

Обработка естественного языка. Лекция 3
Рекуррентные ИНС
и модели памяти

Михаил Пожидаев

20 сентября 2024 г.

Рекуррентные ИНС

Рекуррентная ИНС (*Recurrent neural network*) не забывает значение всех или части выходных нейронов после обработки одного набора входных данных, а использует их для продолжения своей работы.

Вход:

Последовательность элементов (векторов), каждый из которых принято называть time step'ом.

Выход:

Один вектор, кодирующий входную последовательность.

Примем во внимание

Особенности рекуррентной ИНС

1. Появляется множество скрытых состояний h_t , последнее из которых является результирующим вектором.
2. Часто присутствует вектор контекста, который в общем случае не совпадает с вектором скрытого состояния.
3. Поскольку обучающие данные поступают в виде пакета, длина всех последовательностей пакета должна быть одинаковой, но общего требования равной длины для всех последовательностей нет.

Сеть Элмана

$$h_t = f_h(W_h x_t + U_h h_{t-1} + b_h)$$

$$y_t = f_y(W_y h_t + b_y)$$

Seq2seq

Задача преобразования одной последовательности в другую (*sequence to sequence, seq2seq*), в которой вектор контекста связывает две ячейки.

Первая производит кодирование (кодер), а вторая — декодирование (декодер).

Обе являются ячейками рекуррентной ИНС.

Задачи для Seq2seq

Машинный перевод

Исходная и результирующая последовательности содержат текст на разных языках.

Распознавание речи

Исходная последовательность представляет собой звуковой сигнал, а результирующая — текст на естественном языке.

Синтез речи

Исходная последовательность представляет собой текст на естественном языке, а результирующая — звуковой сигнал.

Универсальность Seq2seq ограничена только вычислительной сложностью и наличием достаточного количества обучающих данных.

Структуры Seq2seq

1. В модели Seq2seq входные данные остаются прежними — это последовательность $x = [x_1, x_2, \dots, x_n]$, но выходные данные являются теперь тоже последовательностью: $y = [y_1, y_2, \dots, y_m]$ длины m .
2. Блок кодера иногда заменяется парой рекуррентных ИНС, каждая из которых проводит анализ входной последовательности в одном из двух направлений: первая в прямом, вторая — в обратном. Позволяет отслеживать влияние последующих элементов x_t .
3. Общее состояние кодера в момент времени t примем конкатенацию двух векторов: $h_t = [\vec{h}_t; \overleftarrow{h}_t]$, где $t = 1, \dots, n$.

Особенности декодера

Пометим начало и конец!

Вводятся два дополнительных зарезервированных значения векторов y_t , одно из которых обозначает начало выходной последовательности, второе — её конец.

Перед тем, как выполнить очередной шаг работы, производится вычисление нового элемента выходной последовательности y_t по следующей формуле:

$$y_t = W_r h_{t-1} + b_r$$

Если y_t оказывается равным значению *stop*, то последовательность считается построенной, и работа алгоритма завершается.

Недостатки рекуррентных ИНС

1. Чрезмерная вычислительная сложность.
2. Сугубо последовательный режим работы.
3. Угасание сигналов и чувствительность к шумам.

Долгая краткосрочная память

Проблема!

Значимая информация попадает в вычисляемую весовую матрицу, но на практике часто оказывается затенённой большим количеством сигналов, составляющими шум.

Решение!

Долгая краткосрочная память (*Long short-term memory, LSTM*) с механизмами “забывания” и “акцентирования” реализуемых путём добавления трёх так называемых вентилей (*gates*): вентиля забывания, входного и выходного вентилей.

Входной вентиль и вентиль забывания

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f)$$

Вычисление вектора контекста в LSTM

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)$$

Вычисление выходного вектора и скрытого состояния

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o)$$

$$h_t = o_t \odot \tanh(c_t)$$

Управляемый рекуррентный блок

Управляемый рекуррентный блок (*Gated recurrent unit, GRU*) — это новая модель рекуррентных ИНС, предложенная в 2014 г. как развитие идей LSTM. При сравнимом качестве заметно проще в реализации. Если в LSTM было три вентиля: забывания, входной и выходной, то в GRU их два: обновления и сброса.

Спасибо за внимание!

Всё о курсе: <https://marigostra.ru/materials/nlp.html>

E-mail: msp@luwrain.org

Канал в Телеграм: <https://t.me/MarigostraRu>