

Автоматическое построение нейронной сети оптимальной сложности

Товкес А. А.¹, Бахтеев О. Ю.¹, Стрижов В. В.²

tovkes.aa@phystech.edu; bakhteev@phystech.edu; strijov@phystech.edu

Московский физико-технический институт¹;

Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН^{1,2}

Работа посвящена задаче выбора оптимальной по сложности модели нейросети. Нейросеть представляется в виде вычислительного графа, где ребрам соответствуют базовые функции, а вершинам — промежуточные представления выборки под действием этих функций. Параметры сети разделяются на непосредственно параметры модели, которые определяют итоговое качество классификации; гиперпараметры, определяющие процесс обучения и предотвращение переобучения; структурные параметры, определяющие непосредственно структуру модели. Для решения задачи оптимизации предлагается проводить релаксацию структуры нейросети. Рассмотрено изменение характеристик нейросети при возмущении структурных параметров. Для анализа качества представленного алгоритма проводятся эксперименты на выборках Boston, MNIST и CIFAR-10.

Ключевые слова: *нейронные сети, оптимизация гиперпараметров, оптимальная структура нейронной сети.*

1 Введение

В данной работе рассматривается задача построения оптимальной нейронной сети. В силу большого количества оптимизируемых параметров модели, задача выбора модели глубокого обучения является вычислительно сложной. Поэтому задача выбора структуры модели глубокого обучения включает в себя выбор стратегии параметризации структуры и построения модели, эффективной по вычислительным ресурсам.

Под оптимальной моделью понимается та модель, которая дает приемлемое качество при небольшом числе параметров модели. Нейросеть представляется в виде вычислительного графа, где ребрам соответствуют базовые функции, а вершинам — промежуточные представления выборки под действием этих функций. Сами же структурные параметры — это вектор весов каждой базовой функции, определяющий их вклад в итоговую модель.

Существует ряд подходов к построению нейронной сети. В работе [1] предлагается использовать метод прореживания модели. Он заключается в построении переусложненной модели, с последующим удалением параметров, не влияющих на качество. В [2] используется предсказание структуры модели другой нейросетью. Кроме того в [3] используется метаобучение, которое по некоторой входной выборке, возвращает оптимальные гиперпараметры.

В данной работе исследуется метод построения модели глубокого обучения, позволяющий производить оптимизацию параметров, гиперпараметров и структурных параметров в единой процедуре. В основе разработанного метода лежит алгоритм DARTS, предложенный в работе [4]. Для выбора оптимального набора гиперпараметров предлагается параметризовать структуру модели некоторым действительным вектором, таким образом переходя от дискретного множества значений к непрерывному.

Проверка и анализ метода проводится на выборке Boston Housing [5], MNIST [6], CIFAR-10 [7] и синтетических данных. Результат сравнивается с моделью полученной при помощи

базовых алгоритмов. Критериями качества рассматриваемых алгоритмов выступают качество и полученных моделей и их устойчивость к возмущениям параметров, а также вычислительная сложность методов.

2 Постановка задачи

Пусть заданы обучающая и валидационная выборки:

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

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

состоящие из множеств пар объект-метка,

$$\mathbf{x}_i \in \mathbf{X} \subset \mathbb{R}^n, \quad y_i \in \mathbf{Y} \subset \mathbb{R}.$$

$\mathbf{Y} = \{1, \dots, Z\}$, где Z - количество классов.

Модель задаётся ориентированным графом $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, где для каждого ребра (i, j) задан вектор базовых функций $\mathbf{g}^{i,j}$, с мощностью $|\mathbf{g}^{i,j}| = K^{i,j}$ и весами $\gamma^{i,j}$. Требуется построить такую модель \mathbf{f} с параметрами $\mathbf{W} \in \mathbb{R}^n$:

$$\mathbf{f}(\mathbf{x}, \mathbf{W}) = \{\mathbf{f}_i(\mathbf{x}, \mathbf{w}_i)\}_{i=1}^{|\mathbf{V}|}$$

где $\mathbf{f}_i(\mathbf{x}, \mathbf{w}_i)$ - подмодель с параметрами \mathbf{w}_i задаётся через графовое представление как:

$$\mathbf{f}_i(\mathbf{x}, \mathbf{w}_i) = \sum_{k \in \text{adj}(i)} \langle \gamma^{i,k}, \mathbf{g}^{i,k} \rangle \mathbf{f}_k(\mathbf{x}, \mathbf{w}_k)$$

Тогда параметры модели — конкатенация всех параметров каждой подмодели: $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_{|\mathbf{V}|}]$, а структура модели $\mathbf{\Gamma}$ задаётся вектором $\{\gamma^{i,j}\}_{\mathbf{E}}$.

Функция потерь на обучении L и функция потерь на валидации Q задаются как:

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

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

В итоге получаем 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] Yann Le Cun, John S. Denker and Sara A. Solla. Optimal Brain Damage. 1989.
- [2] Han Cai, Tianyao Chen, Weinan Zhang, Yong Yu, Jun Wang. Efficient Architecture Search by Network Transformation.. 2017. URL: <https://arxiv.org/abs/1707.04873>.

- [3] *D. Maclaurin and D. Duvenaud and R. Adams.* Gradient-based Hyperparameter Optimization Through Reversible Learning 2015.
- [4] *Hanxiao L., Simonyan K., Yang .Y.* DARTS: Differentiable Architecture Search. 2018. URL: <https://arxiv.org/abs/1806.09055>.
- [5] *Harrison Jr. , Rubinfeld D., Daniel L.* Hedonic housing prices and the demand for clean air. 1978. URL: <https://archive.ics.uci.edu/ml/machine-learning-databases/housing/>.
- [6] *Yann LeCun, Corinna Cortes, Christopher J.C. Burges.* The MNIST Database of Handwritten Digits 1998. URL: <http://yann.lecun.com/exdb/mnist/>
- [7] *A. Krizhevsky, V. Nair, G. Hilton.* The CIFAR-10 dataset 2009. URL: <http://www.cs.toronto.edu/~kriz/cifar.html>