

ВВЕДЕНИЕ В СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ

Левченко К.Г.

Финансовый университет при Правительстве РФ,
Россия, г. Москва, ул. Ленинградский проспект, 49
kglevchenko@fa.ru

Аннотация: В докладе описываются популярные методы машинного обучения и области применения нейронных сетей. Раскрываются методы обучения нейронных сетей. Предлагается методика прогнозирования на основе нейронных сетей для моделирования финансовых временных рядов.

Ключевые слова: временные ряды, тренд, нейронный прогноз

Введение

В последние несколько лет основные усилия по исследованию машинного зрения были сосредоточены на свёрточных нейронных сетях (ConvNets, CNNs / СНС). Эти усилия привели к появлению новейших методов решения широкого круга задач классификации. Тем не менее, несмотря на длительную историю подобных подходов, нам всё ещё не хватает теоретического понимания того, как эти системы достигают выдающихся результатов. Действительно, сегодня в области машинного зрения многие прибегают к помощи СНС как некоего «чёрного ящика», который выполняет поставленные перед ним задачи, но имеют очень смутное представление о том, как именно он это делает. Опишем основные подходы к пониманию свёрточных нейронных сетей.

1 Многослойные сети

Сфокусируемся на архитектурах искусственных нейронных сетей ввиду их выдающейся роли. Типичная архитектура нейронной сети состоит из входного слоя x , выходного слоя y и стека из нескольких скрытых слоев h , где каждый слой включает несколько клеток или блоков, как показано на рис. 1.

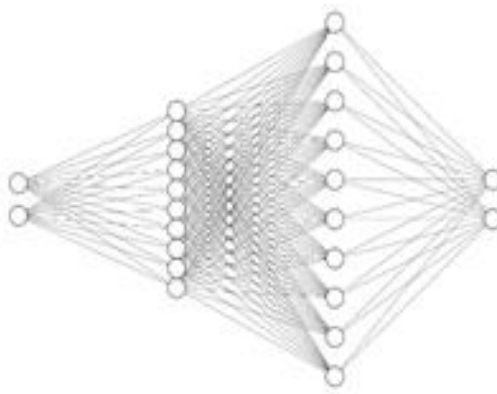


Рис. 1. Нейронная сеть

Обычно каждый скрытый блок h_j получает данные от всех блоков на предыдущем слое и определяется как взвешенная комбинация входных данных, за которой следует нелинейность согласно

$$(1) \quad h_j = F(b_j + \sum_i \omega_{ij} x_i)$$

где ω_{ij} – веса, управляющие прочностью связей между входными блоками и скрытым блоком, b_j – небольшое смещение скрытого блока, а $F(\cdot)$ – некоторая насыщающая нелинейность типа сигмоиды.

Несмотря на то, что нейросетевые модели существуют с 1960-х годов, до недавнего времени они не использовались широко.

Серьёзным препятствием развитию многослойных нейронных сетей является тот факт, что они полагаются на очень большое количество параметров, что, в свою очередь, подразумевает необходимость больших объёмов обучающих данных и вычислительных ресурсов для поддержки обучения по этим параметрам.

Важным достижением, позволившим сделать большой скачок в изучении глубоких нейронных сетей, стало послойное предобучение без учителя с использованием ограниченной машины Больцмана

(RBM/ОМБ). Ограниченные машины Больцмана можно рассматривать как двухслойные нейронные сети, где в ограниченной форме разрешены только прямые связи.

2. Рекуррентные нейронные сети

Для решения задач, основанных на последовательных входах, одной из наиболее удачных многослойных архитектур является рекуррентная нейронная сеть (РНС). Рекуррентная нейронная сеть, представленная на рис. 2, можно рассматривать как особый тип нейронной сети, где каждый скрытый блок принимает на вход данные из наблюдений на текущем временном шаге, а также из своего состояния на предыдущем временном шаге. Выход РНС определяется как

$$(2) \quad h_t = \sigma(\omega_i x_t + u_i h_{t-1})$$

где σ – некоторая нелинейная функция сжатия, а ω_i и u_i – параметры сети, управляющие относительной важностью получаемой в настоящий момент и полученной ранее информации.

Recurrent Neural Network (RNN)

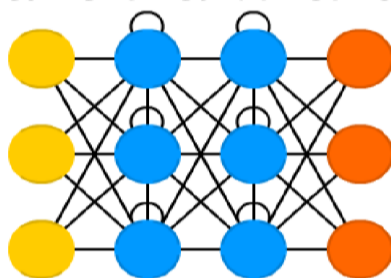


Рис. 2. Рекуррентная нейронная сеть

Хотя РНС создают впечатление мощных архитектур, одной из их главных проблем является ограниченная способность моделировать долгосрочные зависимости. Это ограничение связано с трудностями обучения из-за проблемы взрывающегося или затухающего градиента, которая может возникнуть при обратном распространении ошибки через несколько временных шагов. Чтобы справиться с этой трудностью, были введены сети долгой краткосрочной памяти (LSTM).

LSTM – это рекуррентные сети, дополнительно оснащённые компонентом хранения или памяти, который накапливает информацию с течением времени. Ячейка памяти LSTM снабжена вентилями таким образом, что она позволяет считывать из неё или записывать в неё информацию. Примечательно, что LSTM также содержат вентиль забывания, который позволяет сети стирать информацию, когда она больше не нужна.

3 Свёрточные сети

Свёрточные сети (СНС) – это особый тип нейронных сетей, которые особенно хорошо адаптированы к приложениям компьютерного зрения благодаря их способности к иерархически абстрактным представлениям с локальными операциями. Есть две ключевые идеи проектирования, определяющие успех свёрточных архитектур в области машинного зрения. Во-первых, СНС используют преимущества двумерной структуры изображений и тот факт, что соседние пиксели обычно сильно коррелированы. Поэтому СНС отказываются от использования взаимно однозначных связей между всеми блоками пикселей (как в случае большинства нейронных сетей) в пользу сгруппированных локальных связей. Кроме того, архитектуры СНС основаны на разделении признаков, и каждый канал (или выходная карта признаков), таким образом, генерируется из свёртки с одним и тем же фильтром на всех участках. Эта важная характеристика СНС обуславливает архитектуру, которая опирается на гораздо меньшее количество параметров по сравнению со стандартными нейронными сетями. Во-вторых, в СНС также появляется стадия пулинга, которая обеспечивает определённую степень инвариантности относительно параллельного переноса, делая архитектуру менее подверженной влиянию небольших вариаций расположения. Примечательно, что благодаря пулингу сеть также получает способность постепенно видеть более крупные части входных данных благодаря увеличенному размеру рецептивного поля сети. Увеличение размера рецептивного поля (в сочетании с уменьшением разрешения данных) позволяет сети представлять более абстрактные характеристики входных данных по мере увеличения глубины сети. К примеру, для задачи

распознавания объектов рекомендуется, чтобы слои СНС сначала фокусировались на границах частей объекта, в конечном итоге охватывая весь объект на более высоких слоях иерархии.

Архитектура свёрточных сетей в значительной степени вдохновлена процессом, происходящим в зрительной коре. Неокогнитрон состоит из каскада слоёв K , где каждый слой чередует блоки из клеток S -типа, U_{sl} и блоки из сложных клеток C -типа, U_{cl} , которые отчасти имитируют процесс обработки в биологических простых и сложных клетках соответственно, как показано на рис. 3. Блоки простых клеток выполняют операции, аналогичные локальным свёрткам, за ними следует нелинейность блока линейной ректификации (ReLU),

$$(3) \quad \varphi(x) = \begin{cases} x; & \text{if } x \geq 0 \\ 0; & x < 0 \end{cases},$$

в то время как сложные клетки выполняют операции, аналогичные пулингу усреднением. Модель также включала разделяющую нелинейность, которая обеспечивала тот же эффект, что в современных СНС достигается путём нормализации.

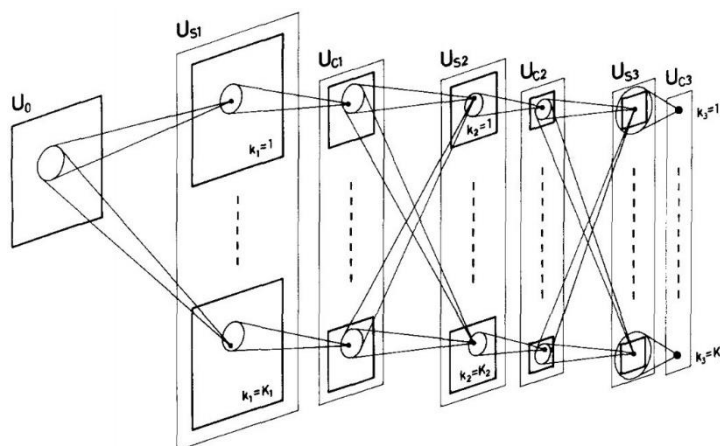


Рис. 3. Структура Неокогнитрона

В отличие от большинства стандартных архитектур сверточных нейронных сетей, неокогнитрон не нуждается в помеченных данных для обучения, поскольку он спроектирован на основе самоорганизующихся карт, где обучение локальных связей между последовательными слоями происходит с помощью повторяющихся представлений набора изображений-стимулов. В частности, Неокогнитрон обучается распознавать связи между входной картой признаков и слоем простых клеток и процедура обучения может быть в целом сведена к двум шагам. На первом шаге каждый раз, когда на входе появляется новый стимул, простые клетки с максимальной реакцией на него выбираются в качестве репрезентативной клетки для данного типа стимула. На втором шаге связи между входом и этими репрезентативными клетками усиливаются каждый раз, когда они реагируют на один и тот же тип входа. Следует отметить, что слои простых клеток организованы в различные группы или плоскости таким образом, что каждая плоскость реагирует только на один тип стимула. Последующие версии Неокогнитрона допускали обучение с учителем, а также нисходящие механизмы внимания.

Большинство архитектур СНС, используемых в последнее время в приложениях машинного зрения, вдохновлены успешной архитектурой для распознавания рукописного текста известной как LeNet.

Порождающие состязательные сети (GANs/ПСС) – это относительно новые модели, использующие преимущества сильной репрезентативной способности многослойных архитектур. ПСС были впервые введены в 2014 году и хотя они не представляли собой новую архитектуру как таковую (например, с точки зрения новых строительных блоков сети), они отличаются некоторыми особенностями, которые делают их несколько иным классом многослойных архитектур. Ключевой проблемой, которую решают ПСС, является внедрение подхода к обучению без учителя, который не требует помеченных данных.

4 Обучение многослойных сетей

Успех различных многослойных архитектур в значительной степени зависит от успешности процесса их обучения. Процедура обучения обычно основывается на обратном распространении ошибки с использованием градиентного спуска. Градиентный спуск широко используется в обучении

многослойных архитектур из-за своей простоты. В его основе – минимизация гладкой функции ошибок $E(w)$ по итерационной процедуре, определённой уравнением:

$$(4) \quad \omega_k = \omega_{k-1} - \alpha \frac{\partial E(\omega)}{\partial \omega}$$

где w представляет параметры сети, α – скорость обучения, которая может контролировать скорость сходимости, а $\partial E(w)/\partial w$ – градиент ошибки, вычисленный по обучающему набору. Этот простой метод градиентного спуска особенно подходит для обучения многослойных сетей благодаря использованию цепного правила для обратного распространения и вычислению производной ошибки по различным параметрам сети на разных слоях.

Одной из основных проблем алгоритма градиентного спуска является выбор скорости обучения, α . Слишком малая скорость обучения ведёт к медленной сходимости, в то время как высокая скорость обучения может привести к промахам или флуктуациям вокруг оптимума. Поэтому для дальнейшего совершенствования метода было предложено несколько вариантов оптимизации простого стохастического градиентного спуска. Простейший метод, называемый стохастическим градиентным спуском с импульсом (momentum), отслеживает величину обновления от одной итерации к другой и даёт импульс процессу обучения, стимулируя дальнейшее обновление, если градиент продолжает указывать то же направление, от одного временного шага к другому, как определяется уравнением:

$$(5) \quad \omega_k = \omega_{k-1} - \alpha \frac{\partial E(\omega)}{\partial \omega} - \gamma \left(\frac{\partial E(\omega)}{\partial \omega} \right)_{t-1}$$

при этом γ управляет импульсом. Ещё один простой метод заключается в том, чтобы настроить коэффициент скорости обучения на снижение по заданному графику, но это далеко от идеала, учитывая, что график необходимо определить до начала учебного процесса и он полностью независим от данных. Другие более усложнённые методы предлагают во время обучения адаптировать скорость обучения к каждому обновляемому параметру w_i , выполняя менее крупные обновления на часто изменяющихся параметрах и более крупные обновления – на нечасто изменяющихся.

Основным недостатком обучения с использованием градиентного спуска, как и его вариантов, является необходимость в больших объемах помеченных данных.

В теории свёрточные сети могут применяться к данным произвольной размерности. Их двумерные реализации хорошо подходят к структуре отдельных изображений и поэтому заслужили особое внимание в области машинного зрения. Благодаря наличию крупномасштабных наборов данных и мощных компьютеров для обучения, в сообществе специалистов по компьютерному зрению в последнее время наблюдается резкий рост использования СНС для различных приложений.

Интерес к архитектурам СНС возобновился после разработки сети AlexNet. AlexNet удалось добиться рекордных результатов распознавания объектов на наборе данных ImageNet. Она состояла в общей сложности из восьми слоёв, 5 свёрточных и 3 полносвязных.

В AlexNet было представлено несколько проекторочных решений, которые позволили эффективно обучать сеть с использованием стандартного стохастического градиентного спуска. В частности, ключом к успеху AlexNet стали четыре важных фактора. Во-первых, в AlexNet применяется нелинейность ReLU вместо таких насыщающих нелинейностей, как сигмоиды, которые использовались в предыдущих передовых архитектурах СНС. Использование ReLU устранило проблему затухающего градиента и ускорило обучение. Во-вторых, с учётом факта, что последние полносвязные слои в сети содержат наибольшее количество параметров, в AlexNet было использовано прореживание (дропаут), впервые внедрённое в контексте нейронных сетей для борьбы с проблемой переобучения. Прореживание, реализованное в AlexNet, заключается в случайном отбрасывании (т.е. установке в ноль) определённого процента параметров слоя. Этот метод позволяет обучать немного различную архитектуру на каждом проходе сети и искусственно сокращать количество параметров, которые должны быть обучены на каждом проходе, что в конечном итоге помогает разбить корреляции между блоками и таким образом справиться с переобучением. В-третьих, AlexNet опиралась на пополнение (аугментацию) данных для улучшения способности сети обучаться инвариантным представлениям. Например, сеть обучалась не только на оригинальных изображениях в обучающем наборе, но и на вариациях, генерируемых случайным сдвигом и отражением обучающих изображений. Наконец, в AlexNet также использовались несколько методов, ускоряющих сходимость процесса обучения, в т.ч. использование импульса и плановое снижение скорости обучения, при котором скорость снижается каждый раз, когда процесс обучения выходит на плато.

Изобретение AlexNet вызвало всплеск научных работ с попытками понять механизм обучения сети, которые они предпринимали разными способами: с помощью визуализации, как это было сделано в сети DeConvNet или с помощью систематических исследований различных архитектур.

Одной из трудностей при работе с СНС является необходимость использовать очень большие наборы данных для обучения всех базовых параметров. Даже такие крупномасштабные наборы данных, как ImageNet, с более чем миллионом изображений, считаются слишком маленькими для обучения некоторых глубоких архитектур. Один из способов справиться с требованиями к набору данных – это искусственно пополнить набор, изменяя изображения, например, путём случайных отражений, вращений или искажений. Главное преимущество состоит в том, что в результате сети становятся более инвариантными к различным преобразованиям. На самом деле этот метод стал одной из главных причин большого успеха AlexNet. Поэтому помимо методов, изменяющих архитектуру сети для облегчения обучения, другое направление работы состоит во внедрении новых строительных блоков, которые обеспечивают лучшее качество обучения.

Помимо простых задач классификации, таких как распознавание объектов, в последнее время СНС успешно справляются и с задачами, требующими точной локализации, такими как семантическая сегментация и обнаружение объектов. Одной из наиболее успешных сетей для семантической сегментации является так называемая полностью свёрточная сеть (FCN). Как следует из названия, FCN не использует полносвязные слои явно, а вместо этого приводит их к форме свёрточных, рецептивные поля которых покрывают всю базовую карту признаков.

Пространственно-временные СНС на основе LSTM, например, были одними из самых ранних попыток расширить двумерные сети для пространственно-временных данных.

Цель СНС на основе LSTM состоит в постепенной интеграции временной информации без строгих ограничений размера входа (во временном отношении). Одним из преимуществ такой архитектуры является появление у сети способности создавать текстовые описания переменного размера. Однако в то время, как LSTM способны улавливать глобальные отношения движения, они могут не справиться с более мелкими паттернами движения. Кроме того, эти модели обычно крупнее, требуют больше данных и поэтому их трудно обучать. На сегодняшний день, за исключением случаев интеграции видео- и текстового анализа, успешность применения LSTM в анализе пространственно-временных изображений ограничена.

Литература

1. *Hadi I., Wildes R. P.* What do we understand about convolutional networks? //arXiv preprint arXiv:1803.08834. – 2018.
2. *Галушкин А. И.* Нейронные сети. Основы теории. Монография. – 2012.
3. *Манжула В. Г., Федяшов Д. С.* Нейронные сети Кохонена и нечеткие нейронные сети в интеллектуальном анализе данных //Фундаментальные исследования. – 2011. – №. 4.