Регуляризация траектории оптимизации параметров модели глубокого обучения на основе дистилляции знаний

М. Горпинич, О. Ю. Бахтеев, В. В. Стрижов
gorpinich4@gmail.com; bakhteev@phystech.edu; strijov@ccas.ru

Исследуется задача оптимизации параметров модели глубокого обучения. Предлагается обобщение методов дистилляции, заключающееся в градиентной оптимизации гиперпараметров. На первом уровне оптимизируются параметры модели, на втором — гиперпараметры, задающие вид оптимизационной задачи. Исследуются свойства оптимизационной задачи и различные виды оператора оптимизации. Предложенное обобщение оптимизации позволяет производить дистилляцию модели с лучшими эксплуатационными характеристиками и за меньшее количество итераций оптимизации. Иллюстрировать применение комбинации данных подходов предлагается с помощью вычислительного эксперимента на выборке CIFAR-10.

Ключевые слова:

DOI:

1 Введение

В работе рассматривается задача оптимизации моделей глубоких нейросетей. Данная задача требует значительных вычислительных мощностей и является затратной по времени. В данной работе предлагается метод оптимизации, позволяющий улучшить эксплуатационные характеристики модели, а также ускорить ее сходимость к точке оптимума.

Предлагается обобщение метода оптимизации на основе дистилляции знаний. Рассматривается *модель-учитель* более сложной структуры, которая была обучена на выборке. Модель более простой структуры предлагается оптимизировать путем переноса знаний модели учителя на более простую модель, называемую *моделью-учеником*, при этом ее качество будет выше по сравнению с качеством, полученным после оптимизации на той же выборке. Примером применения данного подхода является [1]. В работе [2] предложен подход к дистилляции знаний, позволяющий переносить знания на модель с архитектурой, значительно отличающейся от архитектуры модели-учителя.

Предлагается представление задачи в виде двухуровневой оптимизации. На первом уровне оптимизации происходит оптимизация параметров модели, на втором уровне — ее гиперпараметров. Данный подход описан в работах [3–5]. В работе [3] рассматривается жадный градиентный метод оптимизации гиперпараметров, в работе [4] сравниваются различные градиентные методы оптимизации гиперпараметров, а также метод случайного поиска.

В работе рассматривается вид задачи оптимизации, а также различные виды оператора оптимизации. Данный подход с использованием нейросети LSTM описан в работе [6]. Вычислительный эксперимент проводится на выборке изображений CIFAR-10.

2 Постановка задачи

Решается задача классификации вида:

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m},\ \mathbf{x}_i \in \mathbb{R}^n,\ y_i \in \mathbb{Y} = \{1, \dots, K\}, \tag{1}$$

где y_i — это класс объекта, также будем обозначать $\mathbf{y}_i$ вектором вероятности для класса y_i.

Разобьем выборку следующим образом:

$$\mathfrak{D} = \mathfrak{D}_{\text{train}} \sqcup \mathfrak{D}_{\text{val}}. \tag{2}$$

Подвыборку $\mathfrak{D}_{\text{train}}$ будем использовать для оптимизации параметров модели, а подвыборку $\mathfrak{D}_{\text{val}}$ — для оптимизации гиперпараметров.

В качестве внешнего критерия качества рассматривается доля правильных ответов:

$$accuracy = \frac{1}{m}\sum_{i=1}^{m}[\mathbf{g}(\mathbf{x}_i, \mathbf{w}) = y_i], \tag{3}$$

где $\mathbf{g}$ — параметрическая модель классификации с параметрами $\mathbf{w}$.

Пусть задана модель учителя $\mathbf{f}$. Функция потерь $\mathcal{L}_{\text{train}}$, в которой учитывается перенос информации от модели учителя $\mathbf{f}$ к модели ученика $\mathbf{g}$, имеет следующий вид:

$$\mathcal{L}_{\text{train}}(\mathbf{w}, \mathbf{h}) = -\sum_{(\mathbf{x},y)\in\mathfrak{D}_{\text{train}}} \underbrace{\sum_{k=1}^{K} y^k \log \mathbf{g}(\mathbf{x}, \mathbf{w})|_{T=1}}_{\text{исходная функция потерь}} - \beta \sum_{(\mathbf{x},y)\in\mathfrak{D}_{\text{train}}} \underbrace{\sum_{k=1}^{K} \mathbf{f}(\mathbf{x})|_{T=T_0} \log \mathbf{g}(\mathbf{x}, \mathbf{w})|_{T=T_0}}_{\text{слагаемое дистилляции}}, \tag{4}$$

где T — параметр температуры. Параметр температуры T имеет следующие свойства:

1) при $T \to 0$ получаем вектор, в котором один из классов имеет единичную вероятность;
2) при $T \to \infty$ получаем равновероятные классы.

Выражение $\cdot|_{T=t}$ обозначает, что параметр температуры T в предыдущей функции равняется t.

Зададим множество гиперпараметров $\mathbf{h}$ как вектор, состоящий из температуры и коэффициента перед слагаемым дистилляции:

$$\mathbf{h} = [\beta, T].$$

Итоговая оптимизационная задача выглядит следующим образом:

$$\hat{\mathbf{h}} = \arg\max_{\mathbf{h}\in\mathbb{R}^2} \mathcal{L}_{\text{val}}(\hat{\mathbf{w}}, \mathbf{h}), \tag{5}$$

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^s} \mathcal{L}_{\text{train}}(\mathbf{w}, \mathbf{h}), \tag{6}$$

где функция $\mathcal{L}_{\text{val}}$ определяется следующим образом:

$$\mathcal{L}_{\text{val}}(\mathbf{w}, \mathbf{h}) = \sum_{(\mathbf{x},y)\in\mathfrak{D}_{\text{val}}} \sum_{k=1}^{K} y^k \log \mathbf{g}(\mathbf{x}, \mathbf{w})|_{T=1}. \tag{7}$$

Определение 1. Назовем оператором оптимизации алгоритм U выбора вектора параметров $\mathbf{w}'$ по параметрам предыдущего шага $\mathbf{w}$:

$$\mathbf{w}' = U(\mathbf{w}).$$

2.1 Градиентные методы оптимизации параметров дистилляции модели

Примером оператора оптимизации выступает оператор градиентного спуска:

$$U(\mathbf{w}, \mathbf{h}) = \mathbf{w} - \gamma \nabla \mathcal{L}_{\text{train}}(\mathbf{w}, \mathbf{h}), \tag{8}$$

где $\mathbf{h}$ — совокупность гиперпараметров модели, γ — длина шага градиентного спуска.

Оптимизируем параметры $\mathbf{w}$ при помощи η шагов градиентного спуска:

$$\hat{\mathbf{w}} = U \circ U \circ \cdots \circ U(\mathbf{w}_0, \mathbf{h}) = U^\eta(\mathbf{w}_0, \mathbf{h}), \tag{9}$$

где $\mathbf{w}_0$ — начальное значение вектора параметров $\mathbf{w}$.

Переопределим задачу минимизации согласно определению оператора U:

$$\hat{\mathbf{h}} = \arg\max_{\mathbf{h} \in \mathbb{R}^2} \mathcal{L}_{\text{val}}(U^\eta(\mathbf{w}_0, \mathbf{h})). \tag{10}$$

Решим задачу (10) используя градиентный метод. Схема оптимизации гиперпараметров:

1. Для каждого $i = \overline{0, l}$, где l — количество итераций, используемых для оптимизации гиперпараметров:
2. Решим задачу (10) и получим новое значение гиперпараметров $\mathbf{h}'$.
3. Положим $\mathbf{h} = \mathbf{h}'$.

Будем обновлять гиперпараметры $\mathbf{h}$, используя метод градиентного спуска, который зависит только от значений параметров $\mathbf{w}$ на предыдущем шаге. На каждой итерации получим следующее значение гиперпараметров:

$$\mathbf{h}' = \mathbf{h} - \gamma_{\mathbf{h}} \nabla_{\mathbf{h}} \mathcal{L}_{\text{val}}(U(\mathbf{w}, \mathbf{h}), \mathbf{h}) = \mathbf{h} - \gamma_{\mathbf{h}} \nabla_{\mathbf{h}} \mathcal{L}_{\text{val}}(\mathbf{w} - \gamma \nabla \mathcal{L}_{\text{train}}(\mathbf{w}, \mathbf{h}), \mathbf{h}). \tag{11}$$

3 Вычислительный эксперимент

Целью эксперимента является проверка работоспособности предложенного метода дистилляции моделей, а также анализ полученных моделей и их гиперпараметров.

В эксперименте используется выборка CIFAR-10, которая состоит из 60000 цветных изображений размера 32×32 пикселя, разделенных на 10 непересекающихся классов. К каждому классу относится 6000 изображений. Выборка делится на обучающую (50000 изображений) и тестовую (10000 изображений) подвыборки. В тестовой выборке содержится 1000 изображений каждого класса.

Внешним критерием качества модели является *accuracy* (3). В качестве моделей-учителей рассматриваются модели из [2], а именно, ResNet-18 и сверточная нейросеть с тремя сверточными слоями и двумя слоями полносвязной нейросети.

Проведено сравнение среднего качества обучения модели-ученика без дистилляции после 5 запусков, с дистилляцией с моделью-учителем ResNet и сверточной нейросетью после 20 запусков. Значение коэффициента β лежит в пределах от 0 до 1, значение температуры — от 0.1 до 10.

На рисунке 1 изображена зависимость точности от величины коэффициента β. Различные точки отвечают за точность модели без дистилляции, с дистилляцией ResNet и CNN. Можно заметить, что с уменьшением значения коэффициента β значение *accuracy* увеличивается.

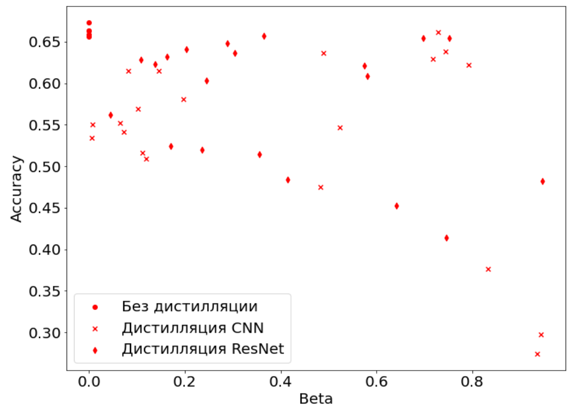

Рис. 1 График зависимости *accuracy* от β

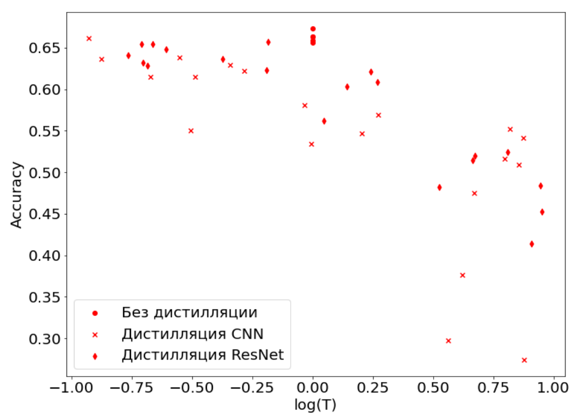

Рис. 2 График зависимости *accuracy* от температуры

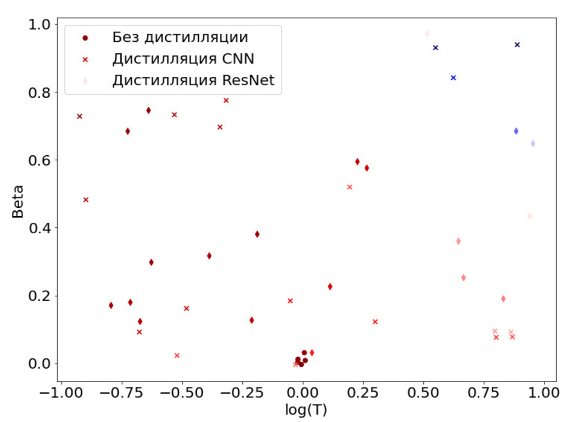

Рис. 3 График зависимости β от температуры с выделенной цветом *accuracy*

На рисунке 2 изображена зависимость точности от T. Для изображения значений температуры используется логарифмическая шкала. По графику видно, что значение температуры уменьшается при увеличении логарифма температуры, но при значениях логарифма от 0.5 до 1 наблюдается резкое уменьшение точности.

На рисунке 3 изображена зависимость β от величины T с выделенной цветом $accuracy$. Заметим, что точки с большим значением $accuracy$ в основном расположены в правом нижнем углу графика, а именно, при значениях β от 0 до 0.5 и значениях $log(T)$ от -1 до 0. Наоборот, точки с низким значением $accuracy$, расположены в правом верхнем углу графика.

Таблица 1 Результаты эксперимента

В таблице 1 приведены результаты эксперимента.

Рис. 4 График зависимости β от количества итераций дистилляции

Рис. 5 График зависимости температуры от количества итераций дистилляции

На рисунке 4 изображена зависимость β от количества итераций дистилляции.

На рисунке 5 изображена зависимость T от количества итераций дистилляции.

3.1 Название параграфа

Разделы и параграфы, за исключением списков литературы, нумеруются.

4 Заключение

Желательно, чтобы этот раздел был, причём он не должен дословно повторять аннотацию. Обычно здесь отмечают, каких результатов удалось добиться, какие проблемы остались открытыми.

Литература

[1] *Hinton Geoffrey E., Vinyals Oriol, Dean Jeffrey.* Distilling the knowledge in a neural network // CoRR, 2015. Vol. abs/1503.02531. URL: `http://arxiv.org/abs/1503.02531`.

[2] *Passalis Nikolaos, Tzelepi Maria, Tefas Anastasios.* Heterogeneous knowledge distillation using information flow modeling // CVPR. — 2020. P. 2336–2345. URL: `https://ieeexplore.ieee.org/xpl/conhome/9142308/proceeding`.

[3] *Luketina Jelena, Berglund Mathias, Greff Klaus, Raiko Tapani.* Scalable gradient-based tuning of continuous regularization hyperparameters // CoRR, 2015. Vol. abs/1511.06727. URL: `http://arxiv.org/abs/1511.06727`.

[4] *Bakhteev Oleg Yu., Strijov Vadim V.* Comprehensive analysis of gradient-based hyperparameter optimization algorithms // Ann. Oper. Res, 2020. Vol. 289. No. 1. P. 51–65.

[5] *Maclaurin Dougal, Duvenaud David, Adams Ryan P.* Gradient-based hyperparameter optimization through reversible learning // CoRR, 2015. Vol. abs/1502.03492. URL: `http://arxiv.org/abs/1502.03492`.

[6] *Andrychowicz Marcin, Denil Misha, Colmenarejo Sergio Gomez, Hoffman Matthew W., Pfau David et al.* Learning to learn by gradient descent by gradient descent // CoRR, 2016. Vol. abs/1606.04474. URL: `http://arxiv.org/abs/1606.04474`.

Received