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, например, нейронная сеть, построенная по восьмислойной архитектуре AlexNet, имеет около 60 миллионов паратером и требует около 729 MFLOPS[1] [1]. Поэтому для уменьшения сложности и размеров используемых моделей ведутся активные исследования в области оптимизации структур нейронных сетей. [1] 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 [2, 3, 4]. Другой подход к проблеме оптимизации заключается в проектировании новых моделей, являющихся результатом работы алгоритмов на базе обучения с подкреплением [5], эволюционных алгоритмов [6] или как результат релаксации решения оптимизационной задачи [7]. Основываясь на [7], предлагается развить описанный там алгоритм сделав его точнее благодаря использованию вариационного вывода.

Проверка алгоритма произведена на таких выборках как MNIST [8] и CIFAR [9], произведено сравнение с существующими моделями сетей, как полученными в результате других оптимизационных алгоритмов, так и выбранных экспертно на основе эвристических

[1]FLOP - единаца измерения работы вычислительной машины, равная одной операцией над числом с плавающей точкой

алгоритмов. В качестве критериев качества рассматриваются в первую очередь размер модели и полное время её проектирования и оптимизации, а также точность.

Постановка задачи

Пусть заданы пространство описания объектов и множество допустимых классов, обозначим их следующим образом:

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

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

$$\mathfrak{D}^{\mathrm{train}} = \{\mathbf{x}_i, y_i\}, \quad i = 1, \ldots, m^{\mathrm{train}}, \quad x \in \mathbb{X}$$

$$\mathfrak{D}^{\mathrm{valid}} = \{\mathbf{x}_i, y_i\}, \quad i = 1, \ldots, m^{\mathrm{valid}}, \quad y \in \mathbb{Y}$$

Требуется построить модель, которая бы реализовывала эту зависимость. Сформулируем эту задачу как поиск обучаемых параметров модели, минимизирующих функцию потерь L_{train} на обучающей выборке и зададим эту функцию потерь следвующим образом:

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

$$L = \log \mathrm{p}(\mathbf{Y}^{train}|\mathbf{X}^{train}, \mathbf{W}, \mathbf{\Gamma}) + \boldsymbol{e}^{(\mathbf{A}||\mathbf{W}||^2)} \tag{1}$$

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

Пусть модель представлена в виде ориентированого, ацикличного графа $\mathbb{G} = (V, E)$, где E - множество всех ребёр графа, а V - множество всех вершин. Как было сказано ранее в вершинах записаны данные, а каждое ребро реализует некоторую функцию, с помощью которой может осуществляется отображение данных из одной вершину в другую, то есть:

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

где F - множество функций, используемых в построении модели, это могут как функции активации так и функции реализующие полносвязные нейронные слои. Данные в каждой вершине являются линейноё комбинации данных, пришедших от входящих в вершину рёбер.

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

Коэффициенты этой линейной комбинации $\gamma_{\mathrm{i,j,k}}$ являются структурными параметрами модели:

$$\sum_{j,k} \gamma_{i,j,k} = 1, \gamma_{i,j,k} \in \mathbf{\Gamma} \quad \gamma_{j,k} \in [0, 1] \tag{2}$$

Также мы накладываем на структурные параметры ограничение (2), чтобы в дальнейшем провести релаксацию, приблизив каждую линейную комбинацию одним из её членов. Далее формулируем задачу оптимизации построеной модели, как минимизацию другой функции потерь по пространству структурных параметров. Зададим эту функцию потерь следующем образом

$$Q = \log \mathrm{p}(\mathbf{Y}^{valid}|\mathbf{X}^{valid}, \mathbf{W}, \mathbf{\Gamma}) \tag{3}$$

Заметим, что пространство структурных параметров $\mathbf{\Gamma}$ представляет собой декартово произведение пространств структурных параметров для отдельных вершин, каждое из которых в силу ограничения (1) является симплексом. Таким образом итоговую задачу можно сформулировать как задачу двухуровневой оптимизации:

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

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

Релаксация модели

Чтобы эффективно решать задачу поиска потимальной структуры, проводится релаксация модели, с помощью которой дискретной задача поиска архитетектуры оптимизации переводится в непрерывную. Релаксация имеет следующий вид:

$$\overline{g}^{(i,j)}(x) = \sum_{k:\gamma_{i,j,k} \in \mathbf{\Gamma}} \frac{exp(\gamma_{i,j,k})}{\sum\limits_{k:\gamma_{i,j,k} \in \mathbf{\Gamma}} exp(\gamma_{(i,j,k)})} g_{i,j,k}(x)$$

где $\overline{g}^{(i,j)}(x)$ - новая базовая функция отаражающая, которую реализует ребро из (v_i, v_j), после завершения поиска оптимальной структуры, дискретные базовые функции будут выбраны как функции имеющие наибольший вес в неприрывной модели $g^*_{i,j} = g_{i,j,k}$, где $k = \arg\max_k \gamma_{i,j,k}$

Регуляризация модели

Для регуляризация структуры модели к фунции потерь (3) прибавляется специальное слагаемого $\lambda P(\Gamma)$, где $P(\Gamma)$ есть произведение всех вероятностей возникновения веса $\gamma_{i,j,k}$. Таким образом функция потерь (3) принимает вид:

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

В качестве этих вероятностей, можно использовать вероятности получаемые из распределений Дирихле и Gumble-Softmax

Оптимизация обучаемых и структурных парамтеров

Как видно из функций потерь сформулированных выше, задача оптимизации как обучаемых, так и структурных параметров зависит и как от текущей архитектуры нейронной сети, так и от её текущих весов. Для решения этой двухуровневой оптимизации используется итеративная процедура описанная в [7], суть которой заключается в попеременном обновлении обучаемых параметров и структурных. Изменение, которое возникает в нашей задачи - это добавление к структурным параметрам веса возникающие из-за структурных соображений, связанных с уменьшением суммарного описания модели, то есть оптимизации её вида.

Литература

[1] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng 0001. Quantized convolutional neural networks for mobile devices. *CoRR*, abs/1512.06473, 2015.

[2] 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.

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

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

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

[6] 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.

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

[8] 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.

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