Случайное сглаживание: Теоретические основы и систематический обзор

К.А. Айрапетьянц, Е.А. Ильюшин

***Аннотация*—На сегодняшний день, когда системы искусственного интеллекта активно используются в различных сферах, проблема безопасности этих систем становится всё более актуальной. И конечно нейросетевые алгоритмы, которые мы на сегодняшней день и отождествляем с понятием «искусственный интеллект», также подвержены преднамеренным и не преднамеренным воздействиям, поэтому получение гарантий устойчивости их работы является важной задачей. Одним из методов позволяющих решать данную задачу является *«случайное сглаживание»* (randomized smoothing). С помощью данного метода мы можем получить формальные гарантии качества работы классификатора на заданном распределении данных. Метод случайного сглаживания, а также его модификации и будут рассмотрены в данном обзоре.**

***Ключевые слова*—случайное сглаживание, робастность нейронных сетей, сертифицированная точность, машинное обучение**

I. Введение

В современном мире, когда системы искусственного интеллекта интегрируются в различные сферы жизни, такие как: медицина, робототехника, промышленность, управление автомобилем т.д., вопрос безопасности систем ИИ становится очень остро.

Как и многие другие информационные системы, системы построенные на базе искусственных нейронных сетей (далее ИНС) подвергаются атакам злоумышленников ради получения своей выгоды, таким образом изучение атак и защит от них на системы данного класса становятся важными областями исследований для научного сообщества. Для решения многих прикладных задач используются классификаторы построенные на базе ИНС. Тогда, в условии, когда классификатор может быть атакован, нужен метод, который будет гарантировать его устойчивость (робастность) к атаке с какой-то вероятностью.

Такой метод *«случайное сглаживание»* (randomized smoothing) был предложен в работе [1], где исследуется возможность построения классификатора, устойчивого к состязательному шуму по норме l_2 и в качестве гарантии предоставляется сертифицированный радиус, внутри которого классификатор будет устойчив.

Далее, метод случайного сглаживания развивается в работах, где исследователи модифицируют процесс состязательного обучения, максимизируя радиус сертификации, пытаются применить случайное сглаживание в задаче сегментации и используют диффузионные модели.

Статья получена 17 ноября 2025

Каринэ Арсеновна Айрапетьянц, МГУ имени М.В. Ломоносова, (email: karine.ayrps@gmail.com).

Евгений Альбинович Ильюшин, МГУ имени М.В. Ломоносова, (email: john.ilyushin@gmail.com).

II. Теоретические основы вероятностной сертификации

A. Сертифицированная состязательная робастность с помощью случайного сглаживания

В работе [1] продемонстрировано, как преобразовать произвольный классификатор, корректно классифицирующий изображения с гауссовским шумом, в новый классификатор с гарантированной робастностью к состязательным возмущениям по норме ℓ_2.

Операция, которая называется *randomized smoothing* (случайное сглаживание) позволяет преобразовать любой базовый классификатор f в *сглаженный* классификатор g, сертифицированно устойчивый по норме ℓ_p. В данной работе рассматривается $p = 2$, также в работах [2] и [3] p принимает значения 1 и ∞ соответственно.

Основным ограничением случайного сглаживания является невозможность точного вычисления вероятностей классификации элементов из распределения $\mathcal{N}(x, \sigma^2 I)$ для нейросетевых классификаторов f. Следовательно, точная оценка предсказаний g для любого входа x и вычисление радиуса сертифицированной устойчивости невозможны. Для решения данных задач применяются алгоритмы Монте-Карло.

Определение II.1. Классификатор называют *сертифицированно устойчивым*, если для любого входа x гарантируется, что предсказание классификатора константно в некоторой области вокруг x.

Методы сертификации подразделяются на точные и консервативные. В контексте возмущений, ограниченных ℓ_p-нормой, точные методы для классификатора g, входа x и радиуса r определяют существование или отсутсвие возмущения δ такого, что $||\delta|| \leq r$, для которого $g(x) \neq g(x + \delta)$. Консервативные методы либо сертифицируют отсутсвие такого возмущения, либо отказываются производить сертификацию (даже если доказано, что такого возмущения не существует).

Точные методы обычно основаны на SMT или целочисленном линейном программировании, а некоторые из них ограничивают глобальную константу Липшица для нейронной сети и характеризуются низкой масштабируемостью в отличие от консервативных методов. Консервативные методы стремятся к наименьшему вмешательству в данные для сохранения их структуры и особенностей, минимизируя искажения.

1) Случайное сглаживание:

Определение II.2. Пусть $f : \mathbb{R}^d \to \mathcal{Y}$ – произвольный классификатор, сопоставляющий входы из $\mathbb{R}^d$ классам из $\mathcal{Y}$. Для любого входа x сглаженный классификатор g возвращает класс, который базовый классификатор f

предсказывает с наибольшей вероятностью при добавлении изотропного гауссовского шума::

$$g(x) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}(f(x+\epsilon) = c), \quad (1)$$

где $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$.

Изотропный гауссовский шум предполагает, что свойства шума одинаковы во всех направлениях пространства. В контексте анализа многомерных сигналов это означает, что шум имеет одинаковую дисперсию и корреляционные свойства во всех направлениях, т.е. его статистические характеристики не зависят от ориентации или направления.

Эквивалентное определение: $g(x)$ возвращает класс c, прообраз которого $\{x' \in \mathbb{R}^d : f(x') = c\}$ имеет наибольшую вероятность на распределении $\mathcal{N}(x, \sigma^2 I)$. Уровень шума σ – гиперпараметр g, регулирующий соотношение между устойчивостью и точностью, и остается постоянным независимо от входа x.

2) Гарантии робастности: Предположим, что при классификации элементов из распределения $\mathcal{N}(x, \sigma^2 I)$ базовый классификатор f возвращает наиболее вероятный класс c_A с вероятностью p_A, а второй по вероятности класс возвращается с вероятностью p_B. Основной результат заключается в том, что сглаженный классификатор g обладает устойчивостью вокруг точки x по норме ℓ_2 с радиусом

$$R = \frac{\sigma}{2}(\Phi^{-1}(p_A) - \Phi^{-1}(p_B)), \quad (2)$$

где Φ^{-1} – функция, обратная к функции распределения стандартного нормального распределения.

Результат остается справедливым при замене p_A его нижней доверительной границей $\underline{p_A}$, а p_B его верхней доверительной границей $\overline{p_B}$.

Теорема II.1. *Пусть $f : \mathbb{R}^d \to \mathcal{Y}$ – произвольная детерминированная или случайная функция и $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$. Определим $g(x) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}(f(x+\epsilon) = c)$.*

Предположим, $c_A \in \mathcal{Y}$ и $\underline{p_A}, \overline{p_B} \in [0,1]$ [1] удовлетворяют условию [2]:

$$\mathbb{P}(f(x+\epsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}(f(x+\epsilon) = c) \quad (3)$$

Тогда, $g(x+\delta) = c_A \quad \forall ||\delta||_2 < R$, где $R = \frac{\sigma}{2}(\Phi^{-1}(p_A) - \Phi^{-1}(p_B))$.

Замечание II.1. Теорема II.1 не накладывает ограничений на природу классификатора f.

Замечание II.2. Сертифицированный радиус $R \to \infty$ при $\underline{p_A} \to 1$ и $\overline{p_B} \to 0$[3] и высоком уровне шума σ.

Теорема II.2. *Пусть $\underline{p_A} + \overline{p_B} \leq 1$. Тогда для любого возмущения δ с $\|\delta\|_2 > R$ существует базовый классификатор f, соответствующий распределению вероятностей классов*

$$\mathbb{P}(f(x+\epsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}(f(x+\epsilon) = c), \quad (4)$$

для которого $g(x+\delta) \neq c_A$.

Теорема II.2 демонстрирует, что гауссовское сглаживание естественным образом влечет ℓ_2-робастность: при отсутствии дополнительных предположений относительно базового классификатора, выходящих за рамки указанных вероятностей классов, множество возмущений, к которым доказуемо устойчив сглаженный по Гауссу классификатор, представляет собой в точности ℓ_2-шар.

3) Масштабируемость: Поскольку формула для радиуса R не зависит явно от размерности данных d, может возникнуть предположение о снижении эффективности случайного сглаживания для изображений большей размерности. Однако изображения высокого разрешения способны выдерживать более высокие уровни изотропного гауссового шума σ, до разрушения классоопределяющих признаков. Следовательно, при большом разрешении сглаживание может применяться с большими значениями σ, что приводит к увеличению радиуса сертификации.

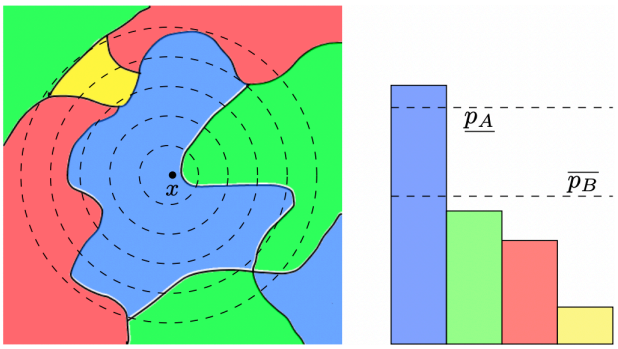


Рис. 1. Результат работы сглаженного классификатора на изображении x. Слева: области решений базового класиификатора f, обозначенные разными цветами. Пунктирные линии – изолинии распределения $\mathcal{N}(x, \sigma^2 I)$. Справа: распределение $f(\mathcal{N}(x, \sigma^2 I))$, $\underline{p_A}$ – нижняя доверительная граница вероятности наиболее вероятного класса, $\overline{p_B}$ – верхняя доверительная граница вероятности остальных классов. Предсказание $g(x)$ отмечено синим цветом.

4) Алгоритм: Для оценки предсказания сглаженного классификатора $g(x)$ необходимо определить класс c_A с максимальным весом в распределении $f(x+\epsilon)$. Рассмотрим вспомогательные функции для предсказания и сертификации:

- SampleUnderNoise(f, x, num, σ);
  - генерирует num шумов из распределения $\epsilon_1, \ldots, \epsilon_{num} \sim \mathcal{N}(0, \sigma^2 I)$;
  - получает предсказания $f(x+\epsilon_1), \ldots f(x+\epsilon_{num})$ путем прогона зашумленных изображений через классификатор;
  - возвращает счетчик количества предсказаний для каждого класса c;
- BinomPValue$(n_A, n_A + n_B, p)$ – возвращает p-значение гипотезы о том, что $n_A \sim$ Binomial$(n_A + n_B, p)$;
- LowerConfBound$(k, n, 1-\alpha)$ – возвращает односторонний нижний $(1-\alpha)$-доверительный интервал для биномиального параметра p при условии что $k \sim$ Binomial(n, p). Другими словами, функция возвращает некоторое число $\underline{p}$ для которого $\underline{p} \leq p$ с вероятностью как минимум $1-\alpha$ по сэмплам $k \sim$ Binomial(n, p).

[1] нижняя граница, при которой классификатор выдает c_A; верхняя граница, при которой классификатор выдает c_B

[2] классификатор f выдает c_A с наибольшей вероятностью среди всех классов

[3] Гауссово распределение определено на всем $\mathbb{R}^d$ и единственный случай, когда $f(x+\epsilon) = c_A$ с вероятностью 1 это когда $f = c_A$ почти всюду.

Алгоритм 1 Сертификация и предсказание

```
// Оценим g на x
function Predict(f, σ, x, n, α)
    counts ← SampleUnderNoise(f, x, n, σ)
    ĉ_A, ĉ_B ← топ 2 класса из counts n_A, n_B ← counts[ĉ_A], counts[ĉ_B]
    if BinomPValue(n_A, n_A + n_B, 0.5) ≤ α then
        return ĉ_A
    else
        return ABSTAIN
    end if
end function

// Сертифицируем робастность g относительно x
function Certify(f, σ, x, n_0, n, α)
    counts0 ← SampleUnderNoise(f, x, n_0, σ)
    ĉ_A ← топ класс в counts0
    counts ← SampleUnderNoise(f, x, n, σ)
    p_A ← LowerConfBound(counts[ĉ_A], n, 1 − α)
    if p_A > 1/2 then
        return предсказание ĉ_A и радиус σ Φ^{-1}(p_A)
    else
        return ABSTAIN
    end if
end function
```

B. Теоретический компромисс между робастностью и точностью

В работе [4] рассматривается бинарный классификатор: пусть модель задаёт отображение $f : \mathbb{R}^d \to \mathbb{R}$, где $\mathbb{R}^d$ – пространство всевозможных входных данных.

Определение II.3. *Робастная ошибка* классификатора f определяется как:

$$\mathcal{R}_{\text{rob}}(f) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\mathbf{1}\{\exists x' \in B(x,\epsilon) \mid f(x')y \le 0\}, \quad (5)$$

где $B(x,\epsilon)$ обозначает шар радиуса ϵ с центром в точке x.

Определение II.4. *Естественная ошибка классификации* определяется как:

$$\mathcal{R}_{\text{nat}}(f) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\mathbf{1}\{f(x)y \le 0\}. \quad (6)$$

Следует отметить, что $\mathcal{R}_{\text{rob}}(f) \ge \mathcal{R}_{\text{nat}}(f)$ для любого классификатора f, при этом $\mathcal{R}_{\text{rob}}(f) = \mathcal{R}_{\text{nat}}(f)$ тогда и только тогда, когда $\epsilon = 0$.

Определение II.5. *Ошибка на границе решения* определяется как:

$$\mathcal{R}_{\text{bdy}}(f) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\mathbf{1}\{x \in \mathbb{B}(\text{DB}(f),\epsilon), f(x)y > 0\}, \quad (7)$$

где $\mathbb{B}(\text{DB}(f),\epsilon)$ обозначает ϵ-окрестность границы решения классификатора f, то есть множество

$$\{x \in \mathbb{R}^d : \exists x' \in B(x,\epsilon) \mid f(x)f(x') \le 0\}.$$

Из введенных обозначений непосредственно следует декомпозиция робастной ошибки:

$$\mathcal{R}_{\text{rob}}(f) = \mathcal{R}_{\text{nat}}(f) + \mathcal{R}_{\text{bdy}}(f). \quad (8)$$

В статье доказываются теоремы II.3, II.4, которые позволяют получить оптимальные верхние и нижние оценки для $\mathcal{R}_{\text{rob}}$.

Предположение 1 (Калиброванная функция потерь)**.** Предположим, что функция потерь ϕ является калиброванной, то есть для $\eta \ne 1/2$ выполняется $H^-(\eta) > H(\eta)$, где

$$H(\eta) := \inf_{\alpha\in\mathbb{R}} C_\eta(\alpha) := \inf_{\alpha\in\mathbb{R}} \left(\eta\phi(\alpha) + (1-\eta)\phi(-\alpha)\right), \quad (9)$$

$$H^-(\eta) := \inf_{\alpha:(2\eta-1)\alpha\le 0} C_\eta(\alpha). \quad (10)$$

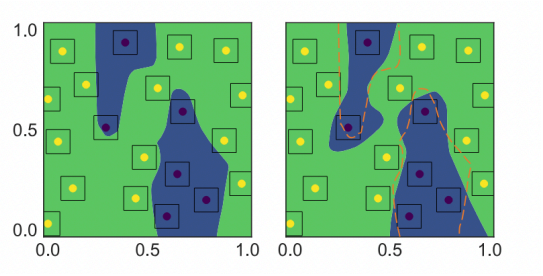


Рис. 2. Сравнение границ решения классификаторов. Слева: граница решения при стандартном обучении. Справа: граница решения при использовании предложенного метода робастного обучения.

Теорема II.3 (Верхняя оценка робастной ошибки)**.** *Пусть* $\mathcal{R}_\phi(f) := \mathbb{E}\phi(f(x)y)$ *и* $\mathcal{R}^*_\phi := \min_f \mathcal{R}_\phi(f)$.

При выполнении предположения 1 для любой неотрицательной функции потерь ϕ *такой, что* $\phi(0) \ge 1$, *любой измеримой функции* $f : \mathbb{R}^d \to \mathbb{R}$, *любого распределения на* $\mathbb{R}^d \times \{\pm 1\}$ *и любого* $\lambda > 0$ *выполняется:*

$$\begin{aligned}\mathcal{R}_{rob}(f) - \mathcal{R}^*_{nat} &\le \psi^{-1}(\mathcal{R}_\phi(f) - \mathcal{R}^*_\phi) \\ &\quad + \mathbb{P}[x \in \mathbb{B}(DB(f),\epsilon), f(x)y > 0] \\ &\le \psi^{-1}(\mathcal{R}_\phi(f) - \mathcal{R}^*_\phi) \\ &\quad + \mathbb{E}\max_{x'\in B(x,\epsilon)} \phi(f(x')f(x)/\lambda),\end{aligned} \quad (11)$$

где ψ *– некоторая функция, определяемая свойствами функции потерь* ϕ.

Теорема II.4 (Нижняя оценка робастной ошибки)**.** *Пусть* $|\mathbb{R}^d| \ge 2$.

При выполнении предположения 1 для любой неотрицательной функции потерь ϕ *такой, что* $\phi(t) \to 0$ *при* $t \to +\infty$, *любого* $\xi > 0$ *и любого* $\theta \in [0,1]$ *существуют распределение на* $\mathbb{R}^d \times \{\pm 1\}$, *функция* $f : \mathbb{R}^d \to \mathbb{R}$ *и параметр* $\lambda > 0$ *такие, что* $\mathcal{R}_{rob}(f) - \mathcal{R}^*_{nat} = \theta$ *и*

$$\begin{aligned}&\psi\left(\theta - \mathbb{E}\max_{x'\in\mathbb{B}(x,\epsilon)} \phi(f(x')f(x)/\lambda)\right) \\ &\le \mathcal{R}_\phi(f) - \mathcal{R}^*_\phi \\ &\le \psi\left(\theta - \mathbb{E}\max_{x'\in\mathbb{B}(x,\epsilon)} \phi(f(x')f(x)/\lambda)\right) + \xi.\end{aligned} \quad (12)$$

На основе полученных теоретических результатов авторы предлагают следующую оптимизационную задачу для обучения робастного классификатора:

$$\min_f \mathbb{E}\Big\{\underbrace{\phi(f(\boldsymbol{x})y)}_{\text{точность}} + \underbrace{\max_{\boldsymbol{x}'\in\mathbb{B}(\boldsymbol{x},\epsilon)} \phi(f(\boldsymbol{x})f(\boldsymbol{x}')/\lambda)}_{\text{робастность}}\Big\}, \quad (13)$$

где первое слагаемое отвечает за точность классификации на исходных данных, а второе – за робастность к состязательным возмущениям.

III. Случайное сглаживание по ℓ_p нормам

A. Неразмеченные данные для улучшения состязательной робастности

В работе [3] теоретически и эмпирическим доказывается, что состязательная робастность может быть значительно повышенна с помощью полуконтролируемого обучения.

В теоретической части используется упрощённая гауссовская модель [5], демонстрирующая разрыв в сложности разработки между стандартным и робастным классификатором –для создания робастного классификатора требуется значительно больше данных. Было доказано, что неразмеченные данные устраняют этот разрыв: простая процедура полуконтролируемого обучения позволяет достичь высокой робастной точности используя то же количество меток, что требуется для достижений стандартной точности.

В экспериментах набор данных CIFAR-10 [6] расширяется с помощью 500 тысяч неразмеченных изображений, полученных из 80 миллионов изображений маленького размера, а также используется робастное самообучение для того, чтобы превзойти state-of-the-art робастной точности по нормам ℓ_2 и ℓ_p:

a) Постановка полуконтролируемой задачи классификации: Рассматривается задача отображения $x \in \mathcal{X} \subseteq \mathbb{R}^d$ на множество меток $y \in \mathcal{Y}$. Пусть $\mathbb{P}_{X,Y}$ обозначает совместное распределение для пар (x, y), а $\mathbb{P}_X$ – маргинальное распределение для $\mathcal{X}$.

Обучающая выборка состоит из:

- Размеченных примеров: $(X, Y) = \{(x_i, y_i)\}_{i=1}^n \sim \mathbb{P}_{X,Y}$
- Неразмеченных примеров: $\tilde{X} = \{\tilde{x}_j\}_{j=1}^{\tilde{n}} \sim \mathbb{P}_X$

Цель состоит в обучении классификатора $f_\theta : \mathcal{X} \to \mathcal{Y}$ из семейства параметризованных моделей $\theta \in \Theta$.

b) Критерии оценки качества: Для оценки эффективности методов используются следующие метрики:

Определение III.1 (Точность)**.** Стандартная ошибка классификации определяется как:

$$\text{err}_{\text{standard}}(f_\theta) := \mathbb{P}_{(x,y)\sim\mathbb{P}_{X,Y}}[f_\theta(x) \neq y] \tag{14}$$

Определение III.2 (Точность на состязательных примерах)**.** Для возмущений в ℓ_p $(p = 2, p = \infty)$ шаре радиуса ϵ робастная ошибка определяется как:

$$\text{err}^{p,\epsilon}_{\text{robust}}(f_\theta) := \mathbb{P}_{(x,y)\sim\mathbb{P}_{X,Y}}[\exists x' \in B^p_\epsilon(x) : f_\theta(x') \neq y] \tag{15}$$

где $B^p_\epsilon(x) := \{x' \in \mathcal{X} \mid \|x' - x\|_{\ell_p} \leq \epsilon\}$ – ℓ_p шар радиуса ϵ с центром в x.

Определение III.3 (Сертифицированная точность)**.** Классификатор f_θ имеет сертифицированную точность ξ по норме ℓ_p, если можно доказать, что:

$$\text{err}^{p,\epsilon}_{\text{robust}}(f_\theta) \leq 1 - \xi \tag{16}$$

1) Алгоритм самообучения: Рассматривается алгоритм обучения с учителем A, который сопоставляет набор данных (X, Y) с параметрами модели θ. Самообучение представляет собой расширение A до полуконтролируемого обучения, состоящее из следующих этапов:

1) получение промежуточной модели $\hat{\theta}_{\text{intermediate}} = A(X, Y)$;
2) генерация псевдоразметки с помощью промежуточной модели $\tilde{y}_i = f_{\hat{\theta}_{\text{intermediate}}}(\tilde{x}_i)$ for $i \in [\tilde{n}]$;
3) объединение исходных данных и данных с псевдоразметкой и обучение финальной модели $\hat{\theta}_{\text{final}} = A([X, \tilde{X}], [Y, \tilde{Y}])$.

B. Случайное сглаживание по норме ℓ_1

В работе [2] рассматривается сертифицированная состязательная робастность сглаженных классификаторов по норме ℓ_1. В отличие от гауссовского случайного сглаживания, приводящего к ℓ_2-гарантиям, авторы используют равномерный шум с носителем ℓ_∞-типа.

Сертификация робастности формулируется не через явный радиус в ℓ_1-норме, а через оценку относительного объёма пересечения носителей распределений шума до и после возмущения входа. Сглаженный классификатор сохраняет предсказание до тех пор, пока объём пересечения двух сдвинутых ℓ_∞-кубов остаётся больше разности вероятностей между наиболее вероятным и вторым по вероятности классами.

Наряду с этим показывается, что ограничения по норме ℓ_1 возникают как достаточное условие ненулевого объёма пересечения, то есть ℓ_1-шар задаёт консервативное описание множества сертифицированно допустимых возмущений, но не является точной формой этого множества.

Ключевым вкладом работы является учет естественных box-ограничений входных данных (например, $x \in [0, 1]^d$ для изображений). Авторы показывают, что учет усеченного носителя шума увеличивает объём пересечения распределений при сдвиге и, как следствие, приводит к более сильным сертификационным гарантиям без изменения базового классификатора.

IV. Метрики

Эффективность сертификации робастности моделей может быть оценена различными способами. Основные метрики, используемые в качестве критериев оценки качества будут приведены в данном разделе.

Определение IV.1. *Сертифицированная точность на тестовом множестве в радиусе r* – доля тестового множества, классифицированная классификатором g корректно с предсказанием, которое сертифицированно робастно в ℓ_2-шаре радиуса r;

Определение IV.2. *Приблизительная сертифицированная точность на тестовом множестве* – доля тестового множества, которую процедура Certify из алгоритма 1 классифицирует корректно и сертифицирует робастность с радиусом $R \geq r$.

Определение IV.3. *Робастная точность* классификатора f определяется как:

$$\mathcal{A}_{\text{rob}}(f) = 1 - \mathcal{R}_{\text{rob}}(f), \tag{17}$$

где $\mathcal{R}_{\text{rob}}(f)$ — робастная ошибка, определенная в уравнении (5).

Определение IV.4. *Естественная точность* классификатора f определяется как:

$$\mathcal{A}_{\text{nat}}(f) = 1 - \mathcal{R}_{\text{nat}}(f), \tag{18}$$

где $\mathcal{R}_{\text{nat}}(f)$ — естественная ошибка классификации, определенная в уравнении (6).

Данные метрики IV.3, IV.4 позволяют количественно оценить компромисс между робастностью и точностью классификации: высокая робастная точность $\mathcal{A}_{\text{rob}}(f)$ указывает на устойчивость модели к состязательным атакам,

в то время как высокая естественная точность $\mathcal{A}_{\text{nat}}(f)$ свидетельствует о хорошей производительности на исходных данных без возмущений.

Определение IV.5. *Эмпирическая робастная точность* – доля тестового множества, атакованного различными атаками (SmoothAdv [7], случайными атаками или адаптивными атаками (Random+ и PGD [8])), которую сглаженный классификатор g классифицирует корректно.

Определение IV.6. *Средний радиус сертификации (ACR)* – для каждого тестового примера (x, y) и модели g оценивается радиус сертификации $CR(g; x, y)$ V.8. Средний радиус сертификации вычисляется как:

$$\text{ACR} = \frac{1}{|\mathcal{S}_{\text{test}}|} \sum_{(x,y)\in\mathcal{S}_{\text{test}}} CR(g; x, y), \tag{19}$$

где $\mathcal{S}_{\text{test}}$ – тестовое множество.

Определение IV.7. *Верифицированная ошибка* – доля тестовых примеров с возмущениями по норме ℓ_∞, на которых модель даёт неверные предсказания.

Определение IV.8. Способ оценки качества для top-k предсказаний VI-C:

- для каждого примера x с меткой l вычисляется сертифицированный радиус $\underline{R_l}$ с помощью алгоритма 6;
- вычисляется сертифицированная top-k точность в радиусе r как доля тестового множества, чей сертифицированный радиус составляет как минимум r.

Помимо описанных выше метрик используются также и другие критерии, в зависимости от решаемой задачи. К примеру, в методах, которые развивают идеи случайного сглаживания в задаче сегментации могут быть использованы следующие метрики: mIoU (mean Intersection over Union), средняя точность по пикселям, процент воздержаний (%∅) (доля пикселей, для которых модель воздержалась от прогноза) Кроме того, для оценки качества методов используется стандартная метрика Accuracy, а также время выполнения.

V. Модификация процесса обучения

A. SmoothAdv: состязательное обучение сглаженных классификаторов

В работе [7] авторы впервые демонстрируют атаку на сглаженный классификатор и ее применение для состязательного обучения сглаженных моделей. Кроме того, представлено более краткое доказательство теоремы II.1 через нелинейное свойство Липшица сглаженного классификатора.

Для демонстрации атаки дается расширенное определение сглаженного классификатора через мягкие классификаторы.

Определение V.1. *Мягкий классификатор* – это функция $F : \mathbb{R}^d \to P(\mathcal{Y})$, где $P(\mathcal{Y})$ обозначает множество вероятностных распределений над множеством классов $\mathcal{Y}$. Нейронные сети обычно выдают мягкое распределение, а затем применяется операция arg max для получения финального предсказания (жесткий классификатор).

Алгоритм 2 SmoothAdv: состязательное обучение сглаженных классификаторов

function TrainMiniBatch($(x^{(1)}, y^{(1)})$, $(x^{(2)}, y^{(2)})$, ..., $(x^{(B)}, y^{(B)})$)
  Attacker $\leftarrow$ ($SmoothADV_{PGD}$ or $SmoothADV_{DDN}$)
  Generate noise samples $\epsilon_i^{(j)} \sim \mathcal{N}(0, \sigma^2 I)$ for $1 \le i \le m$, $1 \le j \le B$
  $L \leftarrow []$ *# Список состязательных примеров для обучения*
  for $1 \le j \le B$ **do**
    $\hat{x}^{(j)} \leftarrow x^{(j)}$ *# Инициализация состязательного примера*
    for $1 \le k \le T$ **do**
      Обновить $\hat{x}^{(j)}$ согласно k-му шагу Attacker, используя
      шумы $\epsilon_1^{(j)}$, $\epsilon_2^{(j)}$, ..., $\epsilon_m^{(j)}$ для оценки градиента функции потерь сглаженной модели согласно уравнению (23)
      # Переиспользование шумов между шагами атаки
    end
    Добавить в L: $((\hat{x}^{(j)} + \epsilon_1^{(j)}, y^{(j)}), (\hat{x}^{(j)} + \epsilon_2^{(j)}, y^{(j)}), \ldots, (\hat{x}^{(j)} + \epsilon_m^{(j)}, y^{(j)}))$
    # Переиспользование шумов для аугментации данных
  end
  Выполнить обратное распространение ошибки на множестве L с соответствующим шагом обучения

Определение V.2. Мягкому классификатору F соответствует *сглаженный мягкий классификатор* $G : \mathbb{R}^d \to P(\mathcal{Y})$, определенный как:

$$G(x) = (F * \mathcal{N}(0, \sigma^2 I))(x) = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I)}[F(x+\epsilon)], \tag{20}$$

где $*$ обозначает операцию свертки с гауссовским ядром.

Пусть $f(x)$ – жесткий классификатор, а g – соответствующий сглаженный жесткий классификатор. Поиск состязательных примеров для сглаженного жесткого классификатора g является сложной задачей из-за недифференцируемости операции arg max. Поэтому авторы предлагают искать состязательные примеры для сглаженного мягкого классификатора G.

Для пары (x, y) требуется найти точку $\hat{x}$, которая максимизирует функцию потерь G в ℓ_2-шаре вокруг x. В качестве функции потерь выбрана кросс-энтропия ℓ_{CE}.

Определение V.3. *Состязательное возмущение* для сглаженного мягкого классификатора определяется как:

$$\begin{aligned}\hat{x} &= \arg\max_{\|x'-x\|_{\ell_2}\le\epsilon} \ell_{\text{CE}}(G(x'), y) \\ &= \arg\max_{\|x'-x\|_{\ell_2}\le\epsilon} \left(-\log \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I)}[(F(x'+\epsilon))_y]\right),\end{aligned} \tag{21}$$

где $(F(x'))_y$ обозначает вероятность класса y согласно выходу классификатора F на входе x'.

Данная формула представляет функцию потерь для SmoothAdv – атаки на сглаженный классификатор. Оптимизация проводится с помощью методов Projected Gradient Descent (PGD) [8] и его вариантов.

1) Алгоритм состязательного обучения: Оптимизация функции потерь проводится методами первого порядка: PGD [8] и DDN [9] (Decoupled Direction and Norm). Основная вычислительная задача при реализации данных методов – вычисление градиента функции потерь по переменной x' для заданной точки (x, y). Пусть $J(x') = \ell_{\text{CE}}(G(x'), y)$ – функция потерь. Тогда:

$$\nabla_{x'} J(x') = \nabla_{x'} \left(-\log \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I)}[(F(x'+\epsilon))_y]\right). \tag{22}$$

Поскольку точное вычисление данного градиента является вычислительно затратным, применяются методы Монте-Карло. Из гауссовского распределения генериру-

ются выборки шума $\epsilon_1, \ldots, \epsilon_m \sim \mathcal{N}(0, \sigma^2 I)$, и математическое ожидание заменяется выборочным средним:

$$\nabla_{x'} J(x') \approx \nabla_{x'} \left(-\log \left(\frac{1}{m} \sum_{i=1}^{m} (F(x' + \epsilon_i))_y \right) \right). \quad (23)$$

Поскольку вычисления становятся затратными с ростом m, на практике для состязательного обучения используются значения $m_{\text{train}} \in \{1, 2, 4, 8\}$. Для оценки качества применяются большие значения $m_{\text{test}} \in \{1, 4, 8, 16, 64, 128\}$. Следует отметить, что хотя оценка функции потерь сходится к истинному значению, она представляет собой смещенную оценку градиента.

2) Альтернативный подход с использованием леммы Штейна: Рассмотрим альтернативный способ оптимизации функции потерь. Поскольку логарифм не изменяет arg max, достаточно минимизировать $G(x')_y$ относительно нормы ℓ_2:

$$\begin{aligned} \nabla_{x'}(G(x')_y) &= \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\sigma^2 I)}[\nabla_{x'}(F(x' + \epsilon))_y] \\ &= \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\sigma^2 I)} \left[\frac{\epsilon}{\epsilon^2} \cdot (F(x' + \epsilon))_y \right], \end{aligned} \quad (24)$$

где последнее равенство следует из леммы Штейна для гауссовских распределений. Для заданного сглаженного классификатора g используются алгоритмы Predict и Certify, аналогичные представленным в разделе 1.

B. MACER: максимизация радиуса сертификации

В работе [10] авторы предлагают непосредственно максимизировать радиус сертификации R без конкретизации каких-либо атак и утверждают, что обученная модель способна достичь доказанной робастности против любой возможной атаки в данном радиусе. В отличие от других методов максимизации радиуса сертификации, предложенный подход применим к архитектурам произвольного размера.

Рассмотрим стандартную задачу классификации с распределением данных p_{data} над парами (x, y), где $x \in \mathcal{X} \subset \mathbb{R}^d$ и $y \in \mathcal{Y} = \{1, 2, \ldots, K\}$. Обычно распределение p_{data} неизвестно, и доступно лишь обучающее множество $\mathcal{S} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. Обозначим $\hat{p}_{\text{data}}$ эмпирическое распределение (равномерное распределение над $\mathcal{S}$).

Определение V.4. Пусть $f_\theta \in \mathcal{F}$ – параметризованный классификатор $f_\theta : \mathcal{X} \to \mathcal{Y}$. *Состязательный пример* $x' = x + \delta$ для классификатора f_θ – это возмущенный вход, такой что f_θ корректно классифицирует исходный пример x, но некорректно классифицирует x', при условии $\|\delta\|_{\ell_2} \leq \epsilon$.

Определение V.5. Модель f_θ называется ℓ_2^ϵ-*робастной* в точке (x, y), если она корректно классифицирует x как y и для всех возмущений δ с $\|\delta\|_{\ell_2} \leq \epsilon$ модель классифицирует $x + \delta$ как y.

В задаче робастной классификации цель состоит в поиске модели, которая является ℓ_2^ϵ-робастной в точке (x, y) с наибольшей вероятностью над распределением $(x, y) \sim p_{\text{data}}$ для заданного $\epsilon > 0$.

По определению, ℓ_2^ϵ-робастность классификатора f_θ в любой точке (x, y) зависит от радиуса наибольшего ℓ_2-шара с центром в x, в котором f_θ не изменяет своего предсказания.

Определение V.6. *Радиус робастности* классификатора f_θ в точке (x, y) определяется как:

$$R(f_\theta; x, y) = \begin{cases} \inf_{f_\theta(x') \neq f_\theta(x)} \|x' - x\|_{\ell_2}, & \text{если } f_\theta(x) = y \\ 0, & \text{если } f_\theta(x) \neq y \end{cases} \quad (25)$$

Поскольку целью является обучение модели, которая является ℓ_2^ϵ-робастной в точке (x, y) с наибольшей вероятностью над $(x, y) \sim p_{\text{data}}$ для заданного $\epsilon > 0$, задача сводится к минимизации математического ожидания 0/1-ошибки:

Определение V.7. *Робастная 0/1-ошибка* определяется как:

$$\ell^{0/1}_{\epsilon\text{-rob}}(f_\theta; x, y) := 1 - \mathbf{1}_{R(f_\theta; x, y) \geq \epsilon}, \quad (26)$$

где $\mathbf{1}_{(\cdot)}$ – индикаторная функция.

Соответственно, минимизируется следующая функция потерь:

$$L^{0/1}_{\epsilon\text{-rob}} := \mathbb{E}_{(x,y) \sim p_{\text{data}}} \ell^{0/1}_{\epsilon\text{-rob}}(f_\theta; x, y). \quad (27)$$

Определение V.8. *Радиус сертификации* $CR(f_\theta; x, y)$ представляет нижнюю границу радиуса робастности $R(f_\theta; x, y)$ и удовлетворяет условию:

$$0 \leq CR(f_\theta; x, y) \leq R(f_\theta; x, y) \quad \forall f_\theta, x, y. \quad (28)$$

Радиус сертификации обеспечивает гарантированную верхнюю границу для 0/1-ошибки робастной классификации. Соответствующая ошибка определяется через радиус сертификации:

$$\ell^{0/1}_{\epsilon\text{-cert}}(f_\theta; x, y) := 1 - \mathbf{1}_{CR(f_\theta; x, y) \geq \epsilon}, \quad (29)$$

то есть пример считается корректно классифицированным только если радиус сертификации достигает значения ϵ.

$$L^{0/1}_{\epsilon\text{-cert}}(f_\theta) := \mathbb{E}_{(x,y) \sim p_{\text{data}}} \ell^{0/1}_{\epsilon\text{-cert}}(f_\theta; x, y). \quad (30)$$

Пусть g_θ – сглаженный классификатор, соответствующий базовому классификатору f_θ.

Для минимизации $\ell^{0/1}_{\epsilon\text{-rob}}$ или $\ell^{0/1}_{\epsilon\text{-cert}}$ предлагается декомпозиция ошибки на две компоненты: ошибку классификации $\ell_C(g_\theta; x, y)$ и ошибку робастности $\ell_R(g_\theta; x, y)$:

$$\ell(g_\theta; x, y) = \ell_C(g_\theta; x, y) + \ell_R(g_\theta; x, y). \quad (31)$$

Пусть $\hat{g}(x)$ – сглаженный мягкий классификатор, а z_θ – выход нейронной сети с примененной функцией softmax.

После наложения определенных ограничений на компоненты ошибки получается следующая функция потерь:

$$\begin{aligned} \ell(\hat{g}_\theta; x, y) &= \ell_C(\hat{g}_\theta; x, y) + \ell_R(\hat{g}_\theta; x, y) \\ &= -\log \hat{z}^y_\theta(x) \\ &\quad + \lambda \max\{\epsilon + \hat{\epsilon} - CR(\hat{g}; x, y), 0\} \cdot \mathbf{1}_{\{\hat{g}_\theta(x) = y\}} \\ &= -\log \hat{z}^y_\theta(x) \\ &\quad + \frac{\lambda\sigma}{2} \max\{\gamma - \hat{\xi}_\theta(x, y), 0\} \cdot \mathbf{1}_{\{\hat{g}_\theta(x) = y\}}, \end{aligned} \quad (32)$$

где:

Алгоритм 3 MACER: Робастное обучение через максимизацию радиуса сертификации

Input: Обучающее множество $\hat{p}_{data}$, уровень шума σ, количество гауссовских примеров k, trade-off фактор λ, hinge фактор γ, обратная температура β, параметры модели θ

for $i = 1, \ldots, l$ **do**

Сэмплировать минибатч $(x_1, y_1), \cdots, (x_n, y_n) \sim \hat{p}_{data}$

Для каждого x_i, сэмплировать k *н.о.р.с.в.* гауссовские примеры

$$x_{i1}, \cdots, x_{ik} \sim \mathcal{N}(x, \sigma^2 I)$$

Вычисить эмпирические матожидания:

$$\hat{z}_\theta(x_i) \leftarrow \sum_{j=1}^{k} z_\theta(x_{ij})/k for i = 1, \cdots, n$$

Вычислить

$$\mathbb{G}_\theta = \{(x_i, y_i) : \tilde{g}_\theta(x_i) = y_i\} : (x_i, y_i) \in \mathbb{G}_\theta \Leftrightarrow y_i = \arg\max_{c \in \mathcal{Y}} \hat{z}^c_\theta(x_i)$$

Для каждого $(x_i, y_i) \in \mathbb{G}_\theta$, вычислить

$$\hat{y}_i : \hat{y}_i \leftarrow \arg\max_{c \in \mathcal{Y} \setminus \{y_i\}} \hat{z}^c_\theta(x_i)$$

Для каждого $(x_i, y_i) \in \mathbb{G}_\theta$, вычислить

$$\hat{\xi}_\theta(x_i, y_i) : \hat{\xi}_\theta(x_i, y_i) \leftarrow \Phi^{-1}(\hat{z}^{y_i}_\theta(x_i)) - \Phi^{-1}(\hat{z}^{\hat{y}_i}_\theta(x_i))$$

Обновить θ с помощью одного шага любого метода оптимизации первого порядка для минимизации

$$-\frac{1}{n}\sum_{i=1}^{n} \log \hat{z}^{y_i}_\theta(x_i) + \frac{\lambda\sigma}{2n} \sum_{(x_i, y_i) \in \mathbb{G}_\theta} \max\{\gamma - \hat{\xi}_\theta(x_i, y_i), 0\}$$

end for

- $\eta_1, \ldots, \eta_k \sim \mathcal{N}(0, \sigma^2 I)$ — независимые одинаково распределенные случайные величины;
- $\hat{z}_\theta = \frac{1}{k}\sum_{j=1}^{k} z_\theta(x + \eta_j)$ — эмпирическое математическое ожидание $z_\theta(x + \eta)$;
- $\hat{\xi}_\theta(x, y) = \Phi^{-1}(\hat{z}^y_\theta(x)) - \Phi^{-1}(\max_{y' \neq y} \hat{z}^{y'}_\theta(x))$.

Во время обучения минимизируется $\mathbb{E}_{(x,y)\sim\hat{p}_{\text{data}}} \ell(\hat{g}_\theta; x, y)$. Для упрощения реализации в качестве гиперпараметра выбирается γ вместо $\hat{\epsilon}$. Обратная температура функции softmax β также является гиперпараметром метода.

C. SmoothMix: обучение откалиброванных по уверенности сглаженных классификаторов для сертифицированной робастности.

В работе [11] предлагается модификация процесса обучения для повышения качества сертифицированной робастности сглаженных классификаторов. Авторы выявляют ограничения метода SmoothAdv [7] и предлагают альтернативный подход к генерации состязательных примеров.

1) Мотивация и ограничения существующих методов: Основным недостатком метода SmoothAdv является то, что состязательные примеры генерируются с жестким ограничением по ℓ_2-расстоянию, что может приводить к переобучению модели к конкретной норме возмущений. Это ограничивает способность модели к обобщению и может снижать достижимый радиус сертификации.

Определение V.9. Пусть $f(x) = \arg\max_{c \in \mathcal{Y}} F(x)$, где $F : \mathbb{R}^d \to P(\mathcal{Y})$ – мягкий классификатор V.1, а g – соответствующий сглаженный классификатор. *Неограниченный состязательный пример* для точки (x, y) определяется как решение оптимизационной задачи:

$$\tilde{x} := \arg\max_{x'} \left(\mathcal{L}(g; x', y) - \beta \cdot \|x' - x\|^2_{\ell_2}\right), \tag{33}$$

где $\mathcal{L}$ – функция кросс-энтропии, $\beta > 0$ – гиперпараметр, регулирующий штраф за отклонение от исходного примера x.

В отличие от стандартного подхода с жестким ограничением $\|x' - x\|_{\ell_2} \leq \epsilon$, данная формулировка позволяет алгоритму адаптивно выбирать оптимальное расстояние возмущения в зависимости от локальных свойств функции потерь.

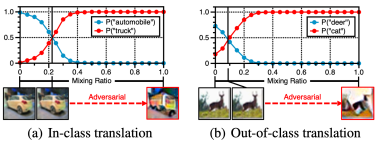
Аналогично методу SmoothAdv [7], для решения задачи (33) используется приближение сглаженного классификатора $G := \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I)}[F(x + \epsilon)]$ V.2.

Процесс поиска неограниченного состязательного примера $\tilde{x}$ реализуется итерационно за T шагов:

$$\tilde{x}^{(t+1)} := \tilde{x}^{(t)} + \alpha \cdot \frac{\nabla_x J(\tilde{x}^{(t)})}{\|\nabla_x J(\tilde{x}^{(t)})\|_{\ell_2}}, \tag{34}$$

где $J(x) := -\log\left(\frac{1}{m}\sum_{i=1}^{m} F_y(x + \epsilon_i)\right)$ и $\epsilon_i \sim \mathcal{N}(0, \sigma^2 I)$ – выборки гауссовского шума для аппроксимации математического ожидания.

2) Калибровка достоверности через интерполяцию: Ключевое наблюдение авторов заключается в том, что при переходе от исходного примера x к неограниченному состязательному примеру $\tilde{x}$ достоверность модели в предсказании класса изменяется резко и нелинейно, что может приводить к плохой калибровке вероятностей.



(a) In-class translation (b) Out-of-class translation

Рис. 3. Изменение достоверности модели при переходе от исходного примера к состязательному. Наблюдается резкое изменение вероятностей, что указывает на необходимость калибровки.

Для решения проблемы плохой калибровки предлагается расширение обучающего множества с использованием интерполяции между исходными и состязательными примерами:

Определение V.10. *SmoothMix интерполяция* определяется как:

$$x^{\text{mix}} := (1 - \lambda) \cdot x + \lambda \cdot \tilde{x}^{(T)}, \tag{35}$$

$$y^{\text{mix}} := (1 - \lambda) \cdot G(x) + \lambda \cdot \frac{1}{C}, \tag{36}$$

где $\lambda \sim \mathcal{U}([0, \frac{1}{2}])$, $C = |\mathcal{Y}|$ – количество классов, и $\frac{1}{C}$ представляет равномерное распределение над классами.

3) Функция потерь SmoothMix: Итоговая функция потерь комбинирует стандартную функцию потерь для естественных примеров с функцией потерь для интерполированных примеров:

Определение V.11. Пусть $L^{\text{nat}} := \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I)}[\mathcal{L}(F(x + \epsilon), y)]$ — функция потерь для естественных примеров. Тогда *общая функция потерь SmoothMix* определяется как:

$$L := L^{\text{nat}} + \eta \cdot L^{\text{mix}}, \tag{37}$$

где $\eta > 0$ — параметр, контролирующий баланс между точностью на естественных примерах и робастностью, а L^{mix} — функция потерь для интерполированных примеров.

Алгоритм 4 SmoothMix обучение

Require: Сэмплировать $(x, y) \sim P$. фактор сглаживания σ. количество шумов m. число шагов T. размер шага α. коэффициент регуляризации $\eta > 0$.

Sample $\epsilon_1, \cdots, \epsilon_m \sim \mathcal{N}(0, \sigma^2 I)$, и $\lambda \sim \mathcal{U}([0, \frac{1}{2}])$
Найти состязательный пример
$\tilde{x}^{(0)}, G(x^{(0)}) \leftarrow x, \frac{1}{m}\sum_{i=1}^{m} F(x + \epsilon_i)$
for $t = 0$ **to** $T - 1$ **do**
  $J(\tilde{x}^{(t)}) \leftarrow -\log G_y(\tilde{x}^{(t)})$
  $\tilde{x}^{(t+1)} \leftarrow \tilde{x}^{(t)} + \alpha \cdot \frac{\nabla_x J(\tilde{x}^{(t)})}{\|\nabla_x J(\tilde{x}^{(t)})\|_2}$
  $G(\tilde{x}^{(t+1)}) \leftarrow \frac{1}{m}\sum_{i=1}^{m} F(\tilde{x}^{(t+1)} + \epsilon_i)$
end for `use_single_step` $x \leftarrow \tilde{x}^{(1)}$
Вычислить лосс SmoothMix
$x^{\text{mix}}, y^{\text{mix}} \leftarrow ((1-\lambda)\cdot x + \lambda \cdot \tilde{x}^{(T)}), ((1-\lambda)\cdot G(x) + \lambda \cdot \frac{y}{C})$
for $i = 1$ **to** m **do**
  $L_i^{\text{nat}}, L_i^{\text{mix}} \leftarrow \mathcal{L}(F(x+\epsilon_i), y), \mathcal{L}(F(x^{\text{mix}} + \delta_i), y^{\text{mix}})$
end for
$L \leftarrow \frac{1}{m}\sum_i (L_i^{\text{nat}} + \eta \cdot L_i^{\text{mix}})$

D. Консистентная регуляризация для сертифицированной робастности

В работе [12] предлагается дополнить стандартную схему обучения консистентной регуляризацией (consistency regularization) с целью уменьшения вариативности предсказаний сглаженного классификатора под действием гауссовского шума для заданного входа x.

1) Декомпозиция робастной ошибки: Аналогично методу MACER [10], авторы минимизируют 0/1-ошибку робастной классификации, определенную в уравнении (26), путем декомпозиции на ошибку классификации и ошибку робастности.

Определение V.12. *Робастная 0/1-ошибка* с использованием радиуса сертификации декомпозируется следующим образом:

$$\begin{aligned}&\mathbb{E}_{(x,y)\sim\mathcal{S}}[1 - \mathbf{1}_{\underline{R}(g;x,y)\geq\epsilon}] \\ &= \underbrace{\mathbb{E}[\mathbf{1}_{g(x)\neq y}]}_{\text{ошибка классификации}} + \underbrace{\mathbb{E}[\mathbf{1}_{g(x)=y, \underline{R}(g;x,y)<\epsilon}]}_{\text{ошибка робастности}},\end{aligned} \tag{38}$$

где:

- $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^n$ – обучающее множество с $x \in \mathbb{R}^d$, $y \in \mathcal{Y} = \{1, \ldots, K\}$;
- g – сглаженный классификатор;
- $\underline{R}(g; x, y)$ – нижняя граница радиуса робастности: $R(g; x, y) \geq \sigma \cdot \Phi^{-1}(p_A) =: \underline{R}(g; x, y)$;
- $\epsilon > 0$ – заданная константа робастности.

2) Мотивация консистентной регуляризации: Предполагая, что ошибка классификации может быть оптимизирована с помощью стандартных функций потерь (например, кросс-энтропии), основное внимание уделяется минимизации ошибки робастности. При этом возникают следующие вычислительные сложности:

1) Точное вычисление сглаженного классификатора g является вычислительно неразрешимой задачей;
2) Сглаженный классификатор g практически недифференцируем при аппроксимации методами Монте-Карло.

Для преодоления данных сложностей рассматривается достаточное условие минимизации 0/1-ошибки робастности.

Лемма V.1 (Достаточное условие для робастности)**.** *Пусть* $f(x) = \arg\max_{k\in\mathcal{Y}} F(x)$ *для дифференцируемой функции* $F : \mathbb{R}^d \to \mathcal{Y}$. *Если* $F(x + \delta)$ *является константой по* δ *для данного* x, *то робастная ошибка равна нулю, поскольку* $\mathbb{P}(f(x + \delta) = g(x)) = 1$ *независимо от* g.

Данное наблюдение приводит к следующей верхней оценке ошибки робастности:

$$\begin{aligned}&\mathbb{E}_{(x,y)\sim\mathcal{S}}[\mathbf{1}_{g(x)=y, \underline{R}(g;x,y)\leq\epsilon}] \\ &= \mathbb{E}[\mathbf{1}_{g(x)=y, \underline{R}(g;x,g(x))\leq\epsilon}] \\ &\leq \mathbb{E}[\mathbf{1}_{\underline{R}(g;x,g(x))\leq\epsilon}] \\ &= \mathbb{E}[\mathbf{1}_{\mathbb{P}(f(x+\delta)=g(x))<\Phi(\epsilon/\sigma)}],\end{aligned} \tag{39}$$

где последнее равенство следует из определения нижней границы радиуса робастности.

Предполагая, что ошибка классификации может быть оптимизирована с помощью стандартной функции потерь, например, кросс-энтропией, сфокусируемся на том, как минимизировать ошибку робастности. При этом возникают следующие сложности: точное вычисление g является трудновыполнимой задачей, g - практически не дифференцируема когда ее приближают методами Монте-Карло.

Чтобы преодолеть эти сложности, заострим внимание на достаточном условии минимизации $0/1$ ошибки робастности. Предположим, что $f(x) = \arg\max_{k\in\mathcal{Y}} F(x)$ (классификатор) для дифференцируемой функции F. Заметим, что робастная ошибка будет равна нулю, если $F(x + \delta)$ равна константе для данного x. Это влечет то, что $\mathbb{P}_\delta(f(x + \delta) = g(x)) = 1$ независимо от g и минимизирует верхнюю грань робастной ошибки из-за следующего:

$$\begin{aligned}&\mathbb{E}_{(x,y)\in\mathcal{S}}[\mathbf{1}_{g(x)} \\ &= y, \underline{R}(g; x, y) \leq \epsilon] = \\ &= [\mathbf{1}_{g(x)=y, \underline{R}(g;x,g(x))\leq\epsilon}] \leq \\ &\leq [\mathbf{1}_{\underline{R}(g;x,g(x))\leq\epsilon}] = [\mathbf{1}_{\mathbb{P}_\delta(f(x+\delta)=g(x))<\Phi(\frac{\epsilon}{\sigma})}],\end{aligned} \tag{40}$$

3) Консистентная регуляризация: На основе данного анализа предлагается оптимизировать сглаженный классификатор g путем регуляризации функции $F(x+\delta)$ для обеспечения её консистентности по возмущению δ.

Определение V.13. *Консистентная регуляризация* определяется как:

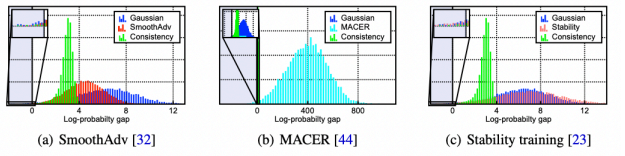
$$L^{\text{con}} := \lambda \cdot \mathbb{E}_{\delta\sim\mathcal{N}(0,\sigma^2 I)}[\text{KL}(G(x)\|F(x+\delta))] + \eta \cdot H(G(x)), \tag{41}$$

где:

- $G(x) := \mathbb{E}_{\delta\sim\mathcal{N}(0,\sigma^2 I)}[F(x + \delta)]$ – усредненное распределение;
- $\text{KL}(\cdot\|\cdot)$ – дивергенция Кульбака-Лейблера;
- $H(\cdot)$ – энтропия Шеннона;
- $\lambda, \eta > 0$ – гиперпараметры регуляризации.

Данная регуляризация принуждает функцию F (и соответственно классификатор f) уменьшать вариативность предсказаний под действием гауссовского шума для заданного входа x.

Замечание V.1. При $\lambda = \eta$ предложенная регуляризация включает в себя кросс-энтропию: $\mathbb{E}_\delta[\mathcal{L}(F(x + \delta), G(x))]$. На практике параметр λ оказывает более значительное влияние на соотношение между точностью и робастностью по сравнению с η.



(a) SmoothAdv [32] (b) MACER [44] (c) Stability training [23]

Рис. 4. Сравнение логарифмических распределений под действием гауссовского шума для фиксированного примера из набора данных MNIST [13]. Консистентная регуляризация приводит к более стабильным предсказаниям.

4) Общая функция потерь: Полная функция потерь комбинирует стандартную функцию потерь для классификации с консистентной регуляризацией:

Определение V.14. *Общая функция потерь с консистентной регуляризацией*:

$$\begin{aligned} L &:= L^{\text{nat}} + L^{\text{con}} \\ &= \mathbb{E}_{\delta\sim\mathcal{N}(0,\sigma^2 I)}[\mathcal{L}(F(x+\delta),y) + \lambda\cdot \mathrm{KL}(G(x)\|F(x+\delta))] + \eta\cdot H(G(x)) \\ &\approx \frac{1}{m}\sum_{i=1}^{m}[\mathcal{L}(F(x+\delta_i),y) + \lambda\cdot \mathrm{KL}(G(x)\|F(x+\delta_i))] + \eta\cdot H(G(x)), \end{aligned} \tag{42}$$

где $\delta_i \sim \mathcal{N}(0,\sigma^2 I)$ – выборки гауссовского шума для аппроксимации математического ожидания методом Монте-Карло.

Данная функция потерь может быть использована с любой базовой функцией потерь для классификации L^{nat}, при условии что она эффективно минимизирует ошибку сглаженного классификатора g.

E. Обучение, основанное на уверенности сглаженного классификатора для сертифицированной робастности

В данной работе [14] предлагается модификация процесса обучения для повышения качества сертифицированной робастности. Рассматриваются два критических случая, для которых вычисленный радиус сертификации является небольшим, и предлагается процедура обучения, штрафующая данные ситуации.

Для удобства введём обозначение:

$$p_f(x,y) := \mathbb{P}(f(x+\epsilon)=y) \tag{43}$$

где $\epsilon\sim\mathcal{N}(0,\sigma^2 I)$ – случайное возмущение.

1) Bottom-K функция потерь для примеров с низкой уверенностью модели: Рассмотрим случай, когда $p_f(x,y)\ll 1$, то есть сглаженный классификатор g демонстрирует низкую уверенность для входа x. В данной ситуации радиус сертификации, получаемый из основной теоремы о случайном сглаживании, будет небольшим. Для исправления этого недостатка авторы предлагают использовать Bottom-K функцию потерь.

Процедура заключается в выборке M независимых одинаково распределённых возмущений $\epsilon_1,\epsilon_2,\ldots,\epsilon_M\sim\mathcal{N}(0,\sigma^2 I)$. Заметим, что индикаторные переменные $\mathbf{1}[f(x+\epsilon_i)=y]$ также являются независимыми случайными величинами с распределением Бернулли с параметром $p_f(x,y)$. Следовательно, количество корректных предсказаний следует биномиальному распределению: $\sum_i \mathbf{1}[f(x+\epsilon_i)=y]\sim\mathrm{Bin}(M,p_f(x,y))$.

В качестве функции потерь для данного случая предлагается минимизировать кросс-энтропию на K из M образцов, упорядоченных по возрастанию значения функции потерь:

$$\mathcal{L}^{\text{low}} := \frac{1}{M}\sum_{i=1}^{K}\mathcal{L}_{\text{CE}}(F(x+\epsilon_{\pi(i)}),y) \tag{44}$$

где $K\sim\mathrm{Bin}(M,p_f(x,y))$, F – модель f без операции $\arg\max$, $\pi(i)$ – индекс i-го наименьшего значения функции потерь среди M примеров.

Поскольку в начале обучения возможна ситуация $p_f(x,y)\approx 0$, предлагается использовать модифицированное значение $K^+ := \max(K,1)$.

2) Функция потерь наихудшего случая для примеров с высокой уверенностью модели: Рассмотрим альтернативный случай, когда $p_f(x,y)\approx 1$, то есть сглаженный классификатор g демонстрирует высокую уверенность для входа x. В процессе обучения с гауссовым шумом, из-за редкости появления других классов в окрестности x, алгоритм обучения может не учитывать минимизацию ошибки на данных примерах. Однако впоследствии они могут появиться при практическом использовании, что приведёт к снижению сертифицированного радиуса.

Для решения данной проблемы предлагается поиск образцов с наибольшей ошибкой. Выполняется выборка $\epsilon_1,\epsilon_2,\ldots,\epsilon_M\sim\mathcal{N}(0,\sigma^2 I)$, но вместо использования значения $F(x+\epsilon_i)$ непосредственно, для каждого ϵ_i производится поиск наихудшего случая в его окрестности. Функция потерь для данного случая определяется как:

$$\mathcal{L}^{\text{high}} := \max_i \max_{\|\epsilon_i^*-\epsilon_i\|_{\ell_2}\le\delta} \mathrm{KL}(F(x+\epsilon_i^*),\hat{y}) \tag{45}$$

где $\mathrm{KL}(\cdot,\cdot)$ – дивергенция Кульбака-Лейблера. Выбор данной метрики обосновывается следующими соображениями:

1) Если $\hat{y}$ представляет собой вектор с единицей на позиции правильной метки и нулями в остальных позициях (жёсткая разметка), то дивергенция KL эквивалентна кросс-энтропии.
2) Данный выбор позволяет использовать мягкую разметку, то есть распределение вероятностей по различным классам.
3) Мотивация подкреплена результатами работы [12], где показано, что устойчивость предсказаний к гауссову шуму различной интенсивности контролирует компромисс между точностью и робастностью.

Для получения разметки $\hat{y}$ используется результат вспомогательной модели f (с softmax-слоем без операции $\arg\max$), предварительно обученной на том же наборе данных. Для решения внутренней задачи максимизации применяется алгоритм проекционного градиентного спуска (PGD) [8] с T итерациями и размером шага $2\delta/T$.

3) Итоговая процедура обучения: Функция $\mathcal{L}^{\text{high}}$ должна применяться только в случае $p_f(x,y)\approx 1$. Однако в процессе обучения точное вычисление $p_f(x,y)$ невозможно, поэтому используется эмпирическая оценка:

$$\hat{p}_f(x,y) := \frac{1}{M}\sum_{i=1}^{M}\mathbf{1}[f(x+\epsilon_i)=y] \tag{46}$$

При условии $K\sim\mathrm{Bin}(M,\hat{p}_f(x,y))$ итоговая функция потерь определяется как:

$$\mathcal{L}^{\text{CAT-RS}} := \mathcal{L}^{\text{low}} + \lambda\cdot\mathbf{1}[K=M]\cdot\mathcal{L}^{\text{high}} \tag{47}$$

Гиперпараметр λ контролирует компромисс между точностью и робастностью модели.

4) Базовые методы для сравнения: В качестве базовых методов рассматриваются: Gaussian training, Stability training, SmoothAdv [7], MACER [10], Consistency [12], SmoothMix [11].

VI. Быстрое сертифицированное робастное обучение с коротким разогревом

В работе [15] предлагается новая инициализация весов для IBP-обучения [16], направленная на сертифицированное робастное обучение. Демонстрируются преимущества использования батч-нормализации и предлагается специализированная регуляризация для стабилизации процесса обучения.

Определение VI.1. IBP (Interval Bound Propagation) [16] – метод для обучения верифицированно робастных классификаторов. Данный подход позволяет определить функцию потерь для минимизации верхней границы максимального расстояния между любыми двумя парами логитов при условии, что входные данные могут быть возмущены в пределах шара по норме ℓ_∞.

1) Сертифицированное робастное обучение: Рассмотрим основные проблемы существующего IBP-обучения:

- взрывной рост границ при инициализации;
- дисбаланс между состояниями ReLU-активаций [17].

Обучение робастной нейронной сети в общем случае можно сформулировать как минимаксную задачу оптимизации:

$$\min_\theta \mathbb{E}_{(x,y)\in\mathcal{X}} \left[\max_{\epsilon\in\Delta(x)} \mathcal{L}(f_\theta(x+\epsilon), y) \right] \tag{48}$$

где f_θ – параметризованная нейронная сеть, $\mathcal{L}$ – функция потерь. Эмпирические методы состязательного обучения решают внутреннюю задачу максимизации с помощью состязательных атак, а затем внешнюю задачу как стандартное обучение глубокой нейронной сети [9], дополненной возмущениями ϵ.

Рассмотрим аффинный слой нейронной сети $h_i = W_i z_{i-1} + b_i$. Вычислим для него IBP-границы согласно определению VI.1:

$$\begin{aligned} \underline{h}_i &= W_{i,+}\underline{z}_{i-1} + W_{i,-}\overline{z}_{i-1} + b_i, \\ \overline{h}_i &= W_{i,+}\overline{z}_{i-1} + W_{i,-}\underline{z}_{i-1} + b_i \end{aligned} \tag{49}$$

где:

- $W_{i,+}$ – положительные элементы матрицы W_i, когда остальные элементы равны нулю; аналогично определяется $W_{i,-}$;
- h_i – функция пост-активаций от предыдущего слоя z_i, то есть $h_i(z_i)$;
- IBP-границы гарантируют выполнение неравенства $\underline{h}_i \le h_i(z_i) \le \overline{h}_i$ для всех $\underline{z}_i \le z_i \le \overline{z}_i$ (поэлементное сравнение).

Разрыв между границами определяется как:

$$\Delta_i = \overline{h}_i - \underline{h}_i = |W_i|(\overline{z}_{i-1} - \underline{z}_{i-1}) = |W_i|\Delta_{i-1} \tag{50}$$

где Δ_i – разрыв между верхней и нижней границами (разность между максимальным и минимальным элементами в слое после активации), $|W_i|$ – поэлементное абсолютное значение.

При инициализации предполагается, что все элементы W_i распределены согласно симметричному относительно нуля распределению с нулевым средним и дисперсией σ_i^2. Пусть $\mathbb{E}[\cdot]$ – математическое ожидание данного распределения. Тогда:

$$\mathbb{E}[\Delta_i] = \frac{n_i}{2}\mathbb{E}[|W_i|]\mathbb{E}[\Delta_{i-1}] \tag{51}$$

Эмпирически можно оценить $\mathbb{E}[\Delta_i]$ на батче данных, вычислив среднее значение. Обозначим эмпирическую оценку как $\hat{\mathbb{E}}[\Delta_i]$.

Определение VI.2. Коэффициент прироста разности при распространении границ от слоя $i-1$ к слою i определяется как:

$$\frac{\mathbb{E}[\Delta_i]}{\mathbb{E}[\Delta_{i-1}]} = \frac{n_i}{2}\mathbb{E}[|W_i|] \tag{52}$$

Границы считаются стабильными, когда данное отношение близко к единице.

Следующая проблема связана с дисбалансом ReLU-активаций [17], проявляющимся в высоком проценте неактивных нейронов.

A. Предложенный метод

Метод включает следующие компоненты:

1) Специализированная инициализация: Каждый элемент W_i инициализируется независимо согласно нормальному распределению $\mathcal{N}(0, \sigma_i^2)$, где параметр σ_i^2 выбирается таким образом, чтобы обеспечить $\frac{n_i}{2}\mathbb{E}[|W_i|] = 1$.

Учитывая, что $\mathbb{E}[|W_i|] = \sqrt{\frac{2}{\pi}}\sigma_i$, получаем:

$$\sigma_i = \frac{\sqrt{2\pi}}{n_i} \tag{53}$$

2) Батч-нормализация: Эмпирически показано, что добавление батч-нормализации для каждого аффинного слоя существенно устраняет дисбаланс ReLU-активаций.

3) Регуляризация разрыва между границами: Предлагается следующий регуляризатор:

$$\mathcal{L}_{\text{tightness}} = \frac{1}{\tau m}\sum_{i=1}^{m} \text{ReLU}\left(\tau - \frac{\hat{\mathbb{E}}[\Delta_i]}{\hat{\mathbb{E}}[\Delta_0]}\right) \tag{54}$$

где $\tau\hat{\mathbb{E}}[\Delta_i] \le \hat{\mathbb{E}}[\Delta_0]$ и $0 < \tau \le 1$. Штраф применяется только при нарушении условия $\tau\hat{\mathbb{E}}[\Delta_i] > \hat{\mathbb{E}}[\Delta_0]$.

4) Балансировка ReLU-активаций: Определим $c_i = (\underline{h}_i + \overline{h}_i)/2$. Пусть α_i – соотношение между вкладом активных и неактивных нейронов в среднее значение $\hat{\mathbb{E}}[c_i]$, а β_i – соотношение между их вкладом в дисперсию $\text{Var}(c_i)$ каждого слоя:

$$\begin{aligned} \alpha_i &= \frac{\sum_j \mathbf{1}(\underline{h}_{i,j} > 0)c_{i,j}}{-\sum_j \mathbf{1}(\overline{h}_{i,j} < 0)c_{i,j}}, \\ \beta_i &= \frac{\sum_j \mathbf{1}(\underline{h}_{i,j} > 0)(c_{i,j} - \hat{\mathbb{E}}[c_i])^2}{\sum_j \mathbf{1}(\overline{h}_{i,j} < 0)(c_{i,j} - \hat{\mathbb{E}}[c_i])^2} \end{aligned} \tag{55}$$

где $\alpha_i, \beta_i > 0$. Активация считается сбалансированной, когда α_i и β_i близки к единице. Накладываются ограничения $\tau \le \alpha_i, \beta_i \le 1/\tau$.

Регуляризация ReLU-активаций определяется как:

$$\mathcal{L}_{\text{relu}} = \frac{1}{\tau m} \sum_{i=1}^{m} \left(\text{ReLU}\left(\tau - \min\left(\alpha_i, \frac{1}{\alpha_i} \right) \right) + \text{ReLU}\left(\tau - \min\left(\beta_i, \frac{1}{\beta_i} \right) \right) \right) \tag{56}$$

B. Процедура обучения

Базовая целевая функция для робастного обучения без регуляризации:

$$\mathcal{L}_{\text{rob}} = \overline{\mathcal{L}}(f_\theta, x, y, \epsilon) \tag{57}$$

где $\overline{\mathcal{L}}(f_\theta, x, y, \epsilon) \ge \max_{\|\delta\|_{\ell_\infty} \le \epsilon} \mathcal{L}(f_\theta(x+\delta), y)$.

В предложенном методе сначала выполняется инициализация параметров согласно IBP-процедуре, затем производится короткий этап разогрева с постепенным увеличением ϵ от 0 до ϵ_{target}, где ϵ_{target} – целевой радиус возмущения, обычно равный или незначительно превышающий максимальный радиус возмущения для тестового множества.

Итоговая целевая функция имеет вид:

$$\mathcal{L} = \mathcal{L}_{\text{rob}} + \lambda(\mathcal{L}_{\text{tightness}} + \mathcal{L}_{\text{relu}}) \tag{58}$$

где λ – коэффициент регуляризации, который в процессе разогрева постепенно уменьшается от λ_0 до 0 с ростом ϵ согласно зависимости $\lambda = \lambda_0(1 - \epsilon/\epsilon_{\text{target}})$.

После завершения этапа разогрева используется только $\mathcal{L} = \mathcal{L}_{\text{rob}}$ для финального обучения с параметром ϵ_{target}.

C. Сертифицированная робастность для top-k предсказаний против состязательных возмущений с помощью случайного сглаживания

В работе [18] авторы получают сертифицированный радиус для top-k предсказаний. Более того, доказывается, что данный сертифицированный радиус является оптимальным для гауссовского шума.

Пусть дан базовый классификатор f. Сглаженный классификатор $g_k(x)$ возвращает набор из k меток с наибольшими вероятностями p_i на входе x. Цель – получить радиус сертификации R_l такой, что $l \in g_k(x+\delta)$ для всех $\|\delta\|_{\ell_2} < R_l$.

Теорема VI.1. *(Сертифицированный радиус для top-k предсказаний). Пусть дан входной пример x, произвольный базовый классификатор f, случайное возмущение $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$, сглаженный классификатор g, произвольная метка $l \in \{1, 2, \dots, c\}$ и границы вероятностей $\underline{p_l}, \overline{p_1}, \dots, \overline{p_{l-1}}, \overline{p_{l+1}}, \dots, \overline{p_c} \in [0,1]$, которые удовлетворяют следующим условиям:*

$$\mathbb{P}(f(x+\epsilon) = l) \ge \underline{p_l} \quad \text{и} \quad \mathbb{P}(f(x+\epsilon) = i) \le \overline{p_i}, \quad \forall i \ne l \tag{59}$$

где $\underline{p}$ и $\overline{p}$ обозначают нижнюю и верхнюю границы p соответственно. Пусть $\overline{p_{b_k}} \ge \overline{p_{b_{k-1}}} \ge \dots \ge \overline{p_{b_1}}$ — наибольшие k вероятностей среди $\{\overline{p_1}, \dots, \overline{p_{l-1}}, \overline{p_{l+1}}, \dots, \overline{p_c}\}$. Обозначим $S_t = \{b_1, b_2, \dots, b_t\}$ — множество меток с наименьшими верхними границами вероятностей среди k наибольших и $\overline{p}_{S_t} = \sum_{j=1}^{t} \overline{p_{b_j}}$ — сумму t верхних границ вероятностей, где $t = 1, 2, \dots, k$. Тогда имеем:

$$l \in g_k(x+\delta), \quad \forall \|\delta\|_{\ell_2} < R_l \tag{60}$$

где R_l – единственное решение следующего уравнения:

$$\Phi\left(\Phi^{-1}(\underline{p_l}) - \frac{R_l}{\sigma} \right) - \min_{t=1}^{k} \frac{\Phi\left(\Phi^{-1}(\overline{p}_{S_t}) + \frac{R_l}{\sigma} \right)}{t} = 0 \tag{61}$$

где Φ и Φ^{-1} – функция стандартного нормального распределения и её обратная функция соответственно.

Теорема VI.2. *(Оптимальность радиуса сертификации). Пусть выполнены условия $\underline{p_l} + \sum_{j=1}^{k} \overline{p_{b_j}} \le 1$ и $\underline{p_l} + \sum_{i=1,\dots,l-1,l+1,\dots,c} \overline{p_i} \ge 1$. Тогда для любого возмущения $\|\delta\|_{\ell_2} > R_l$ существует базовый классификатор f, удовлетворяющий условию*

$$\mathbb{P}(f(x+\epsilon) = l) \ge \underline{p_l} \quad \text{и} \quad \mathbb{P}(f(x+\epsilon) = i) \le \overline{p_i}, \quad \forall i \ne l \tag{62}$$

но при этом $l \notin g_k(x+\delta)$.

Следствия из теорем VI.1 и VI.2:

Следствие VI.1. *Полученный сертифицированный радиус применим к любому базовому классификатору f*

Следствие VI.2. *Согласно уравнению (61), сертифицированный радиус R_l зависит от σ, $\underline{p_l}$ и $\{\overline{p_{b_k}}, \dots, \overline{p_{b_1}}\}$, за исключением $\overline{p_l}$. Когда $\underline{p_l}$ велико, а $\{\overline{p_{b_k}}, \dots, \overline{p_{b_1}}\}$ малы, радиус сертификации R_l больше. При $R_l < 0$ метка l не входит в top-k меток, предсказанных сглаженным классификатором, даже без добавления возмущения, то есть $l \notin g_k(x)$;*

Следствие VI.3. *При использовании случайного сглаживания с гауссовским шумом без дополнительных предположений о базовом классификаторе невозможно найти ℓ_2-радиус сертификации для top-k предсказаний больший, чем R_l;*

Следствие VI.4. *При $k = 1$*

$$R_l = \frac{\sigma}{2}\left(\Phi^{-1}(\underline{p_l}) - \Phi^{-1}(\overline{p_{b_1}}) \right) \tag{63}$$

Алгоритм 5 Predict top-k

Вход: f, k, σ, $\mathbf{x}$, n, α.
Выход: ABSTAIN или предсказанные top-k меток.
$T \leftarrow \emptyset$
counts $\leftarrow$ SampleUnderNoise($f, \sigma, \mathbf{x}, n$)
$c_1, c_2, \dots, c_{k+1} \leftarrow$ top-$(k+1)$ меток по частотам
$n_{c_1}, n_{c_2}, \dots, n_{c_{k+1}} \leftarrow$ counts$[c_1]$, counts$[c_2]$, …, counts$[c_{k+1}]$
for $t \leftarrow 1$ **to** k **do**
  if BinomPValue$(n_{c_t}, n_{c_t} + n_{c_{t+1}}, 0.5) \le \alpha$ **then**
    $T \leftarrow T \cup \{c_t\}$
  else
    return ABSTAIN
  end if
end for
return T

1) Методы оценки вероятностных границ: Рассмотрим методы, используемые в алгоритме 6:

- BinoCP – метод для оценки $\underline{p_l}$ с помощью стандартного одностороннего метода Клоппера–Пирсона. Здесь $\overline{p_i} = 1 - \underline{p_l}$ для всех $i \ne l$. Детальная процедура:
  - выполняется выборка n случайных возмущений $\epsilon_1, \epsilon_2, \dots, \epsilon_n \sim \mathcal{N}(0, \sigma^2 I)$;

Алгоритм 6 Certify top-k

Вход: f, k, σ, $\mathbf{x}$, l, n, μ, α.
Выход: ABSTAIN или $\underline{R_l}$.
counts $\leftarrow$ SampleUnderNoise$(f, \sigma, \mathbf{x}, n, \alpha)$
$[\underline{p_l}, \overline{p}_1, \ldots, \overline{p}_{l-1}, \overline{p}_{l+1}, \ldots, \overline{p}_c] \leftarrow$ BinoCP(counts, α) или SimuEM(counts, α)
$\underline{R_l} \leftarrow 0$
for $t \leftarrow 1$ **to** k **do**
  $\overline{p}_{S_t} \leftarrow \min\left(\sum_{j=1}^{t} \overline{p}_{b_j}, 1 - \underline{p_l}\right)$
  $\underline{R_l}^t \leftarrow$ BinarySearch$(\underline{p_l}, \overline{p}_{S_t}, t, \sigma, \mu)$
  if $\underline{R_l}^t > \underline{R_l}$ **then**
    $\underline{R_l} \leftarrow \underline{R_l}^t$
  end if
end for
if $\underline{R_l} > 0$ **then**
  return $\underline{R_l}$
else
  return ABSTAIN
end if

  - определяется счётчик для метки l: $n_l = \sum_{j=1}^{n} \mathbf{1}(f(x+\epsilon_j) = l)$, где $n_l \sim \text{Bin}(n, p_l)$;
  - согласно методу Клоппера–Пирсона: $\underline{p_l} = B(\alpha; n_l, n - n_l + 1)$, где $1-\alpha$ — уровень доверия и $B(\alpha; u, v)$ — α-квантиль бета-распределения с параметрами u, v.

- SimuEM – метод для совместной оценки $\overline{p_i}$ и $\underline{p_l}$. Пусть $n_i = \sum_{j=1}^{n} \mathbf{1}(f(x+\epsilon_j) = i)$ для всех $i \in \{1, 2, \ldots, c\}$, где $n_i \sim \text{Bin}(n, p_i)$:
  - сначала применяется метод Клоппера–Пирсона для каждой метки i;
  - затем получаются доверительные интервалы с поправкой Бонферрони;

  В результате:

$$\underline{p_l} = B\left(\frac{\alpha}{c}; n_l, n - n_l + 1\right) \tag{64}$$

$$\overline{p_i} = B\left(1 - \frac{\alpha}{c}; n_i + 1, n - n_i\right), \quad \forall i \neq l \tag{65}$$

VII. Случайное сглаживание и диффузионные модели

A. Сглаживание без шума: доказуемая защита для предобученных классификаторов

В работе [19] предлагается подход, при котором к входному изображению сначала применяется модель-денойзер, а затем выполняется предсказание. Пусть дан классификатор f и модель-денойзер $\mathcal{D}_\theta : \mathbb{R}^d \to \mathbb{R}^d$, тогда новый базовый классификатор определяется как композиция: $f \circ \mathcal{D}_\theta : \mathbb{R}^d \to \mathcal{Y}$.

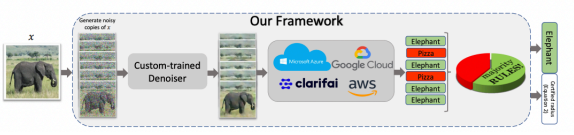


Рис. 5. Схема работы системы с денойзером

Сглаженный классификатор в данном случае определяется следующим образом:

$$g(x) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}[f(\mathcal{D}_\theta(x+\epsilon)) = c], \quad \text{где } \epsilon \sim \mathcal{N}(0, \sigma^2 I) \tag{66}$$

Для каждого уровня шума σ обучается отдельный специализированный денойзер. Исследуются две функции потерь для обучения денойзера $\mathcal{D}_\theta$:

1) Функция потерь среднеквадратичной ошибки: Пусть дан неразмеченный набор данных $\mathcal{S} = \{x_i\}$ чистых изображений. Денойзер обучается минимизации среднеквадратичной ошибки (MSE) между исходным изображением x_i и выходом денойзера $\mathcal{D}_\theta(x_i + \epsilon)$, где $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$. Формально:

$$\mathcal{L}_{\text{MSE}} = \mathbb{E}_{\mathcal{S}, \epsilon}\left[\|\mathcal{D}_\theta(x_i + \epsilon) - x_i\|_{\ell_2}^2\right] \tag{67}$$

2) Функция потерь стабильности: Требуется функция потерь, которая учитывает также ошибку классификации. Пусть дан размеченный датасет $\mathcal{S} = \{(x_i, y_i)\}$. Денойзер обучается с нуля одновременно с задачей классификации зашумлённых изображений:

$$\mathcal{L}_{\text{stab}} = \mathbb{E}_{\mathcal{S}, \epsilon}\left[\mathcal{L}_{\text{CE}}(F(\mathcal{D}_\theta(x_i + \epsilon)), f(x_i))\right], \quad \epsilon \sim \mathcal{N}(0, \sigma^2 I) \tag{68}$$

где $f(x) = \arg\max_{c \in \mathcal{Y}} F(x)$, $\mathcal{L}_{\text{CE}}$ – функция кросс-энтропийных потерь.

Данная функция потерь может использоваться в двух режимах доступа к классификатору:

- **Режим белого ящика**: При наличии доступа к предобученному классификатору возможно выполнение обратного распространения ошибки для $\mathcal{L}_{\text{stab}}$. В данном режиме денойзеры обучаются с нуля путём минимизации ошибки классификации с использованием псевдо-меток, полученных от предобученного классификатора.
- **Режим чёрного ящика**: Используются предобученные суррогатные классификаторы в качестве приближения реальных классификаторов, которые планируется защищать. Денойзеры обучаются минимизировать функцию потерь стабильности (68) суррогатных классификаторов. Эмпирически показано, что денойзеры, обученные в данном режиме, обладают свойством переносимости на другие классификаторы.

VIII. Состязательная робастность бесплатно

В работе [20] демонстрируется, как можно достичь современного уровня (state-of-the-art) сертифицированной робастности против состязательных возмущений по норме ℓ_2, основываясь исключительно на предобученных моделях «из коробки». Для этого комбинируются предобученная диффузионная модель-денойзер и стандартный высокоточный классификатор.

При заданном денойзере выского качетсва (то есть $\mathcal{D}_\theta(x + \epsilon) \approx x$ с высокой вероятностью для $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$), ожидается, что точность базового классификатора f на зашумленных изображениях будет близка к точности этого же классификатора на чистых изображениях.

1) Шумоподавляющие диффузионные вероятностные модели: Диффузионные модели представляют собой класс генеративных моделей, которые обучаются инвертировать время с помощью диффузионного процесса. Прямой диффузионный процесс определяется как:

$$x_t \sim \sqrt{1 - \beta_t} \cdot x_{t-1} + \beta_t \cdot \omega_t, \quad \omega_t \sim \mathcal{N}(0, I), \tag{69}$$

где x_0 берется из распределения данных, β_t – фиксированные (или обучаемые) параметры дисперсии. Обратный диффузионный процесс преобразует изображение из

целевого распределения данных в случайный шум с течением времени, затем синтезирует изображения целевого домена из полученного случайного шума.

В работе используется свойство обучения диффузионных моделей. Пусть дано чистое обучающее изображение $x \in [-1,-1]^{w \cdot h \cdot c}$, диффузионная модель выбирает момент времени $t \in \mathbb{N}^+$, а затем генерирует зашумленное изображение x_t следующим образом:

$$x_t = \sqrt{\alpha_t} \cdot x + \sqrt{1-\alpha_t} \cdot \mathcal{N}(0, I), \tag{70}$$

где α_t – константа, определеяемая моментом времени t, которая задает уровень шума, добавляемого в изображение.

Затем диффузионная модель обучается минимизации расхождения между x и $\mathcal{D}_\theta(x_t)$, чтобы предсказать как должно выглядеть исходное (незашумленное) изображение после применения шума в момент времени t.

Алгоритм 7 Зашумление, денойзинг, классификация

function NoiseAndClassify(x, σ):
$t^\star, \alpha_{t^\star} \leftarrow$ `GetTimestep`(σ)
$x_{t^\star} \leftarrow \sqrt{\alpha_{t^\star}}(x + \mathcal{N}(0, \sigma^2 \mathbf{I}))$
$\hat{x} \leftarrow$ `denoise`($x_{t^\star}; t^\star$)
$y \leftarrow f(\hat{x})$
return y
function GetTimestep(σ):
$t^\star \leftarrow$ find t s.t. $\frac{1-\alpha_t}{\alpha_t} = \sigma^2$
return $t^\star, \alpha_{t^\star}$

2) Денойзированное сглаживание с помощью диффузионной модели: Случайное сглаживание требует примера с наложенным гауссовским шумом $x_{\text{rs}} \sim \mathcal{N}(x, \sigma^2 I)$, в то время как в диффузионной модели зашумлённое изображение имеет распределение $x_t \sim \mathcal{N}(\sqrt{\alpha_t}x, (1-\alpha_t)I)$. Нормируя x_{rs} на $\sqrt{\alpha_t}$ и приравнивая дисперсии, получаем соотношение:

$$\sigma^2 = \frac{1-\alpha_t}{\alpha_t} \tag{71}$$

Чтобы применить диффузионную модель в случайном сглаживании при заданном уровне шума σ, необходимо найти момент времени t^* такой, что $\sigma^2 = \frac{1-\alpha_{t^*}}{\alpha_{t^*}}$. Далее вычисляем:

$$x_{t^*} = \sqrt{\alpha_{t^*}}(x + \delta), \quad \delta \sim \mathcal{N}(0, \sigma^2 I) \tag{72}$$

и применяем диффузионный денойзер к x_{t^*}, чтобы получить оценку:

$$\hat{x} = \mathcal{D}_\theta(x_{t^*}) \tag{73}$$

Затем выполняем классификацию с помощью классификатора «из коробки»:

$$y = f(\hat{x}) \tag{74}$$

Для получения сертифицированной робастности процесс денойзинга повторяется многократно, и вычисляется радиус сертификации согласно стандартной процедуре случайного сглаживания.

Описанный алгоритм представлен в 7.

A. DensePure: понимание диффузионных моделей через призму состязательной робастности

Фреймворк DensePure [21] представляет собой композицию двух компонентов: предобученной диффузионной модели с обратным диффузионным процессом rev и базового классификатора. Данный подход развивает идеи, представленные в разделе VIII где рассматривалась интеграция диффузионных денойзеров с классификаторами для достижения робастности без дополнительного обучения.

1) Алгоритм DensePure:

1) **Обратная диффузия**: Входное изображение x подаётся на вход обратному диффузионному процессу rev, в результате чего получается $\text{rev}(x)$.
2) **Многократная генерация**: Процесс обратной диффузии повторяется K раз независимо, что приводит к получению множества $\{\text{rev}(x)_1, \ldots, \text{rev}(x)_K\}$ стохастически различных версий входного изображения.
3) **Классификация**: Полученные K примеров подаются на вход базовому классификатору f, что даёт множество предсказаний $\{f(\text{rev}(x)_1), \ldots, f(\text{rev}(x)_K)\}$.
4) **Агрегация решений**: Применяется процедура мажоритарного голосования (majority vote, **MV**) для получения итогового предсказания:

$$\begin{aligned} \hat{y} &= \mathbf{MV}\left(\{f(\text{rev}(x)_1), \ldots, f(\text{rev}(x)_K)\}\right) \\ &= \arg\max_c \sum_{i=1}^{K} \mathbf{1}\{f(\text{rev}(x)_i) = c\} \end{aligned} \tag{75}$$

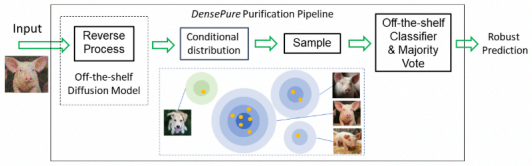


Рис. 6. Схема работы фреймворка DensePure

Ключевое отличие DensePure от подходов раздела VIII заключается в использовании полного обратного диффузионного процесса вместо одношагового денойзинга, что потенциально обеспечивает более качественное восстановление изображений за счёт увеличения вычислительных затрат.

B. Многомасштабное диффузионное сглаживание

В работе [22] выделяются критические случаи, когда сглаженный классификатор g демонстрирует субоптимальную производительность: *сверх-сглаживание* и *сверх-уверенность*. Для решения данных проблем предлагается модифицированный процесс обучения денойзеров, развивающий идеи разделов VII и VIII о композиционных подходах к робастности.

1) Диагностика проблем сглаживания: Проблемные случаи определяются на основе значения уверенности сглаженного классификатора:

$$p_{g_\sigma}(x) := \max_y p_{g_\sigma}(x, y) = \max_y \mathbb{P}_{\epsilon \sim \mathcal{N}(0, \sigma^2 I)}[f(x+\epsilon) = y] \tag{76}$$

Обозначим $p := p_{g_\sigma}(x)$ и введём пороговое значение p_0 для разделения проблемных случаев. По умолчанию используется $p_0 = 0.5$. Эмпирически показано, что значение $p_0 = 0.6$ способно повысить точность классификации за счёт уменьшения сертифицированного радиуса.

2) Сверх-сглаживание ($p \le p_0$): В случае $p \le p_0$ предсказания исходной модели f при заданном уровне шума распределены относительно равномерно по всем классам, что приводит к малой вероятности доминирующего класса. Данная ситуация возникает, когда уровень шума достаточен для изменения семантического содержания входа x. Следствием является малый радиус робастности, поскольку R монотонно убывает при уменьшении p.

Для решения данной проблемы предлагается использование каскада сглаженных классификаторов с различными уровнями шума, где для каждого входа выбирается классификатор, максимизирующий радиус робастности. Пусть заданы K различных параметров $0 < \sigma_1 < \ldots < \sigma_K$ и соответствующие сглаженные модели $g_{\sigma_1}, \ldots, g_{\sigma_K}$. Определим каскадную процедуру $\texttt{casc}(x; \{g_{\sigma_i}\}_{i=1}^K)$:

$$\texttt{casc}(x; \{g_{\sigma_i}\}_{i=1}^K) := \begin{cases} g_{\sigma_K}(x) & \text{если } p_{g_{\sigma_K}}(x) > p_0, \\ \texttt{casc}(x; \{g_{\sigma_i}\}_{i=1}^{K-1}) & \text{если } p_{g_{\sigma_K}}(x) \le p_0 \text{ и } K > 1, \\ \texttt{ABSTAIN} & \text{иначе} \end{cases} \tag{77}$$

Теорема VIII.1 (Гарантии робастности для каскадного сглаживания)**.** *Пусть $g_{\sigma_1}, \ldots, g_{\sigma_K} : \mathcal{X} \to \mathcal{Y}$ – сглаженные классификаторы с соответствующими параметрами $0 < \sigma_1 < \ldots < \sigma_K$. Предположим, что $\texttt{casc}(x; \{g_{\sigma_i}\}_{i=1}^K) =: \hat{y} \in \mathcal{Y}$ достигается при g_{σ_k} для некоторого k. Рассмотрим любые $\underline{p}$ и $\overline{p}_{k',c} \in [0,1]$, которые удовлетворяют условиям:*

$$\underline{p} \le p_{g_{\sigma_k}}(x, \hat{y}), \tag{78}$$

$$\overline{p}_{k',c} \ge p_{g_{\sigma_{k'}}}(x, c) \quad \text{для } k' > k \text{ и } c \in \mathcal{Y} \tag{79}$$

Тогда выполняется $\texttt{casc}(x + \epsilon; \{g_{\sigma_i}\}_{i=1}^K) = \hat{y}$ для любого $\|\epsilon\|_{\ell_2} < R$, где:

$$R := \min \left\{ \sigma_k \cdot \Phi^{-1}(\underline{p}), \min_{\substack{y \ne \hat{y} \\ k' > k}} \left\{ \sigma_{k'} \cdot \Phi^{-1}(1 - \overline{p}_{k',y}) \right\} \right\} \tag{80}$$

Данная теорема обеспечивает формальные гарантии робастности для предсказаний, получаемых каскадной процедурой.

3) Сверх-уверенность ($p \ge p_0$): Альтернативной проблемой при использовании денойзеров является их способность изменять семантическое содержание возмущённого входа $x + \epsilon$ таким образом, что классификатор f с высокой уверенностью относит его к неправильному классу.

Для решения данной проблемы денойзеры дообучаются с использованием модифицированной функции потерь. Обозначим D – денойзер, f_{std} – исходный классификатор, $f := f_{\text{std}} \circ D$ – композицию. Пусть F_{std} – выход последнего слоя f_{std} перед операцией $\arg\max$, то есть $f_{\text{std}}(x) = \arg\max F_{\text{std}}(x)$. Определим $p_{\text{std}} := \max_c F_{\text{std},c}(D(x + \epsilon))$.

Первая компонента функции потерь (Brier score):

$$\mathcal{L}_{\text{Brier}}(x, y) = \\ = \mathbb{E}_\epsilon \left[\mathbf{1}[\hat{y}_\epsilon = y \text{ или } p_{\text{std}}(\epsilon) \le p_0] \cdot \|F_{\text{std}}(D(x + \epsilon)) - \mathbf{e}_y\|_{\ell_2}^2 \right], \tag{81}$$

где $\hat{y}_\epsilon := f(x + \epsilon)$, $\mathbf{e}_y$ – единичный вектор с единицей на позиции y.

Для второй компоненты функции потерь, поскольку проверка условия $p_{g_\sigma}(x) > p_0$ затруднена во время обучения, рассматривается случай, когда для двух независимых случайных величин $\epsilon_1, \epsilon_2 \sim \mathcal{N}(0, \sigma^2 I)$ соответствующие предсказания совпадают: $\hat{y}_1 = \hat{y}_2$, где $\hat{y}_i := f(x + \epsilon_i)$. Обозначим $\mathbf{p}_i := F_{\text{std}}(x + \epsilon_i)$. Тогда:

$$\mathcal{L}_{\text{AC}}(x, y) := \mathbf{1}[\hat{y}_1 = \hat{y}_2 \text{ и } \hat{y}_1 \ne y] \cdot (\|\mathbf{p}_1 - \text{sg}(\mathbf{p}_1)\|_{\ell_2}^2 + \|\mathbf{p}_2\|_{\ell_2}^2), \tag{82}$$

где $\text{sg}(\cdot)$ – операция остановки градиента, возвращающая аргумент без распространения градиента.

Итоговая функция потерь имеет вид:

$$\mathcal{L}(D) := \mathcal{L}_{\text{Denoiser}} + \lambda \cdot (\mathcal{L}_{\text{Brier}} + \alpha \cdot \mathcal{L}_{\text{AC}}) \tag{83}$$

C. Диффузионная модель как сертифицированно робастный классификатор

В данном подразделе рассматривается подход [23], демонстрирующий, что предобученные диффузионные модели обладают внутренней сертифицированной робастностью и могут использоваться как классификаторы без дополнительного обучения. Данный подход развивает идеи предыдущих разделов о композиционных методах робастности и представляет альтернативную перспективу на использование диффузионных моделей.

1) Принципы работы диффузионных моделей: Пусть $x := x_0 \in \mathbb{R}^d$ имеет распределение данных $q(x_0)$. Прямой процесс диффузии постепенно добавляет гауссовский шум к распределению данных, создавая непрерывную последовательность распределений $\{q(x_t) := q_t(x_t)\}_{t=1}^T$:

$$q(x_t) = \int q(x_0) q(x_t | x_0) dx_0, \tag{84}$$

где $q(x_t | x_0) = \mathcal{N}(x_t; x_0, \sigma_t^2 I)$, то есть $x_t = x_0 + \sigma_t \epsilon$ с $\epsilon \sim \mathcal{N}(0, I)$.

Обычно σ_t монотонно возрастает с t, устанавливая взаимно однозначные отображения $t(\sigma)$ от σ к t и $\sigma(t)$ от t к σ. Параметр σ_T выбирается достаточно большим, чтобы $q(x_T)$ было близко к изотропному гауссовскому распределению.

Пусть $p := p_\theta$ – параметризованное обратное распределение с априорным $p(x_T) = \mathcal{N}(x_T; 0, \sigma_T^2 I)$. Процесс диффузии для синтеза данных определяется как марковская цепь с обученными гауссовскими переходами:

$$p(x_{0:T}) = p(x_T) \prod_{t=1}^T p(x_{t-1} | x_t) \tag{85}$$

Обратное гауссовское распределение $p(x_{t-1} | x_t)$ параметризуется нейронной сетью $h_\theta(x_t, t)$ как:

$$p(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \tilde{\sigma}_t^2 I) \tag{86}$$

$$\mu_\theta(x_t, t) = \frac{(\sigma_t^2 - \sigma_{t-1}^2) h_\theta(x_t, \sigma_t) + \sigma_{t-1}^2 x_t}{\sigma_t^2} \tag{87}$$

Параметры θ обучаются путём оптимизации доказанной нижней границы (ELBO) логарифма правдоподобия:

$$\log p(x_0) \geq -\sum_{t=1}^{T} \mathbb{E}_\epsilon \left[w_t \| h_\theta(x_t, \sigma_t) - x_0 \|_{\ell_2}^2 \right] + C_1, \quad (88)$$

где $w_t = \frac{\sigma_{t+1} - \sigma_t}{\sigma_{t+1}^3}$ – вес функции потерь на временном шаге t, C_1 – константа.

Аналогично, условная диффузионная модель $p(x_{t-1}|x_t, y)$ параметризуется как $h_\theta(x_t, \sigma_t, y)$ с соответствующей нижней границей:

$$\log p(x_0|y) \geq -\sum_{t=1}^{T} \mathbb{E}_\epsilon \left[w_t \| h_\theta(x_t, \sigma_t, y) - x_0 \|_{\ell_2}^2 \right] + C_2, \quad (89)$$

где C_2 –константа.

2) Диффузионная модель как классификатор: Диффузионный классификатор представляет собой генеративный классификатор, использующий предобученную диффузионную модель для робастной классификации. Он вычисляет вероятность класса $p(y|x_0) \propto p(x_0|y)p(y)$ через теорему Байеса и аппроксимирует условное правдоподобие через условный ELBO при предположении равномерного распределения $p(y)$:

$$\begin{aligned} \mathrm{DC}(x_0)_y := p(y|x_0) &= \frac{p(x_0|y)p(y)}{\sum_{\hat{y}} p(x_0|\hat{y})p(\hat{y})} = \frac{p(x_0|y)}{\sum_{\hat{y}} p(x_0|\hat{y})} \\ &= \frac{\exp(\log p(x_0|y))}{\sum_{\hat{y}} \exp(\log p(x_0|\hat{y}))} \\ &\approx \frac{\exp\left(-\frac{1}{dT} \sum_{t=1}^{T} \mathbb{E}_\epsilon \left[w_t \| h_\theta(x_t, \sigma_t, y) - x_0 \|_{\ell_2}^2 \right] \right)}{\sum_{\hat{y}} \exp\left(-\frac{1}{dT} \sum_{t=1}^{T} \mathbb{E}_\epsilon \left[w_t \| h_\theta(x_t, \sigma_t, \hat{y}) - x_0 \|_{\ell_2}^2 \right] \right)} \end{aligned} \quad (90)$$

Такой классификатор демонстрирует современный уровень (state-of-the-art) робастности для различных моделей угроз и обобщается на ранее неизвестные атаки без необходимости состязательного дообучения.

3) Теоретические основы: Случайное сглаживание – это независимый от модели метод для определения нижней границы устойчивости к состязательным примерам. Данный подход, подробно рассмотренный ранее, масштабируется до глубоких нейронных сетей и больших наборов данных, достигая современного уровня сертифицированной робастности.

Метод создаёт сглаженный классификатор путём усреднения выходов базового классификатора по гауссовскому шуму. Благодаря липшицевой непрерывности такого классификатора он остаётся стабильным в пределах определённого диапазона возмущений, обеспечивая сертифицированную робастность.

Определение VIII.1 (Липшицева непрерывность)**.** Функция $f : \mathbb{R}^n \to \mathbb{R}$ называется липшицевой с константой Липшица L, если существует $L \geq 0$ такая, что для любых $x_1, x_2 \in \mathbb{R}^n$:

$$|f(x_1) - f(x_2)| \leq L \|x_1 - x_2\|_{\ell_2} \quad (91)$$

Данное свойство формализует идею ограниченного изменения функции относительно изменения её аргумента.

Формально, имея классификатор $f : \mathbb{R}^d \to \mathbb{R}^m$, принимающий d-мерный вход x_0 и предсказывающий вероятности для m классов, y-й выход сглаженного классификатора g определяется как:

$$g(x_0)_y = \mathbb{P}\left(\arg \max_{\hat{y} \in \{1,\ldots,K\}} f(x_0 + \sigma_\tau \cdot \epsilon)_{\hat{y}} = y \right) \quad (92)$$

где $\epsilon \sim \mathcal{N}(0, I)$ – гауссовский шум, σ_τ — уровень шума.

Как показано в разделе V, $\Phi^{-1}(g(x_0)_y)$ является $\frac{1}{\sigma_\tau}$-липшицевой функцией, где Φ^{-1} – обратная функция стандартного нормального распределения. На практике оцениваются нижняя граница $\underline{p_A}$ для $g(x_0)_y$ и верхняя граница $\overline{p_B}$ для $\max_{\hat{y} \neq y} g(x_0)_{\hat{y}}$ с использованием доверительных интервалов, после чего вычисляется нижняя граница сертифицированного радиуса:

$$R = \frac{\sigma_\tau}{2} \left(\Phi^{-1}(\underline{p_A}) - \Phi^{-1}(\overline{p_B}) \right) \quad (93)$$

Обычно, существующие классификаторы обучены классифицировать изображения из исходного распределения $q(x_0)$. Однако входное распределение в уравнении (92) – это $q(x_\tau) = \int q(x_0) q(x_\tau|x_0) dx_0$. Из-за разницы в распределениях, $g(x_0)$, классификатор обученный на исходном распределении $q(x_0)$, показывает низкую точность на $q(x_\tau)$. Вследствие чего невозможно напрямую использовать иффузионный классификатор со случайным сглаживанием. Предлагается подход к созданию диффузионных классификаторов, которые могут напрямую вычислять $p(y|x_\tau)$ с использованием готовой диффузионной модели.

4) Предлагаемый метод:

а) Липшицева непрерывность диффузионного классификатора: Ключевое наблюдение заключается в том, что логиты диффузионного классификатора $-\frac{1}{dT} \sum_{t=1}^{T} w_t \mathbb{E}_\epsilon \left[\| h_\theta(x_t, \sigma_t, y) - x_0 \|_{\ell_2}^2 \right]$ из уравнения (90) могут быть представлены как:

$$-\frac{1}{dT} \sum_{t=1}^{T} w_t \left(\mathbb{E}_\epsilon \left[\| h_\theta(x_t, \sigma_t, y) \|_{\ell_2}^2 \right] + \| x_0 \|_{\ell_2}^2 - 2\mathbb{E}_\epsilon \left[h_\theta(x_t, \sigma_t, y)^\top x_0 \right] \right) \quad (94)$$

Поскольку $\mathbb{E}_\epsilon \left[\| h_\theta(x_t, \sigma_t, y) \|_{\ell_2}^2 \right]$ и $\mathbb{E}_\epsilon \left[h_\theta(x_t, \sigma_t, y) \right]$ сглажены гауссовским шумом, они удовлетворяют условию Липшица. Следовательно, логиты диффузионных классификаторов также липшицевы, что означает робастность всего диффузионного классификатора.

Теорема VIII.2 (Верхняя граница константы Липшица диффузионного классификатора)**.** *Для диффузионного классификатора DC выполняется неравенство:*

$$|DC(x_0+\epsilon)_y - DC(x_0)_y| \leq \frac{1}{2\sqrt{2}} \sum_{t=1}^{T} \frac{w_t}{\sigma_t T} \left(\sqrt{\frac{2}{\pi}} + \frac{2}{\sqrt{d}} \right) \|\epsilon\|_{\ell_2} \quad (95)$$

Если оценить нижнюю границу $\underline{p_A}$ для $DC(x_0)_y$ и верхнюю границу $\overline{p_B}$ для $\max_{\hat{y} \neq y} DC(x_0)_{\hat{y}}$ с помощью концентрационных неравенств, можно получить нижнюю границу радиуса устойчивости:

$$R = \frac{\sqrt{2T}(\underline{p_A} - \overline{p_B})}{(2/\sqrt{d} + \sqrt{2/\pi}) \sum_{t=1}^{T} w_t/\sigma_t} \quad (96)$$

Однако, такая сертифицированная робастность имеет ограничения, поскольку предполагает выполнение максимального условия Липшица на всём пути возмущения.

b) Диффузионный классификатор с точным зашумлённым апостериорным распределением: Случайное сглаживание требует, чтобы базовый классификатор мог классифицировать данные из зашумлённого распределения $q(x_\tau)$. Однако стандартный диффузионный классификатор ограничен классификацией данных только из $q(x_0)$.

Для решения этой проблемы предлагается обобщение диффузионного классификатора для работы с любыми изображениями из $q(x_\tau)$ для произвольного τ. Основная идея заключается в выводе ELBO для $\log p(x_\tau|y)$ и последующем вычислении $p(y|x_\tau)$ через теорему Байеса.

Теорема VIII.3 (ELBO для зашумлённых данных)**.** *ELBO для* $\log p(x_\tau)$ *определяется как:*

$$\begin{aligned}\log p(x_\tau) &\geq C_3 - \sum_{t=\tau+1}^{T} \mathbb{E}\left[D_{KL}(q(x_t|x_{t+1},x_\tau)\|p(x_t|x_{t+1}))\right] \\ &= C_4 + \sum_{t=\tau}^{T} w_t^{(\tau)} \mathbb{E}\left[\|\mathbb{E}[q(x_t|x_{t+1},x_\tau)] - \mathbb{E}[p(x_t|x_{t+1})]\|_{\ell_2}^2\right]\end{aligned} \tag{97}$$

где

- $x_{t+1} \sim q(x_{t+1}|x_\tau)$,
- $w_t^{(\tau)} = \frac{\sigma_{t+1}^2-\sigma_\tau^2}{2(\sigma_t^2-\sigma_\tau^2)(\sigma_{t+1}^2-\sigma_t^2)}$,
- $\mathbb{E}[q(x_t|x_{t+1},x_\tau)] = \frac{(\sigma_{t+1}^2-\sigma_t^2)x_\tau+(\sigma_t^2-\sigma_\tau^2)x_{t+1}}{\sigma_{t+1}^2-\sigma_\tau^2}$,
- $\mathbb{E}[p(x_t|x_{t+1})] = \frac{(\sigma_{t+1}^2-\sigma_t^2)h_\theta(x_{t+1},\sigma_{t+1})+\sigma_t^2 x_{t+1}}{\sigma_{t+1}^2}$

Основываясь на теореме VIII.3, можно аппроксимировать $\log p(x_\tau|y)$ через его ELBO и вычислить $p(y|x_\tau) = \frac{e^{\log p_\theta(x_\tau|y)}}{\sum_{\hat{y}} e^{\log p_\theta(x_\tau|\hat{y})}}$ для классификации.

5) Алгоритмическая реализация: Предлагается эффективный алгоритм «Sift-and-Refine» VIII-C5, который сначала отсеивает маловероятные классы на основе грубых оценок ELBO, а затем уточняет оценки для оставшихся кандидатов.

Алгоритм 8 Sift-and-refine

Require: Функция вычисления ELBO для данного временного шага t и класса y, обозначенная как e_θ; зашумленное входное изображение x_τ; временные шаги просеивания $\{t_i\}_{i=0}^{T_s}$; шаги уточнения $\{t_i\}_{i=0}^{T_r}$; порог τ.

1: Инициализировать список кандидатов классов $C = \{0, 1, \ldots, K\}$.
2: **for** $i = 0$ to T_s **do**
3: **for** все классы y в C **do**
4: Вычислить ELBO для класса y на временном шаге t_i:

$$e_y = e_\theta(x_t, \sigma_{t_i}, y).$$

5: **end for**
6: Найти класс m с минимальным ELBO:

$$m = \arg\min_{y \in C} e_y.$$

7: Обновить C, удалив классы с значением функции ошибки реконструкции τ, превышающей таковую для m:

$$C = \{y \in C : e_y - e_m < \tau\}.$$

8: **end for**
9: Реинициализировать e_y: $e_y = \infty \; \forall y \notin C, 0 \; \forall y \in C$.
10: **for** $i = 0$ to T_r **do**
11: **for** все классы y в C **do**
12: Вычислить и накопить ELBO для класса y на временном шаге t_i:

$$t_i : e_y = e_y + e_\theta(x_t, t_i, y).$$

13: **end for**
14: **end for**
15: **return** $\tilde{y} = \arg\min_y e_y$.

IX. Вероятностная сертификация и трансформации

Данный раздел рассматривает методы вероятностной сертификации робастности против семантических трансформаций, развивая концепции случайного сглаживания для более сложных моделей угроз.

A. TSS: случайное сглаживание против состязательных трансформаций для робастной сертификации

Фреймворк TSS (Transformation-Specific Smoothing) [24] представляет методологию для сертификации робастности систем машинного обучения против общих состязательных трансформаций посредством специализированных функций сглаживания. Данный подход расширяет классическое случайное сглаживание на более широкий класс возмущений.

1) Математические основы и обозначения: Определим основные математические объекты, используемые в данном разделе:

- $\mathcal{X} \subseteq \mathbb{R}^d$ – пространство входных данных размерности d
- $\mathcal{Y} = \{1, \ldots, K\}$ – множество меток классов, где $C \geq 2$ – количество классов
- $\mathcal{Z} \subseteq \mathbb{R}^m$ – пространство параметров трансформации размерности m
- $\mathbb{P}_X$ – вероятностная мера для случайной переменной X
- f_X – плотность вероятности случайной переменной X
- $\mathbb{P}_X(S)$ – вероятность события S относительно меры $\mathbb{P}_X$
- $h : \mathcal{X} \to \mathcal{Y}$ – базовый классификатор, определяемый как $h(x) = \arg\max_{y \in \mathcal{Y}} p(y|x)$

2) Модель угроз и цель сертификации:

a) Семантические трансформации: Семантические трансформации моделируются как детерминированные функции $\phi : \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$, преобразующие изображение $x \in \mathcal{X}$ с использованием параметра $\alpha \in \mathcal{Z}$. Например, функция $\phi_R(x, \alpha)$ моделирует поворот изображения x на α градусов против часовой стрелки с использованием билинейной интерполяции.

Семантические трансформации классифицируются на основе их композиционных свойств. Ключевым критерием является возможность представления композиции трансформации ϕ с самой собой как единичной трансформации с модифицированным параметром, то есть существование такого γ, что $\phi(\phi(x, \alpha), \beta) = \phi(x, \gamma)$ для всех $\alpha, \beta \in \mathcal{Z}$.

Определение IX.1 (Разрешимая трансформация)**.** Трансформация $\phi : \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ называется разрешимой, если для любого $\alpha \in \mathcal{Z}$ существует разрешающая функция $\gamma_\alpha : \mathcal{Z} \to \mathcal{Z}$, которая является инъективной, непрерывно дифференцируемой с неисчезающим якобианом, и для которой выполняется:

$$\phi(\phi(x, \alpha), \beta) = \phi(x, \gamma_\alpha(\beta)), \quad \forall x \in \mathcal{X}, \beta \in \mathcal{Z} \tag{98}$$

Трансформация ϕ называется аддитивной, если $\gamma_\alpha(\beta) = \alpha + \beta$.

Определение IX.2 (Дифференциально разрешимая трансформация)**.** Пусть $\phi : \mathcal{X} \times \mathcal{Z}_\phi \to \mathcal{X}$ – трансформация с пространством параметров $\mathcal{Z}_\phi$, и $\psi : \mathcal{X} \times \mathcal{Z}_\psi \to \mathcal{X}$ – разрешимая трансформация с пространством параметров $\mathcal{Z}_\psi$. Трансформация ϕ

называется дифференциально разрешимой посредством ψ, если для любого $x \in \mathcal{X}$ существует функция $\delta_x : \mathcal{Z}_\psi \times \mathcal{Z}_\phi \to \mathcal{Z}_\psi$ такая, что для любых $\alpha \in \mathcal{Z}_\psi$ и $\beta \in \mathcal{Z}_\phi$:

$$\phi(\psi(x, \alpha), \beta) = \psi(\phi(x, \beta), \delta_x(\alpha, \beta)) \tag{99}$$

b) Модель угроз: Рассматривается состязательная модель, в которой атакующий применяет семантическую трансформацию ϕ с параметром α к входному изображению, преобразуя $x \mapsto \phi(x, \alpha)$. Атакующему разрешается выбирать произвольный параметр α из предопределённого пространства атаки $\mathcal{S} \subseteq \mathcal{Z}$.

c) Цель сертификации: Цель состоит в определении множества параметров, для которых модель гарантированно сохраняет робастность. Формально требуется найти множество $\mathcal{S}_{\text{cert}} \subseteq \mathcal{Z}$ такое, что для классификатора h и состязательной трансформации ϕ выполняется:

$$h(x) = h(\phi(x, \alpha)), \quad \forall \alpha \in \mathcal{S}_{\text{cert}} \tag{100}$$

3) Методология TSS: Имея произвольный базовый классификатор h, конструируется сглаженный классификатор g посредством случайного преобразования входов параметрами, выбираемыми из распределения сглаживания.

Определение IX.3 (Сглаженный классификатор для трансформаций)**.** Пусть $\phi : \mathcal{X} \times \mathcal{Z} \to \mathcal{X}$ – трансформация, $\epsilon \sim \mathbb{P}_\epsilon$ – случайная переменная со значениями в $\mathcal{Z}$, и $h : \mathcal{X} \to \mathcal{Y}$ – базовый классификатор. ϵ-сглаженный классификатор определяется как $g : \mathcal{X} \to \mathcal{Y}$:

$$g(x; \epsilon) = \arg\max_{y \in \mathcal{Y}} q(y|x; \epsilon) \tag{101}$$

где

$$q(y|x; \epsilon) = \mathbb{E}_\epsilon[p(y|\phi(x, \epsilon))] \tag{102}$$

4) Сертификация робастности: Цель заключается в нахождении множества параметров возмущения $\mathcal{S}_{\text{cert}}$, зависящего от вероятностей p_A, p_B и параметра сглаживания ϵ, такого что для всех возможных возмущений $\alpha \in \mathcal{S}_{\text{cert}}$ гарантируется:

$$g(\phi(x, \alpha); \epsilon) = g(x; \epsilon) \tag{103}$$

Данное условие означает, что предсказание сглаженного классификатора не может быть изменено применением трансформации ϕ с параметрами α из робастного множества $\mathcal{S}_{\text{cert}}$.

5) Алгоритмическая реализация: Для сертификации робастности против трансформации ϕ, которая может быть разрешена функцией ψ с параметрами из множества $\mathcal{S} \subseteq \mathcal{Z}_\phi$, применяется следующий алгоритм:

Теорема IX.1 (Гарантии робастности TSS)**.** *Пусть $\phi : \mathcal{X} \times \mathcal{Z}_\phi \to \mathcal{X}$ – трансформация, разрешимая посредством $\psi : \mathcal{X} \times \mathcal{Z}_\psi \to \mathcal{X}$. Пусть $\epsilon \sim \mathbb{P}_\epsilon$ – случайная переменная со значениями в $\mathcal{Z}_\psi$, и сглаженный классификатор $g : \mathcal{X} \to \mathcal{Y}$ задан условием $q(y|x; \epsilon) = \mathbb{E}_\epsilon[p(y|\psi(x, \epsilon))]$ с предсказанием $g(x; \epsilon) = y_A = \arg\max_y q(y|x; \epsilon)$.*

Пусть $\mathcal{S} \subseteq \mathcal{Z}_\phi$ и $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$ – множество параметров трансформации такие, что для любого i вероятности классов удовлетворяют:

$$q(y_A|\phi(x, \alpha_i); \epsilon) \geq p_A^{(i)} \geq p_B^{(i)} \geq \max_{y \neq y_A} q(y|\phi(x, \alpha_i); \epsilon) \tag{104}$$

Алгоритм 9 TSS: Сертификация против трансформаций

Require: Трансформация ϕ, разрешающая трансформация ψ, множество параметров $\mathcal{S}$, классификатор h

Выбрать множество параметров $\{\alpha_i\}_{i=1}^N \subseteq \mathcal{S}$
Вычислить трансформированные входы $\{\phi(x, \alpha_i)\}_{i=1}^N$
Для каждого $\phi(x, \alpha_i)$ вычислить вероятности классов с использованием ψ-сглаженного классификатора
if каждый параметр $\alpha \in \mathcal{S}$ достаточно близок к некоторому α_i ($\delta_x(\alpha, \alpha_i) \in \Delta^*$) **then**
    return Классификатор робастен относительно $\mathcal{S}$
else
    return Робастность не гарантирована
end if

Тогда существует множество $\Delta^* \subseteq \mathcal{Z}_\psi$ такое, что если для любого $\alpha \in \mathcal{S}$ существует α_i с $\delta_x(\alpha, \alpha_i) \in \Delta^*$, то гарантируется:

$$q(y_A|\phi(x, \alpha); \epsilon) > \max_{y \neq y_A} q(y|\phi(x, \alpha); \epsilon) \tag{105}$$

B. Сертифицированная защита с помощью случайного сглаживания от трансформации изображений

В работе [25] рассматривается методы сертификации геометрических трансформаций.

Обобщим теорему о гарантиях робастности для параметризованных трансформаций. Рассмотрим составные трансформации $\psi_\beta : \mathbb{R}^m \to \mathbb{R}^m$, удовлетворяющие $\psi_\beta \circ \psi_\gamma = \psi_{\beta+\gamma}$ для любых $\beta, \gamma \in \mathbb{R}^d$. Теперь можем определить сглаженный классификатор $g : \mathbb{R}^m \to \mathcal{Y}$ для трансформации ψ_β

$$g(x) = \arg\max_c \mathbb{P}_{\beta \sim \mathcal{N}(0, \sigma^2 1)}(f \circ \psi_\beta(x) = c) \tag{106}$$

Теорема IX.2. *(Гарантии робастности). Пусть $x \in \mathbb{R}^m, f : \mathbb{R}^m \to \mathcal{Y}$ - классификатор и $\psi_\beta : \mathbb{R}^m \to \mathbb{R}^m$ - составная трансформация. Если*

$$\mathbb{P}_\beta(f \circ \psi_\beta(x) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c_B \neq c_A} \mathbb{P}_\beta(f \circ \psi_\beta(x) = c_B) \tag{107}$$

тогда $g \circ \psi_\gamma(x) = c_A \forall \gamma : ||\gamma||_2 \leq \frac{\sigma}{2}(\Phi^{-1}(\underline{p_A}) - \Phi^{-1}(\overline{p_B})) =: r_\gamma$

C. Сертифицированная состязательная робастность с дополнительным шумом

В работе [26] авторы предлагают фреймворк, который позволяет вычислить верхнюю грань возмущений, при которых классификатор будет давать корректные предсказания.

Цель – показать, что если алгоритм классифицировал x в класс c, то для любых примеров $\|x - x'\|_{\ell_2} \leq L$, x' также будет классифицирован в класс c. Оценка выводится из следующих лемм и теорем:

Определение IX.4. (Дивергенция Реньи) Для двух распределений вероятностей P и Q над $\mathcal{R}$, дивергенция Реньи порядка $\alpha > 1$ определяется следующим образом:

$$D_\alpha(P||Q) = \frac{1}{\alpha - 1} log \mathbb{E}_{x \sim Q}(\frac{P}{Q})^\alpha \tag{108}$$

Лемма IX.1. *Пусть $P = (p_1, \ldots, p_k)$ и $Q = (q_1, \ldots, q_k)$ - два мультиномиальных распределения над одним и*

тем же множеством индексов $\{1,\dots,k\}$. *Если индексы наибольших вероятностей* P *и* Q *различны, т.е.* $\arg\max_i p_i \neq \arg\max_j q_j$, *тогда*

$$D_\alpha(Q||P) \geq -log(1-p_{(1)}-p_{(2)}+2(\frac{1}{2}(p_{(1)}^{1-\alpha}+p_{(2)}^{1-\alpha}))^{\frac{1}{1-\alpha}}) \tag{109}$$

где $p_{(1)}, p_{(2)}$ *– наибольшая и вторая по величине вероятности среди всех* p_i

Для упрощения обозначим $M_p(x_1,\dots,x_n) = (\frac{1}{n}\sum_{i=1}^n x_i^p)^{1/p}$ как среднее степенное. Тогда, правая часть леммы превратится в $-log(1-2M_1(p_{(1)},p_{(2)}) + 2M_{1-\alpha}(p_{(1)},p_{(2)}))$

Теорема IX.3. *Пусть* $x \in \mathcal{X}$ *и возможный состязательный пример* $x' \in \mathcal{X}$, *такой что* $||x-x'||_2 \leq L$. *Пусть дан классификатор* $f : \mathcal{X} \to \mathcal{Y}$ *и* $f(x+N(0,\sigma^2 I)) \sim (p_1,\dots,p_k)$ *и* $f(x'+N(0,\sigma^2 I)) \sim (p'_1,\dots,p'_k)$. *Если следующее условие выполнено, при условии, что* $p_{(1)}, p_{(2)}$ *- наибольшая и вторая по величине вероятности среди всех* $\{p_i\}$*:*

$$\sup_{\alpha>1} -\frac{2\sigma^2}{\alpha} log(1-2M_1(p_{(1)},p_{(2)})+2M_{1-\alpha}(p_{(1)},p_{(2)})) \geq L^2 \tag{110}$$

то $\arg\max_i p_i = argmax_j p'_j$

Алгоритм 10 Сертифицированно робастный классификатор

Require: Входное изображение **x**; Стандартное отклонение $\sigma > 0$; Классификатор f над $\{1,\dots,k\}$; Количество итераций n ($n = 1$ оптимально только дл робастного классфикатора, ищем c).

1: Пусть $i = 1$.
2: **for** $i \in [n]$ **do**
3: Добавить шум $\epsilon \sim N(0,\sigma^2)$ каждому пикселю x и применить классификатор f на нем. Выход $c_i = f(x+\epsilon)$.
4: **end for**
5: Оценить распределение выхода как $p_j = \frac{\#\{c_i = j : i = 1,\dots,n\}}{n}$.
6: Вычислить верхнюю грань:

$$L = \sup_{\alpha>1}\left(-\frac{2\sigma^2}{\alpha}\log\left(1-p_{(1)}-p_{(2)}+2\left(\frac{1}{2}\left(p_{(1)}^{1-\alpha}+p_{(2)}^{1-\alpha}\right)\right)^{\frac{1}{1-\alpha}}\right)\right)$$

где $p_{(1)}$ и $p_{(2)}$ первое и второе наибольшие значения в $p_1,\dots,p_k$.

7: Вернуть результат классификации $c = \text{argmax}_i\ p_i$ и допустимый размер атаки L.

X. Вероятностная сертификация и задача сегментации

Данный раздел рассматривает развитие концепции случайного сглаживания в задаче семантической сегментации. Семантическая сегментация представляет особый интерес для критически важных приложений, таких как медицинская диагностика и автономное вождение, где робастность является первостепенным требованием.

A. Масштабируемая сертифицированная сегментация с помощью случайного слаживания

Работа [27] предлагает метод, сертифицирующий робастность моделей, решающих задачу семантической сегментации. Модели компьютерного зрения демонстрируют уязвимость к состязательным атакам, что критично для приложений сегментации в областях с высокими требованиями к безопасности. Задача сертификации в семантической сегментации усложняется необходимостью обеспечения гарантий для каждого пикселя изображения, что приводит к экспоненциальному росту вычислительной сложности. В работе авторы сосредоточились на сертификации по l_2 норме, но также отмечается, что предложенный метод можно расширить и на другие нормы l_p.

1) Случайное сглаживание в рамках задачи сегментации: Пусть дан вход $x = \{x_i\}_{i=1}^N$, состоящий из N компонент $x_i \in \mathcal{X}$ (пикселей или точек), и множество классов $\mathcal{Y}$. Семантическую сегментацию можно рассматривать как функцию $f : \mathcal{X}^N \to \mathcal{Y}^N$ такую, что для каждой компоненты x_i определяется $f_i(x) = y_i \in \mathcal{Y}$, где f_i обозначает i-ю компоненту выхода f, полученную из x.

Предполагается, что $\mathcal{X} := \mathbb{R}^m$. Без ограничения общности используется $m = 3$, что соответствует цветовой схеме RGB для изображений и трёхмерным облакам точек.

a) Наивные подходы и их ограничения: **Совместная классификация.** Задача сегментации $f : X^N \to Y^N$ переформулируется через декартово произведение $V := X_{i=1}^N Y$ и введение новой функции $f' : X^N \to V$, которая выполняет классификацию.

В таком случае классификатору f' может быть применена процедура CERTIFY из алгоритма 1. Однако изменение в результате классификации одного компонента x_i изменит класс в V, что усложняет поиск мажоритарного класса $\hat{c}_A$ с высоким значением $\underline{p_A}$.

Независимая классификация. Альтернативный подход состоит в классификации каждой компоненты независимо. Обозначив i-ю компоненту $f(x)$ как $f_i(x)$, применяется процедура CERTIFY из алгоритма 1 N раз для оценки $\hat{f}_i(x)$ и определения классов $\hat{c}_{A,1},\dots,\hat{c}_{A,N}$ и радиусов $R_1,\dots,R_N$. Общий радиус определяется как $R = \min_i R_i$.

Для снижения вычислительных затрат возможно переиспользование входных выборок для всех компонент выходного вектора, выполняя выборку $f(x)$ вместо индивидуальных компонент $f_i(x)$.

Проблема множественного тестирования. Каждый вызов CERTIFY выполняется с вероятностью корректности $1-\alpha$. Общая вероятность ограничена неравенством:

$$\begin{aligned}\mathbb{P}\left(\bigvee_i i\text{-й тест некорректен}\right) \\ \leq \min\left(\sum_{i=1}^N \mathbb{P}(i\text{-й тест некорректен}), 1\right) \\ = \min(N\alpha, 1)\end{aligned} \tag{111}$$

Для больших N это становится критической проблемой, требующей компенсации через выполнение вызовов CERTIFY с $\alpha' = \frac{\alpha}{N}$, что приводит к экспоненциальному росту вычислительной сложности.

b) Ключевые проблемы:

- **«Плохие компоненты»**: Оба алгоритма могут давать неудовлетворительные результаты или определять малый радиус сертификации из-за единственной компоненты x_i, на которой исходный классификатор строит нестабильный прогноз
- **Компромисс множественного тестирования**: Любой алгоритм, сводящий сертификацию сегментации к множеству стохастических тестов, страдает от проблемы множественного тестирования, приводящей к выбору между масштабируемостью и статистической достоверностью

2) Масштабируемая сертификация сегментации: Для решения выявленных проблем предлагается специализированный алгоритм с двумя ключевыми инновациями.

а) Решение проблемы «плохих компонент»: Для уменьшения влияния нестабильных компонент вводится порог $\tau \in [\frac{1}{2}, 1]$ и модель $\hat{f}^\tau : \mathcal{X}^N \to \hat{\mathcal{Y}}^N$ с расширенным множеством классов $\hat{\mathcal{Y}} = \mathcal{Y} \cup \{\varnothing\}$, которая возвращает символ воздержания $\varnothing$, если вероятность наиболее вероятного класса для компоненты x_i ниже порога τ:

$$\hat{f}_i^\tau(x) = \begin{cases} c_{A,i} & \text{если } \mathbb{P}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I)}[f_i(x+\epsilon) = c_{A,i}] > \tau \\ \varnothing & \text{иначе} \end{cases} \tag{112}$$

где $c_{A,i} = \arg\max_{c\in\mathcal{Y}} \mathbb{P}_{\epsilon\sim\mathcal{N}(0,\sigma^2 I)}[f_i(x+\epsilon) = c]$.

Данная модель воздерживается от прогноза для компонент, в которых она не уверена, сохраняя при этом теоретические гарантии.

Теорема X.1 (Гарантии робастности с воздержанием)**.** *Пусть $I_x = \{i \mid \hat{f}_i^\tau(x) \neq \varnothing, i \in \{1,\dots,N\}\}$ обозначает множество индексов, для которых модель построила прогноз. Тогда*

$$\hat{f}_i^\tau(x+\delta) = \hat{f}_i^\tau(x), \quad \forall i \in I_x \tag{113}$$

для любого $\delta \in \mathbb{R}^{N\times m}$ при $\|\delta\|_{\ell_2} \leq R := \sigma\Phi^{-1}(\tau)$.

Как и в оригинальном случае авторы не могут напрямую использовать $\hat{f}^\tau$, а только ее аппроксимацию. Тогда самое простое, это вызывать CERTIFY для каждого компонента и заменить проверку $\underline{p_A} > \frac{1}{2}$ и $\underline{p_A} > \tau$.

Таким образом проблема с «плохими компонентами» решена, остается проблема связанная с множественным тестированием. Для этого предлашается алгоритм SEGCERTIFY 11.

Алгоритм 11 Предсказание и Сертификация

function SegCertify(g, σ, x, n, n_0, δ, α)
  $\texttt{cnts}_1^0, \dots, \texttt{cnts}_N^0 \leftarrow$ Sample(g, x, n_0, σ)
  $\texttt{cnts}_1, \dots, \texttt{cnts}_N \leftarrow$ Sample(g, x, n, σ)
  for $i \leftarrow \{1,\dots,N\}$:
    $\hat{c}_i \leftarrow$ top index in $\texttt{cnts}_i^0$
    $n_i \leftarrow \texttt{cnts}_i[\hat{c}_i]$
    $pval_i \leftarrow$BinPValue(n_i, n, $\leq$, δ)
  $r_1, \dots, r_N \leftarrow$ FwerControl(α, $pval_1, \dots, pval_N$)
  for $i \leftarrow \{1,\dots,N\}$:
    if $\neg r_i$: $\hat{c}_i \leftarrow ABSTAIN$
  $R \leftarrow \sigma\Phi^{-1}(\delta)$
  return $\hat{c}_1, \dots, \hat{c}_N$, R

В алгоритме 11 функция Sample выполняет оценку выборок $f(x+\epsilon)$, где cnts_i обозначает вектор частот классов для i-й компоненты. Как и в алгоритме CERTIFY, используются две выборки cnts и cnts^0 для избежания смещения в выборе модели.

Алгоритм использует n_0 выборок для определения мажоритарного класса $c_{A,i}$ для i-й компоненты, затем определяет количество его появлений n_i при проведении n испытаний. Используя это количество, выполняется односторонний биномиальный тест для получения p-значения.

Функция FwerControl определяет, какие тесты следует отклонить для достижения требуемого уровня достоверности $1-\alpha$. Если i-й тест отклоняется ($r_i = \text{false}$), алгоритм воздерживается от прогноза для соответствующей компоненты.

B. На пути к улучшению сертифицированной сегментации с помощью диффузионных моделей

Работа [28] предлагает подход к повышению качества сертификации для задач сегментации путём интеграции диффузионных моделей в процесс случайного сглаживания. Алгоритм представляет собой модификацию базового подхода SegCertify из раздела X-A, дополненную этапом денойзинга:

Алгоритм 12 Sample функция

function Sample(g, x, n, σ)
  $\texttt{cnts} \leftarrow []$
  for 0 to $n-1$ **do**
    $t^\star, \beta_{t^\star} \leftarrow \texttt{computeTimestep}(\sigma)$
    $x_{t^\star} \leftarrow \sqrt{\beta_{t^\star}}(x + \mathcal{N}(0, \sigma^2\mathbf{I}))$
    $y \leftarrow g(\texttt{denoise}(x_{t^\star}; t^\star))$
    $\texttt{cnts}_y \leftarrow \texttt{cnts}_y + 1$
  return cnts

function computeTimestep(σ)
  $t^\star \leftarrow$ find t s.t. $\frac{1-\beta_t}{\beta_t} = \sigma^2$
  return $t^\star, \beta_{t^\star}$

1) Базовый метод сравнения: В качестве базового метода используется алгоритм SegCertify из раздела X-A.

XI. Случайное сглаживание против некоторых видов атак

Данный раздел исследует применение методов случайного сглаживания для защиты против специализированных типов атак, выходящих за рамки классических ℓ_p возмущений. Развивая теоретические основы II, здесь рассматриваются адаптации сглаживания для противодействия бэкдор-атакам, ℓ_0 атакам и другим специфическим моделям угроз.

A. RAB: доказуемая робастность против бэкдор-атак

Работа [29] представляет подход к обеспечению сертифицированной робастности против бэкдор-атак посредством адаптации методов случайного сглаживания. Данный метод решает критическую проблему защиты от скрытых уязвимостей, внедряемых в процессе обучения модели.

1) Формализация бэкдор-атак: Пусть $\Omega_x \in \mathbb{R}^d$ – бэкдор-паттерн, а $\Delta(\Omega_x) := \{\delta_1, \dots, \delta_r\}$ – множество бэкдор-возмущений. Предполагается, что базовый классификатор обучен на наборе данных с бэкдором, содержащем r примеров, заражённых бэкдор-паттернами из $\Delta(\Omega_x)$.

Цель состоит в том, чтобы предсказание классификатора, обученного на заражённом наборе данных $\mathcal{D}_{BD}(\Delta(\Omega_x))$, для входа $x+\Omega_x$ совпадало с предсказанием сглаженного классификатора, обученного на чистом наборе данных.

Определение XI.1 (Сглаженный классификатор с защитой от бэкдоров)**.** Пусть $f(x, \mathcal{D}) = \arg\max_y p(y|x, \mathcal{D})$ – базовый классификатор. Тогда сглаженный классификатор определяется как:

$$q(y|x, \mathcal{D}) = \mathbb{P}_{X,D}[f(x+X, \mathcal{D}+D) = y] \tag{114}$$

где $X \sim \mathbb{P}_X$ и $D \sim \mathbb{P}_D$ – независимые случайные переменные, выступающие в качестве сглаживающих распределений, причём D представляет собой множество

Алгоритм 13 DNN-RAB для обучения сертифицированно робастных DNN.

Require: Отравленное тренировочное множество $\mathcal{D} = \{(x_i + \delta_i, \tilde{y}_i)_{i=1}^n\}$, уровень шума σ, номер модели N

for $k = 1, \ldots, N$ **do**
  Сэмплировать $\epsilon_{k,1}, \ldots, \epsilon_{k,n} \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2 I_d)$.
  $\mathcal{D}_k = \{(x_i + \delta_i + \epsilon_{k,i}, \tilde{y}_i)_{i=1}^n\}$.
  $h_k = \texttt{train\_model}(\mathcal{D}_k)$.
  Сэмплировать u_k из $\mathcal{N}(0, \sigma^2 I_d)$ детерминированно с random seed, основанным на $hash(h_k)$.
end forreturn Набор моделей $\{(h_1, u_1), \ldots, (h_N, u_N)\}$

Алгоритм 14 Сертифицированный инференс моделей, обученных методом RAB.

Require: Тестировочный пример x, уровень шума σ, модели $\{(h_k, u_k)\}_{k=1}^N$, величина бэкдора $\|\delta\|_2$, количество отравленных тренировочных примеров r

$\texttt{counts} = \{k\colon h_k(x + u_k, \mathcal{D} + \epsilon_k) = y\}$ for $y = 1, \ldots, C$
$y_A, y_B =$ топ 2 индекса из $\texttt{counts}$
$n_A, n_B = \texttt{counts}[y_A], \texttt{counts}[y_B]$
$p_A, p_B = \texttt{calculate\_bound}(n_A, n_B, N, \alpha)$.
if $p_A > p_B$ **then**
  $R = \frac{\sigma}{2\sqrt{r}} \left(\Phi^{-1}(p_A) - \Phi^{-1}(p_B)\right)$
  if $R \geq \|\delta\|_2$ **then return** предсказание y_A, робастный радиус R.
  end if
end ifreturn ABSTAIN

независимых одинаково распределённых случайных величин.

Финальный сглаженный классификатор имеет вид:

$$g(x, \mathcal{D}) = \arg\max_y q(y|x, \mathcal{D}) \tag{115}$$

2) Теоретические гарантии:

Теорема XI.1 (Гарантии робастности против бэкдор атак)**.** *Пусть q – сглаженный классификатор со сглаживающим распределением $Z := (X, D)$. Пусть $\Omega_x \in \mathbb{R}^d$ и $\Delta := (\delta_1, \ldots, \delta_n)$ с $\delta_i \in \mathbb{R}^d$. Пусть $y_A \in \mathcal{Y}$, $p_A, p_B \in [0, 1]$, $y_A = g(x, \mathcal{D})$ и*

$$q(y_A|x, \mathcal{D}) \geq p_A \geq p_B \geq \max_{y \neq y_A} q(y|x, \mathcal{D}) \tag{116}$$

Если оптимальные ошибки второго рода для тестирования $Z \sim \mathbb{P}_0$ против альтернативы $Z + (\Omega_x, \Delta) \sim \mathbb{P}_1$ удовлетворяют:

$$\beta^*(1 - p_A; \mathbb{P}_0, \mathbb{P}_1) + \beta^*(p_B; \mathbb{P}_0, \mathbb{P}_1) > 1 \tag{117}$$

то гарантируется, что $y_A = \arg\max_y q(y|x + \Omega_x, \mathcal{D} + \Delta)$.

α – вероятность ошибки I рода, β – вероятность ошибки II рода, $\beta^*(\alpha_0; \mathbb{P}_0, \mathbb{P}_1) = \inf_{\phi: \alpha(\phi; \mathbb{P}_0) \leq \alpha_0} \beta(\phi; \mathbb{P}_1)$

B. Доказуемая робастность против объединения ℓ_0 состязательных атак

Работа [30] описывает метод предоставления формальных, детерминированных гарантий для предсказаний модели при ℓ_0 атаках, учитывающий как атаки уклонения, так и отравление обучающих данных.

Для произвольного входа (x, y) задача атакующего состоит в том, чтобы добиться $y \neq f(x)$. Атакующий может полностью контролировать некоторые признаки (количество контролируемых признаков определяется ℓ_0 нормой) как во время обучения, так и во время эксплуатации модели.

Определение XI.2 (Сертифицированная робастность признаков)**.** Пусть даны обучающий набор данных (X, y), модель f', обученная на (X', y), и произвольный элемент $x' \in \mathcal{X}$. Сертифицированная робастность признаков $r \in \mathbb{N}$ для произвольного (x, y) определяется как:

$$|X' \ominus X \cup x' \ominus x| \leq r \to y = f'(x') \tag{118}$$

где операция $\ominus$ для матриц признаков возвращает индексы различающихся столбцов, а для векторов – индексы различающихся элементов.

1) Робастность признаков через ансамблевое голосование: Для получения требуемых гарантий предлагается использовать ансамбль моделей, обученных на непересекающихся подмножествах признаков. На основе решений каждой модели с помощью функции решений определяется итоговое предсказание. Пусть ансамбль состоит из T подмоделей, каждая из которых использует признаки $S_1, \ldots, S_T$, где $S_i \subset \{1, 2, \ldots, d\}$ – индексы соответствующих признаков.

Для каждой подмодели t определяются отображения f_t, g_t, где $f_t(x) = \arg\max_{y \in \mathcal{Y}} g_t(x, y)$ и g_t представляет логиты модели t.

Вводятся агрегирующие функции:

$$\begin{aligned} \dot{c}_y(x) &:= \sum_{t=1}^{T} \mathbf{1}[f_t(x) = y] \\ \ddot{c}_y(x, y') &:= \sum_{t=1}^{T} \mathbf{1}[g_t(x, y) > g_t(x, y')] \end{aligned} \tag{119}$$

где $\dot{c}$ показывает количество моделей, которые проголосовали за этот класс, а $\ddot{c}_y$ количество моделей, которые дают большую вероятность классу y относительного y'.

Определяются доминирующие классы:

$$y_{\text{pl}} := \arg\max_{y \in \mathcal{Y}} \dot{c}_y(x) \tag{120}$$

$$y_{\text{ru}} := \arg\max_{y \in \mathcal{Y} \setminus \{y_{\text{pl}}\}} \dot{c}_y(x) \tag{121}$$

Функции разрыва, отображающие разницу уверенности ансамбля, определяются как:

$$\text{Gap}_{\text{vote}}(y, y'; x) := \dot{c}_y(x) - \dot{c}_{y'}(x) - 1[y' < y] \tag{122}$$

$$\text{Gap}_{\text{logit}}(y, y'; x) := \ddot{c}_y(x, y') - \ddot{c}_{y'}(x, y) - 1[y' < y] \tag{123}$$

Теорема XI.2 (Гарантии робастности для простого голосования)**.** *Пусть f – функция голосования для разделения признаков $S_1, S_2, \ldots, S_T$. Тогда для (x, y) сертифицированная робастность признаков равна:*

$$r_{pl} := \left\lfloor \frac{Gap_{vote}(y_{pl}, y_{ru}; x)}{2} \right\rfloor \tag{124}$$

2) Двухэтапное голосование: Предлагается усовершенствованный метод агрегации предсказаний через двухэтапную функцию:

Этап 1: Определение $y_{\text{pl}}, y_{\text{ru}}$ как описано выше.

Этап 2: Определение финального решения:

$$f(x) = y_{\text{RO}} := \begin{cases} y_{\text{pl}} & \text{если } \text{Gap}_{\text{logit}}(y_{\text{pl}}, y_{\text{ru}}; x) \geq 0 \\ y_{\text{ru}} & \text{иначе} \end{cases} \quad (125)$$

Чтобы изменить предсказание модели при такой функции решений, необходимо либо чтобы неправильная метка была выбрана на втором этапе, либо чтобы неправильные метки были выбраны на первом этапе.

Для каждого случая определяется свой радиус, а итоговый радиус –это минимум из двух.

$$r_{\text{RO}}^{\text{Case1}} := \min_{y \in \mathcal{Y} \setminus y_{\text{RO}}} \max \left\{ \left\lfloor \frac{\text{Gap}_{\text{vote}}(\tilde{y}_{\text{RO}}, y)}{2} \right\rfloor, \left\lfloor \frac{\text{Gap}_{\text{vote}}(y_{\text{RO}}, y)}{2} \right\rfloor \right\} \quad (126)$$

$$r_{\text{RO}}^{\text{Case2}} := \min_{y, y' \in \mathcal{Y} \setminus y_{\text{RO}}} \text{dp}\left[\text{Gap}_{\text{vote}}(y_{\text{RO}}, y), \text{Gap}_{\text{vote}}(y_{\text{RO}}, y')\right] \quad (127)$$

где $\text{dp}[\Delta, \Delta'] = 1 + \min\{\text{dp}[\Delta - 2, \Delta' - 1], \text{dp}[\Delta - 1, \Delta' - 2]\}$ находится с помощью динамического программирования, учитывая что $\text{dp}(\Delta, \Delta') = 0$, когда $\max\{\Delta, \Delta'\} \leq 1$.

Теорема XI.3 (Гарантии робастности для двухэтапного голосования)**.** *Пусть f – функция, определённая уравнением 125. Тогда сертифицированная робастность признаков равна:*

$$r_{RO} := \min\{r_{RO}^{Case\ 1}, r_{RO}^{Case\ 2}\} \quad (128)$$

где радиусы для отдельных случаев определяются через динамическое программирование и анализ возможных сценариев изменения голосов.

C. Сертифицированная робастность к состязательным атакам с дифференциальной приватностью

В работе [31] предлагается использовать дифференциальную приватность (differential privacy или DP) [32], чтобы получить гарантии робастности для модели.

1) Основы дифференциальной приватности:

Определение XI.3 (Дифференциальная приватность)**.** Случайный алгоритм A, принимающий обучающие данные d и возвращающий значение из пространства $\mathcal{O}$, удовлетворяет (ϵ, δ)-дифференциальной приватности относительно метрики ρ, если для любых d, d' таких, что $\rho(d, d') \leq 1$, и для любого подмножества $S \subseteq \mathcal{O}$ выполнено:

$$\mathbb{P}[A(d) \in S] \leq e^{\epsilon}\mathbb{P}[A(d') \in S] + \delta \quad (129)$$

где $\epsilon > 0, \delta \in [0, 1]$ – параметры, отражающие уровень приватности.

Из XI.3 следует, что:

Лемма XI.1. *Пусть A – это случайная функция, с ограничением $A(x) \in [0, b], b \in R^+$, удовлетворяющая (ϵ, δ)-DP. Тогда математическое ожидание значения выхода удовлетворяет следующему неравенству:*

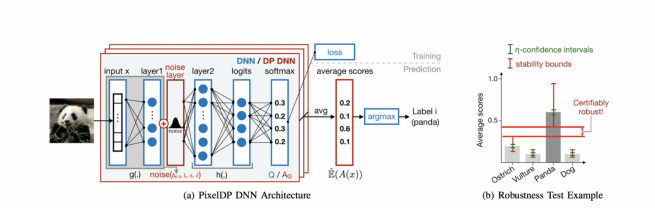
$$\forall \alpha \in B_p(1). \mathbb{E}(A(x)) \leq e^{\epsilon}\mathbb{E}(A(x + \alpha)) + b\delta, \quad (130)$$

где $B_p(b)$ шар с центром в нуле, радиуса b по норме ℓ_p.

Теорема XI.4 (Робастность через дифференциальную приватность)**.** *Пусть A – (ϵ, δ)-PixelDP алгоритм с ℓ_p нормой и параметром L. Для любого x, если существует $k \in \mathcal{K}$ такое, что:*

$$\mathbb{E}(A_k(x)) \geq e^{2\epsilon} \max_{i \neq k} \mathbb{E}(A_i(x)) + (1 + e^{\epsilon})\delta, \quad (131)$$

Тогда мультиклассовая классификация на основе значений $y(x) = (\mathbb{E}(A_1(x)), \ldots, \mathbb{E}(A_K(x)))$ робастна к атакам с наложением $\alpha : ||\alpha||_p \leq 1$ для x.



(a) PixelDP DNN Architecture (b) Robustness Test Example

Рис. 7. Архитектура PixelDP

2) Архитектурные решения: Для применения теории дифференциальной приватности к произвольной модели добавляется слой наложения шума, что позволяет получить требуемый уровень приватности. Выделяются четыре стратегии размещения шумового слоя:

1) Непосредственно после входного слоя;
2) После первого скрытого слоя;
3) В промежуточных слоях модели;
4) После автоэнкодера, добавленного в начало модели.

Для выбранного разделения модели $Q(x) = h(g(x))$ результирующий алгоритм имеет вид:

$$A_Q(x) = h(g(x) + \text{noise}(\Delta, L, \epsilon, \delta)), \quad (132)$$

где чувствительность отображения $g(x)$ определяется как:

$$\Delta_{p,q} = \Delta_{p,q}^{g} = \max_{x \neq x'} \frac{\|g(x) - g(x')\|_{\ell_q}}{\|x - x'\|_{\ell_p}} \quad (133)$$

Шум накладывается из распределения Лапласа или Гаусса. При заданных значениях среднего, равному нулю и $\sigma = \sqrt{2}\Delta_{p,1}L/\epsilon$ для распределения Лапласа получаем $(\epsilon, 0)$-DP. При заданных значениях среднего, равному нулю и $\sigma = \sqrt{\ln\left(\frac{1.25}{\delta}\right)}\Delta_{p,2}L/\epsilon$ для распределения Гаусса получаем (ϵ, δ) для $\epsilon \leq 1$.

3) Обучение: Во время обучения необходимо контролировать чувствительность до слоя с шумом.

4) Получение предсказания и уровня робастности: Оценим $\mathbb{E}(A_k(x))$ с помощью метода доверительных интервалов, получая $\mathbb{E}(A_k(x)) \in [\hat{\mathbb{E}}^{lb}(A_k(x)), \hat{\mathbb{E}}^{up}(A_k(x))]$ с вероятностью η. Также, с помощью метода Монте Карло вычислим значение $\hat{\mathbb{E}}(A_k(x))$. Тогда верна теорема:

Теорема XI.5. *Пусть A – (ϵ, δ)-PixelDP с p нормой и параметром L. Для любого x, если существует $k \in \mathcal{K}$, такое что*

$$\hat{\mathbb{E}}^{lb}(A_k(x)) \geq e^{2\epsilon} \max_{i \neq k} \hat{\mathbb{E}}^{ub}(A_i(x)) + (1 + e^{\epsilon})\delta, \quad (134)$$

Тогда мультиклассовая классификация на основе значений $y(x) = (\hat{\mathbb{E}}(A_1(x)), \ldots, \hat{\mathbb{E}}(A_K(x)))$ робастна к атакам с наложением $\alpha : ||\alpha||_p \leq L$ для x.

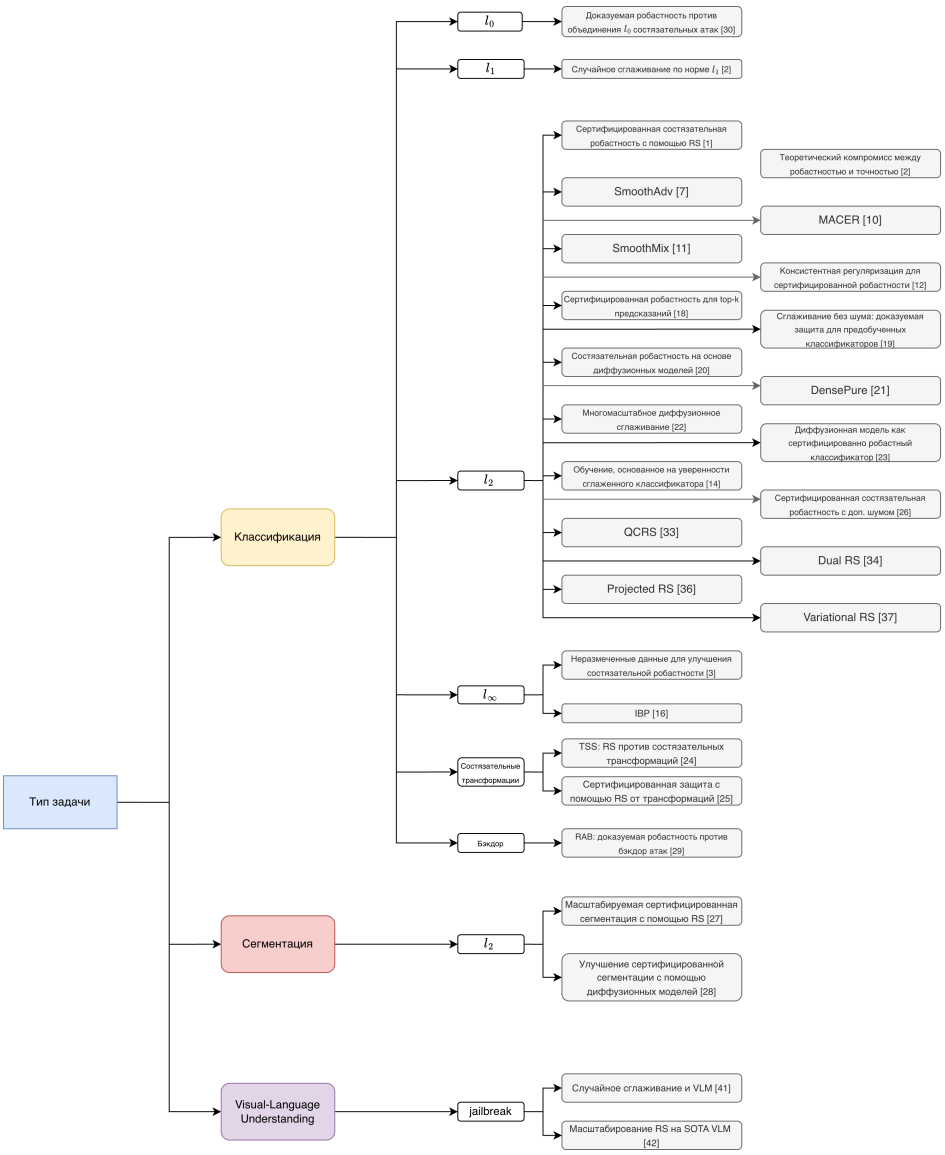


Рис. 8. Классификация методов сертификации робастности модели на основе типа решаемой задачи и типов атак. Здесь под сокращением RS понимается термин *случайное сглаживание* (randomized smoothing).

XII. Тенденции развития метода случайного сглаживания

Рассмотрев фундаментальные принципы и основные алгоритмические реализации метода случайного сглаживания, перейдём к анализу актуальных направлений его дальнейшего развития, отражающих современные тенденции в области сертифицированной робастности.

A. Оптимизационные и теоретические улучшения

Ряд работ сосредоточен на улучшении теоретических границ сертификации и эффективности алгоритмов. В работе QCRS [33] предложена оптимизация, обеспечивающая повышение радиуса сертификации без изменения структуры классификатора. Метод [34] направлен на уменьшение влияния размерности входного пространства и повышение устойчивости оценок. Исследование [35] расширяет применение сглаживания на ядровые методы и задачи обучения с регуляризацией. Работы [36] и [37] развивают идею проекционного и вариационного сглаживания, обеспечивая более гибкую оптимизацию сертификационных радиусов и повышение стабильности обучения.

B. Расширение класса распределений

Современные исследования активно выходят за пределы изотропного гауссовского шума, рассматривая более общие распределения и механизмы сглаживания. Методы [38], UCAN [39] демонстрируют возможность использования смесей, асимметричных и медианных распределений для адаптации сглаживания под свойства данных. Эти подходы позволяют лучше управлять компромиссом между точностью и сертифицированной робастностью в реальных задачах.

C. Статистические и оценочные подходы

Исследование [40] формулирует задачу случайного сглаживания как задачу статистической оценки параметров распределений вероятностей, что открывает возможности для более строгого анализа доверительных интервалов сертификации.

D. Расширение на крупные и мультимодальные модели

Недавние работы [41], [42] демонстрируют масштабирование метода на современные VLM модели и мультимодальные архитектуры, включая трансформеры с миллиардами параметров. Такие исследования подтверждают применимость метода сертификации даже в высокоразмерных и кроссмодальных пространствах признаков.

XIII. Сравнительный анализ

Исходя из рассмотренных в обзоре подходов авторы предлагают выбирать метод для сертификации робастности модели на основе типа решаемой задачи (классификация, сегментаци, vision-language understanding для VLM моделей) и предполагаемых типов атак – состязательных возмущений по нормам $\ell_1, \ell_2, \ell_\infty$, состязательных трансформаций, бэкдор-атак, ℓ_0 состязательных атак и jailbreak атак.

Выбор метода сертификации робастности определяется как типом решаемой задачи, так и предполагаемой моделью угроз. Для задач классификации в условиях плотных аддитивных возмущений наиболее естественным выбором являются методы гауссовского случайного сглаживания, обеспечивающие сертифицированную ℓ_2-робастность и хорошо масштабируемые на данные высокой размерности.

В сценариях разреженных атак более адекватной моделью угроз выступают ℓ_1- и ℓ_0-ограничения, для которых методы случайного сглаживания с равномерным шумом и его модификациями обеспечивают более интерпретируемые и релевантные гарантии устойчивости, особенно при учёте ограничений входных данных.

Для моделей угроз, выходящих за рамки аддитивного шума, включая состязательные трансформации и бэкдор-атаки, применяются специализированные методы сертификации, ориентированные на соответствующую природу атак. Аналогично, для задач семантической сегментации и vision–language understanding требуется адаптация методов сертификации с учётом структурированного выхода модели и мультимодальной природы данных. В частности, для VLM-моделей методы случайного сглаживания рассматриваются как средство повышения устойчивости к jailbreak атакам, направленным на изменение высокоуровневого поведения модели, а не только её точечных предсказаний. Полная классификация представлена на рис. 8.

XIV. Заключение

Данный обзор представляет комплексный анализ современного состояния методов случайного сглаживания для обеспечения сертифицированной робастности систем машинного обучения. Проведённое исследование охватывает широкий спектр подходов — от фундаментальных теоретических основ до специализированных адаптаций для конкретных моделей угроз и задач.

Авторами были рассмотрены различные модификации метода случайного сглаживания, показана его масштабируемость, применение в различных задачах, основные метрики, модели и наборы данных. Главным преимуществом рассмотренных методов является то, что для нейронной сети можно вычислить сертифицированнй радиус, внутри которого предсказание не будет изменяться в условиях атаки. Это, а так же теоретическая обоснованность рассмотренного семейства методов делает их интересными для применения, когда требуется гарантировать устойчивость модели к состязательному возмущению.

Библиография

[1] J. M. Cohen, E. Rosenfeld и J. Z. Kolter, *Certified Adversarial Robustness via Randomized Smoothing*, 2019.

[2] V. Voracek и M. Hein, «Improving l1-Certified Robustness via Randomized Smoothing by Leveraging Box Constraints,» в *Proceedings of the 40th International Conference on Machine Learning*, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato и J. Scarlett, ред., сер. Proceedings of Machine Learning Research, т. 202, PMLR, 23–29 Jul 2023, с. 35 198—35 222. url: https://proceedings.mlr.press/v202/voracek23a.html

[3] Y. Carmon, A. Raghunathan, L. Schmidt, P. Liang и J. C. Duchi, *Unlabeled Data Improves Adversarial Robustness*, 2022.

[4] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui и M. I. Jordan, *Theoretically Principled Trade-off between Robustness and Accuracy*, 2019. url: https://arxiv.org/abs/1901.08573

[5] L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar и A. Mądry, *Adversarially Robust Generalization Requires More Data*, 2018. url: https://arxiv.org/abs/1804.11285

[6] A. Krizhevsky и G. Hinton, «Learning multiple layers of features from tiny images,» University of Toronto, тех. отч. TR-2009, 2009. url: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[7] H. Salman и др., *Provably Robust Deep Learning via Adversarially Trained Smoothed Classifiers*, 2020.

[8] A. Madry, A. Makelov, L. Schmidt, D. Tsipras и A. Vladu, *Towards Deep Learning Models Resistant to Adversarial Attacks*, 2019. url: https://arxiv.org/abs/1706.06083

[9] J. Rony, L. G. Hafemann, L. S. Oliveira, I. B. Ayed, R. Sabourin и E. Granger, *Decoupling Direction and Norm for Efficient Gradient-Based L2 Adversarial Attacks and Defenses*, 2019. url: https://arxiv.org/abs/1811.09600

[10] R. Zhai и др., *MACER: Attack-free and Scalable Robust Training via Maximizing Certified Radius*, 2022.

[11] J. Jeong, S. Park, M. Kim, H.-C. Lee, D. Kim и J. Shin, *SmoothMix: Training Confidence-calibrated Smoothed Classifiers for Certified Robustness*, 2021.

[12] J. Jeong и J. Shin, *Consistency Regularization for Certified Robustness of Smoothed Classifiers*, 2021.

[13] L. Deng, «The mnist database of handwritten digit images for machine learning research,» *IEEE Signal Processing Magazine*, т. 29, № 6, с. 141—142, 2012.

[14] J. Jeong, S. Kim и J. Shin, *Confidence-aware Training of Smoothed Classifiers for Certified Robustness*, 2022.

[15] Z. Shi, Y. Wang, H. Zhang, J. Yi и C.-J. Hsieh, *Fast Certified Robust Training with Short Warmup*, 2021.

[16] S. Gowal и др., *On the Effectiveness of Interval Bound Propagation for Training Verifiably Robust Models*, 2019.

[17] V. Nair и G. E. Hinton, «Rectified linear units improve restricted boltzmann machines,» в *Proceedings of the 27th International Conference on International Conference on Machine Learning*, сер. ICML'10, Haifa, Israel: Omnipress, 2010, с. 807—814.

[18] J. Jia, X. Cao, B. Wang и N. Z. Gong, *Certified Robustness for Top-k Predictions against Adversarial Perturbations via Randomized Smoothing*, 2019.

[19] H. Salman, M. Sun, G. Yang, A. Kapoor и J. Z. Kolter, *Denoised Smoothing: A Provable Defense for Pretrained Classifiers*, 2020.

[20] N. Carlini, F. Tramer, K. D. Dvijotham, L. Rice, M. Sun и J. Z. Kolter, *(Certified!!) Adversarial Robustness for Free!* 2023.

[21] C. Xiao и др., *DensePure: Understanding Diffusion Models towards Adversarial Robustness*, 2022.

[22] J. Jeong и J. Shin, *Multi-scale Diffusion Denoised Smoothing*, 2023.

[23] H. Chen и др., *Your Diffusion Model is Secretly a Certifiably Robust Classifier*, 2024.

[24] L. Li и др., *TSS: Transformation-Specific Smoothing for Robustness Certification*, 2021.

[25] M. Fischer, M. Baader и M. Vechev, *Certified Defense to Image Transformations via Randomized Smoothing*, 2021.

[26] B. Li, C. Chen, W. Wang и L. Carin, *Certified Adversarial Robustness with Additive Noise*, 2019.

[27] M. Fischer, M. Baader и M. Vechev, *Scalable Certified Segmentation via Randomized Smoothing*, 2022.

[28] O. Laousy и др., *Towards Better Certified Segmentation via Diffusion Models*, 2023. url: https://arxiv.org/abs/2306.09949

[29] M. Weber, X. Xu, B. Karlaš, C. Zhang и B. Li, *RAB: Provable Robustness Against Backdoor Attacks*, 2023.

[30] Z. Hammoudeh и D. Lowd, *Provable Robustness Against a Union of ℓ_0 Adversarial Attacks*, 2024.

[31] M. Lecuyer, V. Atlidakis, R. Geambasu, D. Hsu и S. Jana, *Certified Robustness to Adversarial Examples with Differential Privacy*, 2019.

[32] C. Dwork, «Differential Privacy: A Survey of Results,» в *Theory and Applications of Models of Computation*, M. Agrawal, D. Du, Z. Duan и A. Li, ред., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, с. 1—19.

[33] B.-H. Kung и S.-T. Chen, *Towards Large Certified Radius in Randomized Smoothing using Quasiconcave Optimization*, 2023. url: https://arxiv.org/abs/2302.00209

[34] S. Xia, Y. Yu, X. Jiang и H. Ding, *Mitigating the Curse of Dimensionality for Certified Robustness via Dual Randomized Smoothing*, 2024. url: https://arxiv.org/abs/2404.09586

[35] L. Ding, T. Hu, J. Jiang, D. Li, W. Wang и Y. Yao, *Random Smoothing Regularization in Kernel Gradient Descent Learning*, 2023. url: https://arxiv.org/abs/2305.03531

[36] S. Pfrommer, B. G. Anderson и S. Sojoudi, *Projected Randomized Smoothing for Certified Adversarial Robustness*, 2023. url: https://arxiv.org/abs/2309.13794

[37] R. Hase, Y. Wang, T. Koike-Akino, J. Liu и K. Parsons, *Variational Randomized Smoothing for Sample-Wise Adversarial Robustness*, 2024. url: https://arxiv.org/abs/2407.11844

[38] V. Rostermundt и B. G. Anderson, «Certified Adversarial Robustness via Mixture-of-Gaussians Randomized Smoothing,» в *NeurIPS 2025 Workshop: Reliable ML from Unreliable Data*, 2025. url: https://openreview.net/forum?id=ZyGMTcNaio

[39] H. Hong, A. Kundu, A. Payani, B. Wang и Y. Hong, *Towards Strong Certified Defense with Universal Asymmetric Randomization*, 2025. url: https://arxiv.org/abs/2510.19977

[40] V. Voracek, *Treatment of Statistical Estimation Problems in Randomized Smoothing for Adversarial Robustness*, 2025. url: https://arxiv.org/abs/2406.17830

[41] E. Seferis, C. Wu, S. Kollias, S. Bensalem и C.-H. Cheng, *Randomized Smoothing Meets Vision-Language Models*, 2025. url: https://arxiv.org/abs/2509.16088

[42] E. Seferis, «Scaling Randomized Smoothing to state-of-the-art Vision-Language Models,» в *ICLR 2025 Workshop: VerifAI: AI Verification in the Wild*, 2025. url: https://openreview.net/forum?id=hyZePf0jxy

[43] G. Yang, T. Duan, J. E. Hu, H. Salman, I. Razenshteyn и J. Li, *Randomized Smoothing of All Shapes and Sizes*, 2020.

[44] C. Liang и X. Wu, *Mist: Towards Improved Adversarial Examples for Diffusion Models*, 2023.

[45] A. Pal и J. Sulam, *Understanding Noise-Augmented Training for Randomized Smoothing*, 2023.

[46] S. Wu, J. Wang, W. Ping, W. Nie и C. Xiao, *Defending against Adversarial Audio via Diffusion Model*, 2023.

[47] J. Buckman, A. Roy, C. Raffel и I. Goodfellow, «Thermometer Encoding: One Hot Way To Resist Adversarial Examples,» 2018. url: https://openreview.net/pdf?id=S18Su--CW

[48] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li и L. Fei-Fei, «Imagenet: A large-scale hierarchical image database,» в *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, с. 248—255.

[49] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu и A. Ng, «Reading Digits in Natural Images with Unsupervised Feature Learning,» 2011. url: https://api.semanticscholar.org/CorpusID:16852518

[50] K. He, X. Zhang, S. Ren и J. Sun, «Deep residual learning for image recognition,» в *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, с. 770—778.

[51] S. Zagoruyko и N. Komodakis, *Wide Residual Networks*, 2017. url: https://arxiv.org/abs/1605.07146

[52] S. Xie, R. Girshick, P. Dollár, Z. Tu и K. He, *Aggregated Residual Transformations for Deep Neural Networks*, 2017. url: https://arxiv.org/abs/1611.05431

[53] Y. LeCun, L. Bottou, Y. Bengio и P. Haffner, «Gradient-based learning applied to document recognition,» *Proceedings of the IEEE*, т. 86, № 11, с. 2278—2324, 1998.

[54] A. Dosovitskiy и др., *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*, 2021. url: https://arxiv.org/abs/2010.11929

[55] Y. Le и X. S. Yang, «Tiny ImageNet Visual Recognition Challenge,» 2015. url: https : / / api . semanticscholar.org/CorpusID:16664790

[56] A. Radford и др., *Learning Transferable Visual Models From Natural Language Supervision*, 2021. url: https://arxiv.org/abs/2103.00020

[57] J. Wang и др., *Deep High-Resolution Representation Learning for Visual Recognition*, 2020. url: https : // arxiv.org/abs/1908.07919

[58] A. X. Chang и др., *ShapeNet: An Information-Rich 3D Model Repository*, 2015. url: https://arxiv.org/abs/1512.03012

[59] R. Q. Charles, H. Su, M. Kaichun и L. J. Guibas, «PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation,» в *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, с. 77—85. doi: 10.1109/CVPR.2017.16

[60] H. Bao, L. Dong и F. Wei, «BEiT: BERT Pre-Training of Image Transformers,» *CoRR*, т. abs/2106.08254, 2021. url: https://arxiv.org/abs/2106.08254

[61] P. Münch, R. Mreches, M. Binder, H. A. Gündüz, X.-Y. To и A. McHardy, *deepG: Deep Learning for Genome Sequence Data*, R package version 0.3.1, https://deepg.de/, 2024. url: https : / / github . com / GenomeNet/deepG

[62] G. Singh, T. Gehr, M. Püschel и M. Vechev, «An abstract domain for certifying neural networks,» *Proceedings of the ACM on Programming Languages*, т. 3, с. 1—30, янв. 2019. doi: 10.1145/3290354

[63] K. Pei, Y. Cao, J. Yang и S. Jana, «Towards Practical Verification of Machine Learning: The Case of Computer Vision Systems,» дек. 2017. doi: 10 . 48550/arXiv.1712.01785

[64] J. Mohapatra, T.-W. Weng, P.-Y. Chen, S. Liu и L. Daniel, «Towards Verifying Robustness of Neural Networks Against A Family of Semantic Perturbations,» июнь 2020, с. 241—249. doi: 10 . 1109/CVPR42600.2020.00032

[65] M. Fischer, M. Baader и M. T. Vechev, «Certification of Semantic Perturbations via Randomized Smoothing,» *CoRR*, т. abs/2002.12463, 2020. url: https://arxiv.org/abs/2002.12463

[66] M. Cordts и др., «The Cityscapes Dataset for Semantic Urban Scene Understanding,» в *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[67] M. Everingham, L. V. Gool, C. K. I. Williams, J. M. Winn и A. Zisserman, «The Pascal Visual Object Classes (VOC) Challenge.,» *Int. J. Comput. Vis.*, т. 88, № 2, с. 303—338, 2010. url: http://dblp.uni-trier.de/db/journals/ijcv/ijcv88.html#EveringhamGWWZ10

[68] J. Ho, A. Jain и P. Abbeel, «Denoising Diffusion Probabilistic Models,» *CoRR*, т. abs/2006.11239, 2020. url: https://arxiv.org/abs/2006.11239

[69] J. Wang и др., «Deep High-Resolution Representation Learning for Visual Recognition,» *CoRR*, т. abs/1908.07919, 2019. url: http : / / arxiv . org / abs/1908.07919

[70] Z. Chen и др., *Vision Transformer Adapter for Dense Predictions*, 2023. url: https : / / arxiv . org / abs / 2205 . 08534

[71] *GitHub - fastai/imagenette: A smaller subset of 10 easily classified classes from Imagenet, and a little more French — github.com*, https://github.com/fastai/imagenette.

[72] *Weather Dataset — kaggle.com*, https://www.kaggle.com/datasets/muthuj7/weather-dataset.

[73] *Ames Housing Dataset — kaggle.com*, https : //www . kaggle . com / datasets / shashanknecrothapa / ames - housing-dataset.

[74] B. Delattre, P. Caillon, Q. Barthélemy, E. Fagnou и A. Allauzen, *Bridging the Theoretical Gap in Randomized Smoothing*, 2025. url: https : //arxiv . org / abs/2504.02412

Randomized Smoothing in Certified Robustness: Theory and a Systematic Review

Karine Ayrapetyants, Eugene Ilyushin

***Abstract*—Nowadays, as artificial intelligence systems are increasingly applied across various domains, the issue of their security has become ever more relevant. Naturally, neural network algorithms, which we currently associate with the concept of "artificial intelligence," are also susceptible to both intentional and unintentional perturbations. Therefore, providing guarantees for the robustness of their operation is an important task. One of the methods that enables addressing this problem is randomized smoothing. This method allows us to obtain formal guarantees on the performance of a classifier under a given data distribution. Randomized smoothing, as well as its modifications, will be reviewed in this survey.**

***Keywords*—randomized smoothing, neural network robustness, certified accuracy, machine learning**

References

[1] J. M. Cohen, E. Rosenfeld, and J. Z. Kolter, *Certified adversarial robustness via randomized smoothing*, 2019.

[2] V. Voracek and M. Hein, «Improving l1-certified robustness via randomized smoothing by leveraging box constraints», in *Proceedings of the 40th International Conference on Machine Learning*, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., ser. Proceedings of Machine Learning Research, vol. 202, PMLR, 23–29 Jul 2023, pp. 35 198–35 222. [Online]. Available: https://proceedings.mlr.press/v202/voracek23a.html

[3] Y. Carmon, A. Raghunathan, L. Schmidt, P. Liang, and J. C. Duchi, *Unlabeled data improves adversarial robustness*, 2022.

[4] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan, *Theoretically principled trade-off between robustness and accuracy*, 2019. [Online]. Available: https://arxiv.org/abs/1901.08573

[5] L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, and A. Mądry, *Adversarially robust generalization requires more data*, 2018. [Online]. Available: https://arxiv.org/abs/1804.11285

[6] A. Krizhevsky and G. Hinton, «Learning multiple layers of features from tiny images», University of Toronto, Tech. Rep. TR-2009, 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[7] H. Salman et al., *Provably robust deep learning via adversarially trained smoothed classifiers*, 2020.

[8] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, *Towards deep learning models resistant to adversarial attacks*, 2019. [Online]. Available: https://arxiv.org/abs/1706.06083

[9] J. Rony, L. G. Hafemann, L. S. Oliveira, I. B. Ayed, R. Sabourin, and E. Granger, *Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses*, 2019. [Online]. Available: https://arxiv.org/abs/1811.09600

[10] R. Zhai et al., *Macer: Attack-free and scalable robust training via maximizing certified radius*, 2022.

[11] J. Jeong, S. Park, M. Kim, H.-C. Lee, D. Kim, and J. Shin, *Smoothmix: Training confidence-calibrated smoothed classifiers for certified robustness*, 2021.

[12] J. Jeong and J. Shin, *Consistency regularization for certified robustness of smoothed classifiers*, 2021.

[13] L. Deng, «The mnist database of handwritten digit images for machine learning research», *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.

[14] J. Jeong, S. Kim, and J. Shin, *Confidence-aware training of smoothed classifiers for certified robustness*, 2022.

[15] Z. Shi, Y. Wang, H. Zhang, J. Yi, and C.-J. Hsieh, *Fast certified robust training with short warmup*, 2021.

[16] S. Gowal et al., *On the effectiveness of interval bound propagation for training verifiably robust models*, 2019.

[17] V. Nair and G. E. Hinton, «Rectified linear units improve restricted boltzmann machines», in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ser. ICML'10, Haifa, Israel: Omnipress, 2010, pp. 807–814.

[18] J. Jia, X. Cao, B. Wang, and N. Z. Gong, *Certified robustness for top-k predictions against adversarial perturbations via randomized smoothing*, 2019.

[19] H. Salman, M. Sun, G. Yang, A. Kapoor, and J. Z. Kolter, *Denoised smoothing: A provable defense for pretrained classifiers*, 2020.

[20] N. Carlini, F. Tramer, K. D. Dvijotham, L. Rice, M. Sun, and J. Z. Kolter, *(certified!!) adversarial robustness for free!*, 2023.

[21] C. Xiao et al., *Densepure: Understanding diffusion models towards adversarial robustness*, 2022.

[22] J. Jeong and J. Shin, *Multi-scale diffusion denoised smoothing*, 2023.

[23] H. Chen et al., *Your diffusion model is secretly a certifiably robust classifier*, 2024.

[24] L. Li et al., *Tss: Transformation-specific smoothing for robustness certification*, 2021.

[25] M. Fischer, M. Baader, and M. Vechev, *Certified defense to image transformations via randomized smoothing*, 2021.

[26] B. Li, C. Chen, W. Wang, and L. Carin, *Certified adversarial robustness with additive noise*, 2019.

[27] M. Fischer, M. Baader, and M. Vechev, *Scalable certified segmentation via randomized smoothing*, 2022.

[28] O. Laousy et al., *Towards better certified segmentation via diffusion models*, 2023. [Online]. Available: https://arxiv.org/abs/2306.09949

[29] M. Weber, X. Xu, B. Karlaš, C. Zhang, and B. Li, *Rab: Provable robustness against backdoor attacks*, 2023.

[30] Z. Hammoudeh and D. Lowd, *Provable robustness against a union of ℓ_0 adversarial attacks*, 2024.

[31] M. Lecuyer, V. Atlidakis, R. Geambasu, D. Hsu, and S. Jana, *Certified robustness to adversarial examples with differential privacy*, 2019.

[32] C. Dwork, «Differential privacy: A survey of results», in *Theory and Applications of Models of Computation*, M. Agrawal, D. Du, Z. Duan, and A. Li, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–19.

[33] B.-H. Kung and S.-T. Chen, *Towards large certified radius in randomized smoothing using quasiconcave optimization*, 2023. [Online]. Available: https://arxiv.org/abs/2302.00209

[34] S. Xia, Y. Yu, X. Jiang, and H. Ding, *Mitigating the curse of dimensionality for certified robustness via dual randomized smoothing*, 2024. [Online]. Available: https://arxiv.org/abs/2404.09586

[35] L. Ding, T. Hu, J. Jiang, D. Li, W. Wang, and Y. Yao, *Random smoothing regularization in kernel gradient descent learning*, 2023. [Online]. Available: https://arxiv.org/abs/2305.03531

[36] S. Pfrommer, B. G. Anderson, and S. Sojoudi, *Projected randomized smoothing for certified adversarial robustness*, 2023. [Online]. Available: https://arxiv.org/abs/2309.13794

[37] R. Hase, Y. Wang, T. Koike-Akino, J. Liu, and K. Parsons, *Variational randomized smoothing for sample-wise adversarial robustness*, 2024. [Online]. Available: https://arxiv.org/abs/2407.11844

[38] V. Rostermundt and B. G. Anderson, «Certified adversarial robustness via mixture-of-gaussians randomized smoothing», in *NeurIPS 2025 Workshop: Reliable ML from Unreliable Data*, 2025. [Online]. Available: https://openreview.net/forum?id=ZyGMTcNaio

[39] H. Hong, A. Kundu, A. Payani, B. Wang, and Y. Hong, *Towards strong certified defense with universal asymmetric randomization*, 2025. [Online]. Available: https://arxiv.org/abs/2510.19977

[40] V. Voracek, *Treatment of statistical estimation problems in randomized smoothing for adversarial robustness*, 2025. [Online]. Available: https://arxiv.org/abs/2406.17830

[41] E. Seferis, C. Wu, S. Kollias, S. Bensalem, and C.-H. Cheng, *Randomized smoothing meets vision-language models*, 2025. [Online]. Available: https://arxiv.org/abs/2509.16088

[42] E. Seferis, «Scaling randomized smoothing to state-of-the-art vision-language models», in *ICLR 2025 Workshop: VerifAI: AI Verification in the Wild*, 2025. [Online]. Available: https://openreview.net/forum?id=hyZePf0jxy

[43] G. Yang, T. Duan, J. E. Hu, H. Salman, I. Razenshteyn, and J. Li, *Randomized smoothing of all shapes and sizes*, 2020.

[44] C. Liang and X. Wu, *Mist: Towards improved adversarial examples for diffusion models*, 2023.

[45] A. Pal and J. Sulam, *Understanding noise-augmented training for randomized smoothing*, 2023.

[46] S. Wu, J. Wang, W. Ping, W. Nie, and C. Xiao, *Defending against adversarial audio via diffusion model*, 2023.

[47] J. Buckman, A. Roy, C. Raffel, and I. Goodfellow, «Thermometer encoding: One hot way to resist adversarial examples», 2018. [Online]. Available: https://openreview.net/pdf?id=S18Su--CW

[48] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, «Imagenet: A large-scale hierarchical image database», in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.

[49] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Ng, «Reading digits in natural images with unsupervised feature learning», 2011. [Online]. Available: https://api.semanticscholar.org/CorpusID:16852518

[50] K. He, X. Zhang, S. Ren, and J. Sun, «Deep residual learning for image recognition», in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[51] S. Zagoruyko and N. Komodakis, *Wide residual networks*, 2017. [Online]. Available: https://arxiv.org/abs/1605.07146

[52] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, *Aggregated residual transformations for deep neural networks*, 2017. [Online]. Available: https://arxiv.org/abs/1611.05431

[53] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, «Gradient-based learning applied to document recognition», *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[54] A. Dosovitskiy et al., *An image is worth 16x16 words: Transformers for image recognition at scale*, 2021. [Online]. Available: https://arxiv.org/abs/2010.11929

[55] Y. Le and X. S. Yang, «Tiny imagenet visual recognition challenge», 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:16664790

[56] A. Radford et al., *Learning transferable visual models from natural language supervision*, 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[57] J. Wang et al., *Deep high-resolution representation learning for visual recognition*, 2020. [Online]. Available: https://arxiv.org/abs/1908.07919

[58] A. X. Chang et al., *Shapenet: An information-rich 3d model repository*, 2015. [Online]. Available: https://arxiv.org/abs/1512.03012

[59] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, «Pointnet: Deep learning on point sets for 3d classification and segmentation», in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 77–85. doi: 10.1109/CVPR.2017.16

[60] H. Bao, L. Dong, and F. Wei, «Beit: BERT pre-training of image transformers», *CoRR*,

vol. abs/2106.08254, 2021. [Online]. Available: https://arxiv.org/abs/2106.08254

[61] P. Münch, R. Mreches, M. Binder, H. A. Gündüz, X.-Y. To, and A. McHardy, *Deepg: Deep learning for genome sequence data*, R package version 0.3.1, https://deepg.de/, 2024. [Online]. Available: https://github.com/GenomeNet/deepG

[62] G. Singh, T. Gehr, M. Püschel, and M. Vechev, «An abstract domain for certifying neural networks», *Proceedings of the ACM on Programming Languages*, vol. 3, pp. 1–30, Jan. 2019. doi: 10.1145/3290354

[63] K. Pei, Y. Cao, J. Yang, and S. Jana, «Towards practical verification of machine learning: The case of computer vision systems», Dec. 2017. doi: 10.48550/arXiv.1712.01785

[64] J. Mohapatra, T.-W. Weng, P.-Y. Chen, S. Liu, and L. Daniel, «Towards verifying robustness of neural networks against a family of semantic perturbations», Jun. 2020, pp. 241–249. doi: 10.1109/CVPR42600.2020.00032

[65] M. Fischer, M. Baader, and M. T. Vechev, «Certification of semantic perturbations via randomized smoothing», *CoRR*, vol. abs/2002.12463, 2020. [Online]. Available: https://arxiv.org/abs/2002.12463

[66] M. Cordts et al., «The cityscapes dataset for semantic urban scene understanding», in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[67] M. Everingham, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, «The pascal visual object classes (voc) challenge.», *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010. [Online]. Available: http://dblp.uni-trier.de/db/journals/ijcv/ijcv88.html#EveringhamGWWZ10

[68] J. Ho, A. Jain, and P. Abbeel, «Denoising diffusion probabilistic models», *CoRR*, vol. abs/2006.11239, 2020. [Online]. Available: https://arxiv.org/abs/2006.11239

[69] J. Wang et al., «Deep high-resolution representation learning for visual recognition», *CoRR*, vol. abs/1908.07919, 2019. [Online]. Available: http://arxiv.org/abs/1908.07919

[70] Z. Chen et al., *Vision transformer adapter for dense predictions*, 2023. [Online]. Available: https://arxiv.org/abs/2205.08534

[71] *GitHub - fastai/imagenette: A smaller subset of 10 easily classified classes from Imagenet, and a little more French — github.com*, https://github.com/fastai/imagenette.

[72] *Weather Dataset — kaggle.com*, https://www.kaggle.com/datasets/muthuj7/weather-dataset.

[73] *Ames Housing Dataset — kaggle.com*, https://www.kaggle.com/datasets/shashanknecrothapa/ames-housing-dataset.

[74] B. Delattre, P. Caillon, Q. Barthélemy, E. Fagnou, and A. Allauzen, *Bridging the theoretical gap in randomized smoothing*, 2025. [Online]. Available: https://arxiv.org/abs/2504.02412