Глава 9.

Логически прозрачные нейронные сети и производство явных знаний из данных

Е.М.Миркес

Вычислительный центр СО РАН в г. Красноярске[1]

Производство явных знаний из накопленных данных - проблема, которая намного старше чем компьютеры. Обучаемые нейронные сети могут производить из данных скрытые знания: создается навык предсказания, классификации, распознавания образов и т.п., но его логическая структура обычно остается скрытой от пользователя. Проблема проявления (контрастирования) этой скрытой логической структуры решается в работе путем приведения нейронных сетей к специальному "логически прозрачному" разреженному виду.

Исследуются два вопроса, встающие перед каждым исследователем, решившим использовать нейронные сети: "Сколько нейронов необходимо для решения задачи?" и "Какова должна быть структура нейронной сети?" Объединяя эти два вопроса, мы получаем третий: "Как сделать работу нейронной сети понятной для пользователя (логически прозрачной) и какие выгоды может принести такое понимание?" Описаны способы получения логически прозрачных нейронных сетей. Приведен пример из области социально-политических предсказаний. Для определенности рассматриваются только нейронные сети, обучение которых есть минимизация оценок (ошибок) с использованием градиента. Градиент оценки вычисляется методом двойственности (его частный случай - метод обратного распространения ошибки).

1. Сколько нейронов нужно использовать?

При ответе на этот вопрос существует две противоположные точки зрения. Одна из них утверждает, что чем больше нейронов использовать, тем более

[1] 660036, Красноярск-36, ВЦК СО РАН, E-mail: amse@cckr.krasnoyarsk.su

надежная сеть получится. Сторонники этой позиции ссылаются на пример человеческого мозга. Действительно, чем больше нейронов, тем больше число связей между ними, и тем более сложные задачи способна решить нейронная сеть. Кроме того, если использовать заведомо большее число нейронов, чем необходимо для решения задачи, то нейронная сеть точно обучится. Если же начинать с небольшого числа нейронов, то сеть может оказаться неспособной обучиться решению задачи, и весь процесс придется повторять сначала с большим числом нейронов. Эта точка зрения (чем больше - тем лучше) популярна среди разработчиков нейросетевого программного обеспечения. Так, многие из них как одно из основных достоинств своих программ называют возможность использования любого числа нейронов.

Вторая точка зрения опирается на такое “эмпирическое” правило: чем больше подгоночных параметров, тем хуже аппроксимация функции в тех областях, где ее значения были заранее неизвестны. С математической точки зрения задачи обучения нейронных сетей сводятся к продолжению функции заданной в конечном числе точек на всю область определения. При таком подходе входные данные сети считаются аргументами функции, а ответ сети - значением функции. На рис. 1 приведен пример аппроксимации табличной функции полиномами 3-й (рис. 1.а) и 8-й (рис. 1.б) степеней. Очевидно, что аппроксимация, полученная с помощью полинома 3-ей степени больше соответствует внутреннему представлению о “правильной” аппроксимации. Несмотря на свою простоту, этот пример достаточно наглядно демонстрирует суть

X	1	2	3	4
F(X)	5	4	6	3

Рис. 1. Аппроксимация табличной функции

проблемы.

Второй подход определяет нужное число нейронов как минимально необходимое. Основным недостатком является то, что это, минимально необходимое число, заранее неизвестно, а процедура его определения путем постепенного наращивания числа нейронов весьма трудоемка. Опираясь на опыт работы группы НейроКомп в области медицинской диагностики [4,5,9], космической навигации и психологии [10] можно отметить, что во всех этих задачах ни разу не потребовалось более нескольких десятков нейронов.

Подводя итог анализу двух крайних позиций, можно сказать следующее: сеть с минимальным числом нейронов должна лучше (“правильнее”, более гладко) аппроксимировать функцию, но выяснение этого минимального числа нейронов требует больших интеллектуальных затрат и экспериментов по обучению сетей. Если число нейронов избыточно, то можно получить результат с первой попытки, но существует риск построить “плохую” аппроксимацию. Истина, как всегда бывает в таких случаях, лежит посередине: нужно выбирать число нейронов большим чем необходимо, но не намного. Это можно осуществить путем удвоения числа нейронов в сети после каждой неудачной попытки обучения. Однако существует более надежный способ оценки минимального числа нейронов - использование процедуры контрастирования [1]. Кроме того, процедура контрастирования позволяет ответить и на второй вопрос: какова должна быть структура сети.

2. Процедура контрастирования

Процедура контрастирования основана на оценке значимости весов связей в сети. Впервые процедура контрастирования нейронных сетей на основе показателей чувствительности описана одновременно в [1] и (существенно более частный вариант) в [2]. В книге [1] указаны основные цели контрастирования: упростить техническую реализацию сети и сделать навык сети более понятным - явизовать (сделать явным) знание, полученное сетью в ходе обучения.

Результаты экспериментов по контрастированию нейронных сетей опубликованы в [7,8]. Существуют также подходы, не использующие показатели чувствительности [3]. Уже в [1] описано несколько способов вычисления показателей чувствительности. Приведем два наиболее широко используемых.

2.1. Контрастирование на основе оценки

Рассмотрим сеть, правильно решающую все примеры обучающего множества. Обозначим через w_p, $p = 1,\text{K}\ ,n$ веса всех связей. При обратном функционировании сети по принципу двойственности или методу обратного распространения ошибки сеть вычисляет вектор градиента функции оценки H по весам связей - $\text{Grad}(H) = \left\{\partial H / \partial w_p\right\}_{p=1,\text{K}\ ,n}$. Пусть w^0 - текущий набор весов связей, а оценка текущего примера равна H^0. Тогда в линейном приближении можно записать функцию оценки в точке w как $H(w) = H^0 + \sum_{p=1}^{n} \frac{\partial H}{\partial w_p}(w_p - w_p^0)$. Используя это приближение можно оценить изменение оценки при замене w_p^0 на как $\chi(p,q) = \left|\frac{\partial H}{\partial w_p}\right| \cdot \left|w_p^* - w_p^0\right|$, где q - номер примера обучающего множества, для которого были вычислены оценка и градиент. Величину $\chi(p,q)$ будем называть показателем чувствительности к замене w_p на w_p^* для примера q. Далее необходимо вычислить показатель чувствительности, не зависящий от номера примера. Для этого можно воспользоваться любой нормой. Обычно используется равномерная норма (максимум модуля): $\chi(p) = \max_q \chi(p,q)$. Умея вычислять показатели чувствительности, можно приступать к процедуре контрастирования. Приведем простейший вариант этой процедуры:

1. Вычисляем показатели чувствительности.
2. Находим минимальный среди показателей чувствительности - χ_{p^*}.

3. Заменим соответствующий этому показателю чувствительности вес $w^0_{p^*}$ на $w^*_{p^*}$, и исключаем его из процедуры обучения.
4. Предъявим сети все примеры обучающего множества. Если сеть не допустила ни одной ошибки, то переходим ко второму шагу процедуры.
5. Пытаемся обучить отконтрастированную сеть. Если сеть обучилась безошибочному решению задачи, то переходим к первому шагу процедуры, в противном случае переходим к шестому шагу.
6. Восстанавливаем сеть в состояние до последнего выполнения третьего шага. Если в ходе выполнения шагов со второго по пятый был отконтрастирован хотя бы один вес, (число обучаемых весов изменилось), то переходим к первому шагу. Если ни один вес не был отконтрастирован, то получена минимальная сеть.

Возможно использование различных обобщений этой процедуры. Например, контрастировать за один шаг процедуры не один вес, а заданное пользователем число весов. Наиболее радикальная процедура состоит в контрастировании половины весов связей. Если половину весов отконтрастировать не удается, то пытаемся отконтрастировать четверть и т.д. Отметим, что при описанном методе вычисления показателей чувствительности, предполагается возможным вычисление функции оценки и проведения процедуры обучения сети, а также предполагается известным обучающее множество. Возможен и другой путь.

2.2. Контрастирование без ухудшения

Пусть нам дана только обученная нейронная сеть и обучающее множество. Допустим, что вид функции оценки и процедура обучения нейронной сети неизвестны. В этом случае так же возможно контрастирование сети. Предположим, что данная сеть идеально решает задачу. Тогда нам необходимо так отконтрастировать веса связей, чтобы выходные сигналы сети при решении всех задач изменились не более чем на заданную величину. В этом случае

контрастирование весов производится понейронно. На входе каждого нейрона стоит адаптивный сумматор, который суммирует входные сигналы нейрона, умноженные на соответствующие веса связей. Для нейрона наименее чувствительным будет тот вес, который при решении примера даст наименьший вклад в сумму. Обозначив через x_p^q входные сигналы рассматриваемого нейрона при решении q-го примера получаем формулу для показателя чувствительности весов: $\chi(p,q) = \left|\left(w_p - w_p^*\right)\cdot x_p^q\right|$. Аналогично ранее рассмотренному получаем $\chi(p) = \left|\left(w_p - w_p^*\right)\right|\cdot \max_q\left|x_p^q\right|$. В самой процедуре контрастирования есть только одно отличие - вместо проверки на наличие ошибок при предъявлении всех примеров проверяется, что новые выходные сигналы сети отличаются от первоначальных не более чем на заданную величину.

3. Логически прозрачные нейронные сети

Одним из основных недостатков нейронных сетей, с точки зрения многих пользователей, является то, что нейронная сеть решает задачу, но не может рассказать как. Иными словами из обученной нейронной сети нельзя извлечь алгоритм решения задачи. Однако специальным образом построенная процедура контрастирования позволяет решить и эту задачу.

Зададимся классом сетей, которые будем считать логически прозрачными (то есть такими, которые решают задачу понятным для нас способом, для которого легко сформулировать словесное описания в виде явного алгоритма). Например потребуем, чтобы все нейроны имели не более трех входных сигналов.

Зададимся нейронной сетью у которой все входные сигналы подаются на все нейроны входного слоя, а все нейроны каждого следующего слоя принимают выходные сигналы всех нейронов предыдущего слоя. Обучим сеть безошибочному решению задачи.

После этого будем производить контрастирование в несколько этапов. На первом этапе будем контрастировать только веса связей нейронов входного слоя.

Если после контрастирования у некоторых нейронов осталось больше трех входных сигналов, то увеличим число входных нейронов. Затем аналогичную процедуру произведем поочередно для всех остальных слоев. После завершения описанной процедуры будет получена логически прозрачная сеть. Можно произвести дополнительное контрастирование сети, чтобы получить минимальную сеть. На рис. 2 приведены восемь минимальных сетей. Если под логически прозрачными сетями понимать сети, у которых каждый нейрон имеет не более трех входов, то все сети кроме пятой и седьмой являются логически прозрачными. Пятая и седьмая сети демонстрируют тот факт, что минимальность сети не влечет за собой логической прозрачности.

В качестве примера приведем интерпретацию алгоритма рассуждений, полученного по второй сети приведенной на рис. 2. Постановка задачи: по ответам на 12 вопросов необходимо предсказать победу правящей или оппозиционной партии. Ниже приведен список вопросов.

1. Правящая партия была у власти более одного срока?
2. Правящая партия получила больше 50% голосов на прошлых выборах?
3. В год выборов была активна третья партия?
4. Была серьезная конкуренция при выдвижении от правящей партии?
5. Кандидат от правящей партии был президентом в год выборов?
6. Был ли год выборов временем спада или депрессии?
7. Был ли рост среднего национального валового продукта на душу населения больше 2.1%?
8. Произвел ли правящий президент существенные изменения в политике?
9. Во время правления были существенные социальные волнения?
10. Администрация правящей партии виновна в серьезной ошибке или скандале?
11. Кандидат от правящей партии - национальный герой?
12. Кандидат от оппозиционной партии - национальный герой?

Ответы на вопросы описывают ситуацию на момент, предшествующий выборам. Ответы кодировались следующим образом: “да” - единица, “нет” -

минус единица. Отрицательный сигнал на выходе сети интерпретируется как предсказание победы правящей партии. В противном случае ответом считается победа оппозиционной партии. Все нейроны реализовывали пороговую функцию, равную 1, если алгебраическая сумма входных сигналов нейрона больше либо равна 0, и -1 при сумме меньшей 0. Ответ сети базируется на проявлениях двух синдромов: синдрома политической нестабильности (сумма ответов на вопросы 3, 4 и 9) и синдрома плохой политики (ответы на вопросы 4, 8 и 6). Заметим что симптом несогласия в правящей партии вошел в оба синдрома. Таким образом, для победы правящей партии необходимо отсутствие (-1) обоих синдромов.

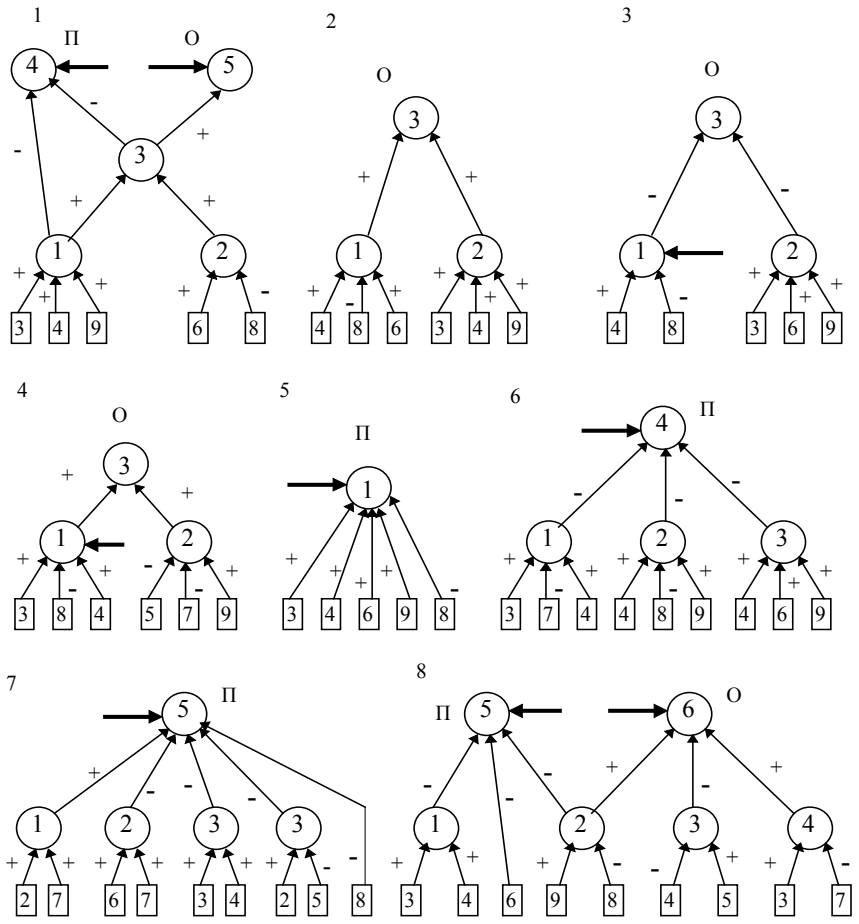

Рис. 2. Набор минимальных сетей для решения задачи о предсказании результатов выборов президента США. В рисунке использованы следующие обозначения: буквы "П" и "О" - обозначают вид ответа, выдаваемый нейроном: "П" - положительный сигнал означает победу правящей партии, а отрицательный - оппозиционной; "О" - положительный сигнал означает победу оппозиционной партии, а отрицательный - правящей.

(1) Нейрон; [1] Входной сигнал; ⟶ Связь между элементами сети;
"+","-" - знак связи; ⟶ Подача постоянного сигнала.

На рис. 2 приведены структуры шести логически прозрачных нейронных сетей,

решающих задачу о предсказании результатов выборов президента США [6,11]. Все сети, приведенные на этом рисунке минимальны в том смысле, что из них нельзя удалить ни одной связи так, чтобы сеть могла обучиться правильно решать задачу. По числу нейронов минимальна пятая сеть.

Заметим, что все попытки авторов обучить нейронные сети со структурами, изображенными на рис. 2, и случайно сгенерированными начальными весами связей закончились провалом. Все сети, приведенные на рис. 2, были получены из существенно больших сетей с помощью процедуры контрастирования. Сети 1, 2, 3 и 4 были получены из трехслойных сетей с десятью нейронами во входном и скрытом слоях. Сети 5, 6, 7 и 8 были получены из двухслойных сетей с десятью нейронами во входном слое. Легко заметить, что в сетях 2, 3, 4 и 5 изменилось не только число нейронов в слоях, но и число слоев. Кроме того, почти все веса связей во всех восьми сетях равны либо 1, либо -1.

4. Заключение

Технология получения явных знаний из данных с помощью обучаемых нейронных сетей выглядит довольно просто и вроде бы не вызывает проблем - необходимо ее просто реализовывать и пользоваться.

Первый этап: обучаем нейронную сеть решать базовую задачу. Обычно базовой является задача распознавания, предсказания (как в предыдущем разделе) и т.п. В большинстве случаев ее можно трактовать как ***задачу о восполнении пробелов в данных***. Такими пробелами являются и имя образа при распознавании, и номер класса, и результат прогноза, и др.

Второй этап: с помощью анализа показателей значимости, контрастирования и доучивания (все это применяется, чаще всего, неоднократно) приводим нейронную сеть к логически прозрачному виду - так, чтобы полученный навык можно было “прочитать”.

Полученный результат неоднозначен - если стартовать с другой начальной карты, то можно получить другую логически прозрачную структуру. ***Каждой базе данных отвечает несколько вариантов явных знаний.*** Можно считать это недостатком технологии, но мы полагаем, что, наоборот, технология, дающая единственный вариант явных знаний, недостоверна, а ***неединственность результата является фундаментальным свойством производства явных знаний из данных.***

Работа выполнена при поддержке Красноярского краевого фонда науки, грант 6F0124.

ЛИТЕРАТУРА

1. Горбань А.Н. Обучение нейронных сетей. М.": изд. СССР-США СП "ParaGraph", 1990. 160 с. (English Translation: AMSE Transaction, Scientific Siberian, A, 1993, Vol. 6. Neurocomputing, PP. 1-134).
2. Le Cun Y., Denker J.S., Solla S.A. Optimal Brain Damage // Advances in Neural Information Processing Systems II (Denver 1989). San Mateo, Morgan Kaufman, pp. 598-605 (1990)
3. Prechelt L. Comparing Adaptive and Non-Adaptive Connection Pruning With Pure Early Stopping // Progress in Neural Information Processing (Hong Kong, September 24-27, 1996), Springer, Vol. 1 pp. 46-52.
4. Gilev S.E., Gorban A.N., Kochenov D.A., Mirkes Ye.M., Golovenkin S.E., Dogadin S.A., Maslennikova E.V., Matyushin G.V., Nozdrachev K.G., Rossiev D.A., Shulman V.A., Savchenko A.A. "MULTINEURON" neural simulator and its medical applications // Proceedings of the International Conference on Neural Information Processing (Oct. 17-20, Seoul, Korea). V. 2. PP. 1261-1266.
5. Rossiev et al, The Employment of Neural-Network Classifier for Diagnostics of Different phases of Immunodeficiency // Modelling, Measurment & Control, C. Vol.42, No.2, 1994. PP. 55-63.

6. Gorban A.N., Waxman C. How many neurons are sufficient to elect the U.S.A. President? // AMSE Transaction, Scientific Siberian, A, 1993. Vol. 6. Neurocomputing. PP. 168-188
7. Gordienko P. Construction of efficient neural networks: Algorithms and tests // Proceedings of the International joint Conference on Neural Networks IJCNN'93, Nagoya, Japan, 1993. PP. 313-316.
8. Еремин Д.И. Контрастирование // Нейропрограммы/ под. ред. А.Н.Горбаня. Красноярск: изд. КГТУ, 1994. С. 88-108
9. Gorban A.N., Rossiyev D.A., Butakova E.V., Gilev S.E., Golovenkin S.E., Dogadin S.A., Kochenov D.A., Maslennikova E.V., Matyushin G.V., Mirkes Ye.M., Nazarov B.V., Nozdrachev K.G., Savchenko A.A., Smirnova S.V., Shulman V.A. Medical and Physiological Applications of MultiNeuron Neural Simulator. Proceedings of the WCNN'95 (World Congress on Neural Networks'95, Washington DC, July 1995). PP. 170-175.
10. Dorrer M.G., Gorban A.N., Kopytov A.G., Zenkin V.I. Psychological Intuition of Neural Networks. Proceedings of the WCNN'95 (World Congress on Neural Networks'95, Washington DC, July 1995). PP. 193-196.
11. Gorban A.N., Waxman Cory, Neural Networks For Political Forecast. Proceedings of the WCNN'95 (World Congress on Neural Networks'95, Washington DC, July 1995). PP. 176-178.
12. Горбань А.Н., Россиев Д.А. Нейронные сети на персональном компьютере. Новосибирск: Наука, 1996. 276 с.