Автоматическое определение релевантности параметров нейросети

Таранов[1] *С.К. Бахтеев*[1] *О. Ю. Стрижов*[1,2] *В. В.*
taranov.sk@phystech.edu; bakhteev@phystech.edu; strijov@phystech.edu
[1]Московский физико-технический институт
[2]Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН

В данной работе исследуется выбор оптимальной структуры нейронной сети. Модели нейронных сетей зачастую содержат большое число обучаемых параметров, предполагается, что их число можно снизить с сохранением точности прогноза. Предлагается метод, корректирующий модель в процессе обучения на основе идеи представления сети в виде графа, рёбра которого являются примитивными функциями, а вершины — промежуточными представлениями выборки, полученные под действием этих функций. Для решения задачи оптимизации предлагается проводить релаксацию структуры нейросети, так чтобы соответствующая модель удовлетворяла требованиям точности для данной задачи. Также проводятся численные эксперименты на выборках данных Boston, MNIST, CIFAR-10.

Ключевые слова: *нейронные сети, оптимизация гиперпараметров, релаксация графа.*

Введение

Решается задача построения нейронных сетей оптимальной сложности. Нейронные сети, очень распространённые в текущее время, обладают большой вычислительной сложностью и большим количеством обучаемых параметров. Это усложняет их обучение и использование, особенно на устройствах с ограниченными ресурсами. Поэтому для уменьшения сложности и размеров используемых моделей ведутся активные исследования в области оптимизации структур нейронных сетей.

В данной работе оптимальными будут считаться структуры, обладающие небольшим числом структурных и обучаемых параметров, при условии сохранения достаточного уровня точности и устойчивости модели. Под структурными параметрами будем понимать те из них, которые описывают структуры модели, например количество нейронных слоёв, количество нейроннов, содержащихся в них, а также функции активации.

Существует несколько подходов к оптимизации структуры нейронных сетей, один из них — обработка уже существующих моделей, например, прореживание уже существующих избыточных моделей с последующим дообучением [1, 2, 3]. Другой подход к проблеме оптимизации заключается в проектировании новых моделей, как результат работы алгоритмов на базе обучения с подкреплением [4], эволюционных алгоритмов [5] или как результат релаксации решения оптимизационной задачи [6]. Основываясь на [6], мы предлагаем развить описанный там алгоритм сделав его точнее благодаря использованию вариационного вывода.

Проверка алгоритма произведена на таких выборках как MNIST [7] и CIFAR [8], произведено сравнение с существующими моделями сетей, как полученными в результате других оптимизационных алгоритмов, так и выбранных экспертно на основе эвристических алгоритмов. В качестве критериев качества рассматриваются в первую очередь размер модели и полное время её проектирования и оптимизации, а также точность.

Постановка задачи

Постановка нашей задачи состоит из 2 частей - постановки задачи, для которой будет построена модель, и задача оптимизации этой модели. В качестве первой части будем рассматривать постановку задачи классификации объектов. Пусть заданы пространства описания объектов и допустимых классов, обозначим их следующим образом:

$$\mathbb{X} \subset \mathbb{R}^{\mathrm{n}}, \quad \mathbb{Y} \subset \mathbb{R}$$

Также заданы обучающая и тестовый выборки над декартовым произведением этих пространств, состоящие соответственно из описания объекта и класса к которому он принадлежит:

$$\mathfrak{D}^{\text{train}} = \{\mathbf{x}_i, y_i\}, \quad i = 1, \ldots, m^{\text{train}},$$

$$\mathfrak{D}^{\text{valid}} = \{\mathbf{x}_i, y_i\}, \quad i = 1, \ldots, m^{\text{valid}},$$

Утверждается, что существует зависимость, сопоставляющая по описанию объекта, соответствующий ему класс, обозначим её как $y^* : \mathbb{X} \to Y$ Известно что описанные выше выборки удовлетворяют этой зависимости. Требуется построить модель, которая бы реализовывала эту зависимость. Что формулируется как задача поиска обучаемых параметров модели, минимизирующих функцию потерьL_{train} на обучающей выборке:

$$\mathbf{W}^*(\mathbf{\Gamma}) = \arg\min_{\mathbf{W}} L_{\text{train}}(\mathbf{W}, \mathbf{\Gamma})$$

$$L_{train} = \log p(\mathbf{Y}^{train} | \mathbf{X}^{train}, \mathbf{W}, \mathbf{\Gamma}) + \boldsymbol{exp}^{(\mathbf{A}||\mathbf{W}||^2)}$$

где $\mathbb{W}$ - совокупность всех обучаемых параметров, $\mathbf{\Gamma}$ - множество всех структурных парметров модели, а $\mathbf{A}$ - регуляризационное слагаемое.

Вторая часть заключается в постановке задачи оптимизации модели, решающий ранее поставленную задачу. Пусть наша модель представлена в виде ориентированого, ациклического графа $\mathbb{G} = (V, E)$. E - множество всех ребёр графа, каждое из который реализует некоторую функцию, с помощью которой реализуется отображение данных из одной вершину в другую, то есть:

$$e_{\mathrm{i,j}} \in E : e_{\mathrm{i,j,k}} = G(f_k, v_i, V_j), \quad f_k \in F, v_i \in V \quad G(f, v, V) = f(v) \in V$$

где V - множество всех вершин, а каждая вершина v_i характеризуется данными записаными в ней, которые получаются путём линейноё комбинации, пришедших от входящих в вершину рёбер с соответствующими коэффициентами, которые и являются структурными параметрами модели:

$$v_i \in V : v_i = \sum_{j,k} e_{\mathrm{i,j,k}} \cdot \gamma_{\mathrm{i,j,k}}$$

$$\sum_{j,k} \gamma_{i,j,k} = 1, \gamma_{i,j,k} \in \mathbf{\Gamma} \tag{1}$$

$\mathbb{F}$ - множество функций, используемых в построении модели, это могут как функции активации так и функции реализующие полносвязные нейронные слои, элементы которого $\gamma_{i,j,k} in [0, 1]$. Кроме всего прочего мы накладываем ограничение (1), чтобы в дальнейшем

провести релаксацию приблизив линейную комбинацию одним из её членов. Таким образом мы можем сформулировать задачу потимизации построеной модели, как минимизацию следующей функции по пространству структурных гиперпараметров, которое представляет собой декартово произведение симплексов, содержащих структурные параметры для каждой из вершин.

$$Q = \log \mathrm{p}(\mathbf{Y}^{valid}|\mathbf{X}^{valid}, \mathbf{W}, \mathbf{\Gamma})$$

Таким образом итоговую задачу можно сформулировать как задачу двухуровневой оптимизации:

$$\mathbf{W}^*(\mathbf{\Gamma}) = \arg\min_{\mathbf{W}} L(\mathbf{W}, \mathbf{\Gamma})$$

$$\mathbf{\Gamma}, \mathbf{A} = \min_{\mathbf{\Gamma}} Q(\mathbf{W}^*(\mathbf{\Gamma}), \mathbf{\Gamma})$$

Литература

[1] Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural networks, October 30 2015. Comment: Published as a conference paper at NIPS 2015.

[2] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *CoRR*, abs/1608.08710, 2016.

[3] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Training pruned neural networks. *CoRR*, abs/1803.03635, 2018.

[4] Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. *CoRR*, abs/1611.01578, 2016.

[5] Esteban Real, Sherry Moore, Andrew Selle, Saurabh Saxena, Yutaka Leon Suematsu, Quoc V. Le, and Alex Kurakin. Large-scale evolution of image classifiers. *CoRR*, abs/1703.01041, 2017.

[6] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: differentiable architecture search. *CoRR*, abs/1806.09055, 2018.

[7] L. Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Process. Mag*, 29(6):141–142, 2012.

[8] Alex Krizhevsky. *Learning Multiple Layers of Features from Tiny Images*, volume 8. 2009.