

Глубокое обучение

В последние годы популярность глубокого обучения в мире сильно возросла в силу роста вычислительных мощностей и появления новых архитектур. В частности, интерес представляют нейронные сети. Нейронная сеть - это модель машинного обучения, построенная по принципу организации и функционирования нервных клеток живого существа.

Нейронные сети состоят из ряда соединённых друг с другом узлов, называемых нейронами, которые, в общем виде, моделируют нейроны человеческого мозга. Соединяются нейроны друг с другом посредством синапсов. Благодаря подобной структуре, нейронные сети способны не только анализировать входящую информацию, но и воспроизводить её из своей памяти.

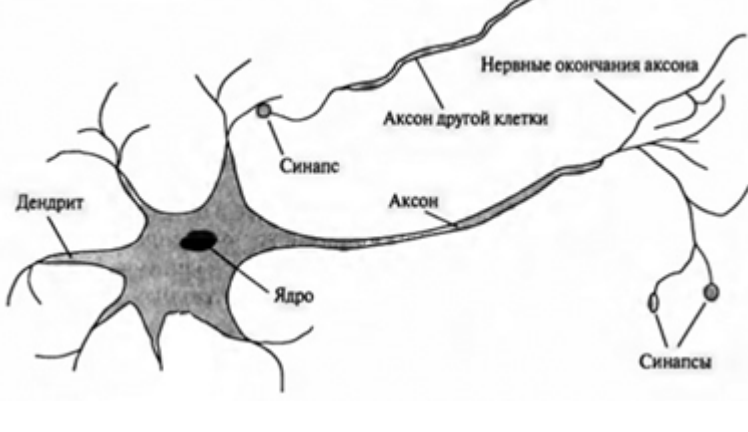


Рисунок 1. Схема нейрона.

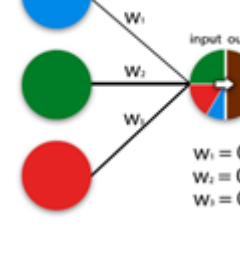


Рисунок 2. Синапс нейронной сети.

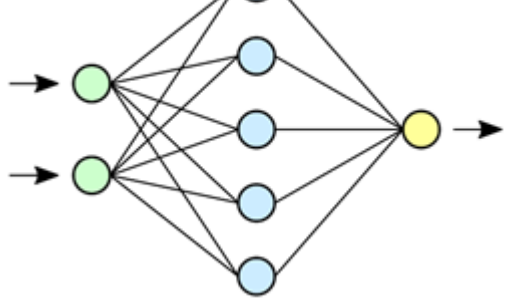
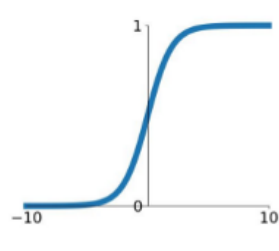


Рисунок 3. Полноценная нейронная сеть

На входные нейроны (зеленые) поступают входные данные, на выходные (желтые) поступают выходные данные, и в процессе обучения у связей между нейронами формируются веса. Голубые нейроны – скрытый слой, их количество задается человеком, соответственно, чем больше скрытых слоев, тем больше связей будет у нейронной сети, и тем большее количество весов ей нужно определить. Таким образом нейронная сеть определяет, насколько значимыми в том или ином случае являются те или иные нейроны. Выходные данные зависят от функции активации. Некоторые из функций активаций представлены ниже:

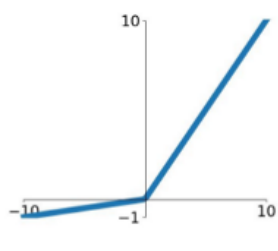
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



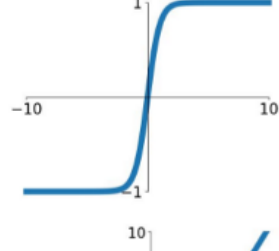
Leaky ReLU

$$\max(0.1x, x)$$



tanh

$$\tanh(x)$$

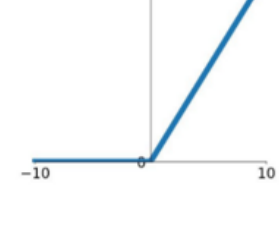


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ReLU

$$\max(0, x)$$



ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$

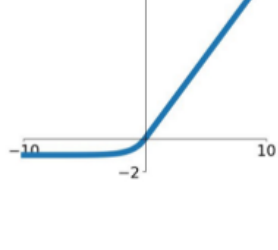


Рисунок 4. Основные функции активации

Архитектуры нейронных сетей

Сверточные нейронные сети

Для решения разных задач существует огромное количество архитектур нейронных сетей. Например, для работы с изображением используются сверточные (конволюционные) нейронные сети:

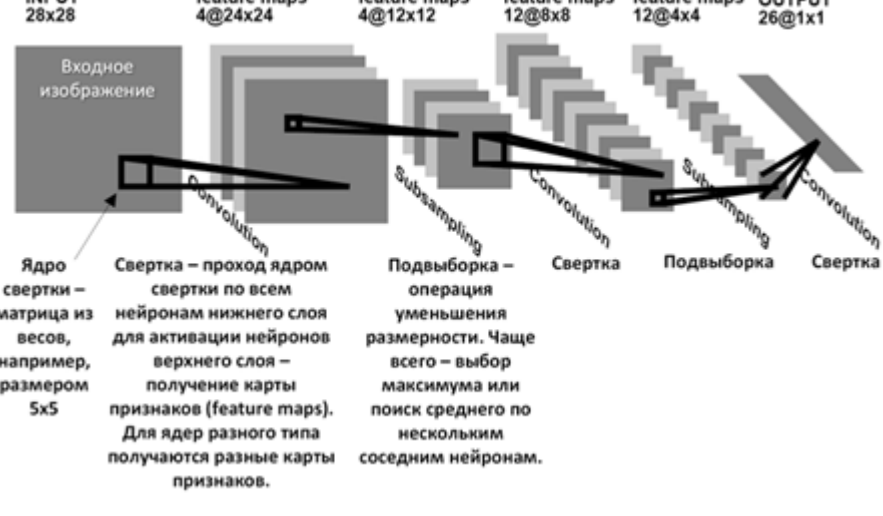


Рисунок 5. Сверточная нейронная сеть

Рекуррентные нейронные сети

Для решения задач регрессии, в частности задач прогнозирования, а то есть предсказания временных последовательностей, используют рекуррентные сети. В силу их особенности, они запоминают данные на всех временных шагах, а также благодаря принципу авторегрессии учитывают все предыдущие значения при предсказании последующих. Такие сети также зачастую используются при работе с текстами, например, при переводе, а также продолжении слов при помощи нейронных сетей (к примеру, сервис Яндексa «Балабоба»).

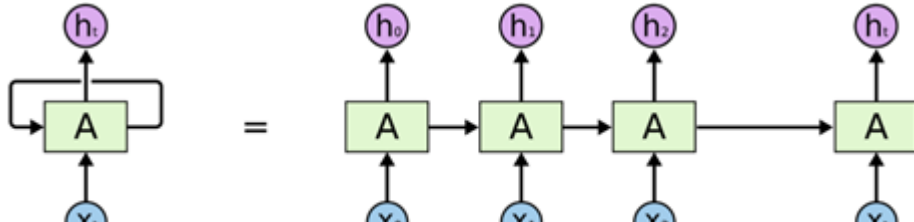


Рисунок 6. Рекуррентные нейронные сети.

Сети с использованием архитектуры encoder-decoder

Также для работы с текстом(и аналогичных задач, которые включают в себя работу с последовательностями элементов) используются нейронные сети с архитектурой encoder-decoder, которые иначе называют трансформерами. Например, такую сеть использует сервис «Google Translate». Именно благодаря использованию подобных нейронных сетей в какой-то момент качество перевода резко возросло и на текущий момент в большинстве случаев является не хуже человеческого.

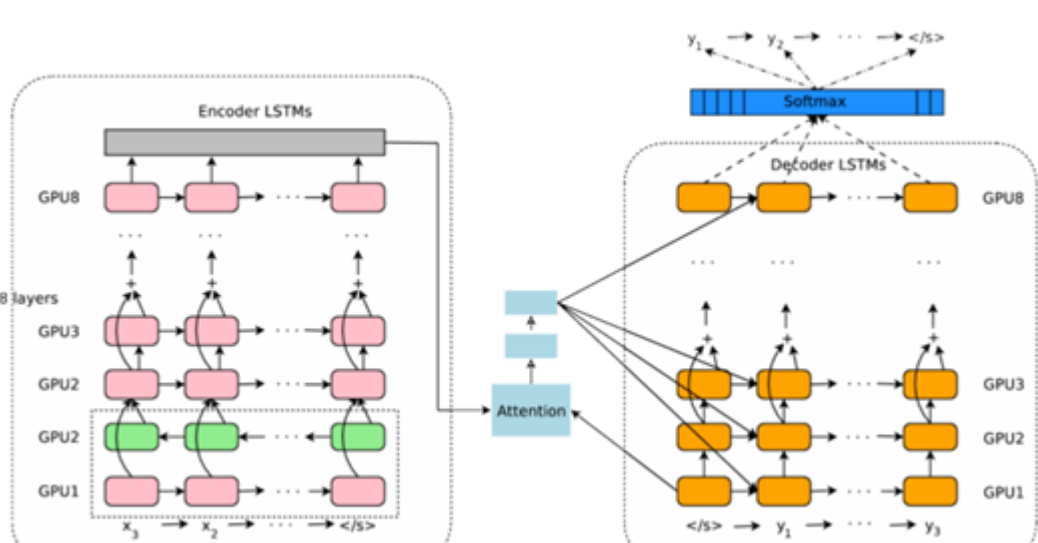


Рисунок 7. Архитектура Google's neural machine translation.

Обучение нейронных сетей

При обучении нейронных сетей невозможно сказать, каким именно образом формируются веса между нейронами. Это, по своей сути, черный ящик, который и вызывает недоверие у множества ученых и коммерческих деятелей. При обучении фокус на формирование весов выстраивается случайным образом. Это приводит к недообучению или переобучению. Для отслеживания этого используют зачастую так называемые обучающие кривые, которые выражают ошибку на обучающей и на тестовой выборке. Если они начинают расходиться – это первый признак того, что что-то идет не так. Таким образом, при недообучении модель строит слишком простую зависимость между данными, не отражающую реальный результат

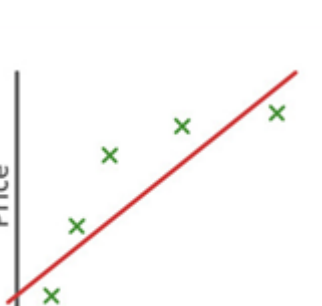


Рисунок 8. Пример недообученной модели

С другой стороны, переобучение строит слишком сложную модель при предсказании данных. Тут может возникнуть вполне резонный вопрос – разве не чем больше точность модели, тем лучше результат? Ответ прост – безусловно да, но при переобучении модель скорее не учится предсказывать, а запоминает данные. Таким образом, если взять пример, которого не было в обучающей выборке – результат предсказания будет плохой.

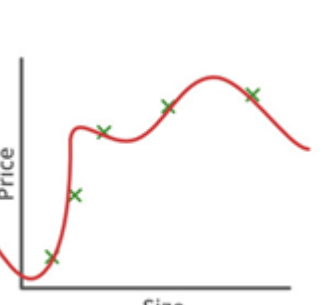


Рисунок 9. Пример переобученной модели

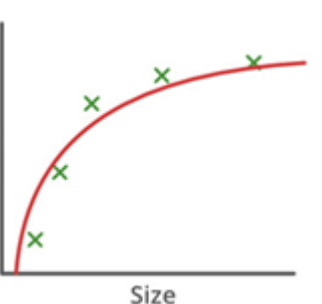


Рисунок 10. Пример хорошо обученной модели

Что касается обучающей кривой, выглядит она следующим образом:



Рисунок 11. Обучающая кривая.

Из рисунка видно, что при больших значениях функции потерь происходит недообучение. Когда ошибка на тестовой выборке начинает уходить в отрыв от обучающей выборки, происходит переобучение. Идеал – где-то посередине. Однако далеко не всегда обучающая кривая выглядит так гладко, как показано выше. Порой она выглядит совсем иначе:

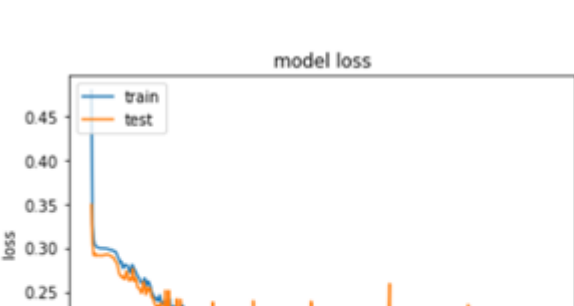


Рисунок 12. Пример реальной обучающей кривой