Locality-Sensitive Hashing (Reformer)

Подготовила Курченко Лилия, группа 191

Проблемы трансформеров

- N слоёв – в N раз больше памяти (хранение активации для обратного распространения)
- Промежуточные слои FF часто достаточно большие.

Проблемы трансформеров

- Матрица attention на последовательностях длины L часто требует $O(L^2)$ как в памяти, так и во времени.

Как их решить?

- Реформер

Что нового в реформере?

- скалярное произведение attention -> LSH
  $O(L^2)$ -> O(LlogL)

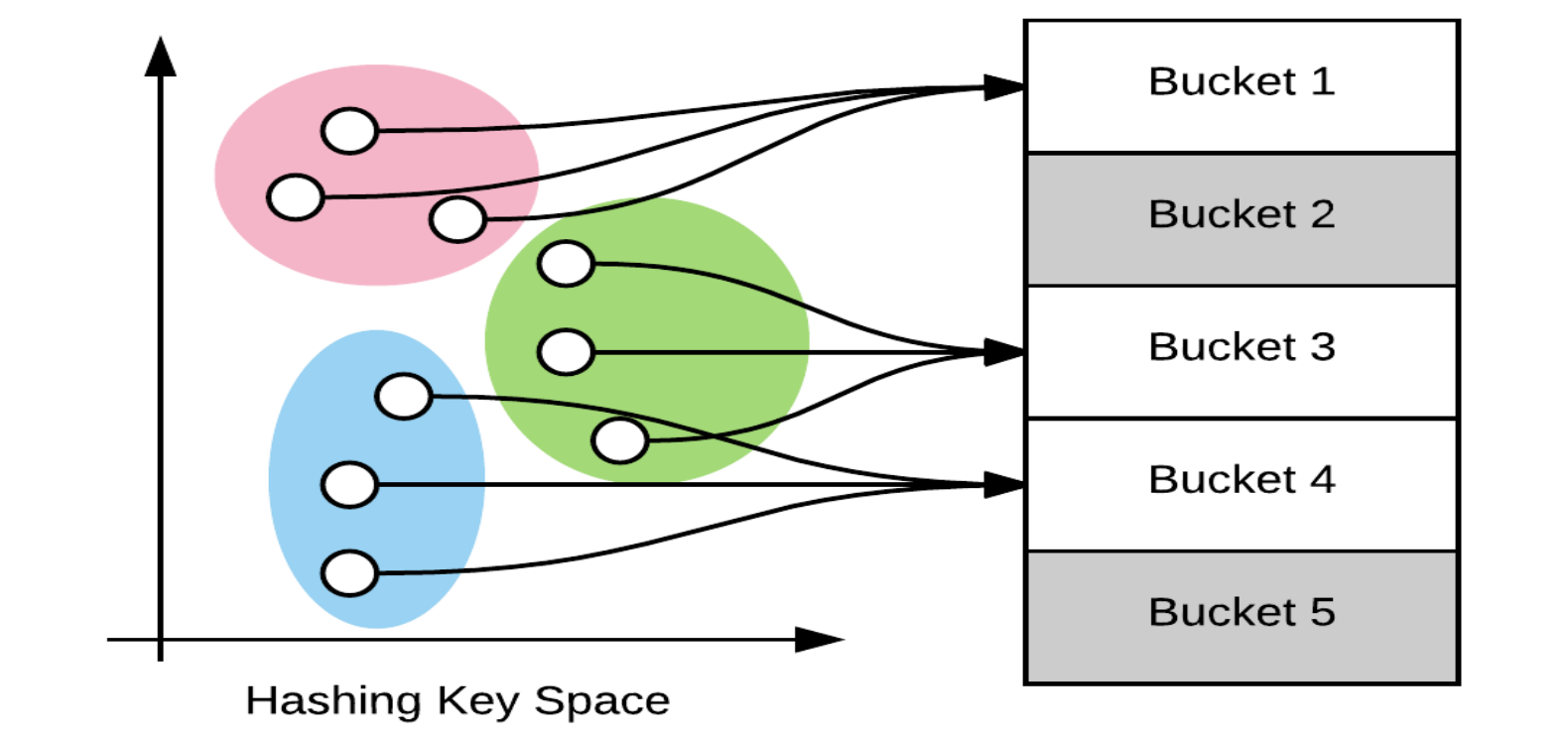
- Стандартные остаточные блоки -> обратимые остаточные слои
  N активаций -> 1 активация сохраняется

Locality-Sensitive Hashing Attention

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}})\mathbf{V}$$

- Нас интересуют только самые большие элементы в QK.
- Для qi∈Q ищем ближайшие к qi строки в K.
- Для быстрого поиска используем Locality-Sensitive Hashing (LSH) в attention

Locality-Sensitive Hashing Attention



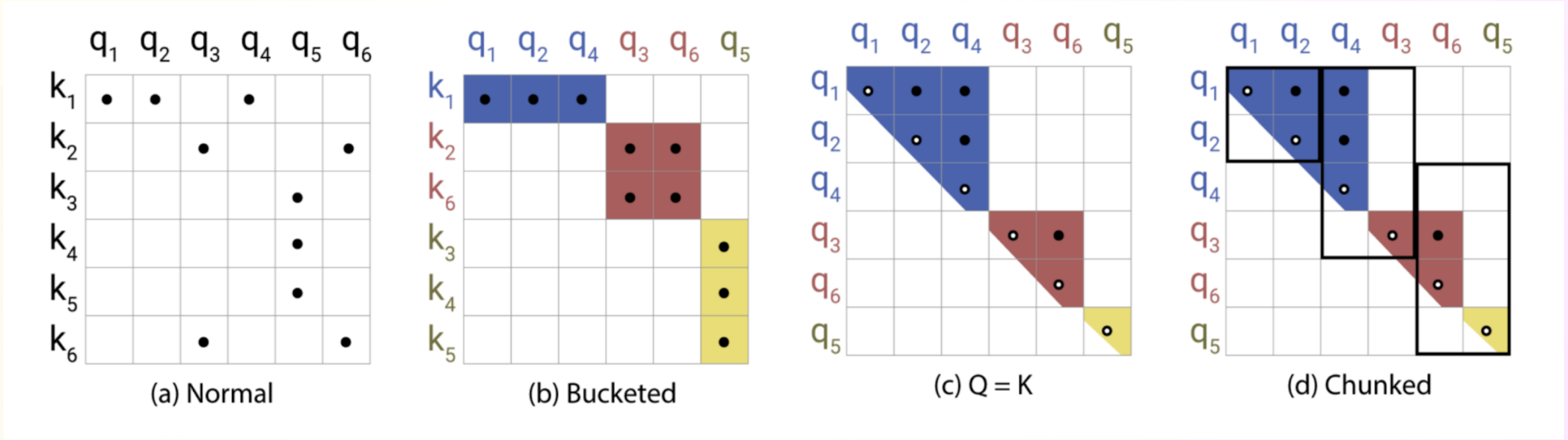
- Схема хэширования x↦h(x) является locality-sensitive, если сохраняет информацию о расстоянии между данными так, что близкие вектора получают одинаковые хэши, а далекие получают совершенно разные

Locality-Sensitive Hashing Attention

- Реформер адаптирует схему хэширования

- Фиксированная случайная матрица $R \in R^{d \times b/2}$, b - гиперпараметр
- $h(x)=\text{argmax}([xR;-xR])$

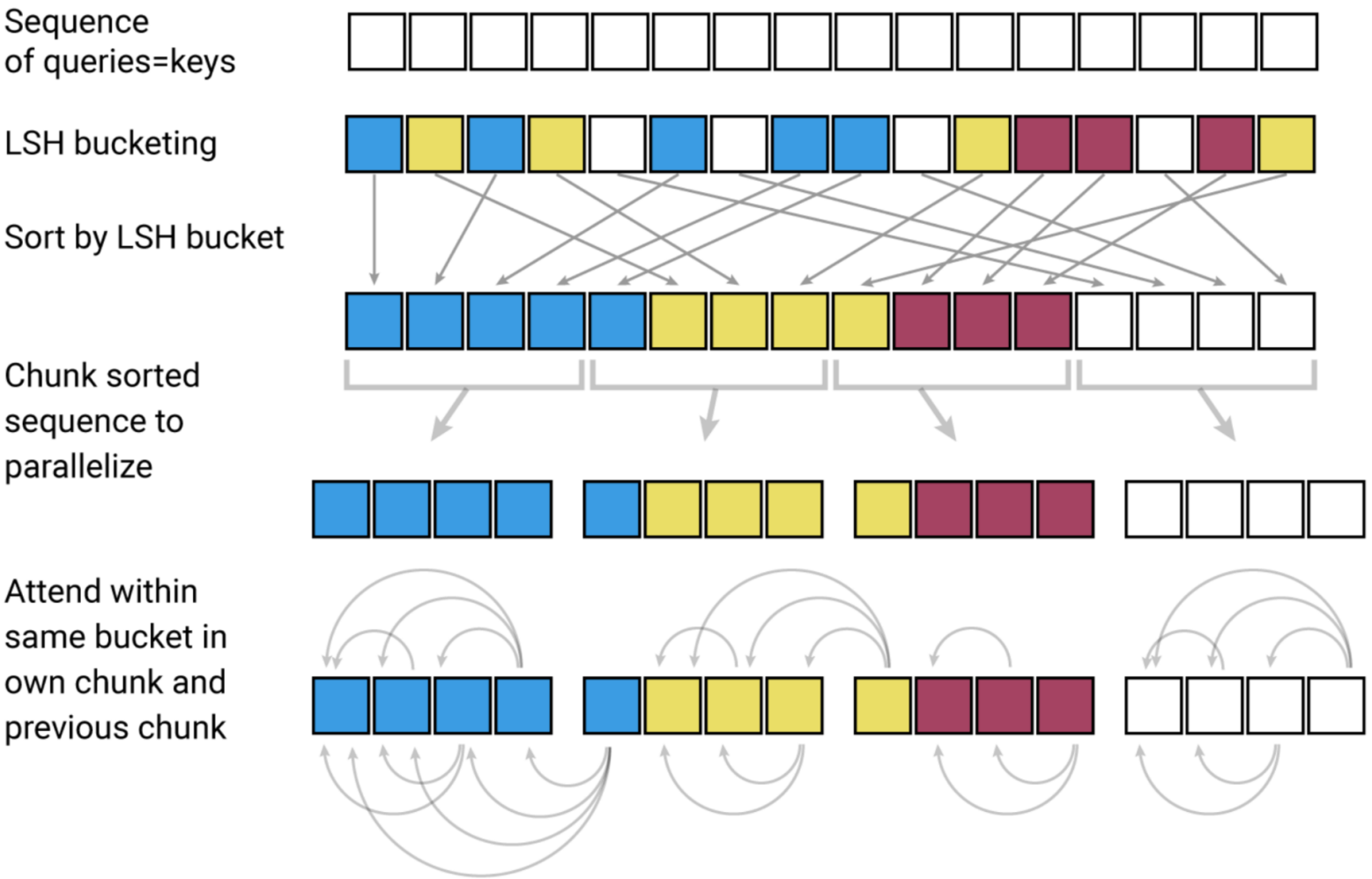
Locality-Sensitive Hashing Attention



Locality-Sensitive Hashing Attention

- Запрос относится только к позициям из той же группы
- Матрица attention для full attention часто разреженная
- Сортируем ключи и запросы относительно их хэш групп.

Locality-Sensitive Hashing Attention



LSH attention:

- группировка,
- сортировка,
- разделение,
- вычисление attention

Обратимые остаточные нейросети

- Слой $x \mapsto y$, normal residual layer: $y=x+F(x)$
- Обратимые слои делят input и output на пары $(x1,x2) \mapsto (y1,y2)$

Вычисляем:

$y1=x1+F(x2),\ y2=x2+G(y1)$

Обратить легко:

$x2=y2-G(y1), x1=y1-F(x2)$

Обратимые остаточные нейросети

- Реформер применяет ту же идею к трансформеру

- Комбинируем attention (F) и слои с прямой связью (G) внутри реверсивного блока:

- $Y1=X1+Attention(X2)$, $Y2=X2+FeedForward(Y1)$

Обратимые остаточные нейросети

- Память может быть еще уменьшена
- Разделим для этого вычисления с прямой связью:

$$Y_2 = [Y^{(1)}_2;\ldots;Y^{(c)}_2] =$$

$$[X^{(1)}_2 + \mathrm{FeedForward}(Y^{(1)}_1); \ldots ; X^{(c)}_2 + \mathrm{FeedForward}(Y^{(c)}_1)]$$

- Полученный обратимый трансформер может не хранить активацию в каждом слое

Результаты:

- Благодаря LSH attention уменьшили сложность до O(LlogL).
- Благодаря обратимости остаточных слоев храним активации только 1 раз за обучение, вместо N