

Т. Н. Кондратьева (Ростов-на-Дону, ДГТУ). **Прогнозирование с помощью многослойной рекуррентной нейронной сети LSTM.**

Возможности прогнозирования нейронной сети напрямую связаны с ее способностью обобщения и выделения скрытых зависимостей между входными и выходными данными. После обучения сеть способна предсказать будущее значение некой последовательности на основе нескольких предыдущих значений [1]. Я буду использовать многослойную рекуррентную нейронную сеть LSTM для предсказания последнего значения на момент времени t , учитывая значения за $(t - 1)$ временной период. RNN сети решают этот вопрос, поскольку содержат в себе обратные связи, позволяющие сохранять информацию.

Задача сводится к определению функции для загрузки данных, компиляции модели, обучения и построения результатов [2]. В качестве основного источника наборов данных будем использовать репозиторий машинного обучения UCI Machine Learning Repository, применение которого позволяет анализировать алгоритмы машинного обучения.

Логика работы сети LSTM представлена на рис., $\{x_0, x_1, \dots, x_n\}$ — последовательность входных данных, $\{h_0, h_1, \dots, h_n\}$ — последовательность выходных данных.

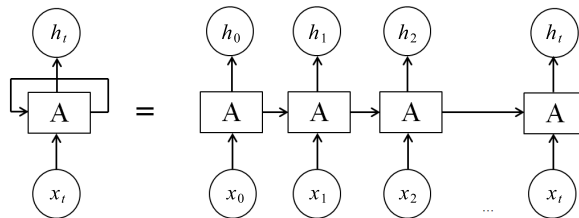


Рис. Архитектура нейронной сети LSTM

Первый слой вычисляет, насколько на данном шаге ему нужно забыть предыдущую информацию, множители к компонентам вектора памяти (1).

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (1)$$

Второй слой вычисляет, насколько ему интересна новая информация, пришедшая с сигналом — такой же множитель, но уже для наблюдения (2).

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \quad (2)$$

$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C).$$

На третьем слое вычисляется линейная комбинация памяти и наблюдения с только вычисленными весами для каждой из компонент. Так получается новое состояние памяти, которое в таком же виде передается далее (3).

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t. \quad (3)$$

Осталось вычислить output. Но поскольку часть входного сигнала уже в памяти, не нужно считать активацию по всему сигналу. Сначала сигнал проходит через сигмоиду, которая решает, какая его часть важна для дальнейших решений, затем гиперболический тангенс «размазывает» вектор памяти на отрезке $[-1;1]$, и в конце эти два вектора перемножаются (4).

$$\begin{aligned} O_t &= \sigma(W_o[h_{t-1}, x_t] + b_o), \\ h_t &= O_t \tanh(t). \end{aligned} \quad (4)$$

Для обучения сети используем алгоритм BPTT [3]. Построение модели Sequential сводится к сложению слоев в одном объекте. Создадим сеть с одномерным входом, двумя скрытыми слоями и одномерным выходным слоем [4]. После инициализации модели создаем первый слой, в данном случае слой LSTM. Здесь мы используем параметры по умолчанию, поэтому он ведет себя как стандартный повторяющийся слой. При создании второго слоя используем рекурсию Keras. Последний слой — это плотный слой, поскольку активация его регрессии, является линейной. Наконец, мы компилируем модель, используя среднеквадратичную ошибку и оптимизатор RMSprop. При обучении модели используем метод подгонки модели. В заключение, можно сказать, что в целом данная архитектура нейронной сети позволяет решать такие задачи, как, предсказание элемента h_i последовательности $\{h_i\}_{i=0}^n$, на примере последовательности $\{x_i\}_{i=0}^n$.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 17-01-00888 а.

СПИСОК ЛИТЕРАТУРЫ

1. Беркинблит М. Б. Нейронные сети. М.: МИРОС и ВЗМШ РАО, 1993, 96 с.
2. Вороновский Г. К., Мазотило К. В., Петрашев С. Н., Сергеев С. А. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности. Харьков: Основа, 1997, 112 с.
3. Горбань А. Н. Обучение нейронных сетей. М.: СССР-США СП «Параграф», 1990, 160 с.
4. Белявский Г. И., Лила В. Б., Пучков Е. В. Алгоритм и программная реализация гибридного метода обучения искусственных нейронных сетей. — Программные продукты и системы, 2012, № 4, с. 96–100. [Текст].