**СЛАЙД 1**

Сегодня у нас урок №4 – и мы продолжаем наше краткое знакомство с нейронными сетями.

**СЛАЙД 2**

На прошлом уроке мы остановились на схеме многослойной нейронной сети и задали вопрос: а где же в этой схеме заключена способность к обучению? Что мы должны в ней регулировать, реагируя на изменяющиеся входные данные? Ведь мы знаем - нейронные сети должны обучаться!

**СЛАЙД 3**

Наиболее очевидной величиной, которую мы могли бы регулировать, является сила связи между узлами. На схеме вновь показаны соединенные между собой узлы, но на этот раз с каждым соединением ассоциируется определенный вес. Низкий весовой коэффициент ослабляет сигнал, высокий — усиливает его.

Следует сказать несколько слов о небольших индексах, указанных рядом с коэффициентами. Например, символ W23 обозначает весовой коэффициент, связанный с сигналом, который передается от узла 2 данного слоя к узлу 3 следующего слоя. Следовательно, W12 — это

весовой коэффициент, который ослабляет или усиливает сигнал, передаваемый от узла 1 к узлу 2 следующего слоя.

Приведенная картинка с тремя слоями нейронов, каждый из которых связан с каждым из нейронов в предыдущем и следующем слоях, выглядит довольно привлекательно. Однако интуиция подсказывает нам, что рассчитать распространение входных сигналов по всем слоям не так-то просто.

**СЛАЙД 4**

Поэтому мы попытаемся проделать необходимые вычисления на примере меньшей нейронной сети, состоящей всего лишь из двух слоев, каждый из которых включает по 2 нейрона.

Как и раньше, каждый узел превращает сумму двух входных сигналов в один выходной с помощью функции активации. Мы также будем использовать сигмоиду.

А что насчет весовых коэффициентов? Это очень хороший вопрос: с какого значения следует начать? Обычно начинают со случайных значений.

Первый слой узлов — входной, и его единственное назначение — представлять входные сигналы. Таким образом, во входных узлах функция активации не применяется, т.е. с первым слоем все просто — никаких вычислений. Далее мы должны заняться вторым слоем, в котором потребуется выполнить некоторые вычисления. Нам предстоит определить входной сигнал для каждого узла в этом слое.

Входной сигнал для второго слоя вычисляется с помощью следующего выражения:

*х = ( выход первого узла \* вес связи) + ( выход второго узла \* вес связи)*

**СЛАЙД 5**

Итак, допустим нам удалось успешно описать распространение сигналов по нейронной сети, т.е. определить величину выходных сигналов при заданных величинах входных сигналов.

Что дальше?

Наш следующий шаг заключается в сравнении выходных сигналов нейронной сети с тренировочными данными (т.е. с заранее известными сигналами) для определения ошибки. Нам необходимо знать величину этой ошибки, чтобы можно было улучшить выходные результаты путем изменения параметров сети. Т.е. нам нужна корректировка весовых коэффициентов в процессе обучения нейронной сети.

Как нам обновлять весовые коэффициенты связей в случае, если выходной сигнал и его ошибка формируются за счет вкладов более чем одного узла? Эту задачу иллюстрирует приведенная схема.

Если на выходной узел поступает сигнал только от одного узла, то все намного проще. Но как использовать ошибку выходного сигнала при наличии двух входных узлов?

Наиболее оптимальной является идея распределять ошибку между узлами пропорционально связям между ними. Большая доля ошибки приписывается вкладам тех связей, которые имеют больший вес. Почему? Потому что они оказывают большее влияние на величину ошибки. Эту идею и иллюстрирует схема. В данном случае сигнал, поступающий на выходной узел, формируется за счет двух узлов. Весовые коэффициенты связей равны 3,0 и 1,0. Распределив ошибку между двумя узлами пропорционально их весам, мы увидим, что для обновления значения первого, большего веса следует использовать 3/4 величины ошибки, тогда как для обновления значения второго, меньшего веса — 1/4.

**СЛАЙД 6**

Мы можем расширить эту идею на случаи с намного большим количеством узлов. Если бы выходной узел был связан со ста входными узлами, мы распределили бы выходную ошибку между всеми ста

связями пропорционально их вкладам, размер которых определяется весами соответствующих связей.

В данной схеме мы используем веса для распространения ошибки от выходного слоя вглубь сети. этот метод называется **обратным распространением ошибки** в процессе обучения нейронной сети.

Итак подытожим:

• Нейронные сети обучаются посредством уточнения весовых коэффициентов своих связей. Этот процесс управляется ошибкой — разностью между правильным ответом, предоставляемым тренировочными данными, и фактическим выходным значением.

• Ошибка на выходных узлах определяется простой разностью между желаемым

и фактическим выходными значениями.

Важно то, что при обратном распространении ошибок учитываются весовые коэффициенты

связей, и это наилучший показатель того, что мы пытаемся справедливо распределить ответственность за возникающие ошибки.

Идем дальше!

**СЛАЙД 7**

Мы с вами пока что не приступили к решению главной задачи — обновлению весов связей в нейронной сети. Мы понимаем, что ошибки подсказывают нам, как должны быть изменены

веса связей, чтобы улучшить результирующий общий ответ на выходе нейронной сети.

Первое, что приходит в голову: а почему бы не использовать алгебраические методы для непосредственного вычисления весов? Но, к сожалению, этот путь нам не подходит ввиду громоздкости соответствующих вычислений. Существует слишком много комбинаций весов и слишком много функций, зависящих от функций, зависящих от других функций и т.д., которые мы должны комбинировать в ходе анализа распространения сигнала по сети.

Чтобы убедиться в том, насколько они не тривиальны, достаточно взглянуть на приведенное устрашающее выражение, которое представляет выходной сигнал как функцию входных сигналов и весовых коэффициентов для простой нейронной сети с тремя слоями по три узла. Ничего себе! Лучше держаться от этого подальше!

Еще вариант: а не могли бы мы просто перебирать случайные сочетания весовых коэффициентов, пока не будет получен устраивающий нас результат? Но представьте, что каждый весовой коэффициент может иметь 1000 возможных значений в диапазоне от -1 до +1, например 0,501,

-0,203 или 0,999. Тогда в случае нейронной сети с тремя слоями по три узла, насчитывающей 18 весовых коэффициентов, мы должны были бы протестировать 18 тысяч возможностей. Если бы у нас

была более типичная нейронная сеть с 500 узлами в каждом слое, то нам пришлось бы протестировать 500 миллионов различных значений весов. Если бы для расчета каждого набора комбинаций требовалась одна секунда, то для обновления весов с помощью всего лишь одного тренировочного примера понадобилось бы примерно 16 лет. Тысяча тренировочных примеров — и мы имели бы 16 тысяч лет! Как говорится – не вариант!

**СЛАЙД 8**

В общем, эта головоломка не поддавалась математикам на протяжении многих лет и получила реальное практическое разрешение лишь в 60-70-х годах прошлого века. Итак, как же нам разрешить эту явно трудную проблему?

Об этом мы узнаем уже на следующем уроке – это будет заключительный урок по нейросетям.

И после этого мы перейдем к компьютерному зрению.

А на сегодня все, всем пока!