Deep Learning School

Трансформеры

Главные идеи

План лекции

- Трансформеры. Главные идеи:
  - Убираем рекуррентность
  - Используем Multi-head attention:
    - Self-attention
    - Cross-attention
- Трансформеры. Вспомогательные элементы:
  - Word & Position embeddings
  - LayerNorm
  - Dense layers
  - Residual connections
- Трансформеры. Особенности обучения, вариации

План лекции

- **Трансформеры. Главные идеи:**
  - Убираем рекуррентность
  - Используем Multi-head attention:
    - Self-attention
    - Cross-attention
- Трансформеры. Вспомогательные элементы:
  - Word & Position embeddings
  - LayerNorm
  - Dense layers
  - Residual connections
- Трансформеры. Особенности обучения, вариации

Трансформеры. Главные идеи

Недостатки рекуррентных моделей:

- 🥱 Медленные
- 😠 Плохо утилизируют GPU/TPU
- 🤔 Проблемы с обучением
- 🥴 Проблемы с памятью

Трансформеры. Главные идеи

💡 Attention Is All You Need 💡

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

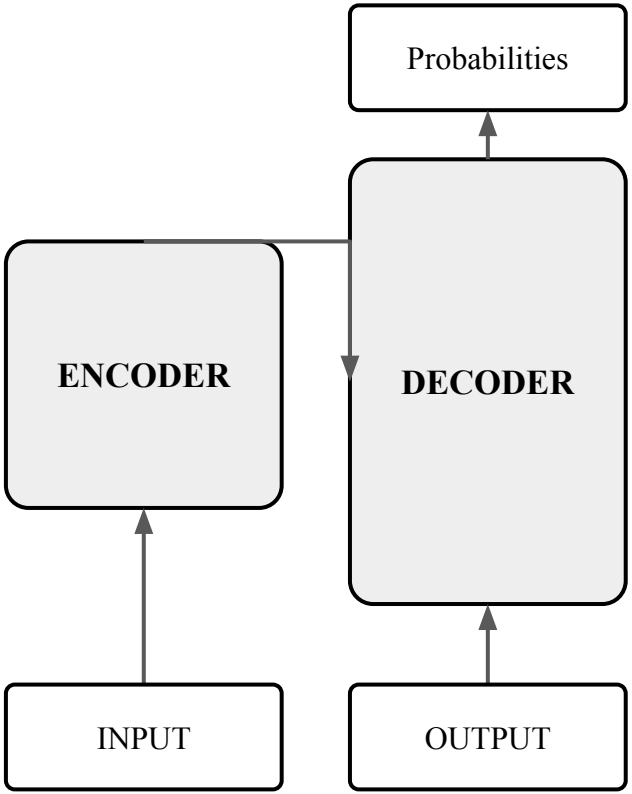
Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

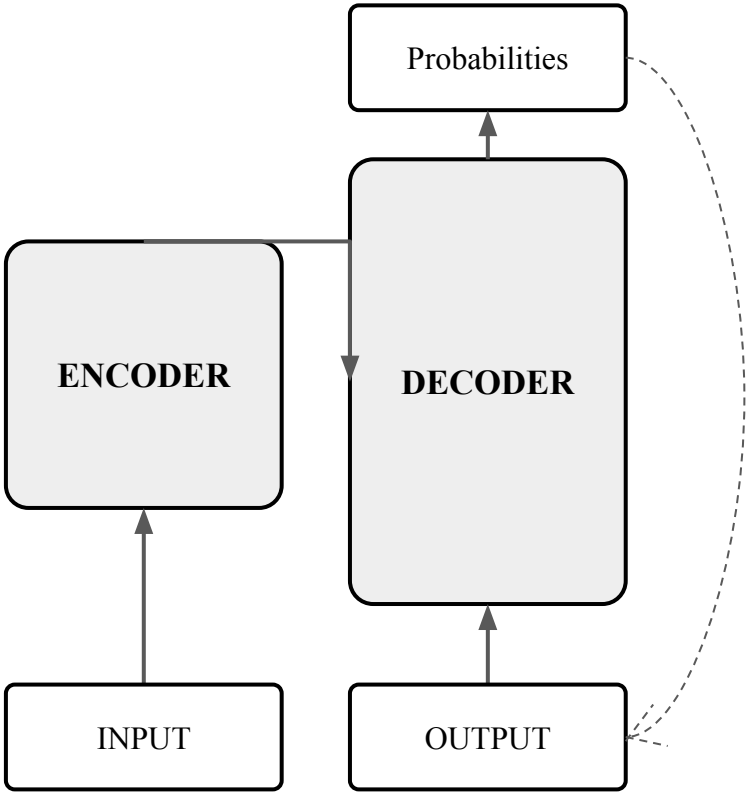
Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Трансформеры. Главные идеи

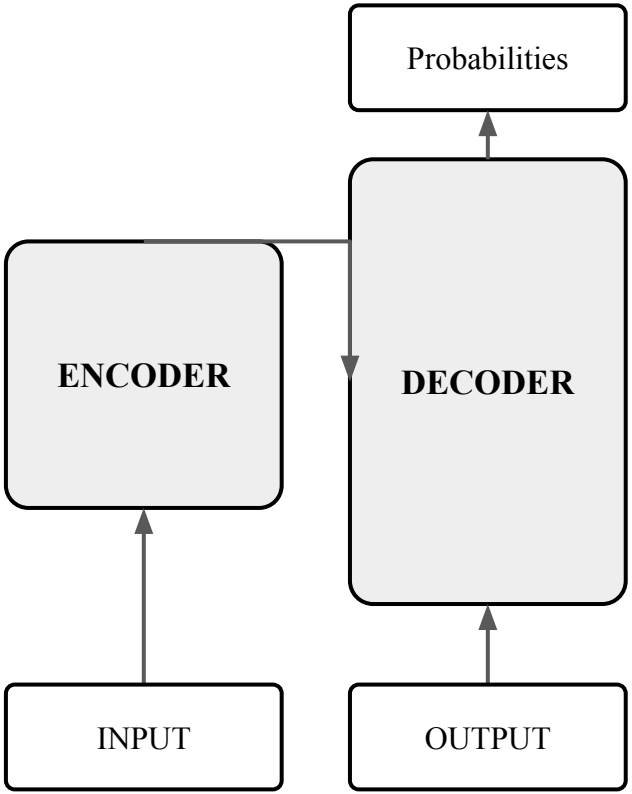


Трансформеры. Главные идеи



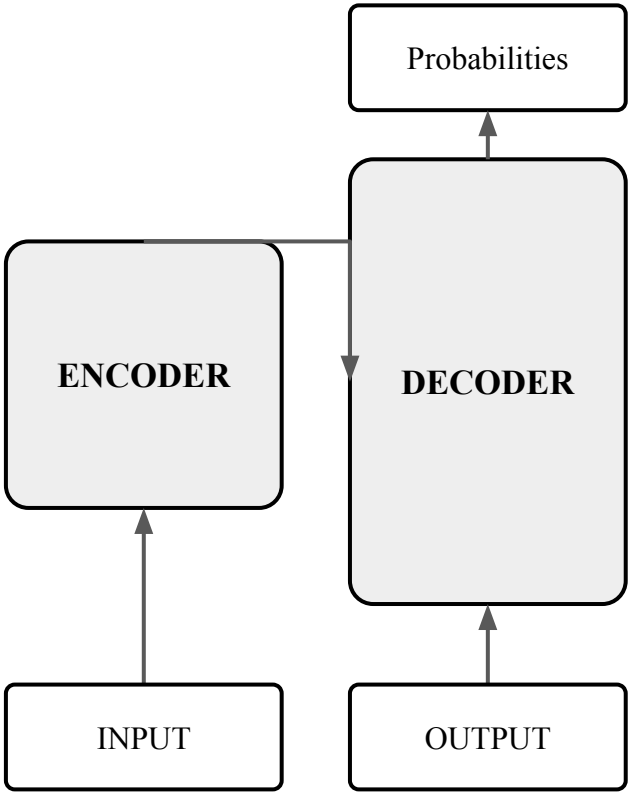
Трансформеры. Главные идеи

В энкодере (слева) и декодере (справа) живут разные виды attention



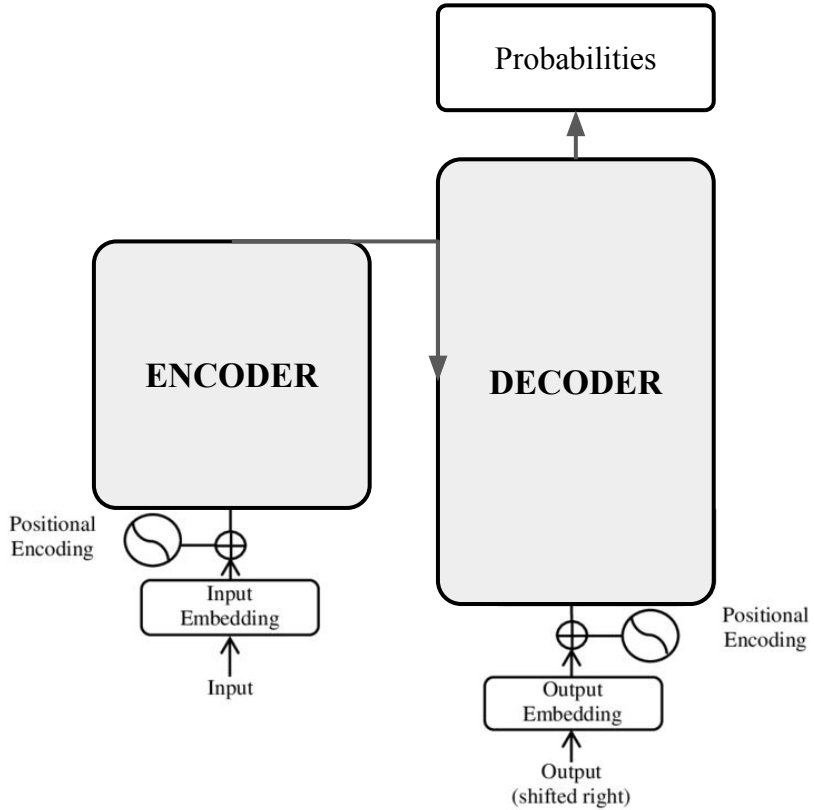
Трансформеры. Главные идеи

В энкодере (слева) и декодере (справа) живут разные виды attention



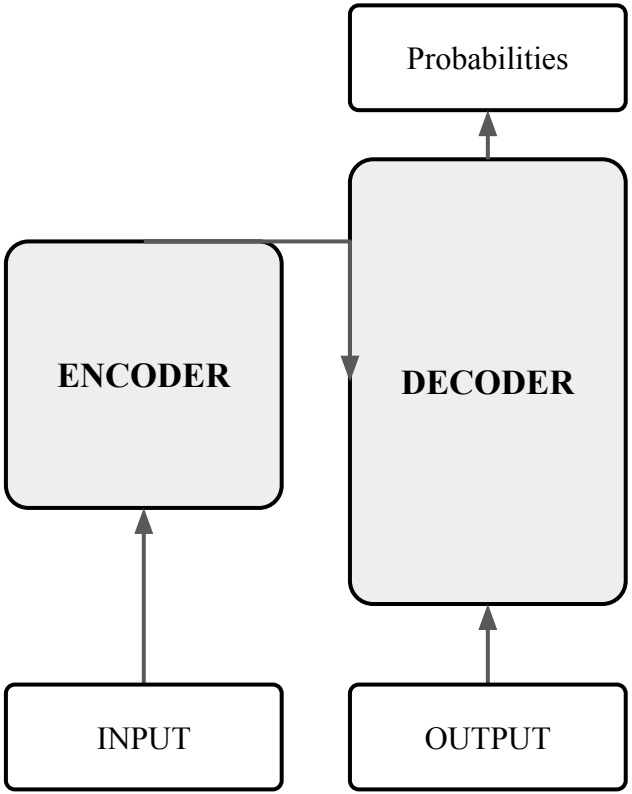
RNN-блоки больше не используются

Трансформеры. Главные идеи

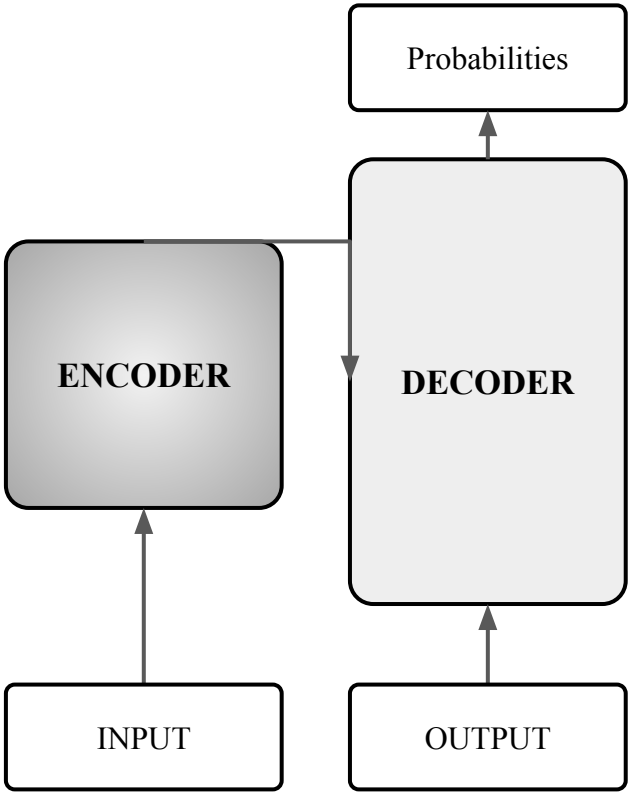


RNN-блоки больше не используются

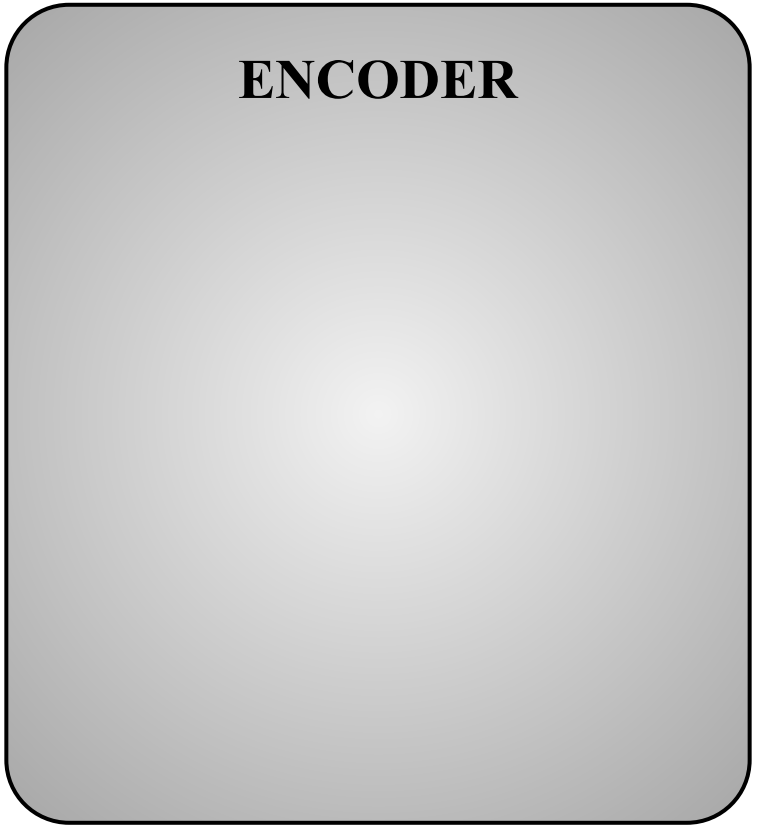
Трансформеры. Главные идеи



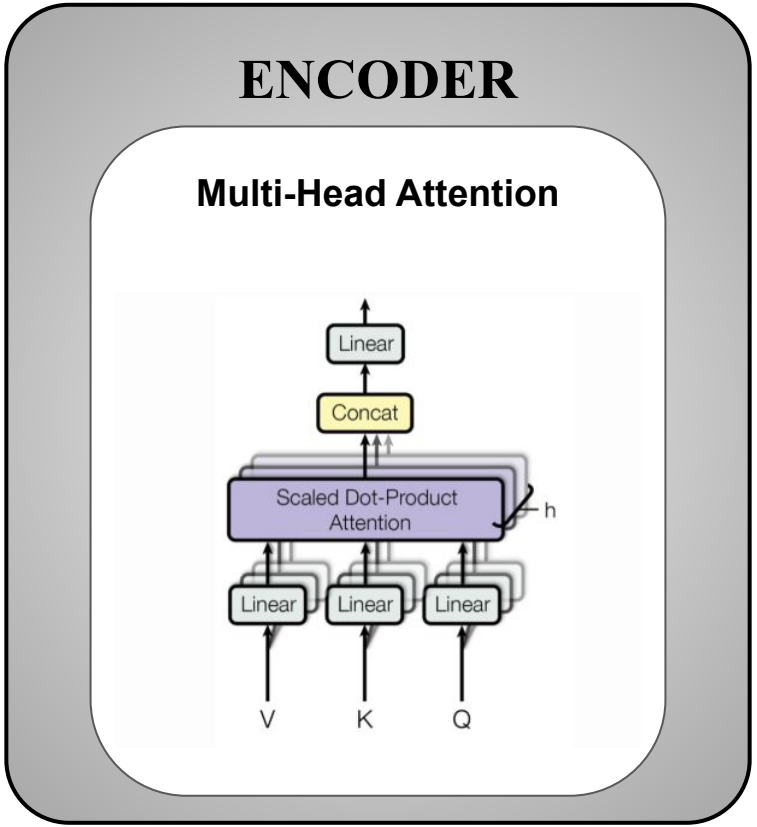
Трансформеры. Главные идеи



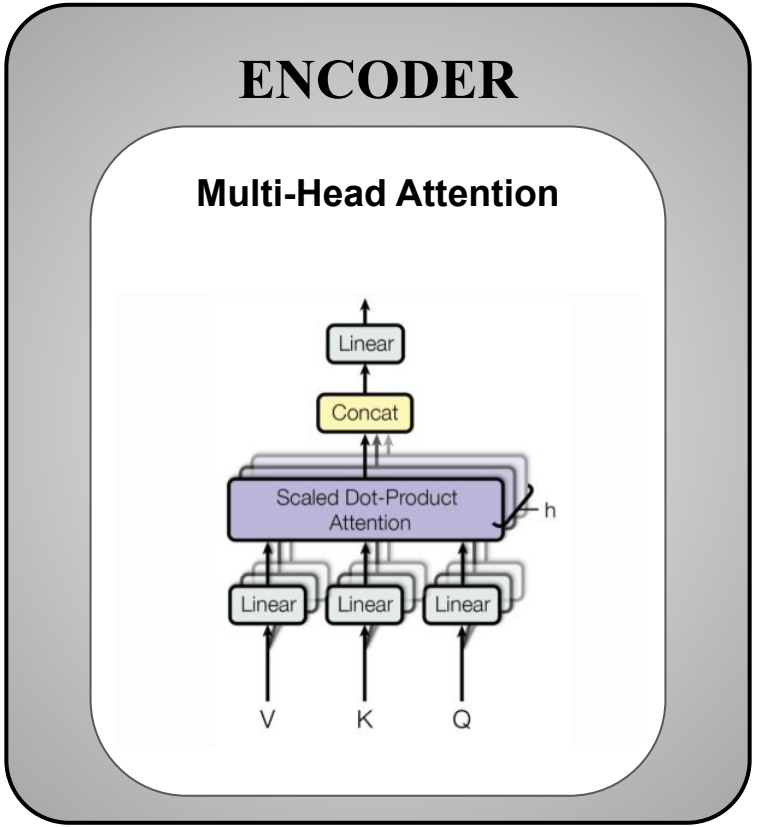
Трансформеры. Главные идеи



Трансформеры. Главные идеи

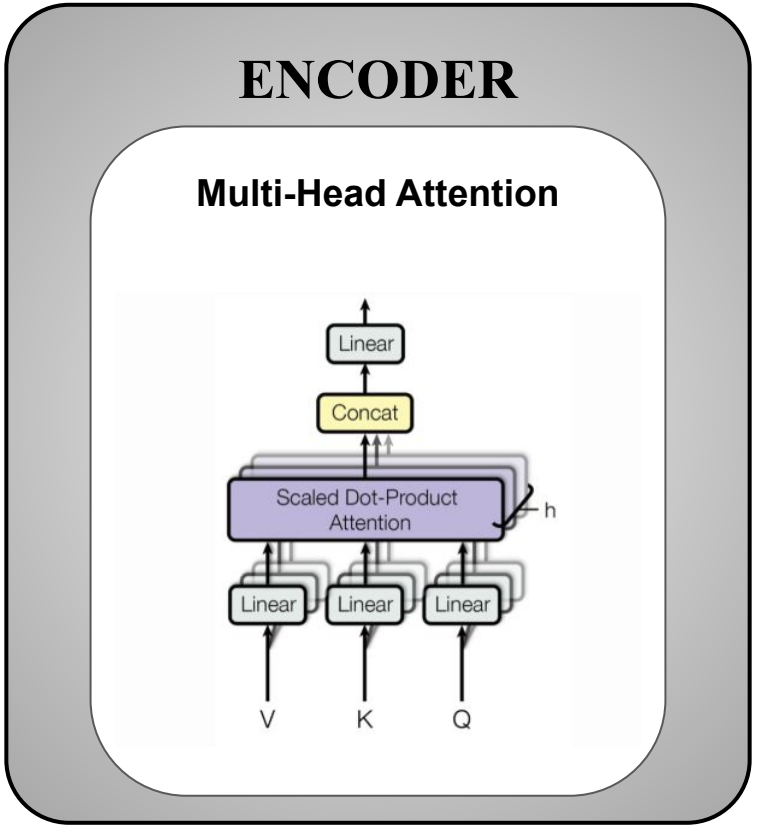


Трансформеры. Главные идеи



Блок Multi Head Attention для **энкодера**:

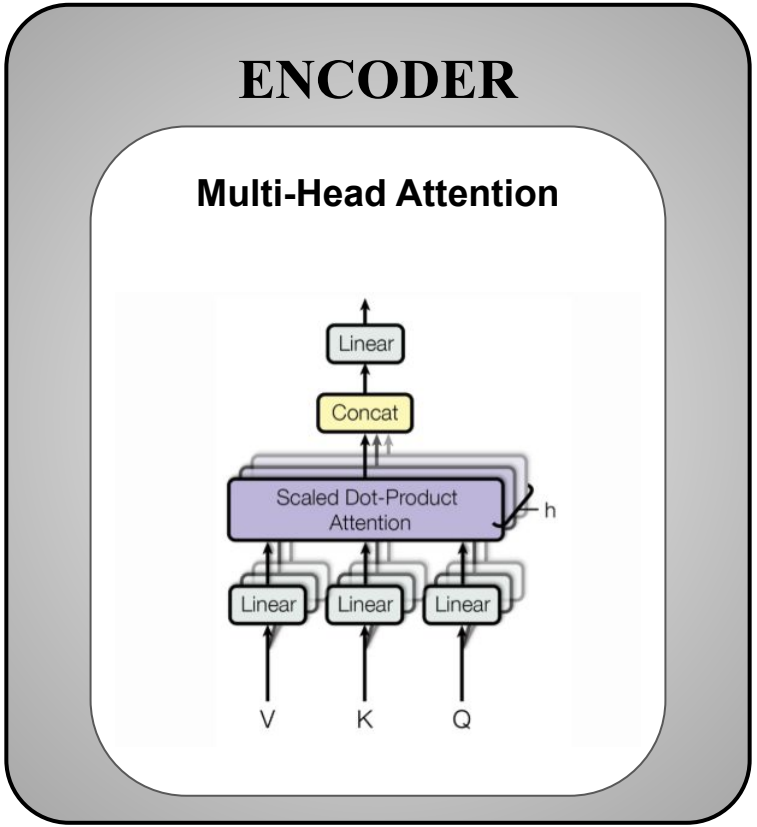
Трансформеры. Главные идеи



Блок Multi Head Attention для **энкодера**:

- H голов внимания

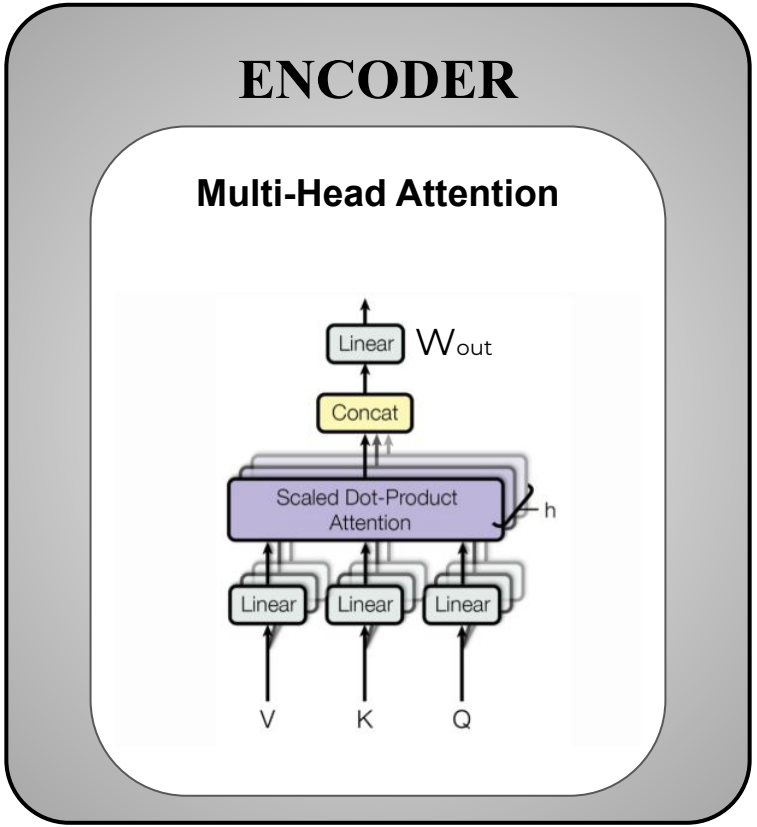
Трансформеры. Главные идеи



Блок Multi Head Attention для **энкодера**:

- H голов внимания
- Каждая голова содержит веса:
  - Матриц проекций W_V, W_K, W_Q
  - Выходной матрицы W_{out}

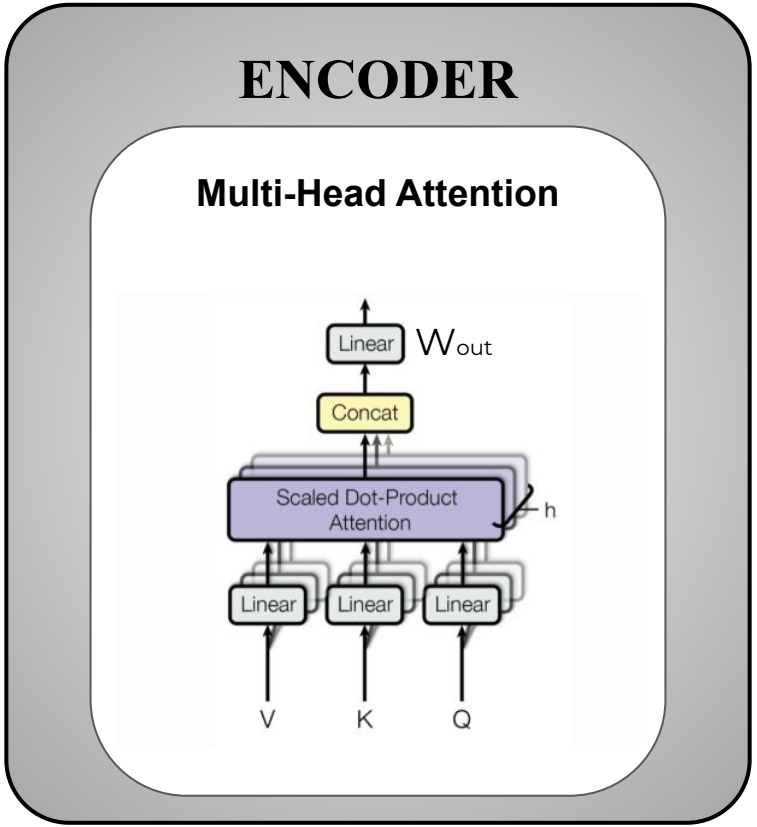
Трансформеры. Главные идеи



Блок Multi Head Attention для **энкодера**:

- H голов внимания
- Каждая голова содержит веса:
  - Матриц проекций W_V, W_K, W_Q
  - Выходной матрицы W_{out}

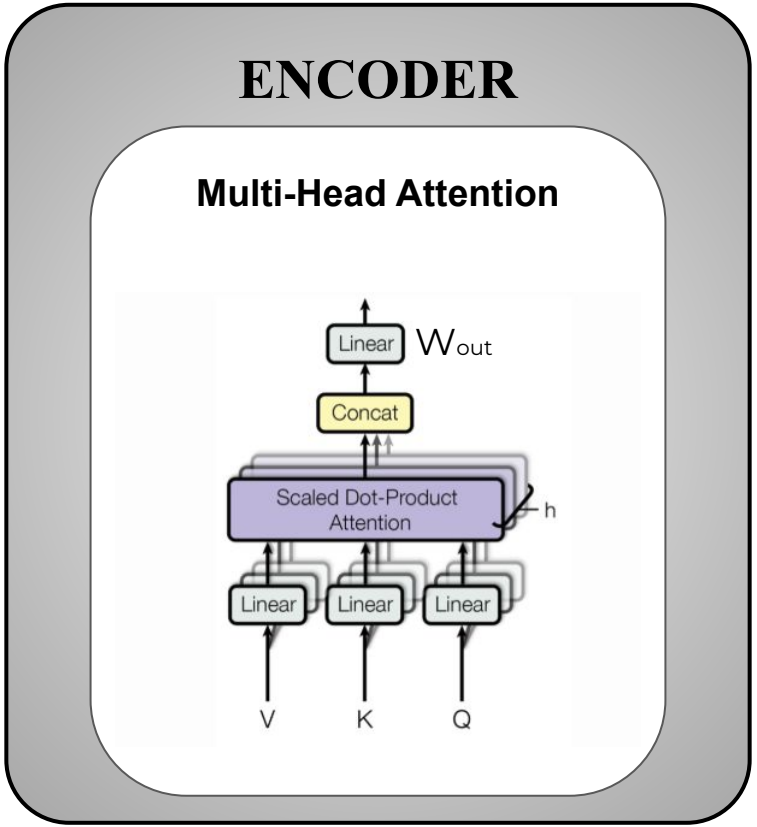
Трансформеры. Главные идеи



Блок Multi Head Attention для **энкодера**:

- H голов внимания
- Каждая голова содержит веса:
  - Матриц проекций W_V, W_K, W_Q
  - Выходной матрицы W_{out}
- Scaled Dot-Product Attention:
  - Для **энкодера** это Self-attention

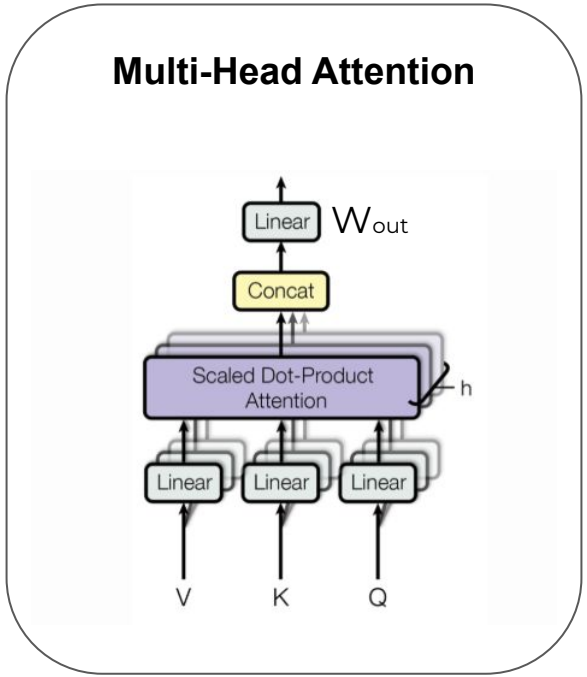
Трансформеры. Главные идеи



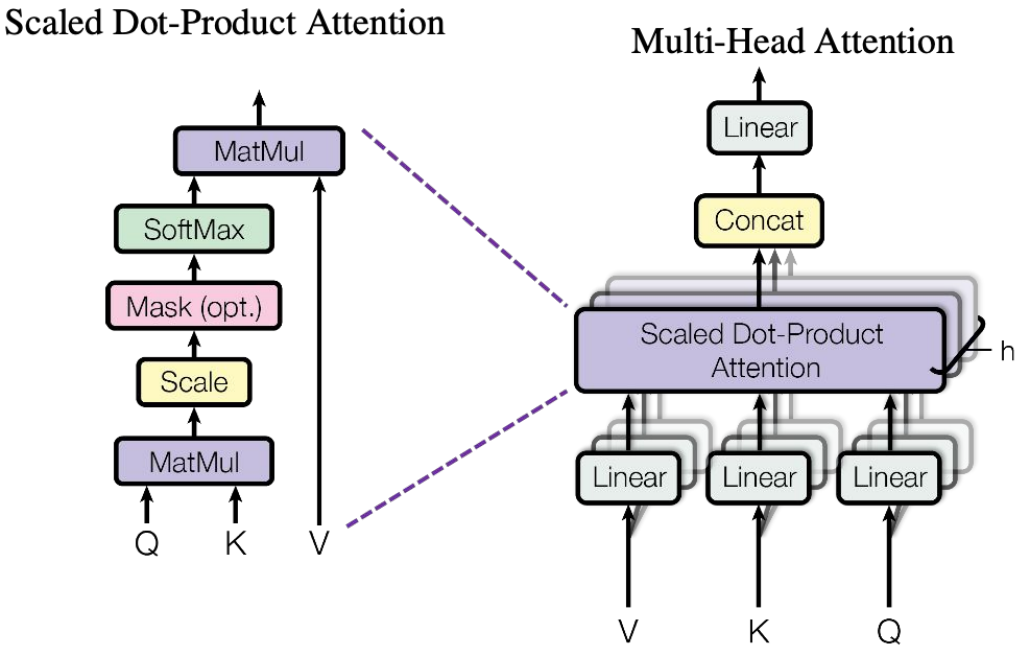
Блок Multi Head Attention для **энкодера**:

- H голов внимания
- Каждая голова содержит веса:
  - Матриц проекций W_V, W_K, W_Q
  - Выходной матрицы W_{out}
- Scaled Dot-Product Attention:
  - Для **энкодера** это Self-attention
- Головы работают вместе, как ансамбль! Каждая голова выучивает свое независимое преобразование, которое вносит вклад в общее решение.

Трансформеры. Главные идеи

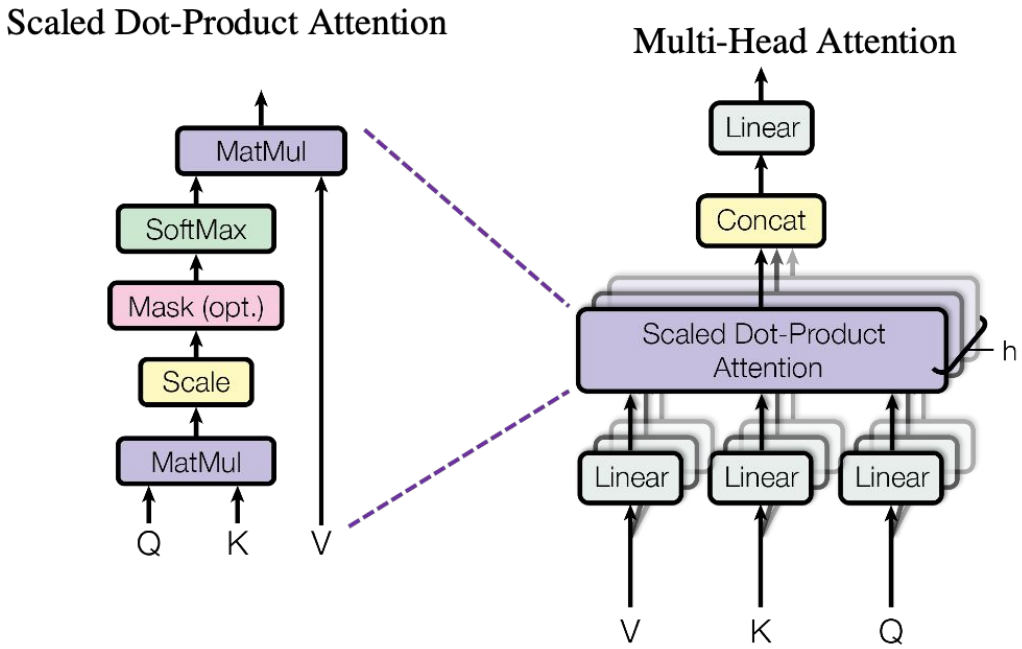


Трансформеры. Главные идеи



Трансформеры. Главные идеи

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



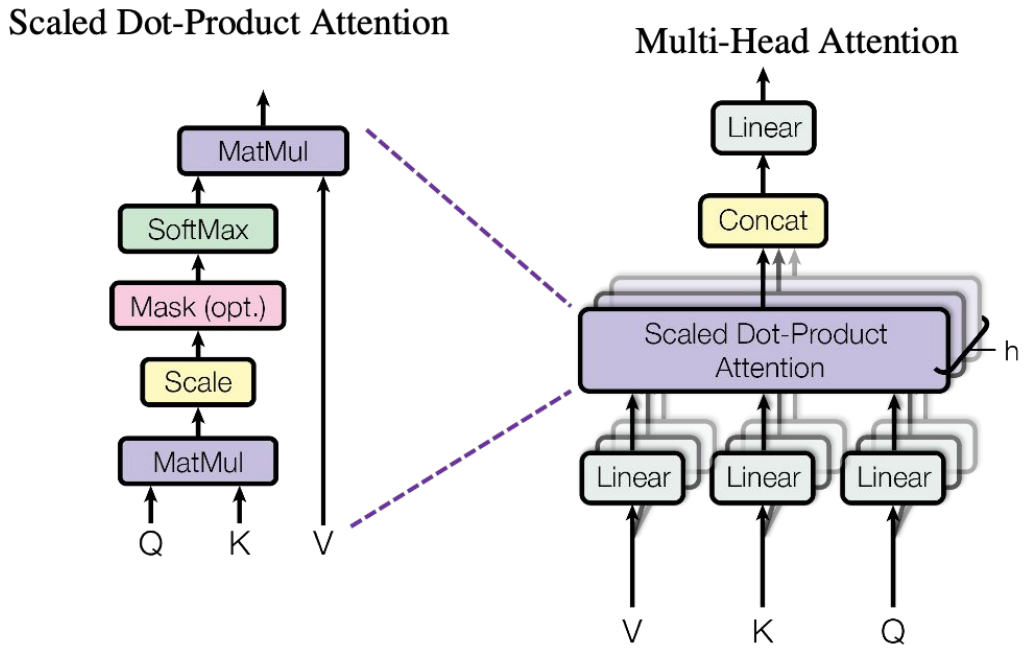
Трансформеры. Главные идеи

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

- **Self-attention:**

$$X^{\text{out}} = W^{\text{attn}}(XW^{\text{V}})$$

$$\text{with } W^{\text{attn}} = \text{softmax}\left(\frac{(XW^{\text{Q}})(XW^{\text{K}})^{\text{T}}}{\sqrt{d}}\right)$$



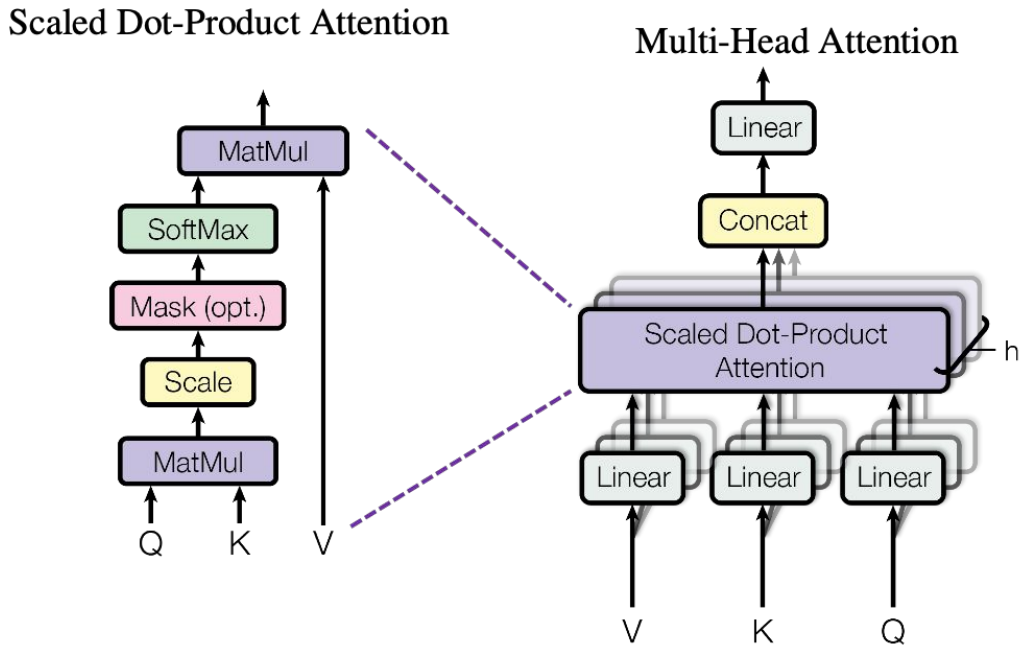
Трансформеры. Главные идеи

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

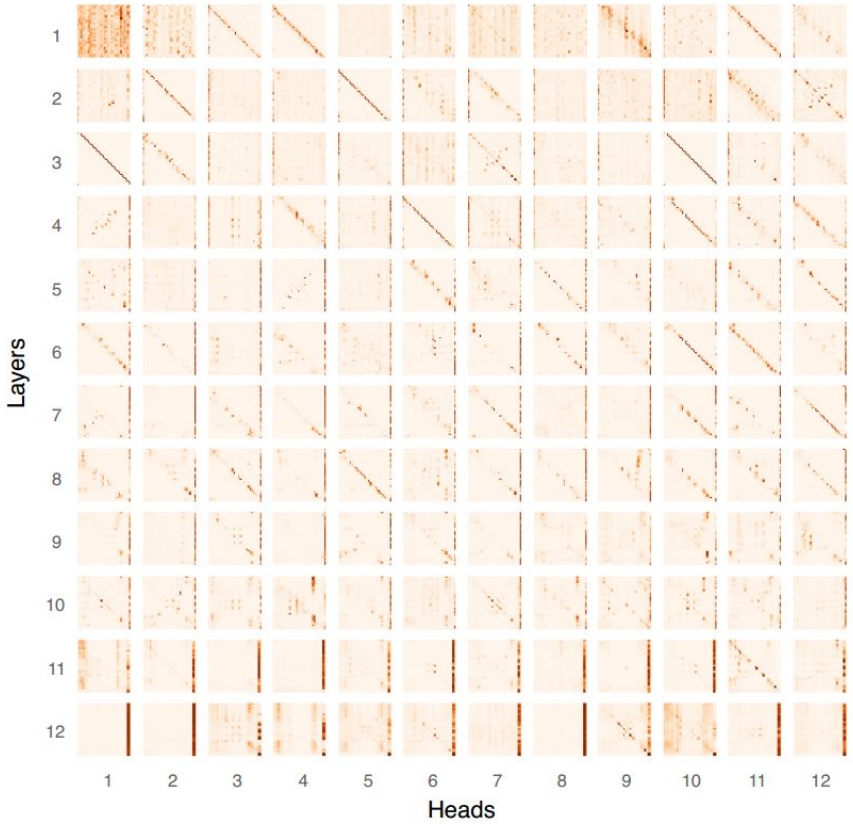
- **Self-attention:**

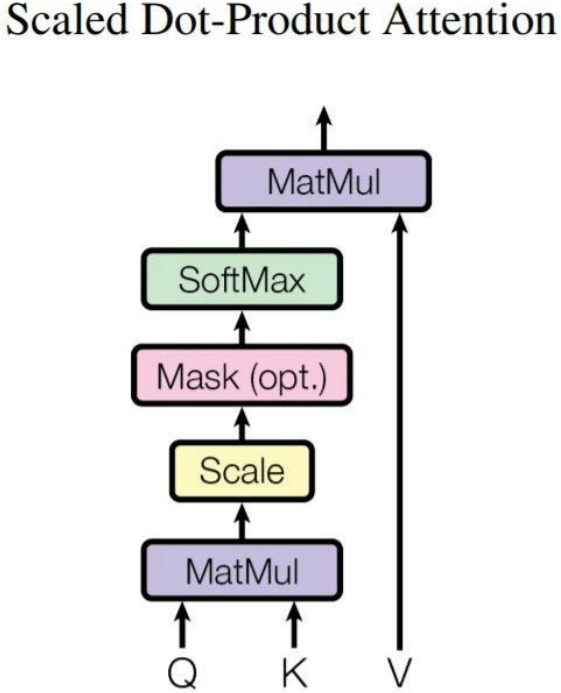
$$X^{\text{out}} = W^{\text{attn}}(\overbrace{XW^{\text{V}}}^{\textbf{V}})$$

$$\text{with } W^{\text{attn}} = \text{softmax}\left(\frac{(\overbrace{XW^{\text{Q}}}^{\textbf{Q}})(\overbrace{XW^{\text{K}}}^{\textbf{K}})^{\text{T}}}{\sqrt{d}}\right)$$

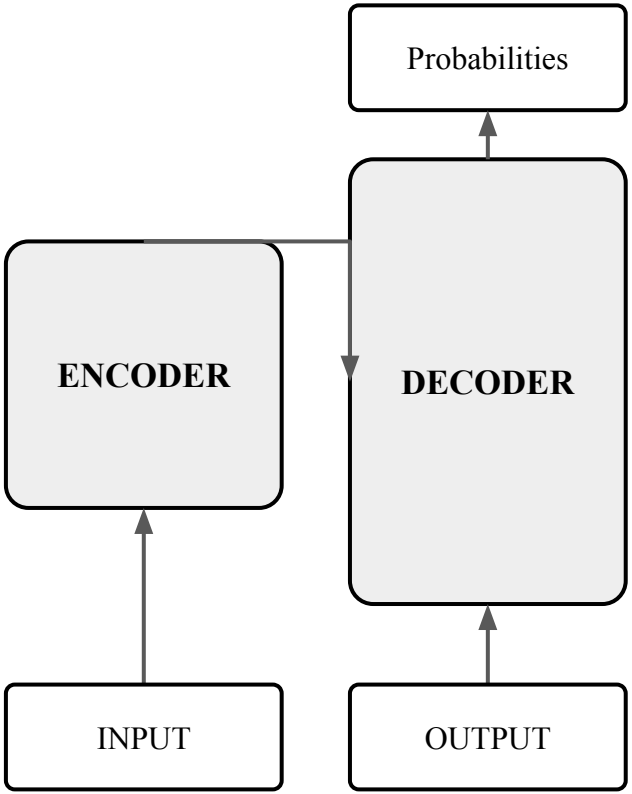


Трансформеры. Главные идеи

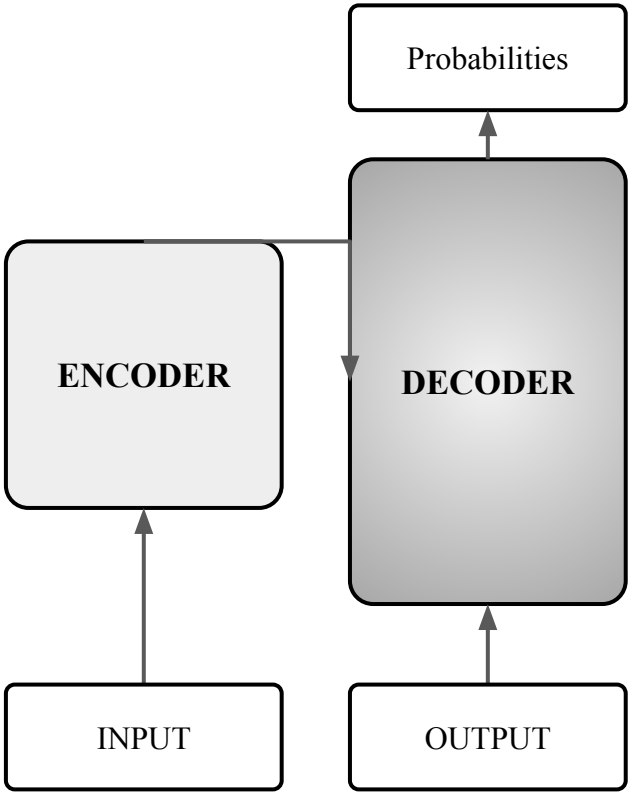




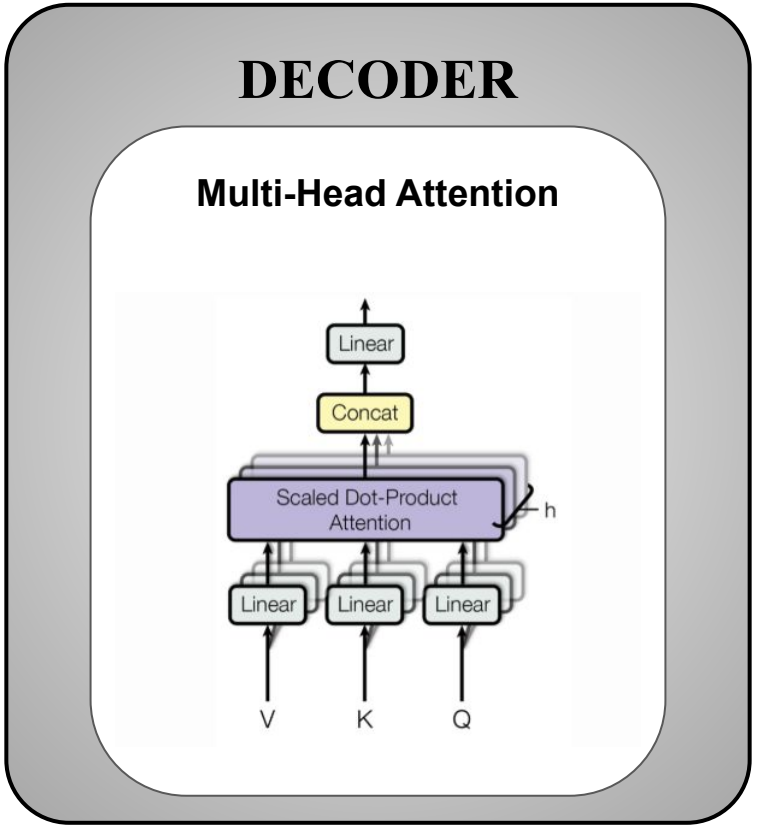
Трансформеры. Главные идеи



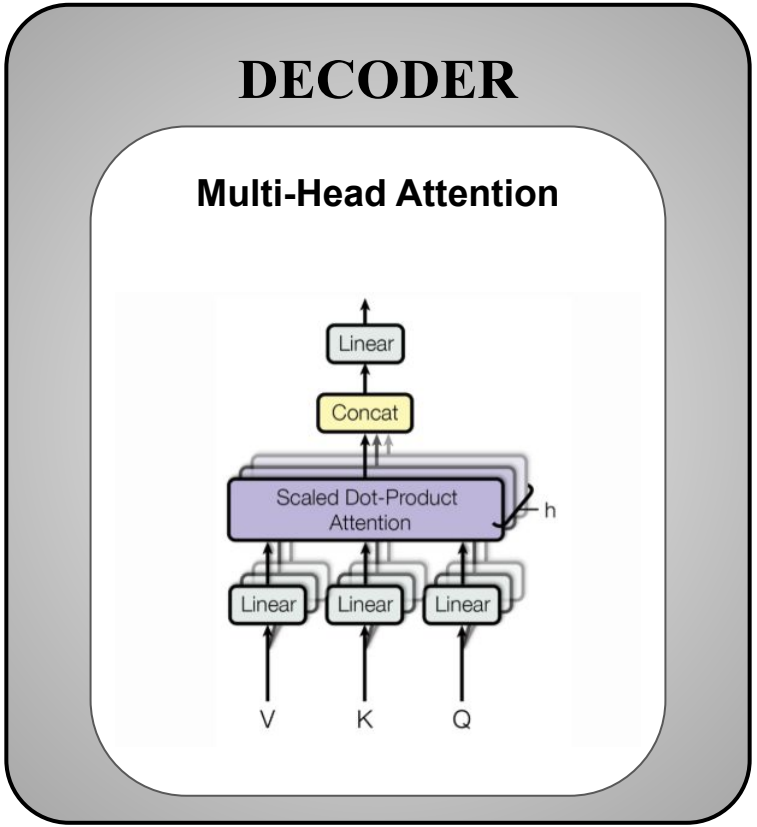
Трансформеры. Главные идеи



Трансформеры. Главные идеи



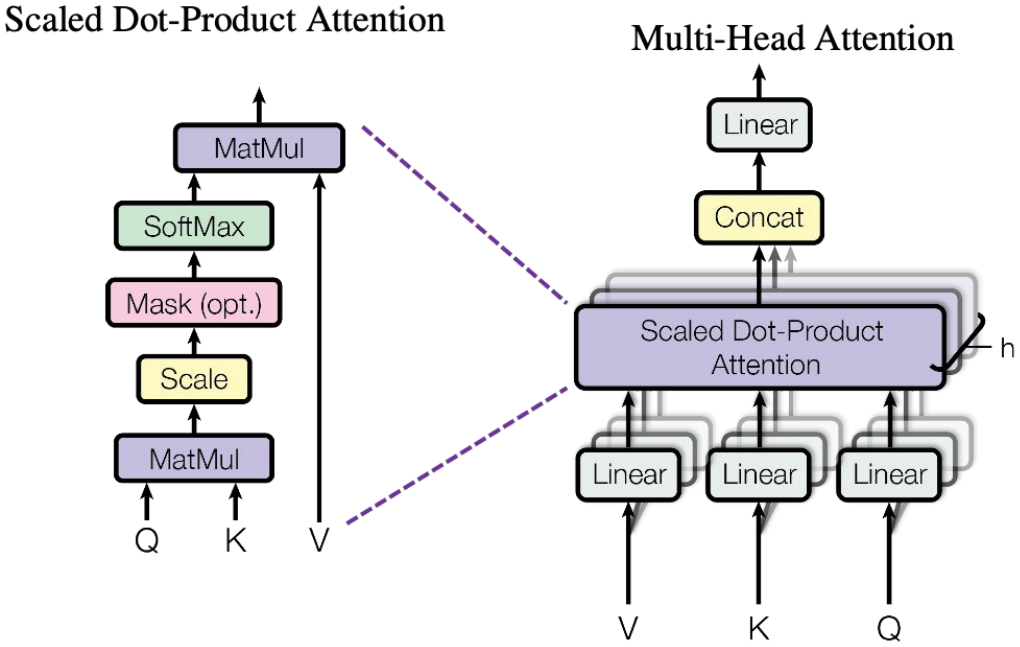
Трансформеры. Главные идеи



Блок Multi Head Attention для **декодера**:

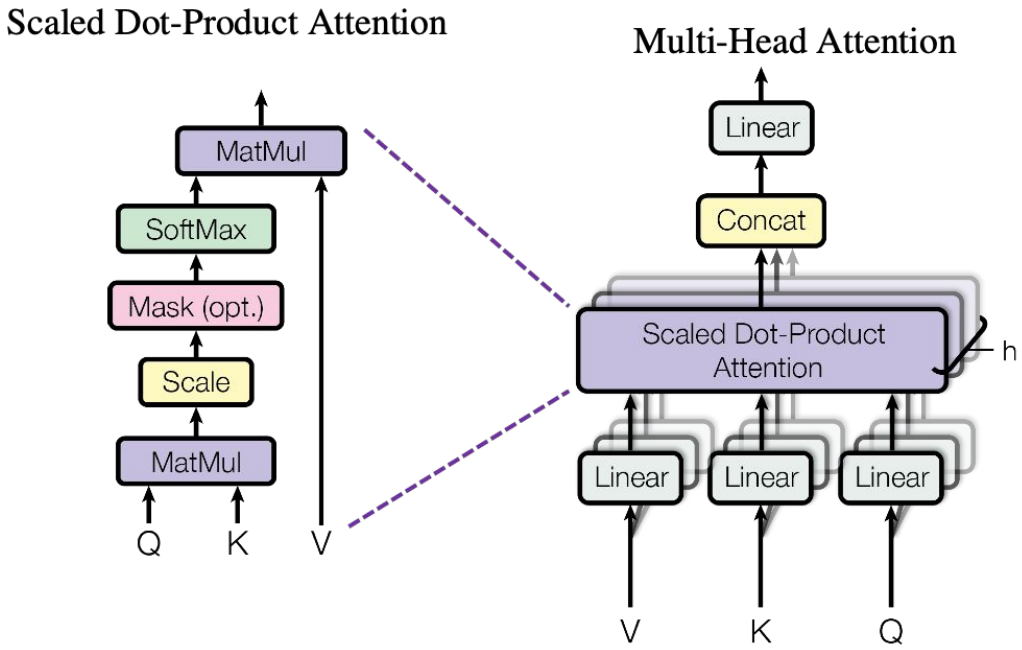
- H голов внимания
- Каждая голова содержит веса:
  - Матриц проекций W_V, W_K, W_Q
  - Выходной матрицы W_{out}
- Scaled Dot-Product Attention:
  - Содержит слои с **Cross-attention,** где смешивается информация из input и output

Трансформеры. Главные идеи



Трансформеры. Главные идеи

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



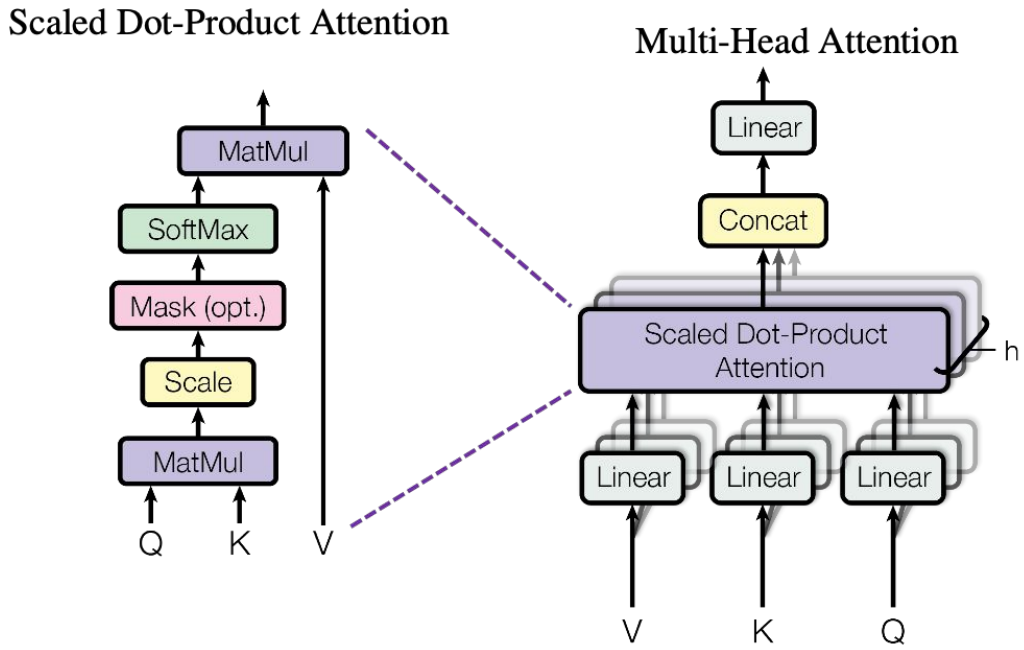
Трансформеры. Главные идеи

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

- **Cross-attention:**

$$X^{\text{out}} = W^{\text{attn}}(X'W^{\text{V}})$$

$$\text{with } W^{\text{attn}} = \text{softmax}\left(\frac{(XW^{\text{Q}})(X'W^{\text{K}})^{\text{T}}}{\sqrt{d}}\right)$$



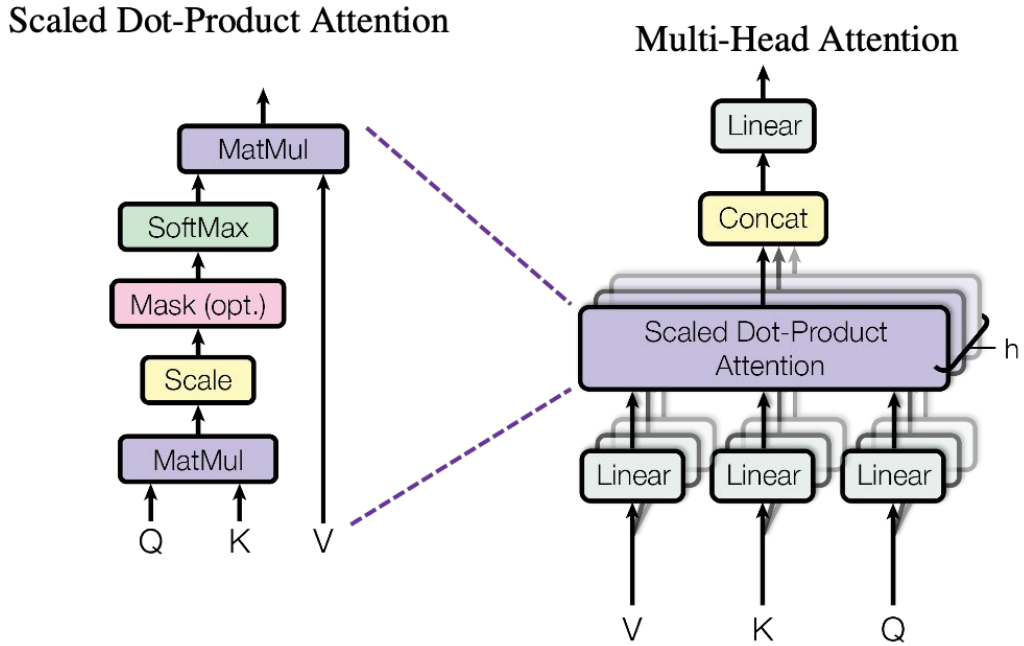
Трансформеры. Главные идеи

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

- **Cross-attention:**

$$X^{\text{out}} = W^{\text{attn}}(X'W^{\text{V}})$$

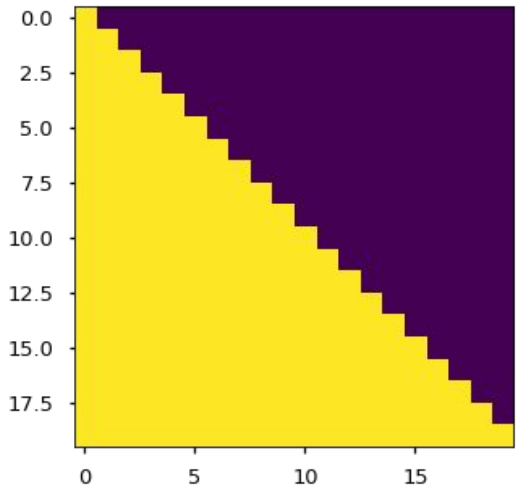
$$\text{with } W^{\text{attn}} = \text{softmax}\left(\frac{(XW^{\text{Q}})(X'W^{\text{K}})^{\text{T}}}{\sqrt{d}}\right)$$

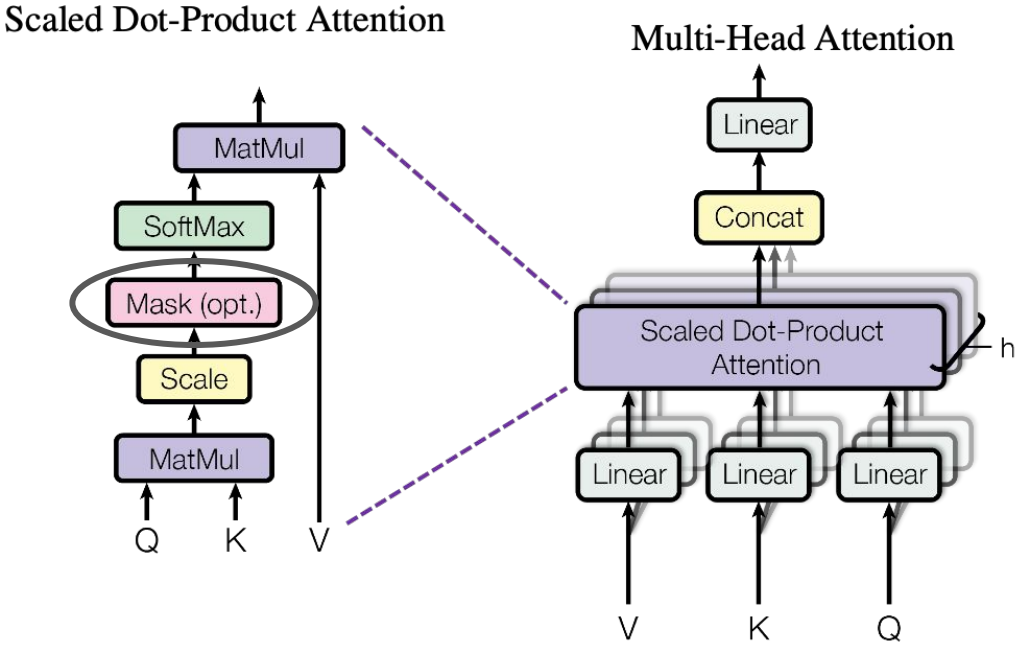


- X - преобразованный input (с выхода энкодера)
- X' - преобразованный output (с выхода предыдущего слоя декодера)

Трансформеры. Главные идеи

- **Masking:**





Итоги видео

В этом видео мы познакомились с основной идеей и базовой схемой трансформера.

В следующих видео мы более подробно разберемся в устройстве и обучении трансформерных моделей.

Deep Learning School

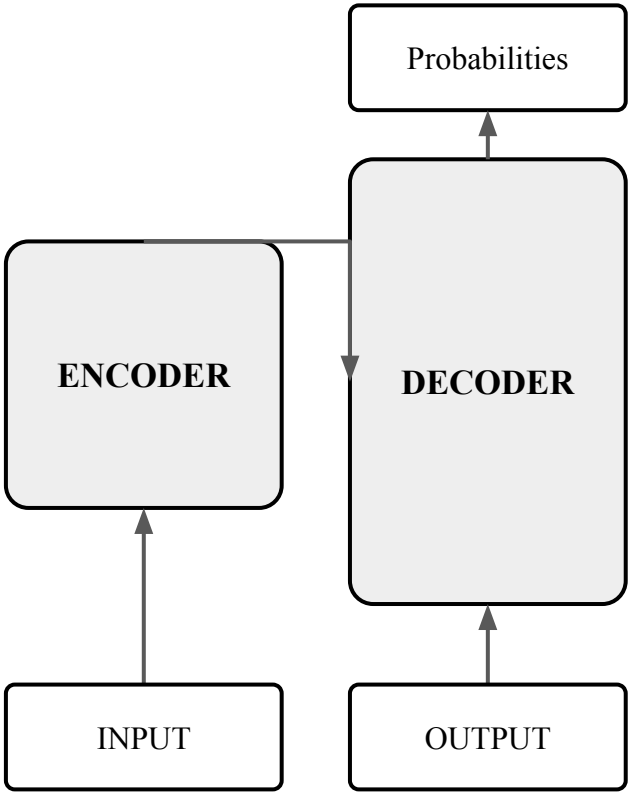
Трансформеры

Вспомогательные элементы

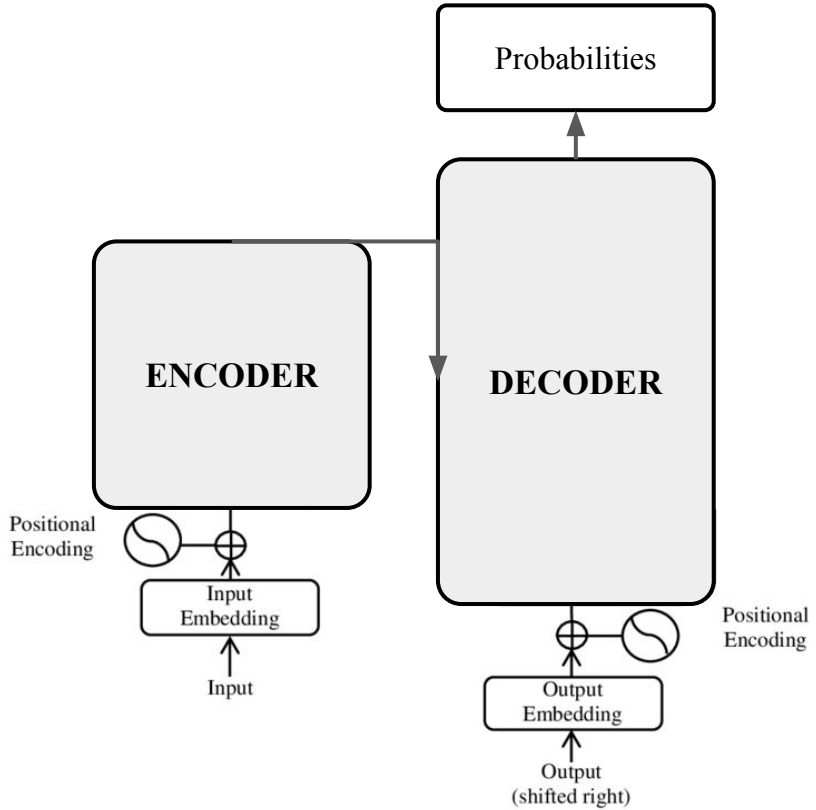
План лекции

- Трансформеры. Главные идеи:
  - Убираем рекуррентность
  - Используем Multi-head attention:
    - Self-attention
    - Cross-attention
- **Трансформеры. Вспомогательные элементы:**
  - Word & Position embeddings
  - LayerNorm
  - Dense layers
  - Residual connections
- Трансформеры. Особенности обучения, вариации

Трансформеры. Вспомогательные элементы

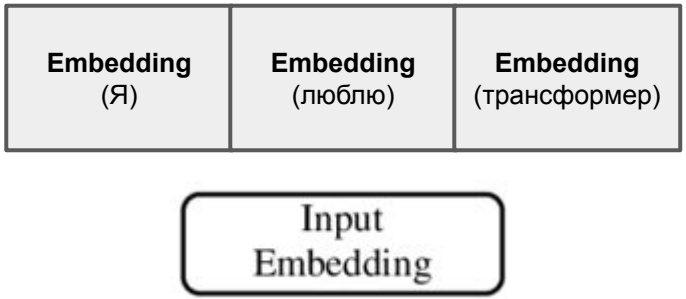


Трансформеры. Вспомогательные элементы

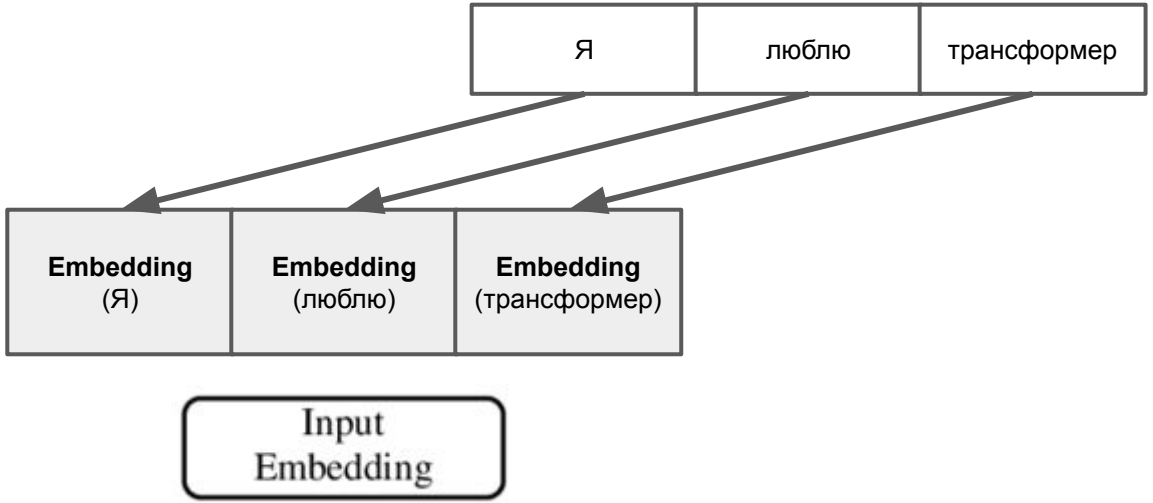


Трансформеры. Вспомогательные элементы

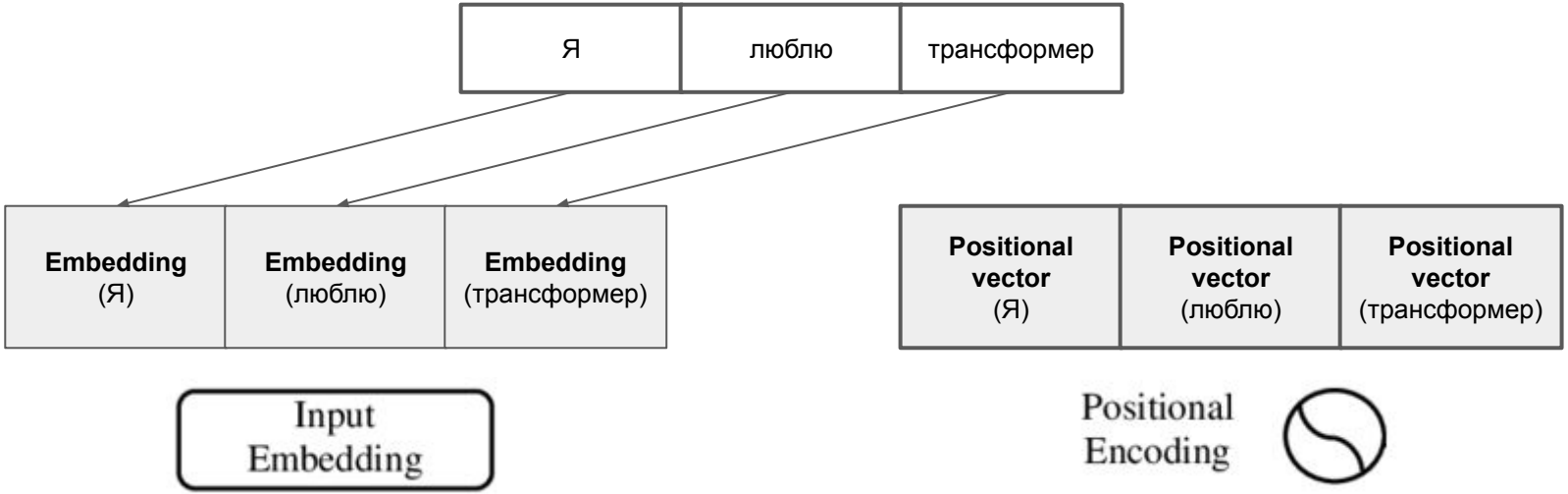
Трансформеры. Вспомогательные элементы



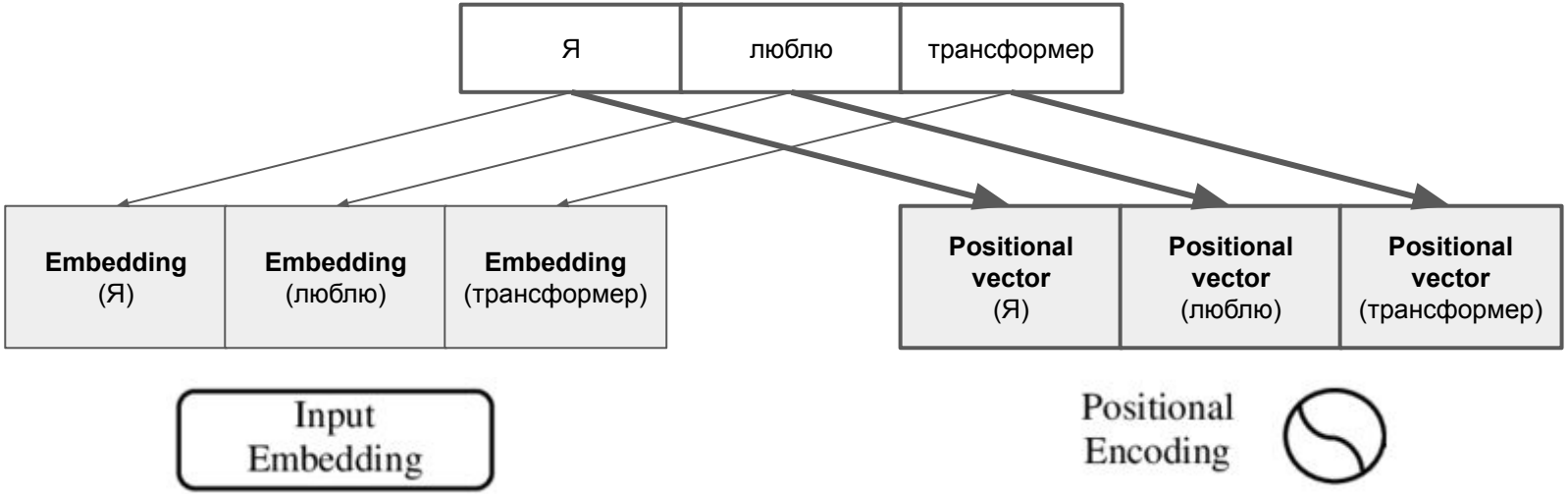
Трансформеры. Вспомогательные элементы



Трансформеры. Вспомогательные элементы



Трансформеры. Вспомогательные элементы



Трансформеры. Вспомогательные элементы

Я	люблю	трансформер

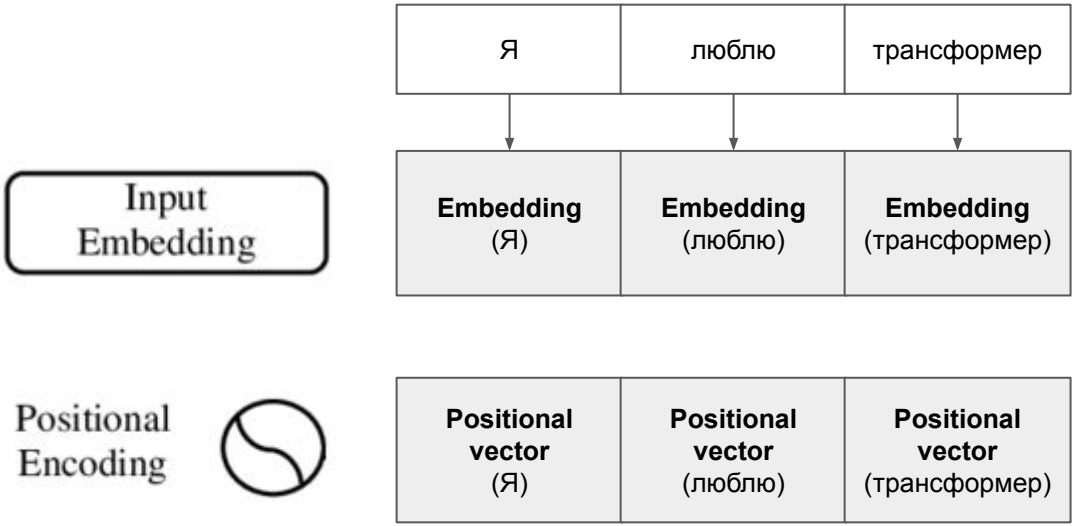
Input Embedding

Embedding (Я)	**Embedding** (люблю)	**Embedding** (трансформер)

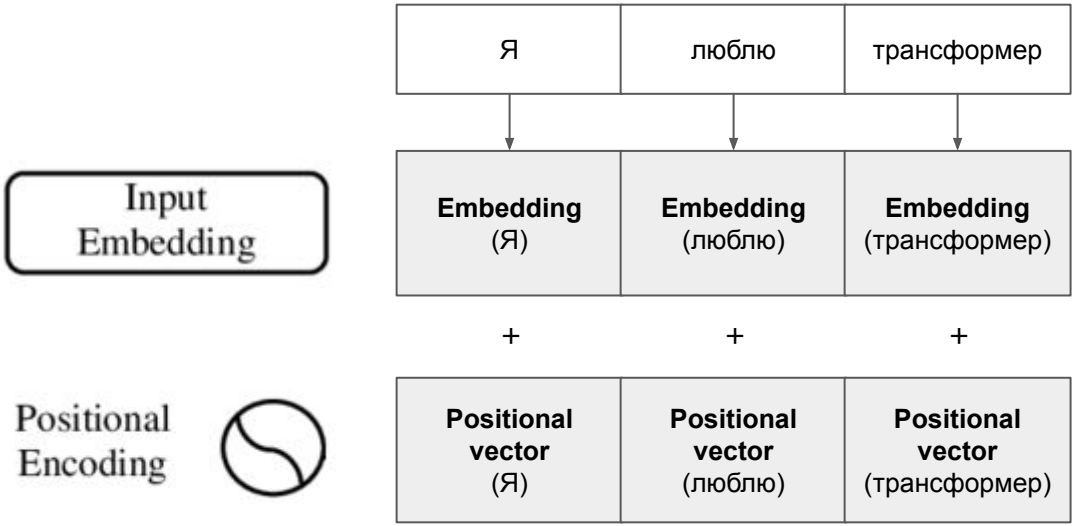
Positional Encoding

Positional vector (Я)	**Positional vector** (люблю)	**Positional vector** (трансформер)

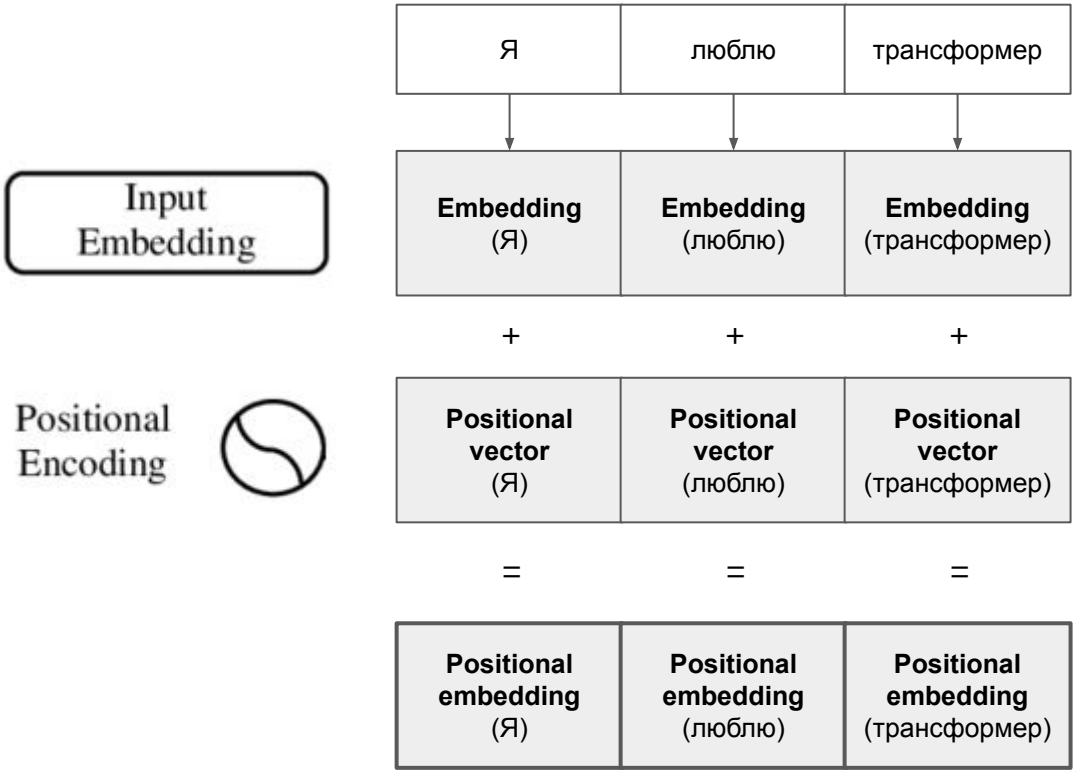
Трансформеры. Вспомогательные элементы



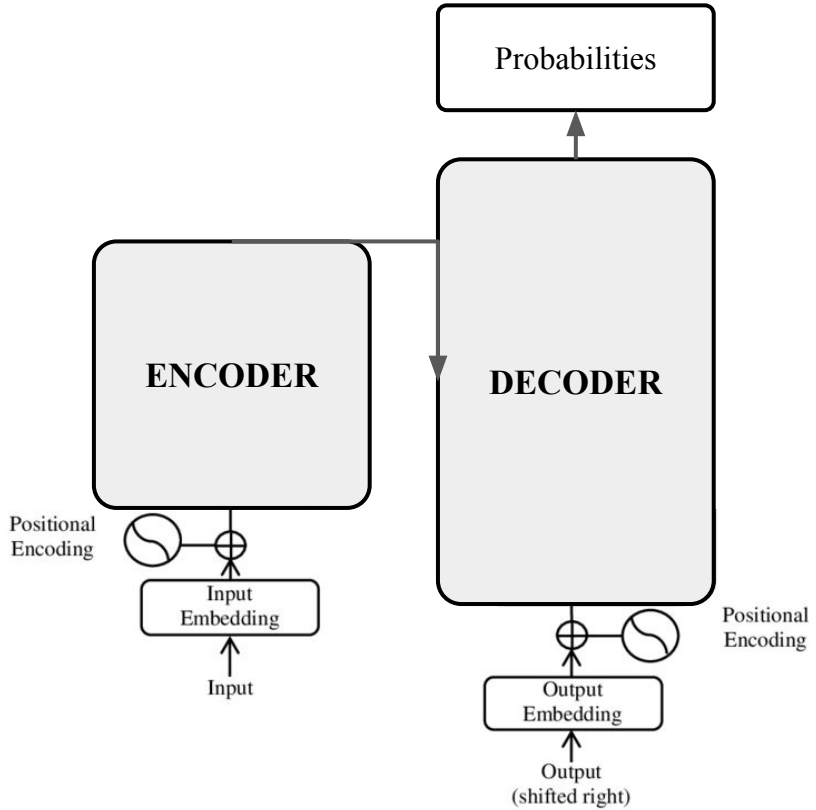
Трансформеры. Вспомогательные элементы



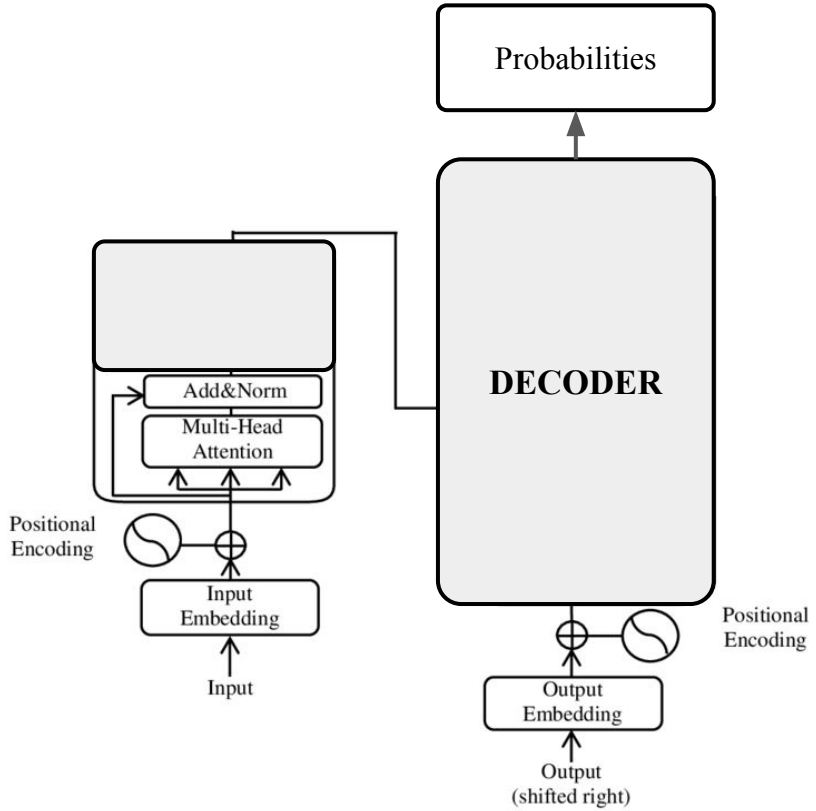
Трансформеры. Вспомогательные элементы



Трансформеры. Вспомогательные элементы

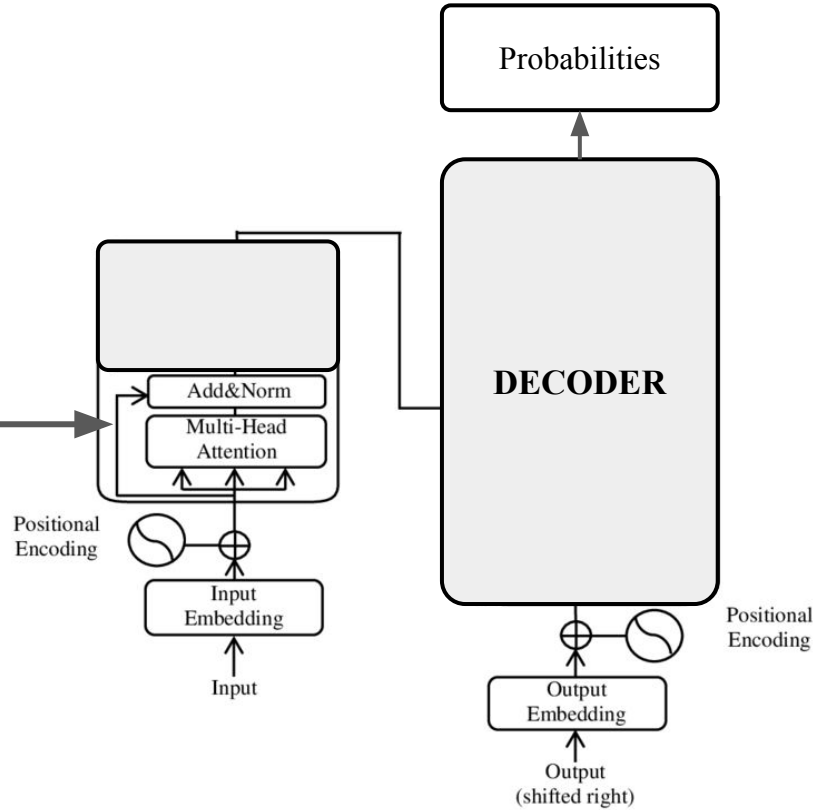


Трансформеры. Вспомогательные элементы

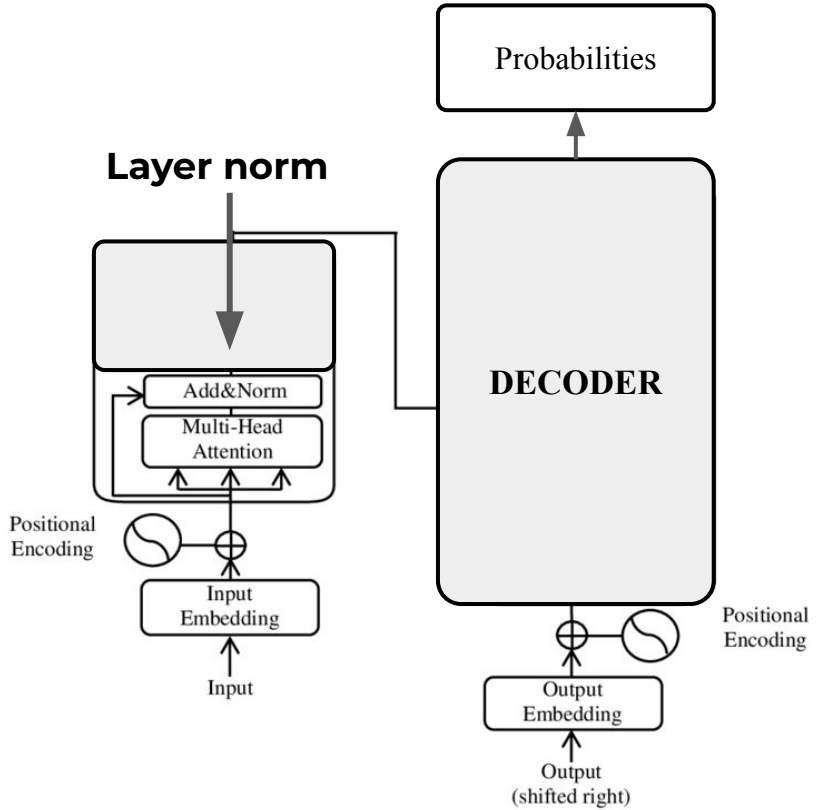


Трансформеры. Вспомогательные элементы

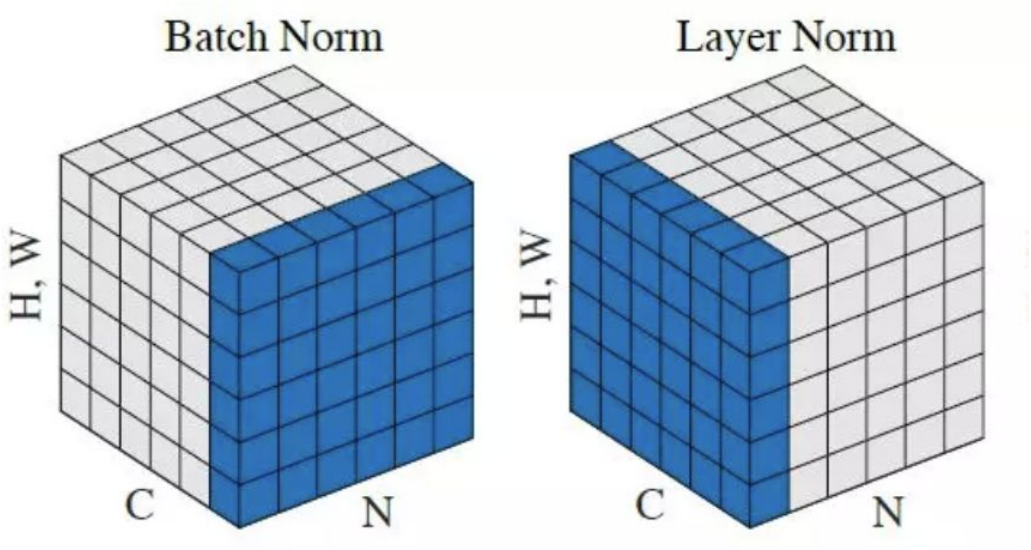
Residual connection (вспоминаем ResNet)



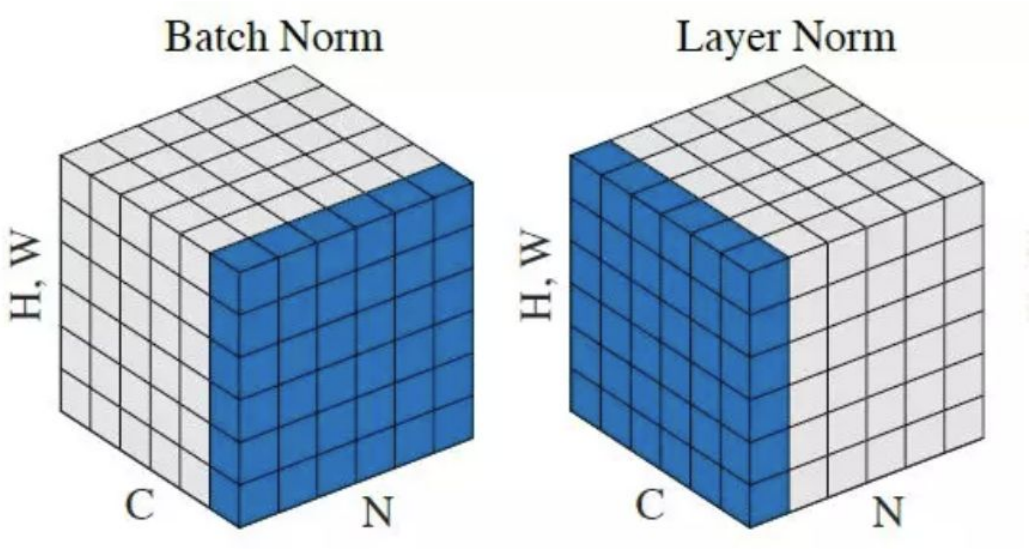
Трансформеры. Вспомогательные элементы



Трансформеры. Вспомогательные элементы

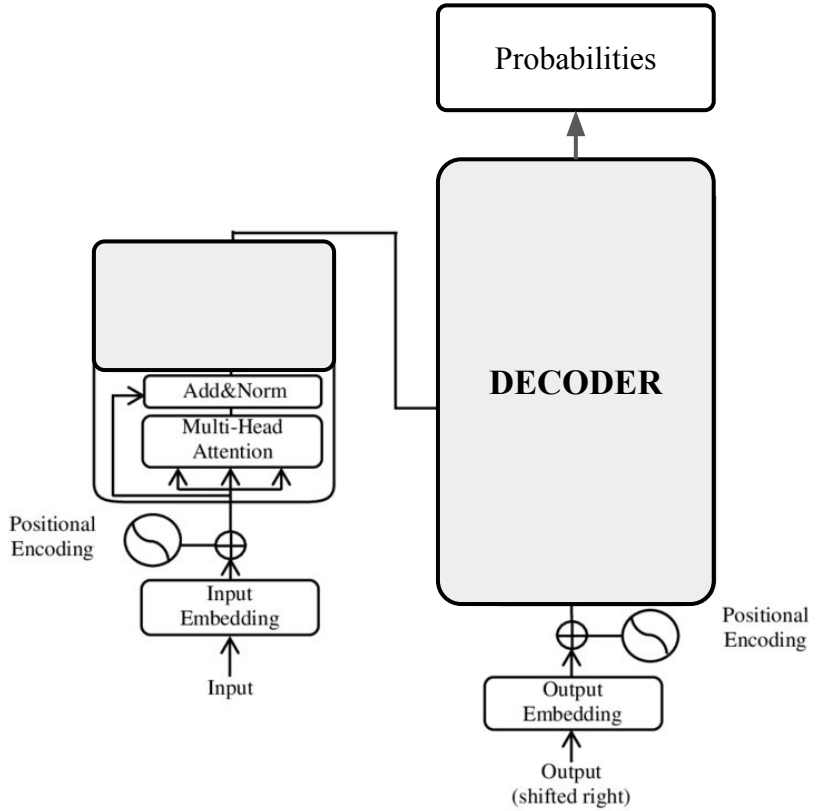


Трансформеры. Вспомогательные элементы

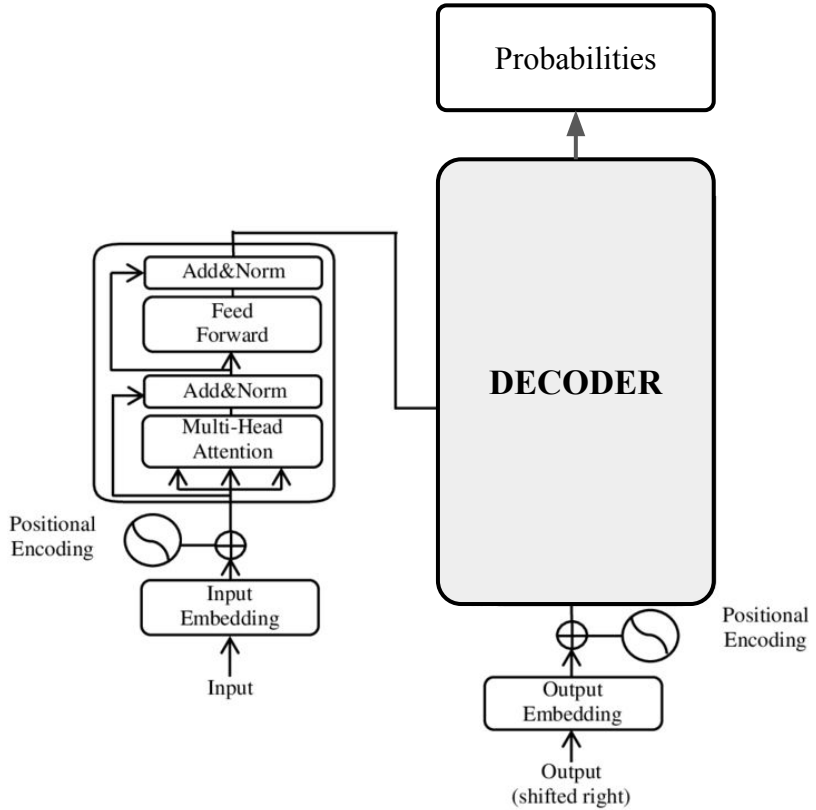


- Оба метода преобразуют данные так, чтобы среднее было равно нулю, а стандартное отклонение - единице
- Layer Norm не вводит неявной зависимости между разными примерами в батче
- Layer Norm актуален даже если в батче один пример

Трансформеры. Вспомогательные элементы

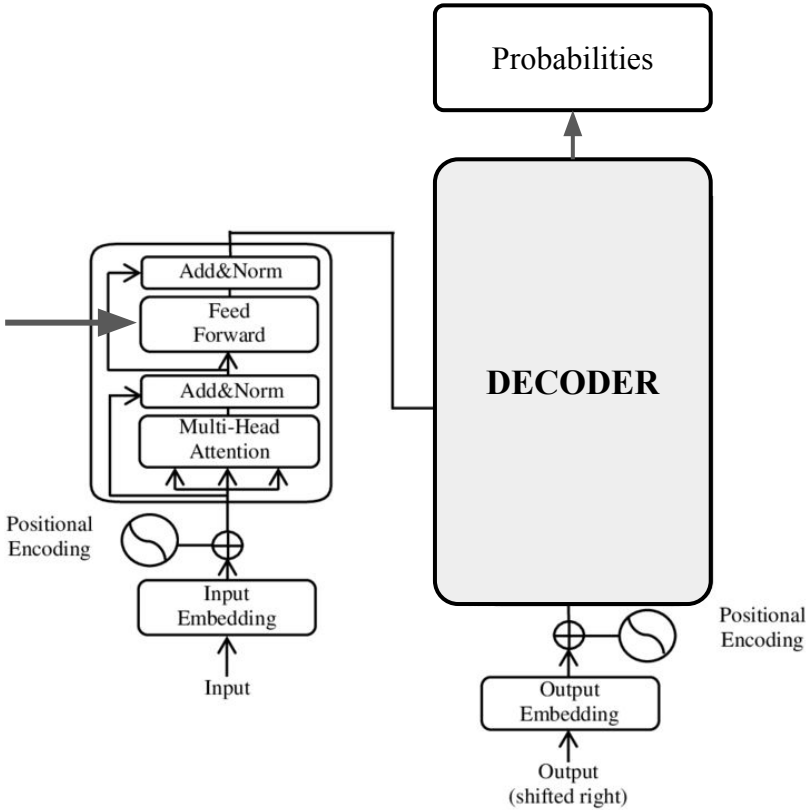


Трансформеры. Вспомогательные элементы

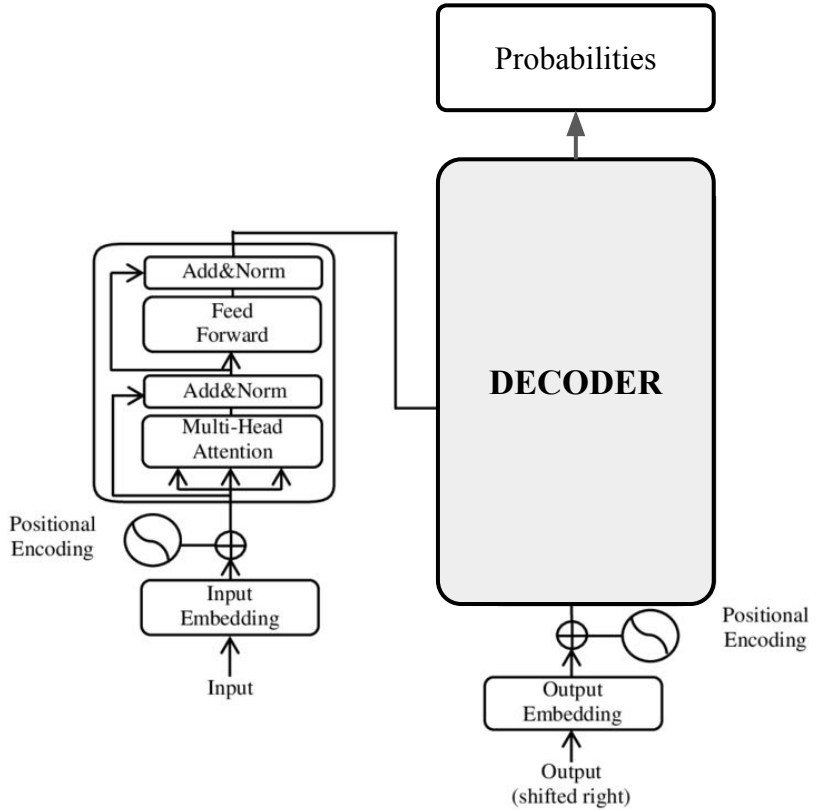


Трансформеры. Вспомогательные элементы

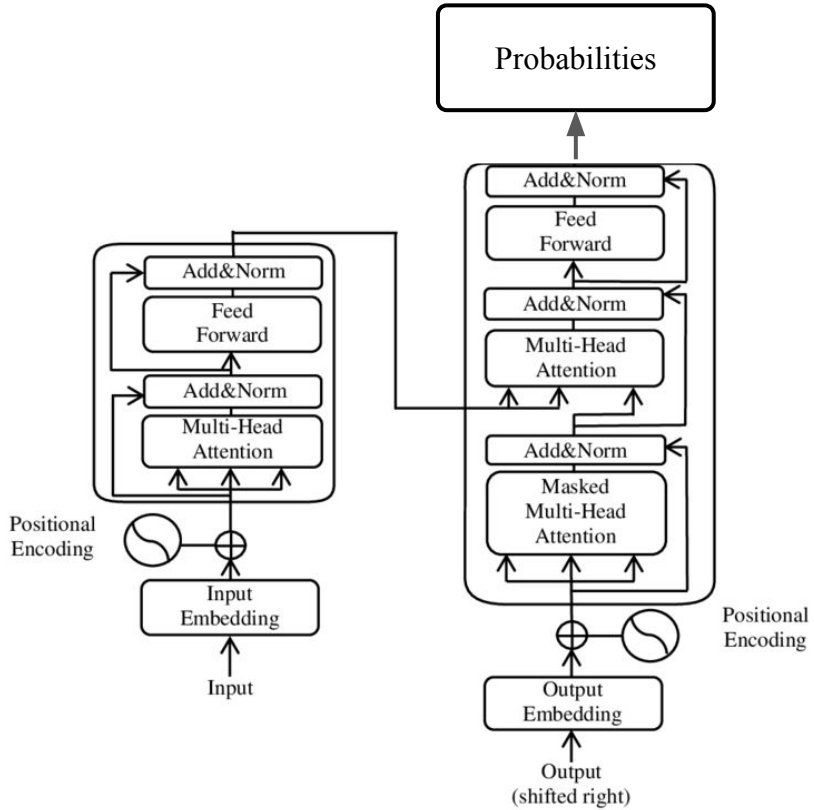
Полносвязные слои
(могут называться “output”, “intermediate”)



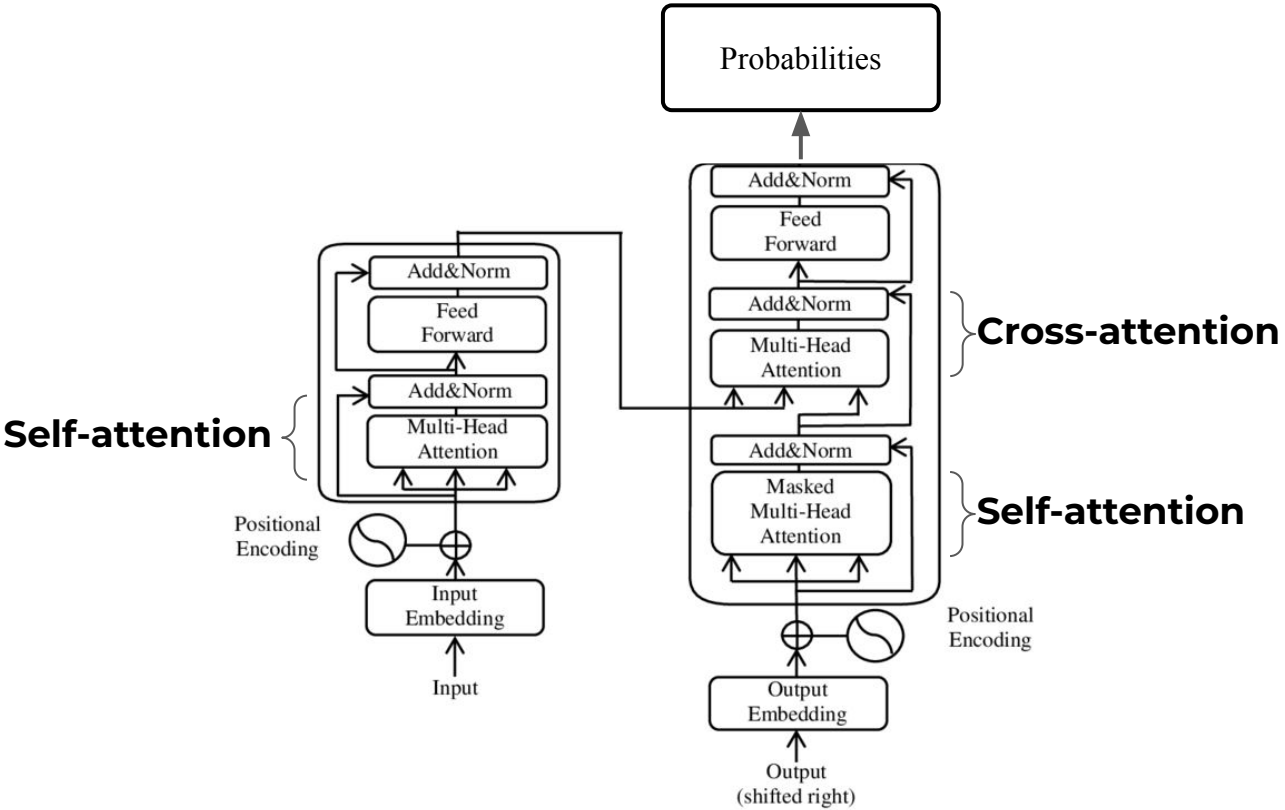
Трансформеры. Вспомогательные элементы



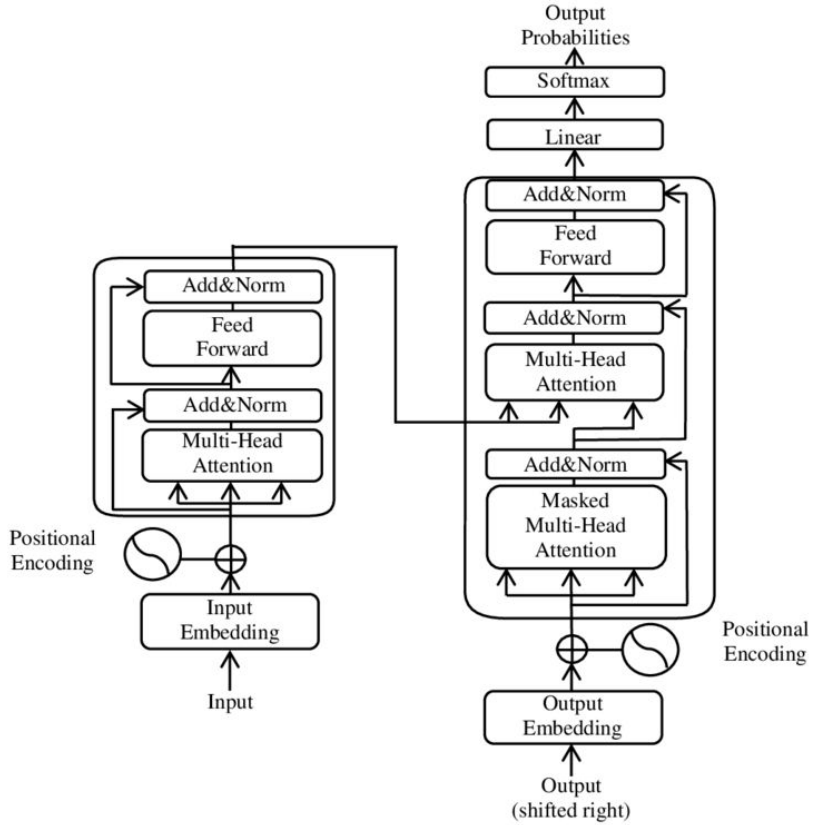
Трансформеры. Вспомогательные элементы



Трансформеры. Вспомогательные элементы



Трансформеры. Вспомогательные элементы



Итоги видео

В этом видео мы познакомились с полной архитектурой трансформера.

В оставшемся видео мы посмотрим на особенности обучения трансформерных моделей.

Deep Learning School

Трансформеры

Особенности обучения, вариации

План лекции

- Трансформеры. Главные идеи:
  - Убираем рекуррентность
  - Используем Multi-head attention:
    - Self-attention
    - Cross-attention
- Трансформеры. Вспомогательные элементы:
  - Word & Position embeddings
  - LayerNorm
  - Dense layers
  - Residual connections
- **Трансформеры. Особенности обучения, вариации**

Трансформеры. Особенности обучения, вариации

Как эффективно обучать трансформеры?

Трансформеры. Особенности обучения, вариации

Как эффективно обучать трансформеры?

Обычно обучение происходит в два этапа:

Трансформеры. Особенности обучения, вариации

Как эффективно обучать трансформеры?

Обычно обучение происходит в два этапа:

1. **Предобучение** (Pretraining):
    - Обычно это Masked Language Modelling и его более продвинутые вариации (маскирование целых слов, сущностей или предложений)
2. **Дообучение** (Finetuning):
    - Classic Finetuning for downstream task
    - Reinforcement Learning from Human Feedback

Трансформеры. Особенности обучения, вариации

Как эффективно обучать трансформеры?

Обычно обучение происходит в два этапа:

1. **Предобучение** (Pretraining)
2. **Дообучение** (Finetuning)

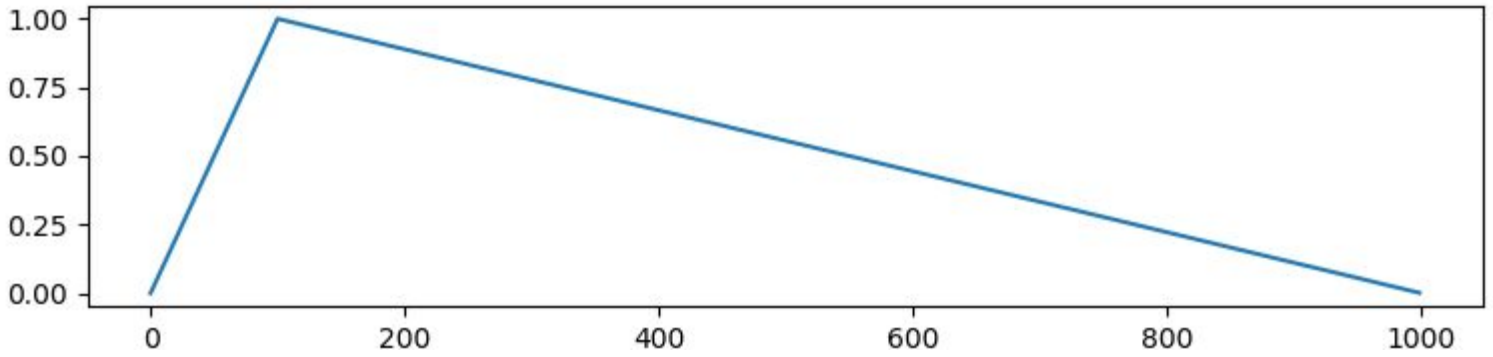
Для каждого этапа используются:

- **Адаптивные методы** (Adam)
- **Планировщики обучения** (schedulers) с “прогревом” (warmup)

Трансформеры. Особенности обучения, вариации

Прогрев улучшает стабильность обучения с помощью адаптивных методов (Adam, AdamW).

Стандартный планировщик - Linear scheduler with warmup:



(График величины learning rate в зависимости от количества шагов)

Трансформеры. Особенности обучения, вариации

Некоторые из сложностей, возникающих при обучении трансформеров:

- Предобучение требует больших вычислительных ресурсов
- Результат обучения зависит от инициализации (т.е. от seed RNG)
- Нужно подбирать настройки оптимизатора и планировщика

Трансформеры. Особенности обучения, вариации

Некоторые из сложностей, возникающих при обучении трансформеров:

- Предобучение требует больших вычислительных ресурсов
- Результат обучения зависит от инициализации (т.е. от seed RNG)
- Нужно подбирать настройки оптимизатора и планировщика
- Для очень больших моделей (миллиарды параметров) требуются ухищрения, чтобы разнести модель на несколько GPU/TPU

Трансформеры. Особенности обучения, вариации

Некоторые из сложностей, возникающих при обучении трансформеров:

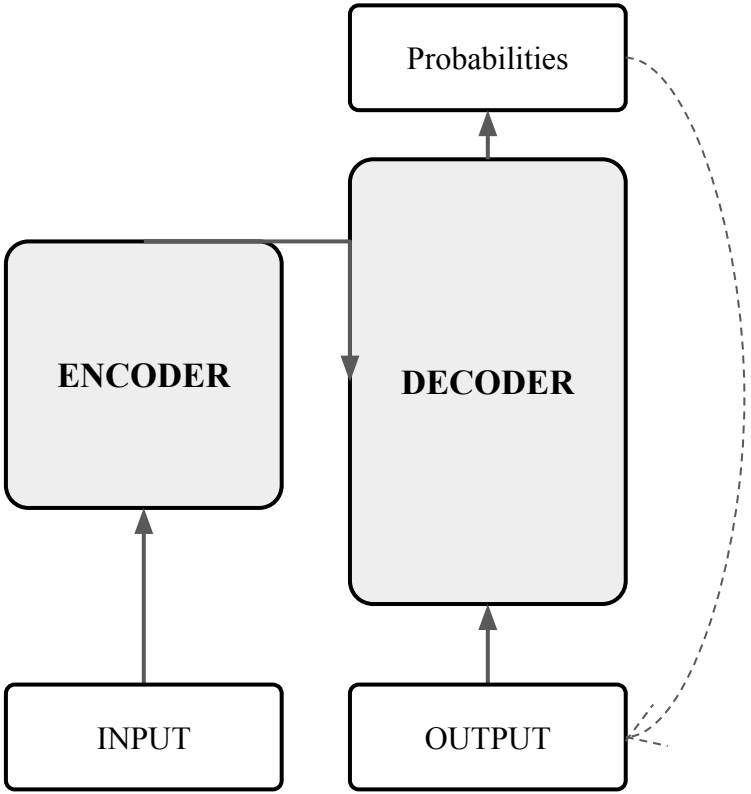
- Предобучение требует больших вычислительных ресурсов
- Результат обучения зависит от инициализации (т.е. от seed RNG)
- Нужно подбирать настройки оптимизатора и планировщика
- Для очень больших моделей (миллиарды параметров) требуются ухищрения, чтобы разнести модель на несколько GPU/TPU
  (N инженеров сойдут с ума)

Трансформеры. Особенности обучения, вариации

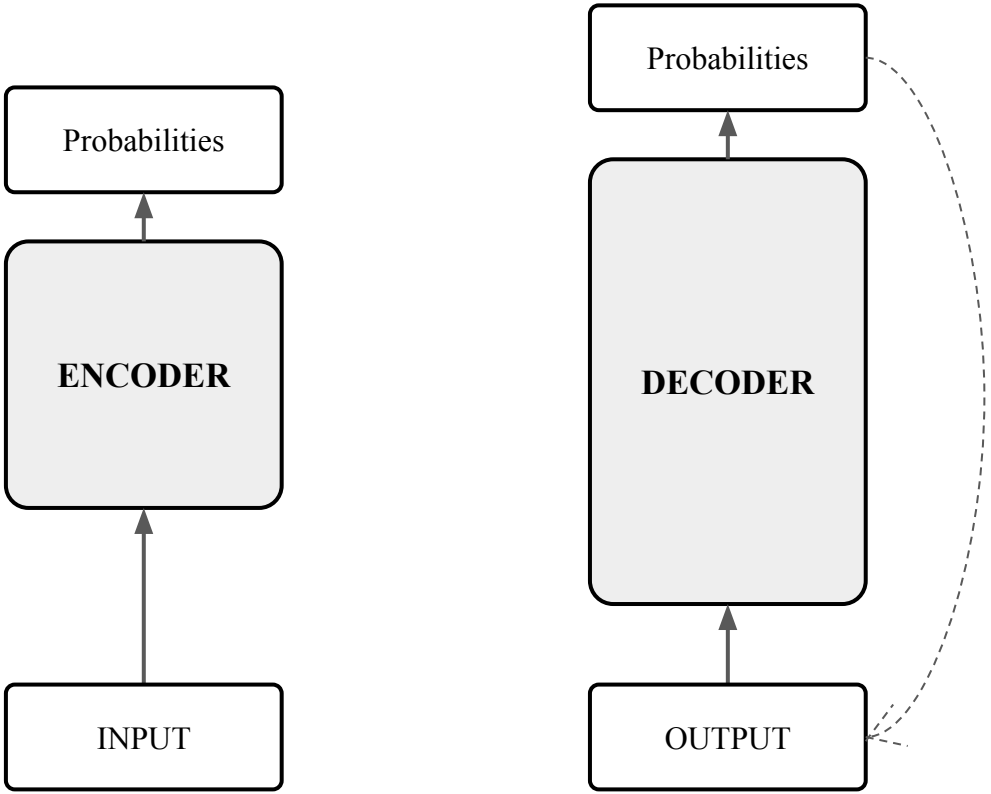
Некоторые из сложностей, возникающих при обучении трансформеров:

- Предобучение требует больших вычислительных ресурсов
- Результат обучения зависит от инициализации (т.е. от seed RNG)
- Нужно подбирать настройки оптимизатора и планировщика
- Для очень больших моделей (миллиарды параметров) требуются ухищрения, чтобы разнести модель на несколько GPU/TPU
- Для очень больших моделей проблема с нестабильностью обучения до конца не решена, иногда приходится делать рестарты

Трансформеры. Особенности обучения, вариации

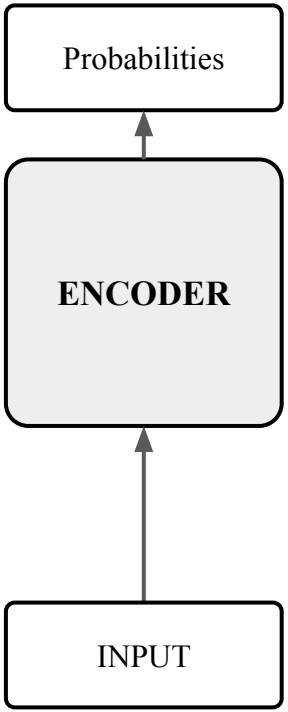


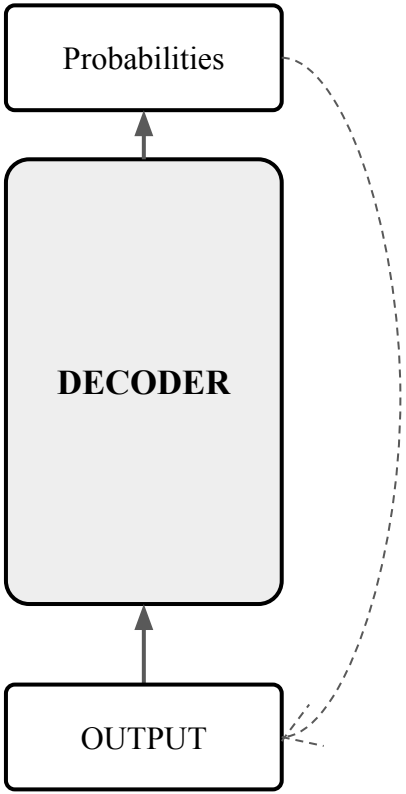
Трансформеры. Особенности обучения, вариации



Трансформеры. Особенности обучения, вариации

BERT-like модели





GPT-like модели

Итоги видео

В этом видео мы познакомились с особенностями обучения трансформеров и были заинтригованы возможными вариациями этих моделей.