УДК 621.391

МЕТОД ОТЖИГА В ЗАДАЧЕ ОБУЧЕНИЯ НЕЙРОННЫХ СЕТЕЙ

аспирант В. В. МАЦКЕВИЧ
(Белорусский государственный университет, Минск)

Аннотация. *В работе рассматривается актуальная проблема, связанная с обучением нейронных сетей. Предложен алгоритм обучения на основе метода отжига и показано (на примере решения задачи сжатия изображений), что он является более эффективным (по качеству) в сравнении с существующими градиентными методами.*

Ключевые слова: *нейронная сеть, метод отжига, метод градиентного спуска, обучение.*

Введение. В настоящее время происходит стремительное развитие цифровых технологий. В результате возникает множество прикладных задач, связанных с обработкой больших объемов различного рода информации. Для решения данной проблемы очень часто используются нейросетевые технологии. Широкое распространение нейронные сети получили благодаря своей универсальности.

В отличие от классических алгоритмов обработки данных, нейронная сеть требует настройки ее параметров под конкретно решаемую задачу. Процесс адаптации нейронной сети к задаче производится на основе ее обучения. Эффективность полученного решения напрямую зависит от успешности процесса адаптации сети.

Несмотря на достигнутые в данном направлении результаты, проблема обучения по-прежнему является актуальной [1, 2]. Для обучения нейронных сетей наибольшее распространение получили различные модификации метода градиентного спуска. Высокая скорость сходимости метода позволяет обучать нейронные сети очень сложной архитектуры [3]. Однако качество полученного решения может оказаться недостаточно высоким.

В работе представлен алгоритм обучения нейронных сетей на основе метода отжига. Эффективность данного подхода показана на примере решения задачи сжатия цветных изображений.

Анализ проблемы. Процесс обучения связан с вычислением параметров нейронной сети, обеспечивающих получение наилучшего решения по заранее заданному функционалу качества. Таким образом, обучение нейронной сети является оптимизационной задачей.

Традиционно для реализации процесса обучения нейронных сетей используют метод градиентного спуска. Однако, последний обладает серьезным недостатком – на процесс накладываются сильные ограничения по сходимости. Во-первых,

некоторые модификации градиентного метода имеют строгие ограничения на множество начальных приближений, например, метод Ньютона. Во-вторых, любой градиентный метод сходится в точках, где производная от целевой функции равна нулю. Это значит, что полученное решение может оказаться точкой перегиба или локального минимума. Данный факт приводит к тому, что полученное решение может оказаться гораздо хуже оптимального.

Метод отжига, напротив, лишен вышеуказанных недостатков. Он всегда сходится к оптимальному решению, причем из любого начального приближения [4]. Единственным недостатком данного метода является его логарифмическая скорость сходимости. Это означает, что для обучения требуется большое количество вычислительных ресурсов. Поэтому на начальном этапе развития нейронных сетей из-за малых мощностей компьютеров метод не получил практического распространения.

Алгоритм обучения на основе метода отжига. Любая нейронная сеть задается множествами соответствующих параметров. Фиксация количества этих множеств и их мощностей, грубо говоря, и определяют архитектуру нейронной сети. Например, ограниченная машина Больцмана типа Бернулли-Бернулли характеризуется тремя типами параметров: множеством весов *W*, множествами смещений видимого и скрытого слоев – *B*, *FB* соответственно. Выбор по одному элементу из каждого множества параметров для достижения наилучшего решения и есть задача обучения нейронной сети.

Для реализации процесса обучения предлагается одна из возможных алгоритмических реализаций метода отжига. Опишем состав такого алгоритма.

Алгоритм реализуется в два основных этапа.

Предварительный этап. Производится инициализация (задание) начальных значений параметров сети и температуры T_0.

Основной этап. На данном этапе реализуется процедура последовательного обновления значений параметров с использованием заданного функционала качества.

Опишем данную процедуру более подробно. Для простоты изложения рассмотрим ее на примере множества параметров *W*. Для других множеств эта процедура полностью совпадает.

Множеству параметров *W* поставим в соответствие отрезок L_w длины *l*. После этого каждый элемент множества *W* последовательно помещаем в центр заданного отрезка. Для определения направления изменения значений параметров генерируем случайное число от 0 до 1. Если полученная реализация числа больше 0,5, то значение параметра увеличивается, в противном случае – уменьшается.

Новое значение параметра определяется в результате реализации равномерно распределенной случайной величины на отрезке, концами которого являются текущее значение параметра, и конец отрезка, в сторону которого осуществляется изменение.

Аналогично действия последовательно выполняются для других параметров сети.

Для вновь полученных значений параметров вычисляется функционал качества.

Далее принимается решение о переходе в новое состояние.

Если значение функционала больше текущего, то переходим в новое состояние с вероятностью:

$$P(y \mid x) = \exp\left(-\frac{F(y) - F(x)}{T_i}\right), \quad (1)$$

где x – текущее состояние;

y – выбранное для перехода состояние;

F – минимизируемая целевая функция;

T_i – температура i-й итерации.

В противном случае происходит безусловный переход в новое состояние.

Охлаждение производится по правилу:

$$T_{k+1} = \frac{T_0}{\ln(k+1)}, \quad (2)$$

где k – количество совершенных итераций.

После охлаждения производится проверка полученного решения на оптимальность. Решение является оптимальным, если в течение последних S итераций не проводился переход в новое состояние, либо время, отведенное на обучение, истекло.

Если полученное решение оптимально, то алгоритм завершает свою работу, в противном случае производится переход на следующую итерацию.

Главным достоинством предлагаемого алгоритма является гарантированная сходимость результата к точке глобального минимума, т. е. к оптимальному решению. К недостаткам можно отнести традиционную логарифмическую скорость сходимости метода.

Эксперименты. Основной целью проводимых экспериментов является сравнение эффективности описанного алгоритма, реализующего метод отжига, с методом градиентного спуска в задачах обучения нейронных сетей. В качестве градиентного метода используется наилучшая его модификация – метод адаптивного градиента [5].

Эксперименты были разделены на две основные части.

В первой части осуществлялось 16 кратное сжатие изображений выборки CIFAR-10 [6]. Т.к. степень сжатия не высока и объекты на изображениях относятся к одному из 10 классов, то задача сжатия данной выборки не является очень

сложной. Для экспериментов была спроектирована нейронная сеть из каскада ограниченных машин Больцмана типа Гаусс-Бернулли. Входной слой состоял из 48 нейронов, что соответствует фрагменту изображения 4 на 4 пикселя. Выходной слой – из 24 нейронов, что задает требуемую степень сжатия.

Во второй части экспериментов осуществлялось 32 кратное сжатие изображений выборки STL-10 [7]. Т. к. степень сжатия в данном случае заметно выше, чем в первой части, и изображения могут содержать произвольные объекты, то сжатие таких изображений является достаточно сложной задачей. В данной части экспериментов использовалась архитектура нейронной сети аналогичная предыдущей. Единственное отличие – для повышения степени сжатия количество нейронов в выходном слое было снижено до 12 нейронов.

Для вычисления градиента для ограниченных машин Больцмана был выбран наиболее распространенный метод сравнительной разности (CD) [8]. Стоит отметить, что данный метод в качестве параметра использует число семплирований. Для достижения максимальной скорости обучения число семплирований было установлено равной единице.

Эксперименты проводились на операционной системе Lubuntu 20.04 с процессором intel i7-4770k, видеокартой nvidia gtx 1070 ti.

Как показывают результаты (табл. 1, 2) метод отжига превзошел градиент по всем параметрам. Для эксперимента были выбраны параметры, обеспечивающие наиболее быструю сходимость.

Таблица 1. Результаты обучения на выборке CIFAR-10

Алгоритм обучения	Метод отжига				Метод градиентного спуска			
Время обучения, ч	0,125	0,0625	0,03125	0,01563	0,0381	0,0382	0,0315	0,0166
MSE	585	653	791	1160	2194	2195	2195	4963
PSNR	20,6	20,1	19,3	17,7	14,8	14,8	14,8	11,5
PSNR-HVS	20,8	20,3	19,5	17,9	14,9	14,9	14,9	11,8
SSIM	0,7	0,675	0,628	0,524	0,312	0,311	0,311	0,179

Таблица 2. Результаты обучения на выборке STL-10

Алгоритм обучения	Метод отжига				Метод градиентного спуска			
Время обучения, ч	1	0,5	0,25	0,125	0,439	0,433	0,253	0,132
MSE	733	796	934	1517	1727	1727	2956	7107
PSNR	19,6	19,2	18,6	16,5	15,9	15,9	13,6	9,96
PSNR-HVS	19,8	19,4	18,8	16,7	16	16	13,7	10,1
SSIM	0,51	0,483	0,439	0,342	0,308	0,308	0,25	0,148

Из приведенных результатов видно, что при одинаковом времени обучения алгоритм отжига по качеству оказался лучше метода градиентного спуска (на выборке CIFAR-10 в среднем 3,5 раза и на более сложной выборке STL-10 в 3,9 раза).

По мере увеличения времени обучения обобщающая способность нейронной сети градиентным методом снижалась на валидационной выборке, что привело к останову алгоритма. В то же время алгоритм, реализующий метод отжига, главным недостатком которого является медленная сходимость, продолжил повышать качество обучения.

Заключение. Полученные в экспериментах результаты показывают возможную перспективу использования метода отжига в задачах обучения нейронных сетей. Это, как было показано выше, может привести к росту качества обучения. Потенциал данного подхода раскрывается по мере роста вычислительных мощностей компьютеров. Предложенный в работе алгоритм обучения обладает всеми преимуществами метода отжига и может быть использован для обучения глубоких нейронных сетей.

ЛИТЕРАТУРА

1. Krasnoproshin, V. V. Annealing method in training restricted Boltzmann machine / V. V. Krasnoproshin, V. V. Matskevich // Proceedings of the 14-th International Conference. – PRIP'2019, Minsk, 2019. – P. 264–268.
2. Krasnoproshin, V. V. Statistical approach to image compression based on a restricted Boltzmann machine / V. V. Krasnoproshin, V. V. Matskevich // Proceedings of the 12-th International Conference "Computer Data Analysis and Modeling" – CDAM'2019, Minsk, 2019. – P. 207–213.
3. Hamis, S. Image Compression at Very Low Bitrate Based on Deep Learned Super-Resolution / S. Hamis, T. Zaharia, O. Rousseau // IEEE 23rd International Symposium on Consumer Technologies (ISCT). – P. 128–133.
4. Hajek, B. Cooling schedules for optimal annealing / B. Hajek // Mathematics of operations research. – Vol. 13, iss. 2. – 1988.
5. Kingma, D. P. Adam: A Method for Stochastic Optimization / D. P. Kingma, J. L. Ba, // Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015). – P. 1–15.
6. Выборка CIFAR-10 [Электронный ресурс]: – Режим доступа: https://www.cs.toronto.edu/~kriz/cifar.html. – Дата доступа: 04.03.2020.
7. Выборка STL-10 [Электронный ресурс]: – Режим доступа: https://web.archive.org/web/20110803194852/http://www.stanford.edu/~acoates//stl10/. – Дата доступа: 24.04.2019.
8. Li, X. A Novel Restricted Boltzmann Machine Training Algorithm With Dynamic Tempering Chains / X. Li, X. Gao, Ch. Wang // IEEE ACCESS. – Vol. 9. – 2021. – P. 21939–21950.