-искажения можно использовать для улучшения качества классификации. Yoo и др. в [195] предложили метод улучшения классификации, путём добавления небольших искажений к исходному изображению.

**Заключение.**

В соответствии с классификацией, предложенной в разделе 2, все описанные методы атак будут отличаться своей направленностью (targeted/non-targeted), информированностью атакующего (black-box/white-box), свойствами искажения (универсальность, используемая норма) и количеством шагов при его получении (одношаговые/итеративные). Все описанные атаки направлены на вызов ошибки 2 рода – изменение верного предсказания модели на ложное.

Таблица 1 – Классификация атак части 3.1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Атака | Информированность атакующего | Направленность | Количество шагов | Искажение | |
| Универсальность | Норма |
| L-BFGS | White-box | Targeted | Одношаговый | Индивидуальное |  |
| FGSM | White-box | Targeted | Одношаговый | Индивидуальное |  |
| BIM&ILCM | White-box | Non-targeted | Итеративный | Индивидуальное |  |
| JSMA | White-box | Targeted | Итеративный | Индивидуальное |  |
| One-Pixel | Black-box | Non-targeted | Итеративный | Индивидуальное |  |
| C&W | White-box | Targeted | Итеративный | Индивидуальное |  |
| DeepFool | White-box | Non-targeted | Итеративный | Индивидуальное |  |
| UAP | White-box | Non-targeted | Итеративный | Универсальное |  |
| UPSET | Black-box | Targeted | Итеративный | Универсальное |  |
| ANGRY | Black-box | Targeted | Итеративный | Индивидуальное |  |
| Houdini | Black-box | Targeted | Итеративный | Индивидуальное |  |
| ATNs | White-box | Targeted | Итеративный | Индивидуальное |  |

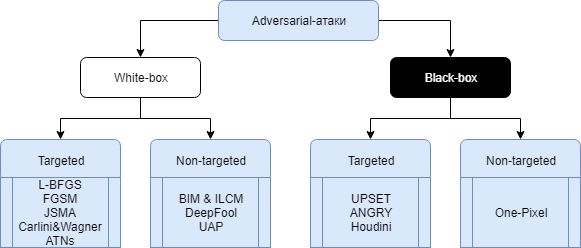


Рис. 3. Классификация наиболее известных атак в соответствии с их направленностью и информированностью атакующего.

**3.2. Известные adversarial-атаки в реальных условиях.**

**Список использованных источников.**