

Вопрос 1

Выполнен

Баллов: 1,00 из 1,00

Сопоставьте вид модулей и стоимость учёта зависимостей длины n . $O(n)$ означает, что нужно порядка $O(n)$ слоёв такого типа, чтобы иметь возможность связать слова, отстоящие друг от друга примерно на n позиций во входном тексте.

"Обычные" свёртки ($dilation = 1$) $O(n/k)$, где k - размер ядра

Рекуррентные модули

 $O(n)$ "Прореженные" свёртки ($dilation = 2^i$) $O(\log(n))$

Ваш ответ верный.

Вопрос 2

Верно

Баллов: 1,00 из 1,00

Механизм self-attention принимает на вход матрицу $Input \in \mathbb{R}^{InLen \times EmbSize}$, где $InLen$ - количество строк, соответствующее длине входной последовательности, а $EmbSize$ - количество столбцов, соответствующее количеству признаков для каждого элемента.

Найдите матрицу попарного сходства элементов $Logits = Input \cdot Input^T$.

Входная матрица имеет следующий вид

$Input = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}$ Результат запишите в виде матрицы, на одной строке - элементы одной строки матрицы, разделённые пробелами.

Нужные операции можно выполнить в `numpy`, используя функцию `dot` или `matmul`, а также оператор матричного умножения `a @ b`.

Ответ: (штрафной режим: 0 %)

```
1 0 1 0
0 1 1 0
1 1 2 0
0 0 0 0
```

	Comment
✓	Верное решение

Прошли все тесты! ✓

Верно

Баллы за эту попытку: 1,00/1,00.

Вопрос 3

Верно

Баллов: 1,00 из 1,00

Используя матрицу $Logits$, полученную в результате решения предыдущей задачи, **найдите выходную матрицу** $Result \in \mathbb{R}^{InLen \times EmbSize}$ **для механизма self-attention.**

Для этого нужно

1. матрицу $Logits$ нормировать с помощью softmax по строкам $AttScores = softmax(Logits, rows)$, в результате $0 \leq AttScores[i, j] \leq 1$ и $\sum_{j=0}^{InLen-1} AttScores[i, j] = 1$
2. найти взвешенную сумму исходных признаков с учётом найденных весов: $Result = AttScores \cdot Input$ (с помощью матричного произведения)

Результат запишите в виде матрицы, на одной строке - элементы одной строки матрицы, разделённые пробелами. В качестве десятичного разделителя используйте точку. Ответ округлите до не менее чем двух знаков после запятой.

Ответ: (штрафной режим: 0 %)

0.73 0.5
0.5 0.73
0.73 0.73
0.5 0.5

	Comment
✓	Верное решение

Прошли все тесты! ✓

Верно

Баллы за эту попытку: 1,00/1,00.

Вопрос 4

Верно

Баллов: 1,00 из 1,00

Давайте теперь посмотрим, как работает более общий вариант self-attention - когда в качестве ключей, запросов и значений используются разные матрицы.

На вход мы получаем всё ту же матрицу

$$Input = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}$$

Общий алгоритм состоит из следующих основных шагов:

1. Найти значения ключей, запросов и значений, используя линейное преобразование

$$Keys = Input \cdot Proj_K + Bias_K$$

$$Queries = Input \cdot Proj_Q + Bias_Q$$

$$Values = Input \cdot Proj_V + Bias_V$$

2. Найти матрицу попарного сходства, используя полученные матричное произведение запросов и ключей

$$Logits = Queries \cdot Keys^T$$

3. Найти коэффициенты усреднения, нормировав матрицу попарного сходства с помощью softmax по строкам

$$AttScores = softmax(Logits, rows)$$

4. Найти результат с помощью матричного произведения матриц значений и коэффициентов

$$Result = AttScores \cdot Values$$

Вам требуется найти значение матрицы $Result \in \mathbb{R}^{InLen \times EmbSize}$ с учётом следующих параметров преобразования:

$$Proj_K = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

$$Proj_Q = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

$$Proj_V = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$Bias_K = Bias_Q = Bias_V = (0 \quad 0)$$

Результат запишите в виде матрицы, на одной строке - элементы одной строки матрицы, разделённые пробелами. В качестве десятичного разделителя используйте точку. Ответ округлите до не менее чем двух знаков после запятой.

Ответ: (штрафной режим: 0 %)

0.5 0.5
0.73 0.5
0.73 0.5
0.5 0.5

	Comment
✓	Верное решение

Прошли все тесты! ✓

Верно

Баллы за эту попытку: 1,00/1,00.

Вопрос 5

Верно

Баллов: 1,00 из 1,00

Ещё один шаг - перейдём от простого self-attention к multihead self-attention.

Общий алгоритм - точно такой же, как и в предыдущей задаче. Отличие в том, что нам нужно несколько раз применить механизм внимания с разными параметрами преобразований $Result^i = SelfAttention(Input, Proj_K^i, Proj_Q^i, Proj_V^i)$,

$$Result^i \in \mathbb{R}^{InLen \times \frac{EmbSize}{HeadsN}}.$$

Результат $MHResult \in \mathbb{R}^{InLen \times EmbSize}$ получается конкатенацией $Result^i$ по столбцам:

$$MHResult = [Result^1, Result^2, \dots, Result^{HeadsN}],$$

Вам требуется найти $MHResult$ для входных данных

$$Input = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix} \text{ с учётом количества "голов" } HeadsN = 2 \text{ и параметров преобразований}$$

$$Proj_K^1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, Proj_K^2 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix},$$

$$Proj_Q^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Proj_Q^2 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix},$$

$$Proj_V^1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, Proj_V^2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$Bias_K^i = Bias_Q^i = \begin{pmatrix} 0 & 0 \end{pmatrix}, \quad Bias_V^i = 0$$

Результат запишите в виде матрицы, на одной строке - элементы одной строки матрицы, разделённые пробелами. В качестве десятичного разделителя используйте точку. Ответ округлите до не менее чем двух знаков после запятой.

Ответ: (штрафной режим: 0 %)

0.5 0.73
0.73 0.73
0.73 0.88
0.5 0.5

	Comment
✓	Верное решение

Прошли все тесты! ✓

Верно

Баллы за эту попытку: 1,00/1,00.