Бустинг глубоких нейросетевых ансамблей

Шокоров Вячеслав Александрович

Московский физико-технический институт
Факультет управления и прикладной математики
Кафедра интеллектуальных систем

Научный руководитель к.ф.-м.н. Д. П. Ветров

Москва,
2023 г.

Решаемая задача

Глубокий ансамбль

Ансамблирование глубоких нейросетевых моделей, через усреднение предсказаний, является одним из лучших методов повышения обобщающей способности итогового алгоритма.

Проблема

Существует алгоритм бустинга, который показывает результаты лучше, чем ансамбль через усреднение. Но он работает только с слабыми моделями, которые не показывают 100%-ю точность на обучении. Нейронные сети таковыми не являются.

Решение

Предлагается использовать дисконтированную функцию потерь (1), для обучения моделей, а в качестве ошибки, необходимой для бустинга, использовать марджин (2).

Дисконтированная функция потерь

В качестве нейронной сети понимается параметрическая функция $f(x, \theta)$, где $x \in \mathbb{R}^{\mathrm{h} \times \mathrm{w} \times 3}$ — матрица изображения, $\theta \in \mathbb{R}^t$ — вектор параметров модели. $f(\cdot, \theta) : \mathbb{R}^{\mathrm{h} \times \mathrm{w} \times 3} \to \mathbb{R}^{\mathrm{K}}$, где K — число классов классификации.
Причем:

$$f(\cdot, \theta) = W \circ \mathrm{BN} \circ \dots$$

W — матрица весов последнего слоя, BN — слой нормировки по данным (батчу).

Дисконтированная функция потерь

Функционал $L(f, d)$ назовем *дисконтированной функцией потерь*, d — значение дисконта:

$$L(f, d) = \frac{1}{N} \sum_{x_i} -\log \frac{e^{f(x_i)_{y_i} - d}}{e^{f(x_i)_{y_i} - d} + \sum_{j \neq y_i} e^{f(x_i)_j}}, \tag{1}$$

Задачи решаемые дисконтированной функцией потерь

Проблема

В пространстве весов глубокой, перепараметризованной нейронной сети существует множество точек глобального минимума функции потерь. Минимумы отличаются по генерализации.

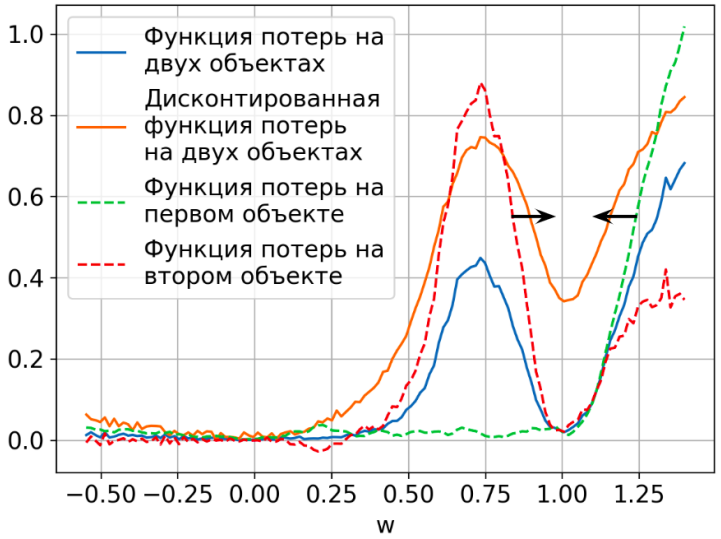
Решение

Дисконтированная функция потерь дополнительно штрафует модель за попадание в узкий минимум.

Также дисконтированная функция потерь позволяет обучать модель с требуемым итоговым значением марджина.

Сравнение узких и широких минимумов

(a) Визуализация преимущества широкого минимума: модель, которая сошлась в узкий минимум, будет иметь большую ошибку на тестовой выборке, чем модель, которая сошлась в широкий минимум.



(b) Эффект, который получаем от дисконтированной функции потерь, узкий, глобальный минимум перестает был глобальным.

Требования к архитектуре модели

Лемма 1

Пусть дана функция $f(x)$, и $f \not\equiv 0$ тогда $\forall g > 0$ и $\alpha > 0$ верно:

$$\lim_{\alpha \to \infty} |L(\alpha f, g) - L(\alpha f, 0)| = 0.$$

При увеличении нормы весов последнего слоя W уменьшается вклад, который достигается дисконтированной функцией потерь при ненулевом значении гэпа. Для компенсации данного эффекта предлагается замораживать веса последнего слоя. Такая модификация нейронной сети используется во всех последующих рассуждениях и экспериментах.

Марджин

Марджином модели $f(\cdot, \theta)$ назовем:

$$m(\theta, x, y) = f(x, \theta)_y - \max_{j \neq y} f(x, \theta)_j. \tag{2}$$

Теорема 1 (Шокоров 2023)

Пусть дана функция $f(x)$, с замороженными W — весами последнего слоя. $\mathcal{X} = \{(x, y) | x \in \mathbb{R}^{\mathrm{h} \times \mathrm{w} \times 3}, y \in \{1, 2 \dots K\}\}$ — множество объектов многоклассовой классификации, причем выборка равновесна, т.е. $\mathrm{P}_{(x,y)\sim\mathcal{X}}(y = k) = 1/K$. Тогда максимальное значение среднего марджина достигаемого функцией:

$$\bar{m} = \frac{1}{|\mathcal{X}|} \sum_{(x,y)\sim\mathcal{X}} m(\theta, x, y) \leq \frac{2}{K} \|W\|_1.$$

Теорема 2 (Шокоров 2023)

Для достаточно больших g_1 и g_2, для некоторой функции $f(x)$ с замороженными W — весами последнего слоя и множества объектов $\mathcal{X}$ верно, что существует предел:

$$\exists \lim_{d \to \infty} L(f, d) - d < \infty$$

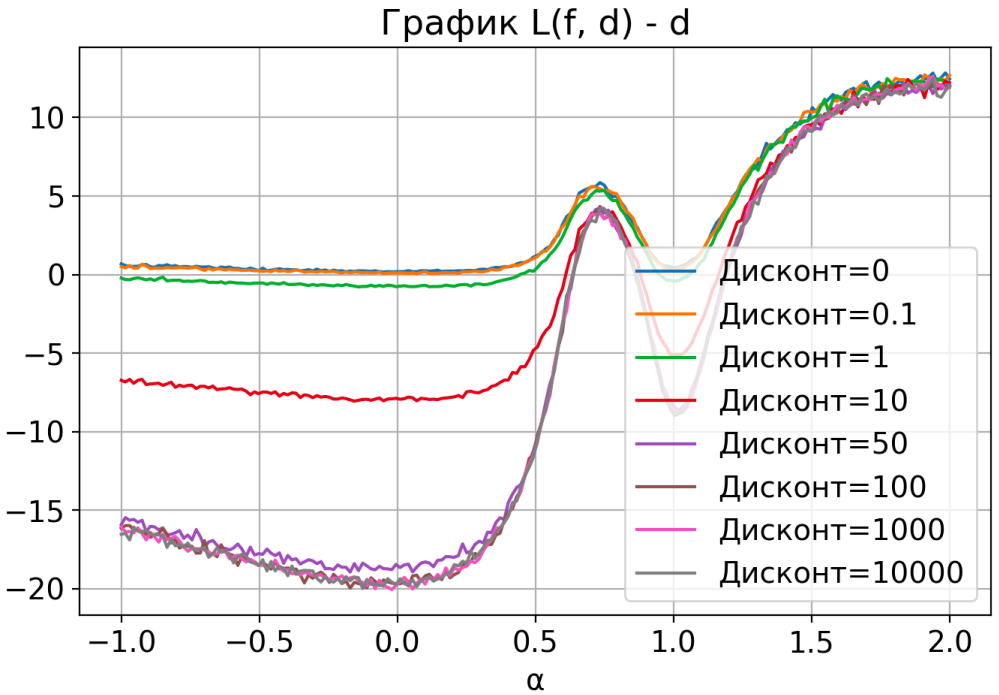
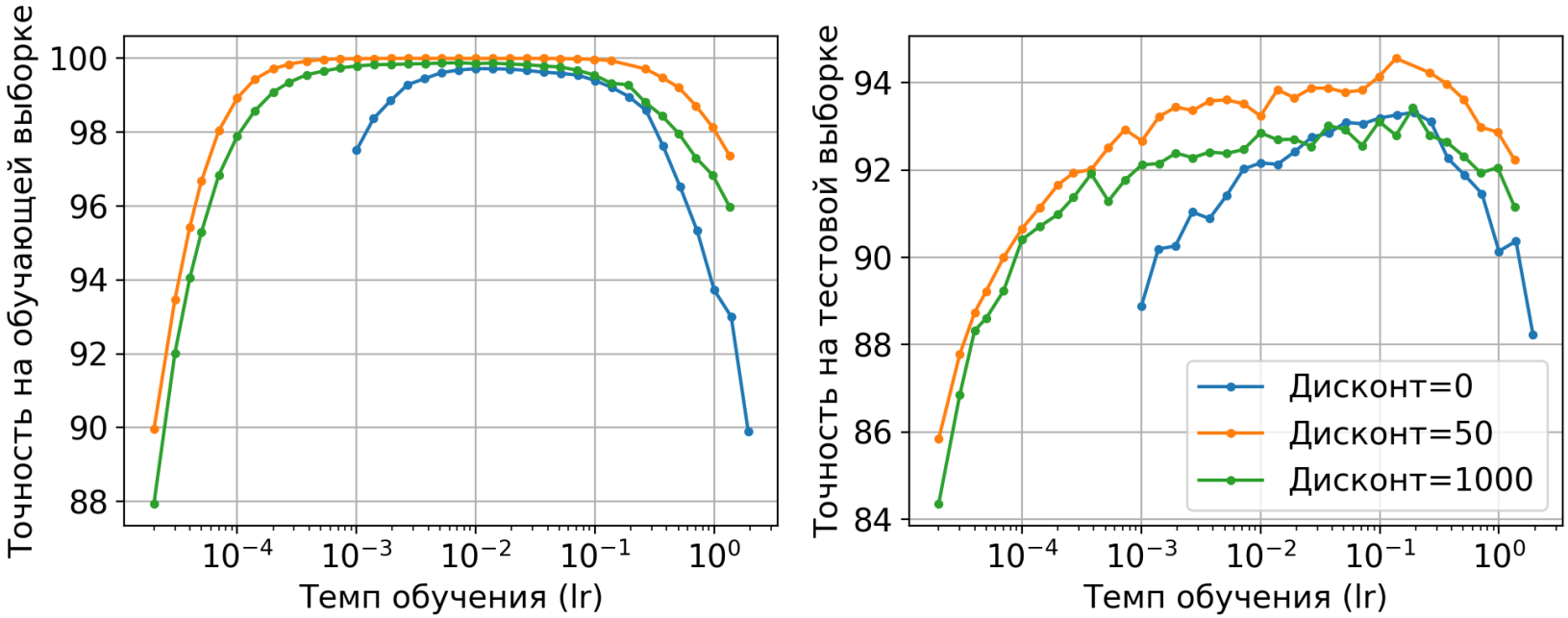


График $L(f, d) - d$ для различных значений гэпа, при движении вдоль некоторого вектора в пространстве весов. Шаг вдоль этого направления описывает коэффициент α.

Результаты бучения с дисконтированной функцией



Оптимальные параметры нейронной сети f подбираются методом стохастического градиентного спуска при минимизации:

$$L(f(\cdot,\theta),d) + \lambda\|\theta\|_2^2 \to \min_{\theta}, \tag{3}$$

где λ — коэффициент L_2 регуляризации весов ($\|\theta\|_2^2$).

Оценка ширины минимума

$$SoftMax(\mathrm{v})_i = \frac{e^{\mathrm{v}_i}}{\sum_j e^{\mathrm{v}_j}}$$

Для оценки ширины минимума предлагается использовать среднюю норму стохастического градиента, то есть:

$$\mathbb{E}_{(x,y)\sim\mathcal{X}} \left\| \nabla_{\bar{\theta}} - \log SoftMax\big(f(x,\bar{\theta})\big)_y \right\|,$$

где $\bar{\theta} = \theta / \|\theta\|$

Дисконт	Темп обучения	Ср. норма градиента
1000	0.00721	116.3
50	0.01924	11.9
0	0.01	22.9

Таблица: Оценка ширина минимума на обучающей выборке.

Ансамбль глубоких нейросетевых моделей.

Ансамблем глубоких нейросетевых моделей (базовый ансамбль) называется ансамбль состоящий из множества моделей $f(\cdot, \theta)$ полученных при минимизации (3) при различной начальной инициализации.

Предсказание ансамбля $\{f(\cdot, \theta_i)\}_{i=1}^{M}$ — усреднённое предсказание моделей.

$$y_{pred} = \frac{1}{M} \sum_{i=1}^{M} SoftMax(f(x, \theta_i))$$

- **Цель:** Улучшить метод построения базового ансамбля.
- **Идея:** Использовать бустинг: обучать следующую модель компенсировать ошибки предыдущих моделей. В качестве оценки ошибки использовать марджин.

Algorithm Алгоритм бустинга для построения ансамбля

Input: $\mathcal{X}$ — множество объектов для обучения, T — размер ансамбля, $\text{get\_next\_discount}(\cdot)$ — функция генерирующая следующее значение дисконта, λ.
Output: $\{\hat{\theta}^1, \hat{\theta}^2, \ldots, \hat{\theta}^T\}$.
$\mathbf{d} \leftarrow \text{get\_next\_discount}(\mathbf{0})$ // Начальное значение дисконта
$\mathcal{M} \leftarrow \{\}$ // Множество значений марджинов моделей ансамбля
for $t \leftarrow 1$ to T **do**
    $\hat{\theta}^t \leftarrow \arg\min_\theta L(f(\cdot, \theta), \mathbf{d}) + \lambda\|\theta\|_2^2$
    $\mathcal{M} \leftarrow \mathcal{M} \cup \{m(\hat{\theta}^t)\}$
    $\mathbf{d} \leftarrow \text{get\_next\_discount}(\mathcal{M})$
end for

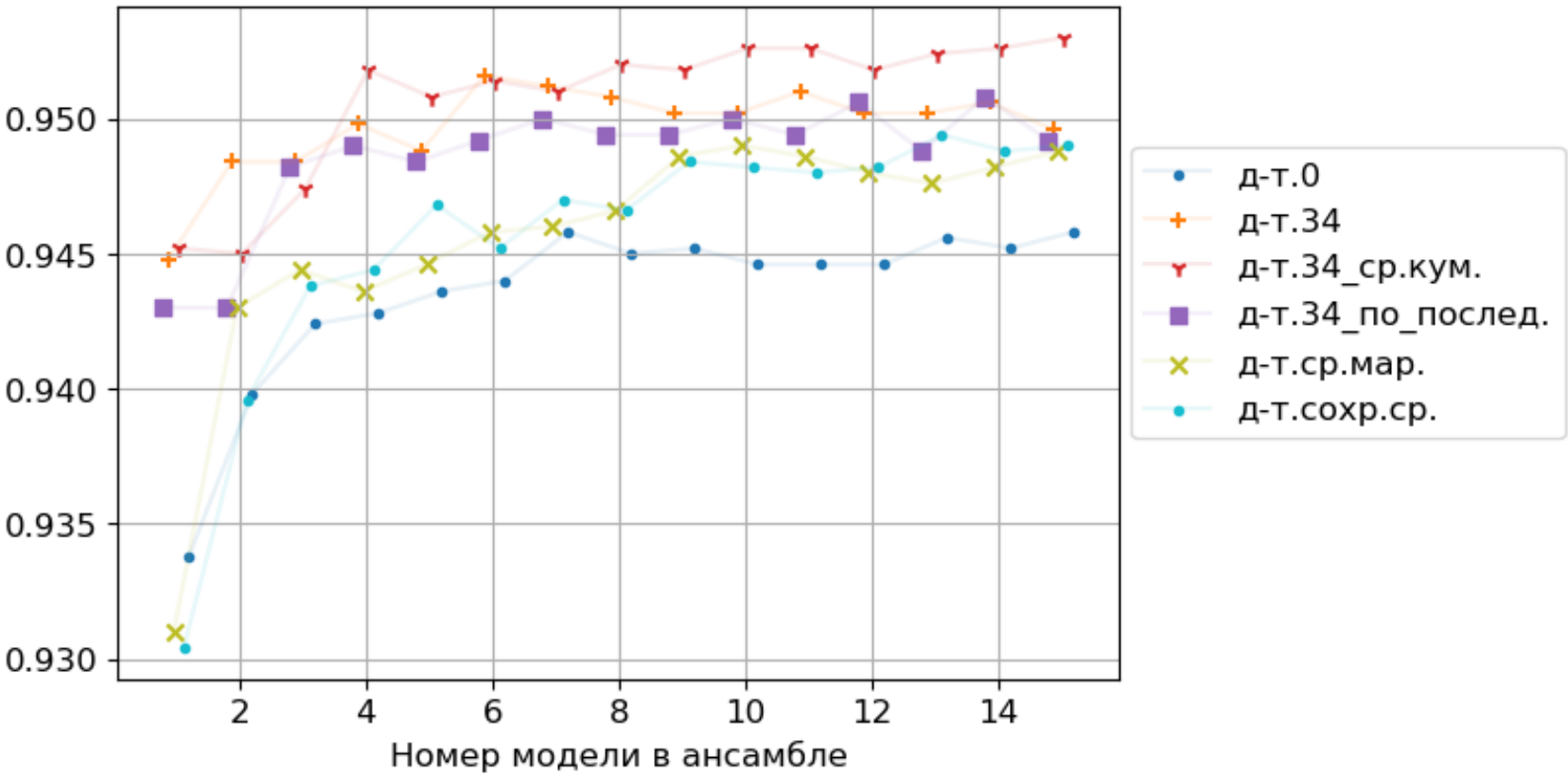
Переменная $\mathbf{d}$ является вектором, описывает значение дисконта для каждого объекта.

Описание функции $\mathrm{get\_next\_discount}(\cdot)$

Название	Формула
д-т.0	0
д-т.34	34
д-т.34_по_послед.	$34 + \frac{1}{N}\sum_{j=1}(\mathcal{M}_{T,j}) - \mathcal{M}_{T,i}$
д-т.34_ср.кум.	$34 + \frac{1}{T}\sum_t \left(\frac{1}{N}\sum_j \mathcal{M}_{t,j} - \mathcal{M}_{t,i}\right)$
д-т.ср.мар.	$\frac{1}{N}\sum_j \mathcal{M}_{T,j}$
д-т.сохр.ср.	$\frac{2}{N}\sum_{j=1}(\mathcal{M}_{T,j}) - \mathcal{M}_{T,i}$

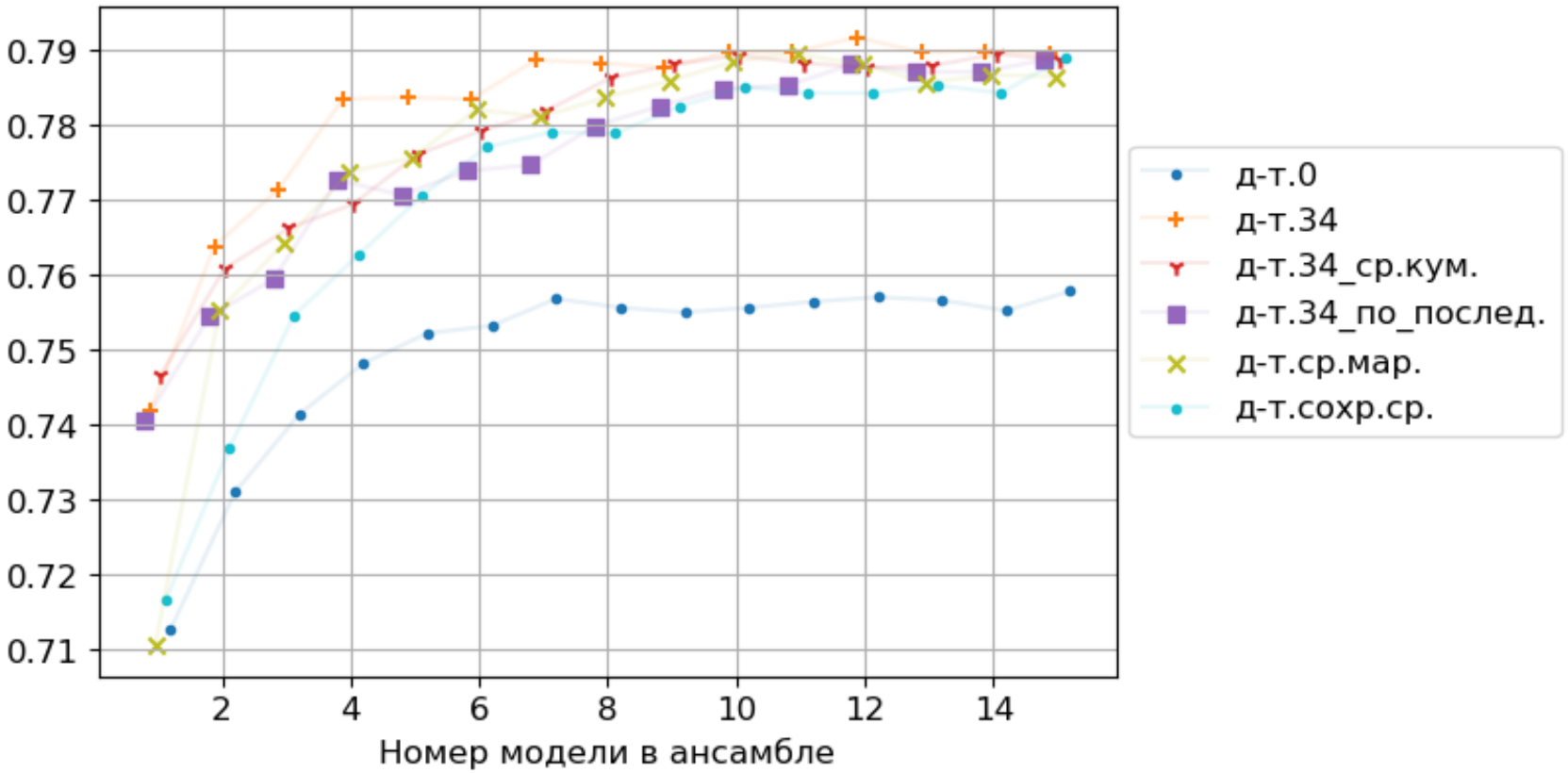
Таблица: Различные методы построения функции $\mathrm{get\_next\_discount}(\cdot)$. Для краткости, в столбце формула указывается только правая часть равенства, левая соответственно: $\mathrm{get\_next\_discount}(\mathcal{M})_i$. Значение дисконта 34 выбрано эмпирически, как значение на котором достигается максимальная точность при обучении с дисконтированной функцией потерь.

Вычислительный эксперимент



Эксперимент проводился на датасете CIFAR10, на архитектуре ResNet с замороженным последним слоем.

Вычислительный эксперимент



Эксперимент проводился на датасете CIFAR100, на архитектуре ResNet с замороженным последним слоем.

Результаты, выносимые на защиту

1. Предложена дисконтированная функция потерь, позволяющая увеличивать генерализацию модели.
2. Предоставлено теоретическое обоснование корректности функции.
3. Предложен метод построения бустинга на глубоких нейронных сетях, который показывает себя лучше чем базовый ансамбль.

Также данная работа получила призерство на 65-ой Всероссийской научной конференции МФТИ.