

Задача машинного перевода

Дано:

Предложение на исходном языке (**source language**)

Выход:

Предложение на целевом языке (**target language**)

The screenshot shows a machine translation interface with two input fields and a central output area. The top bar has language selection dropdowns: 'ОПРЕДЕЛИТЬ ЯЗЫК' (Detect language), 'АНГЛИЙСКИЙ' (English), 'РУССКИЙ' (Russian) which is underlined in blue, 'ФРАНЦУЗСКИЙ' (French), and a dropdown arrow. Between the first two dropdowns is a double-headed arrow icon. To its right is another set of dropdowns: 'РУССКИЙ' (Russian), 'АНГЛИЙСКИЙ' (English) which is underlined in blue, 'УКРАИНСКИЙ' (Ukrainian), and a dropdown arrow. Below these are two input fields separated by a horizontal line. The left input field contains the Russian sentence 'Сбербанк -- всегда рядом' with a small 'x' icon to its right. The right input field contains the English translation 'Sberbank - always there'. Below the inputs is a smaller line of text: 'Sberbank – vsegda ryadom'. At the bottom of the interface are several icons: a microphone, a speaker, a magnifying glass, a share symbol, and a refresh symbol.

Машинный перевод в 50-е годы

Дитя холодной войны: переводчик с русского на английский IBM 701 Translator

Доктор Достерт предсказывал, что «пять, возможно, три года спустя, преобразование межязыковых значений электронным процессом в важных функциональных областях для нескольких языков будет свершившимся фактом." (1954)

Подход основан на наборе правил и использовал словарь для перевода русских слов в английские.

Проект был закрыт через несколько лет, так как требовал слишком много ресурсов при слабых результатах.



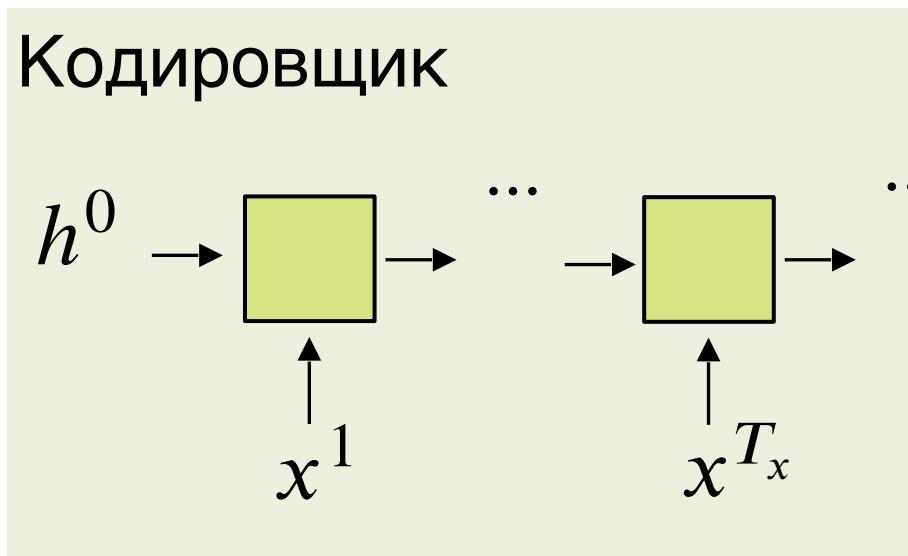
Статистический перевод – основной способ до 2014 года

- Использует сложные и громоздкие модели
- Много отдельных компонент
- Сложная генерация входных признаков
- Необходимость поддержки сложной системы
- Да и качество не очень

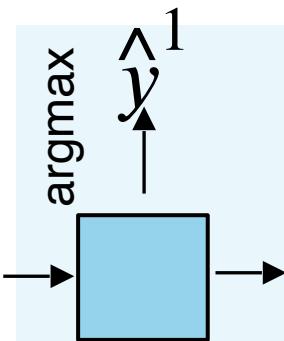
Нейронная сеть для машинного перевода

Возьмем две рекуррентных нейронных сети

Сделаем из них **seq2seq** (sequence to sequence) архитектуру



Вероятности каждого слова



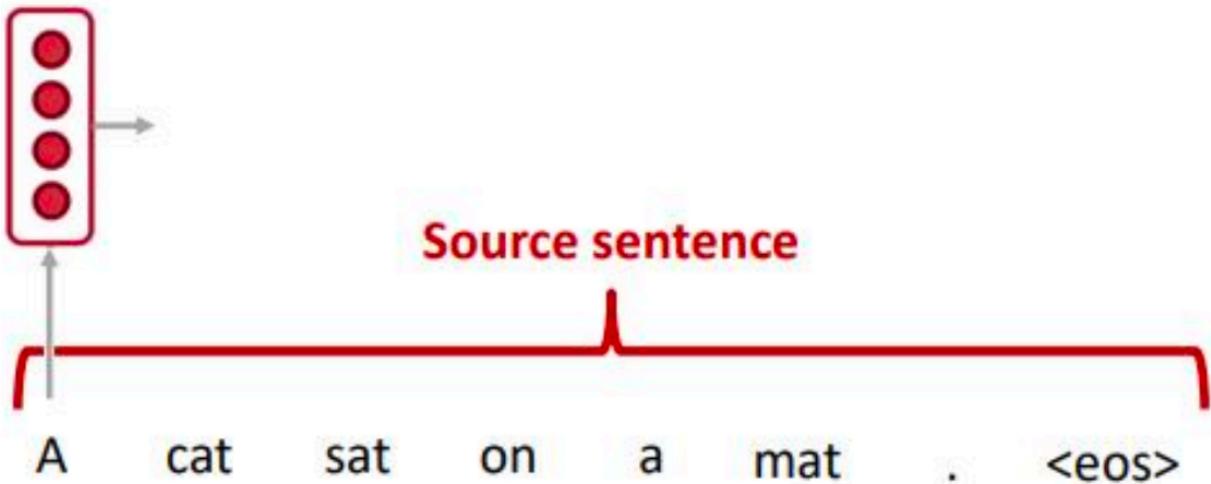
Декодировщик

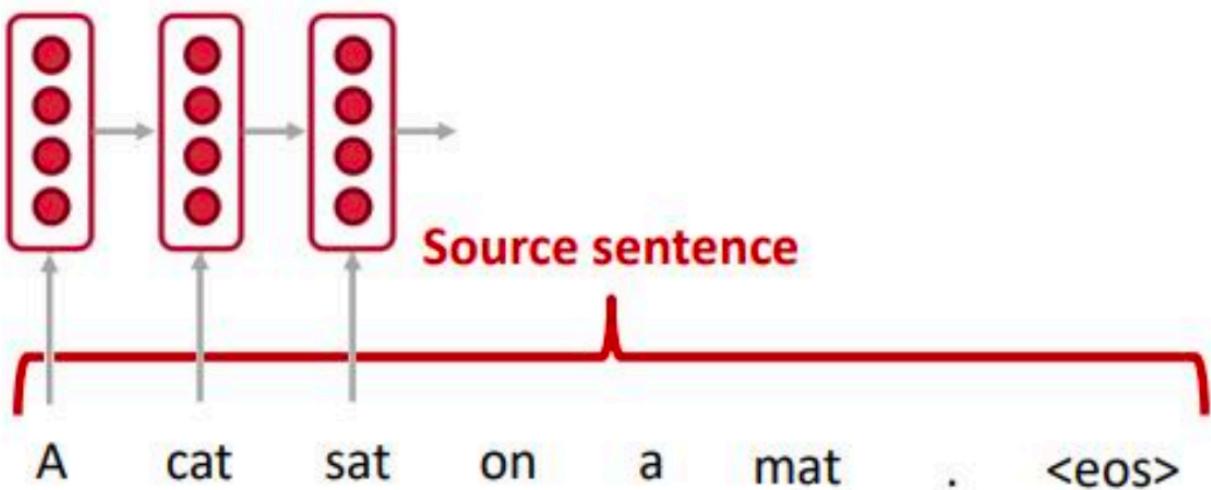


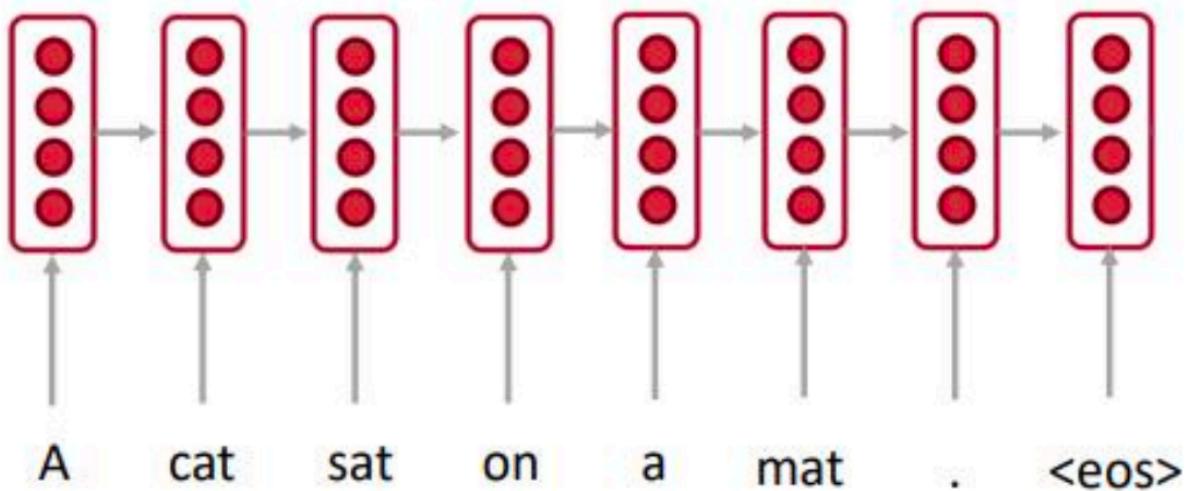
Source sentence

A cat sat on a mat . <eos>

A red bracket is positioned below the text, underlining the words "A", "cat", "sat", "on", "a", "mat", and the period. A small red tick mark is placed above the word "on". After the bracket, there is a space followed by "<eos>".

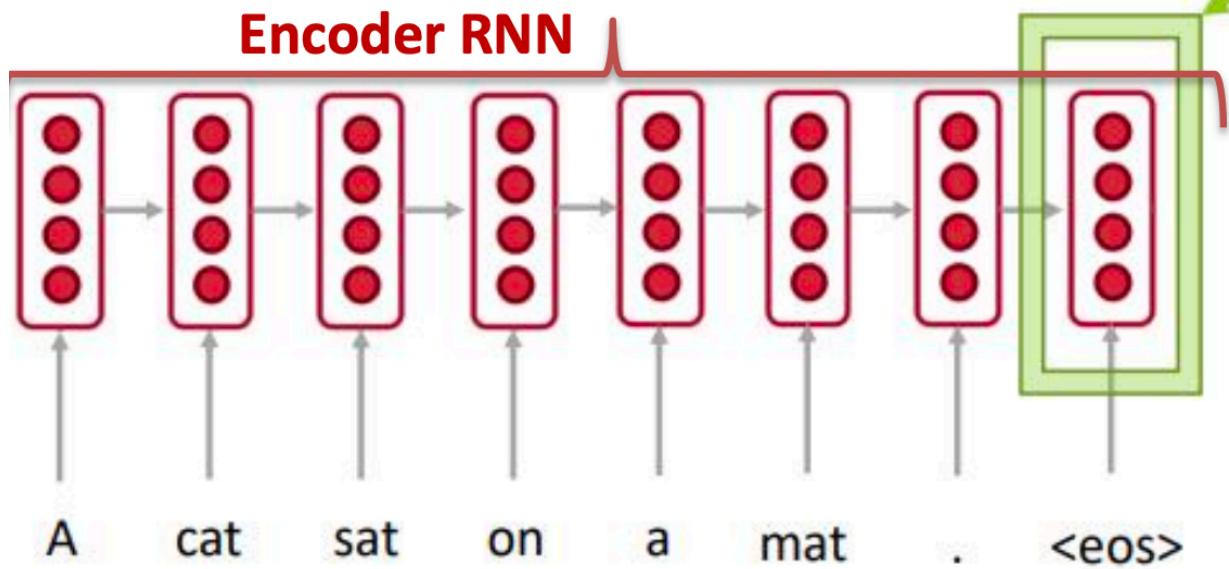




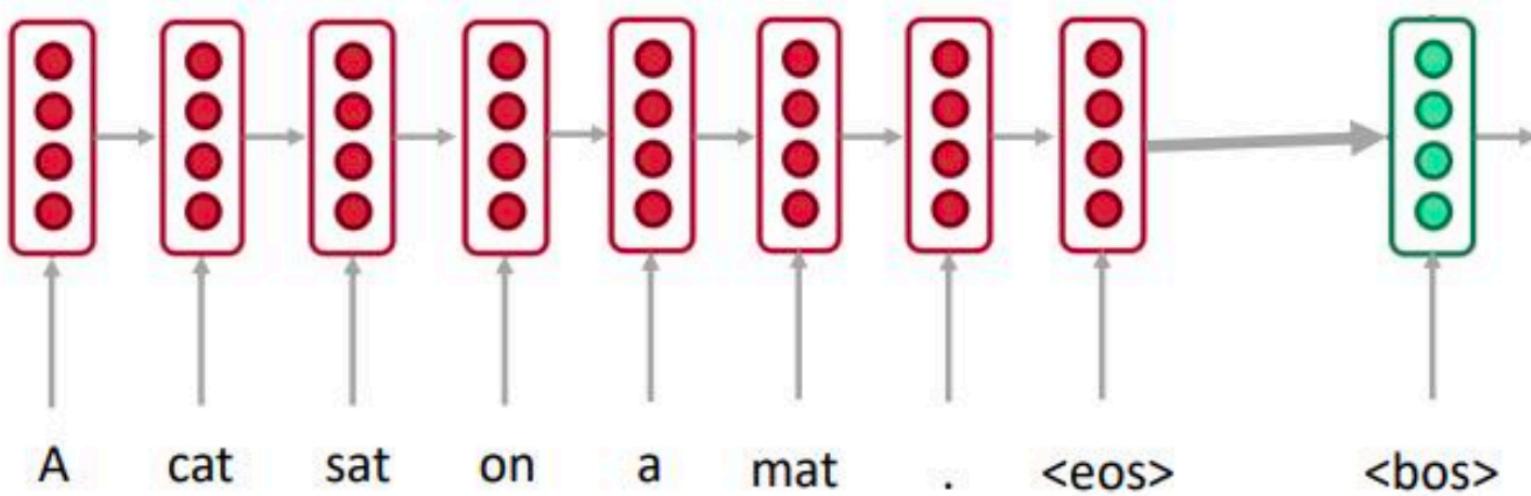


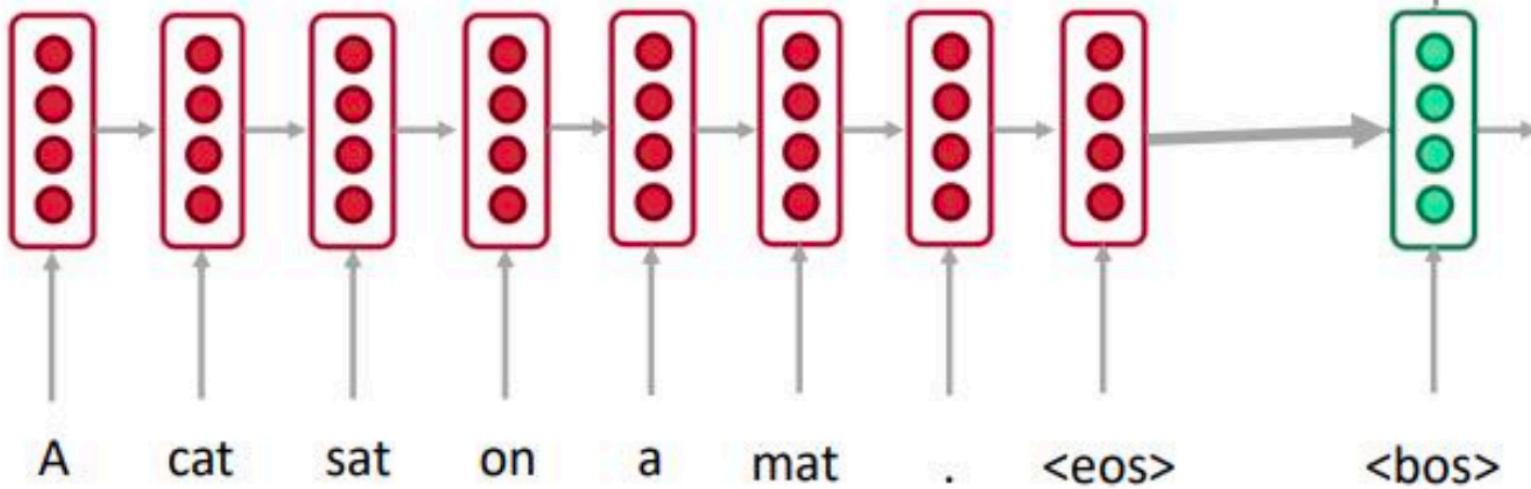
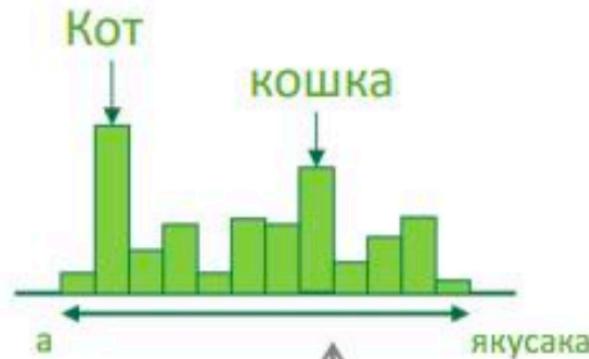


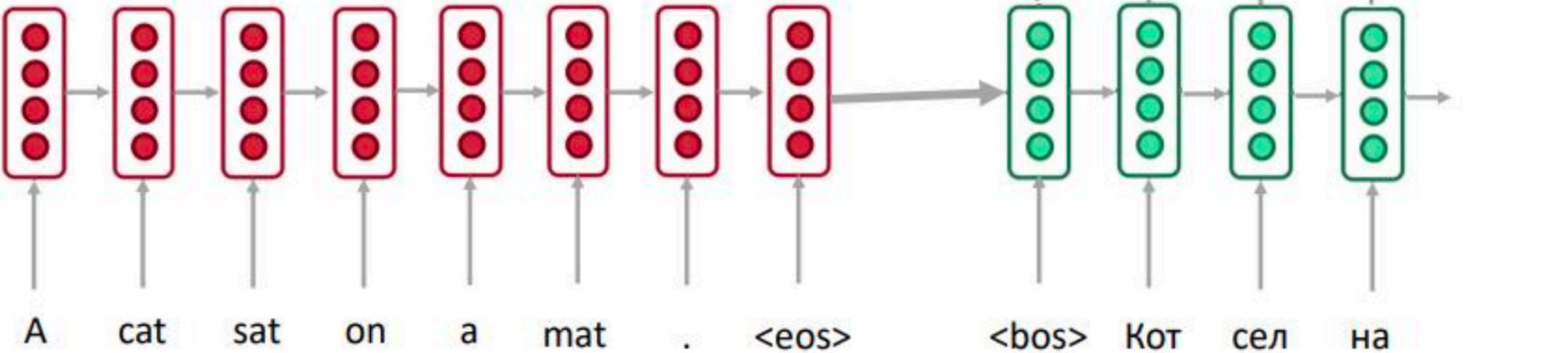
Encoder RNN

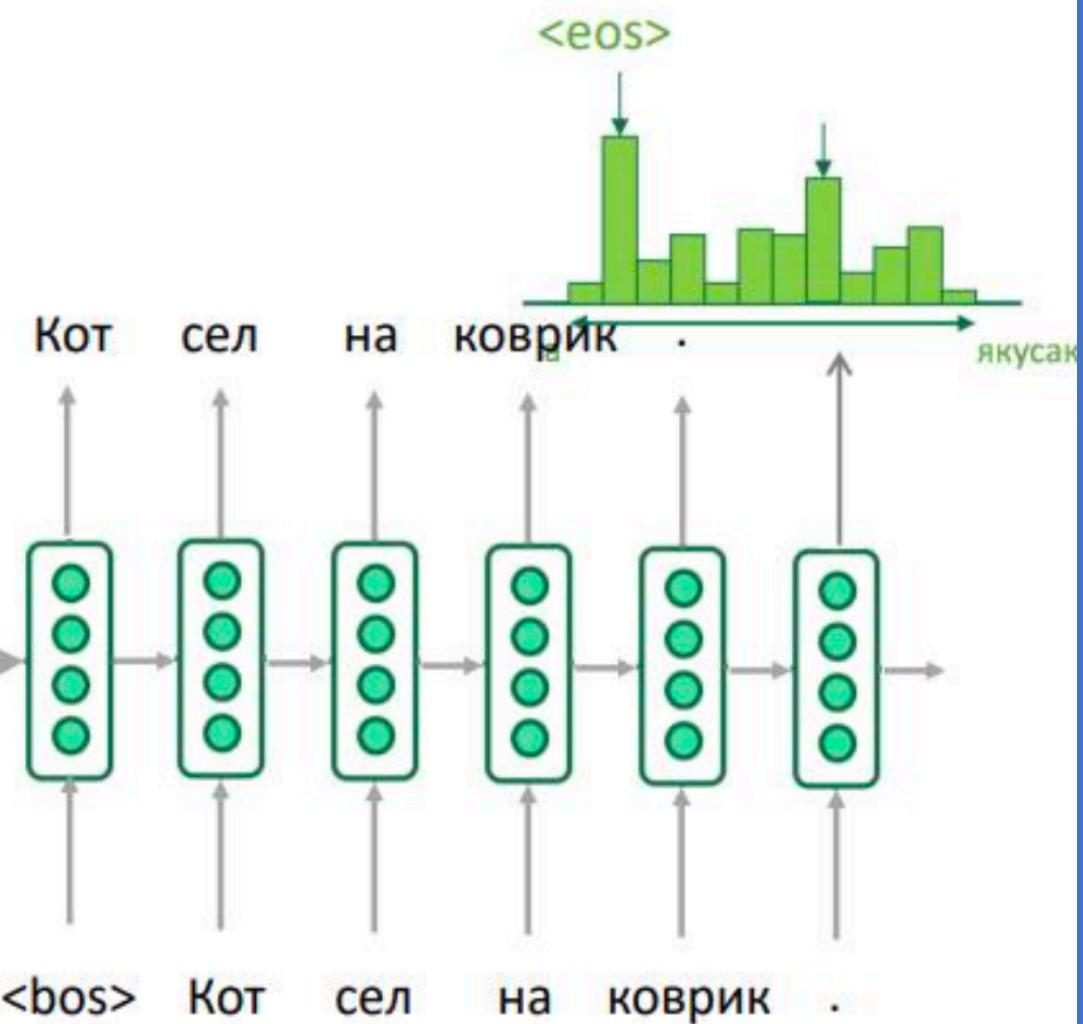
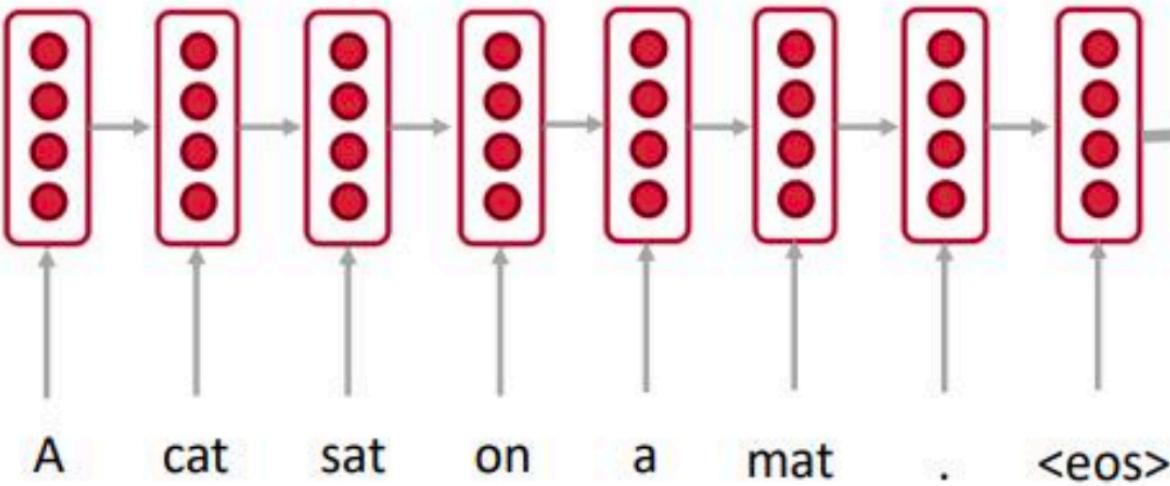


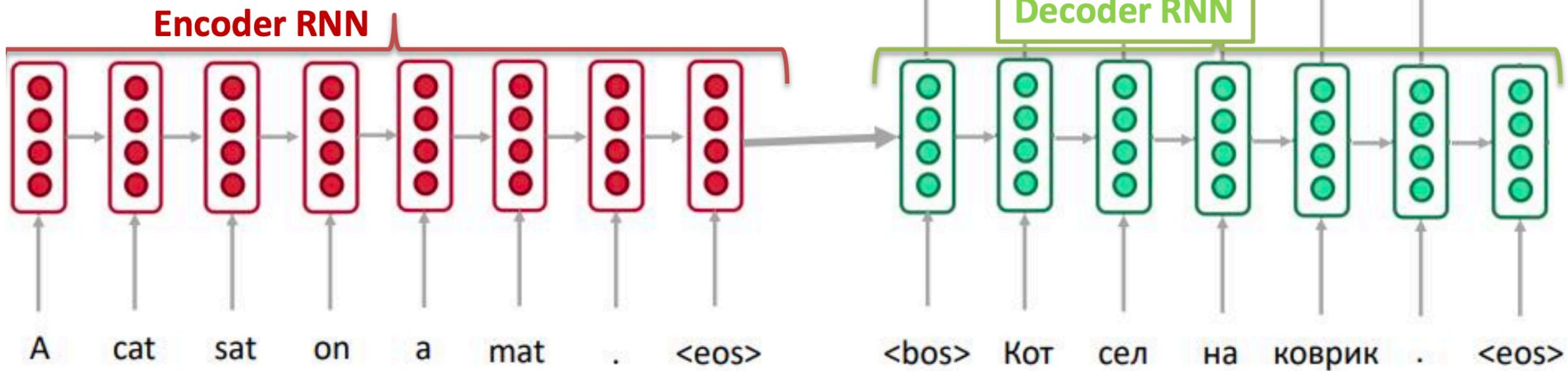
Закодированное исходное предложение.
Используем в качестве исходного состояния для *decoder RNN*



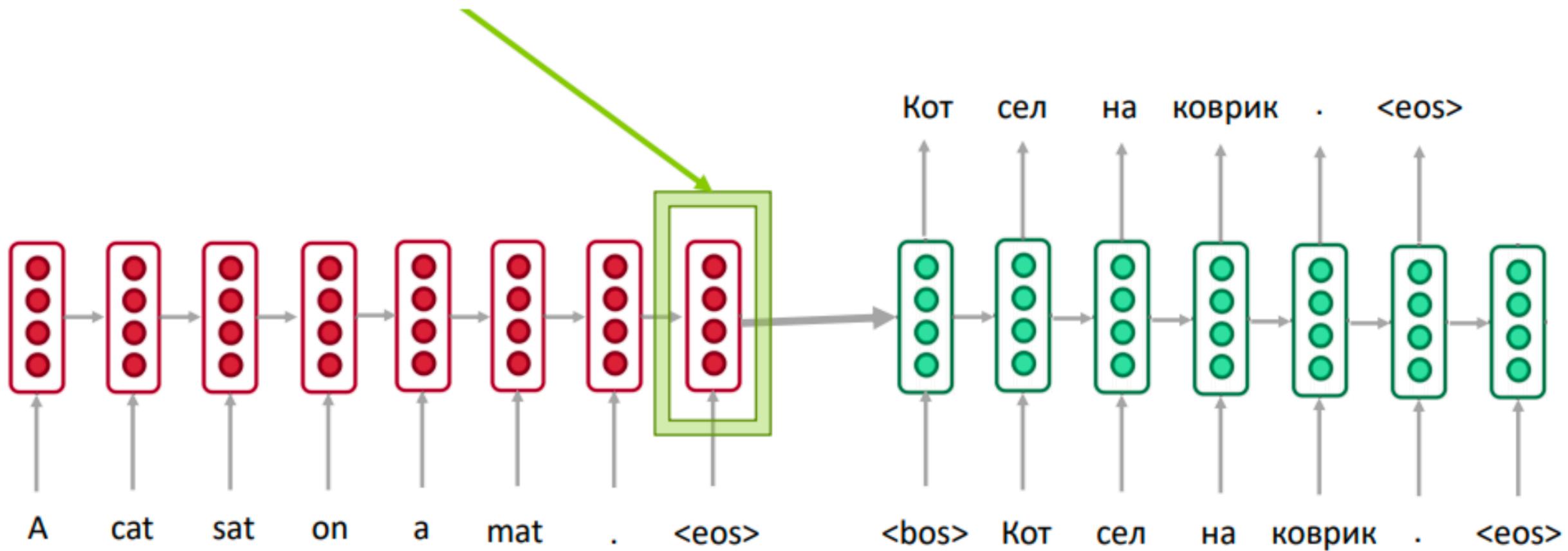




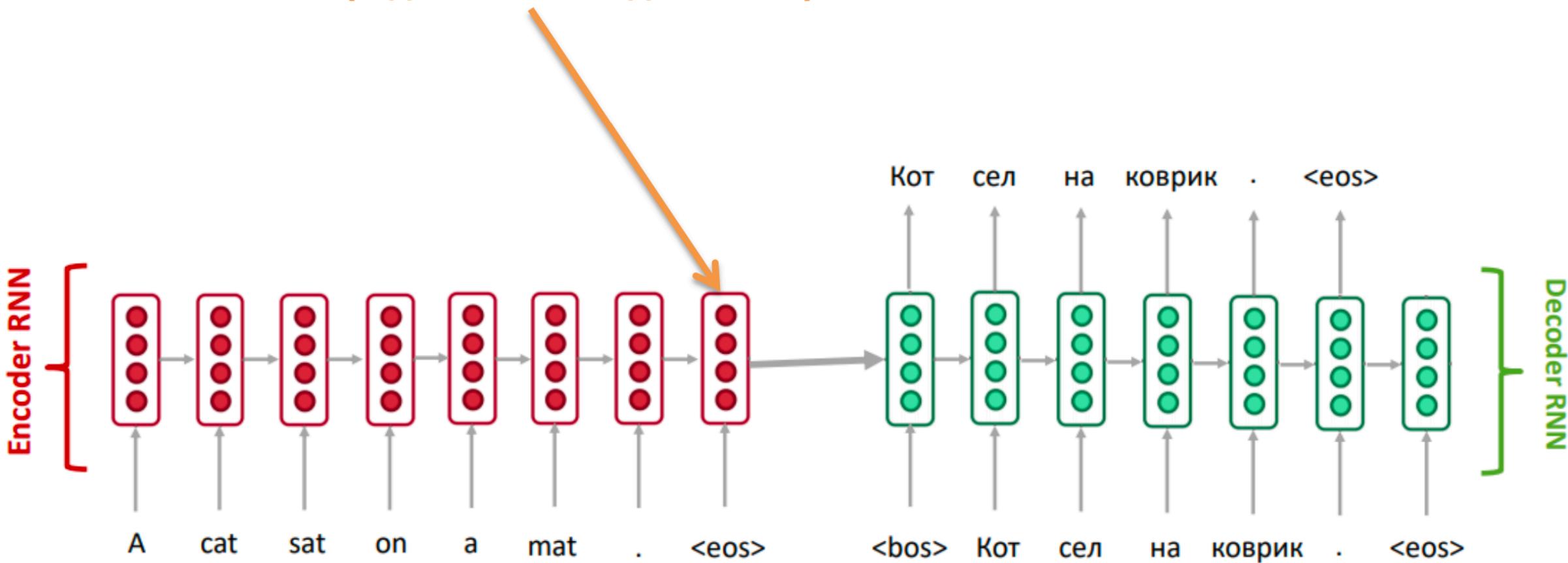


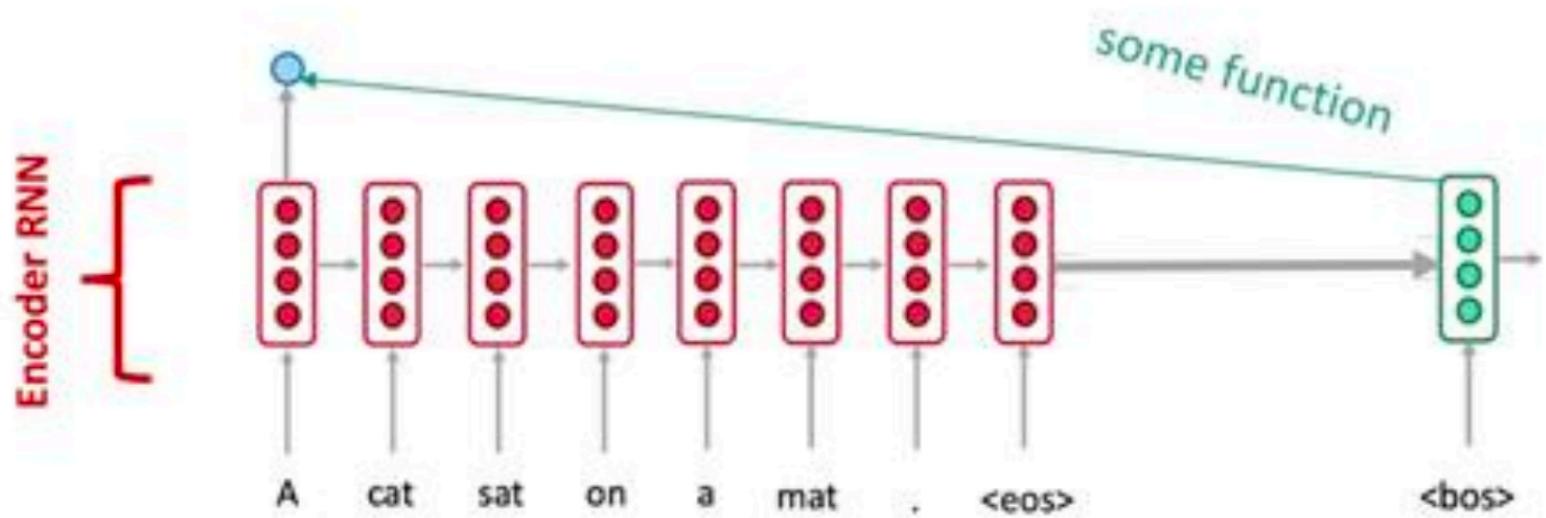


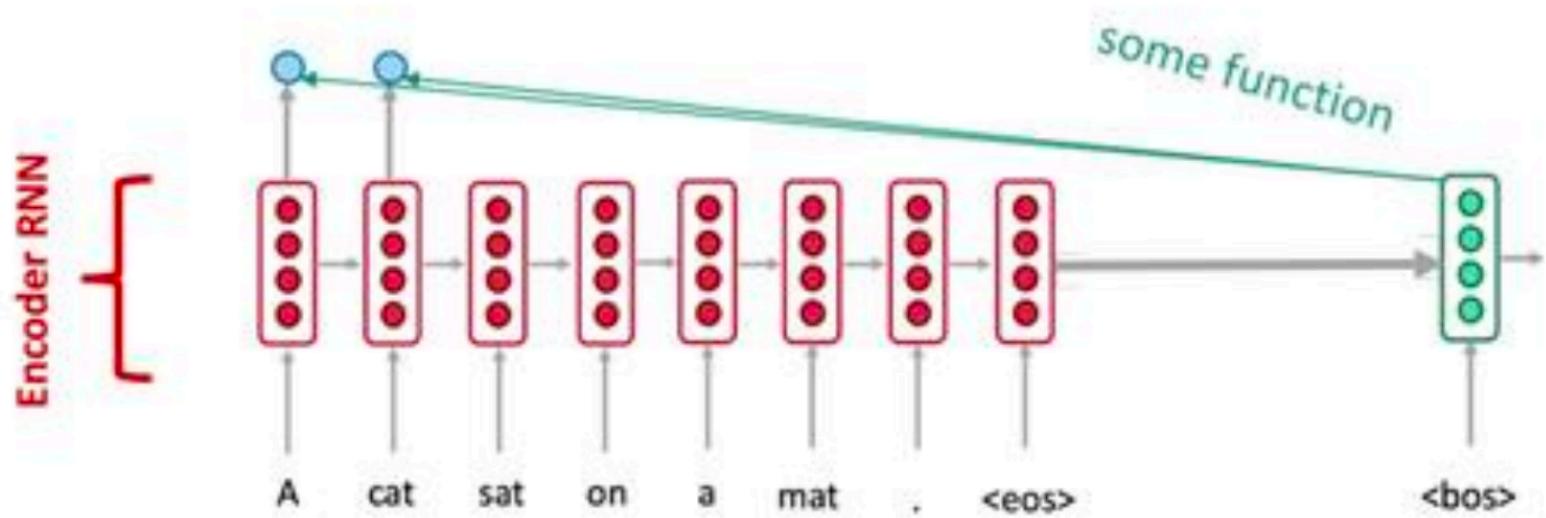
Исходное предложение
encoded sentence

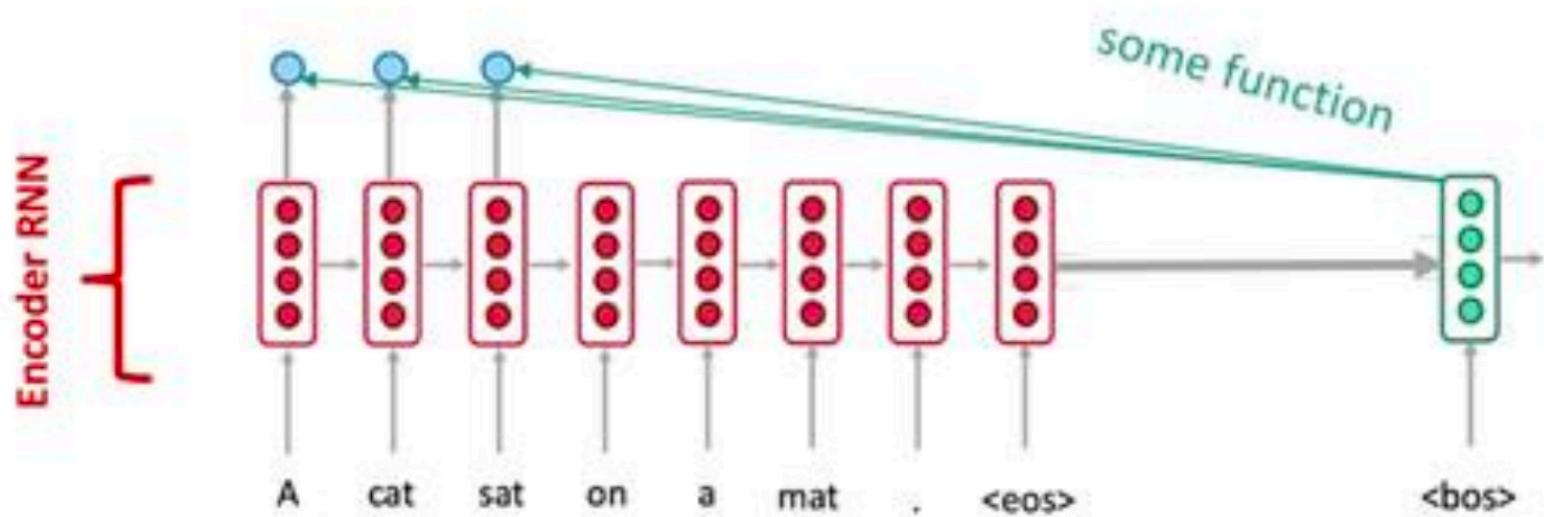


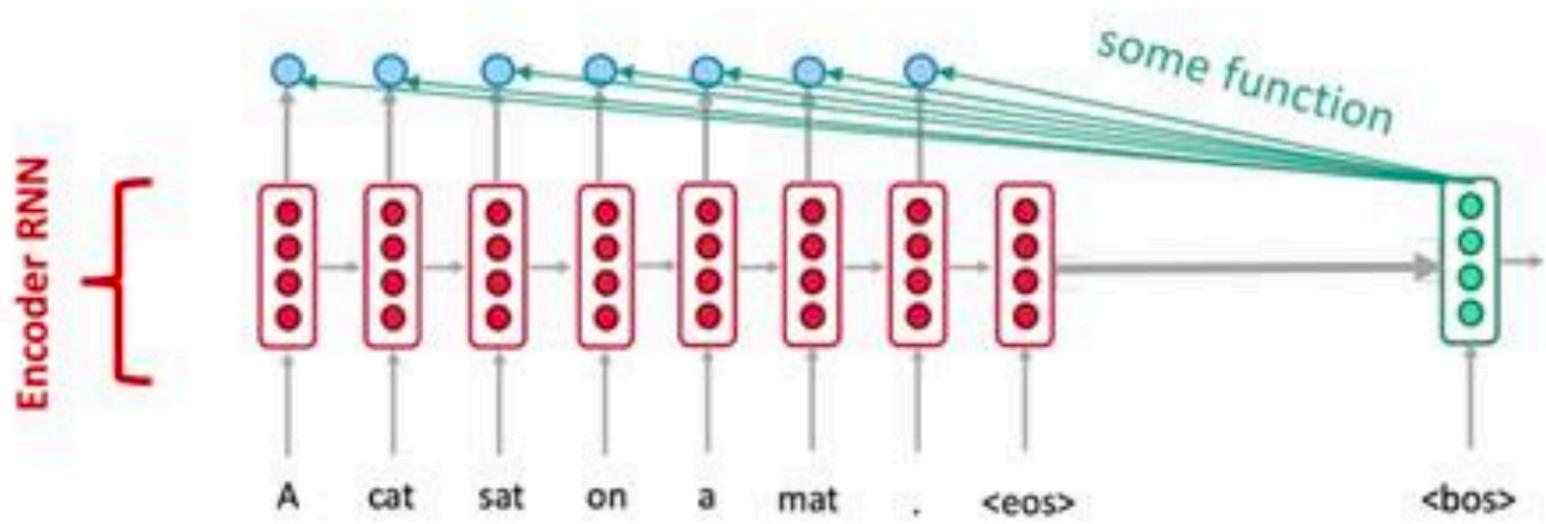
Information bottleneck:
модель должна помещать всю информацию о
предложении в один вектор

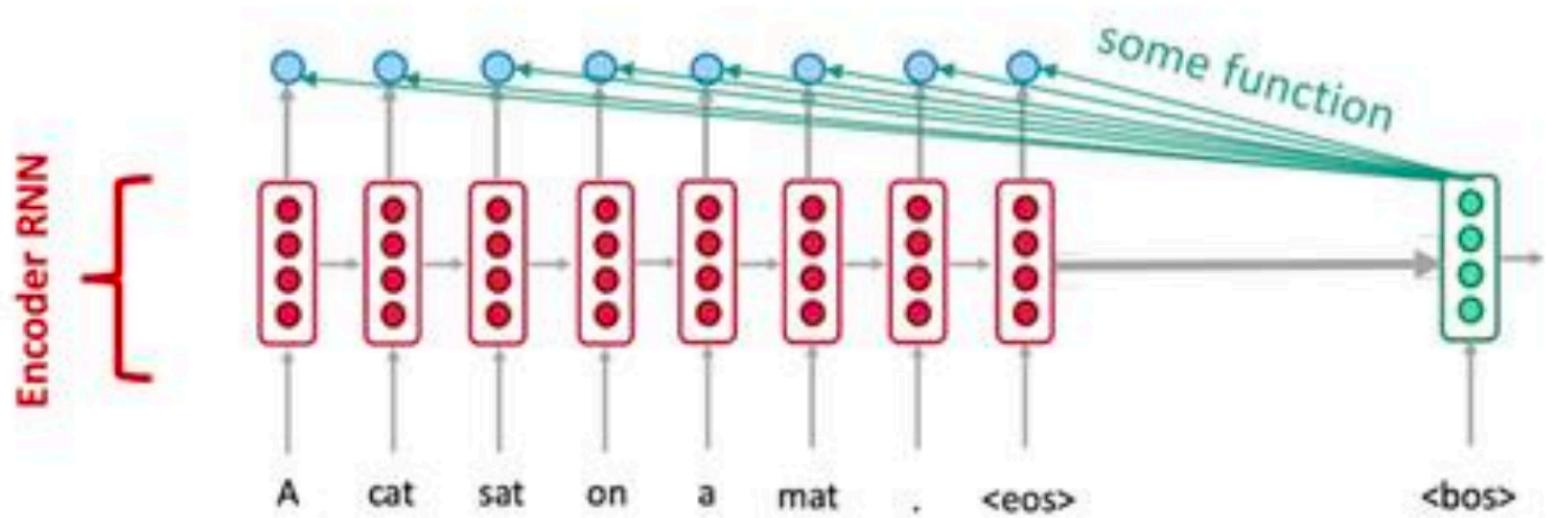


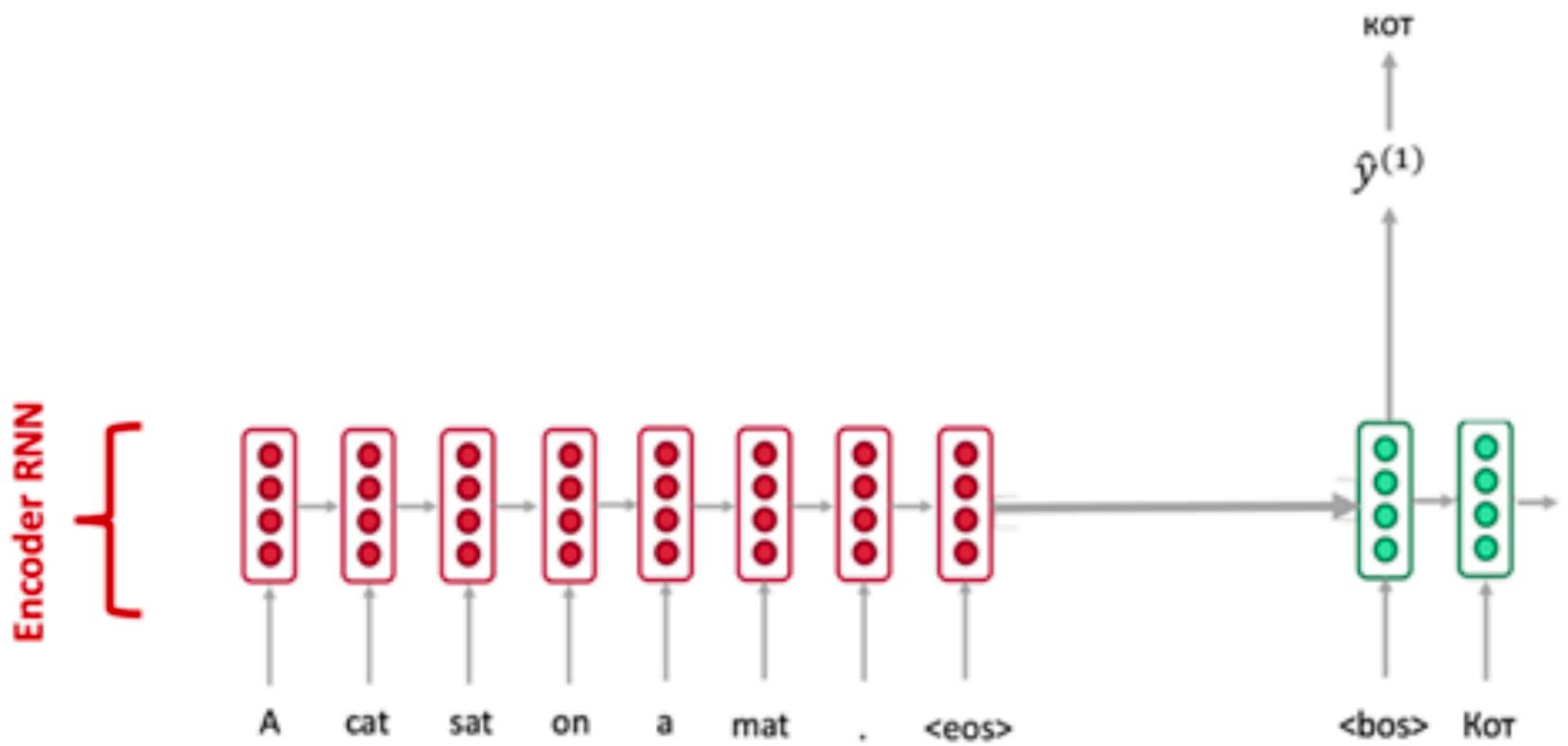


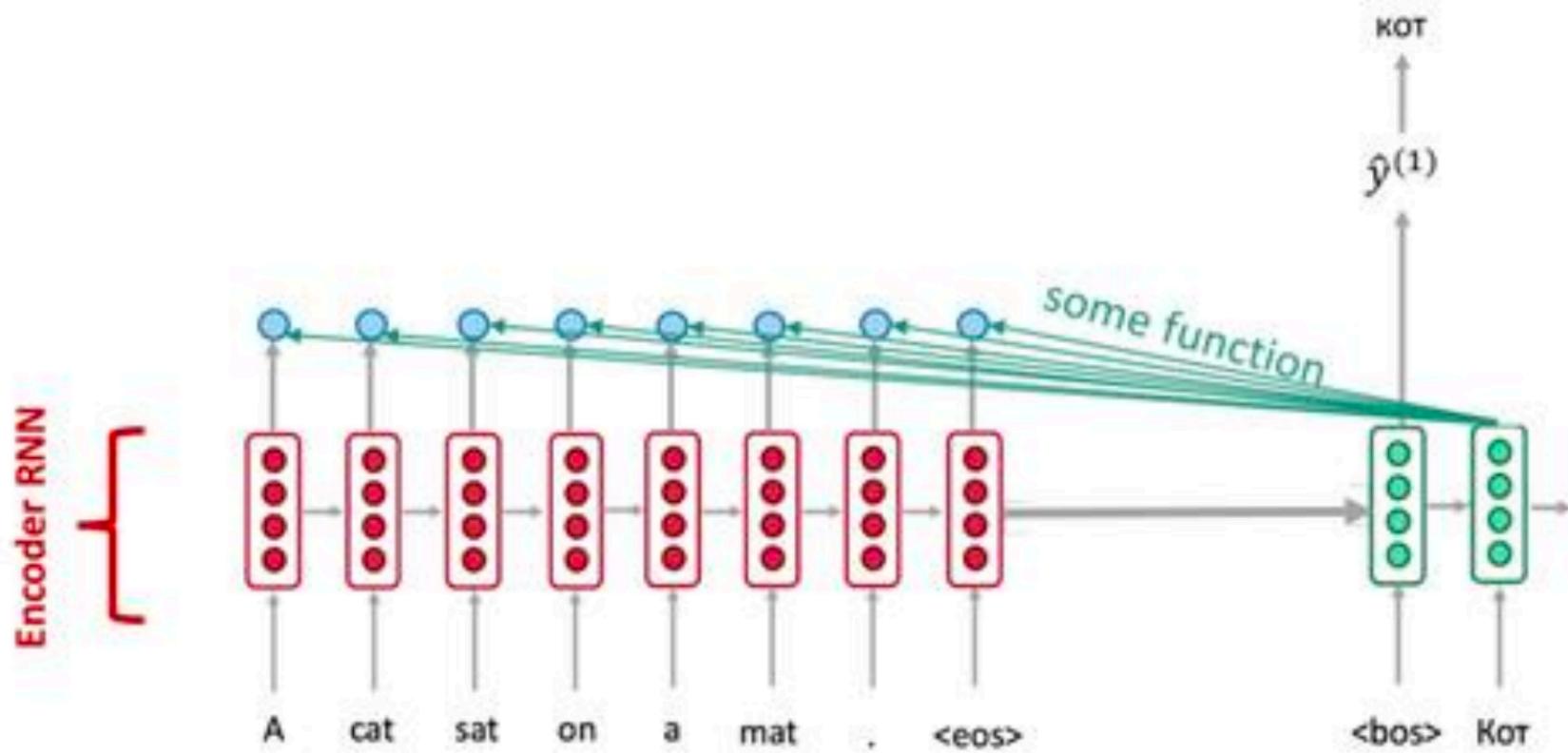


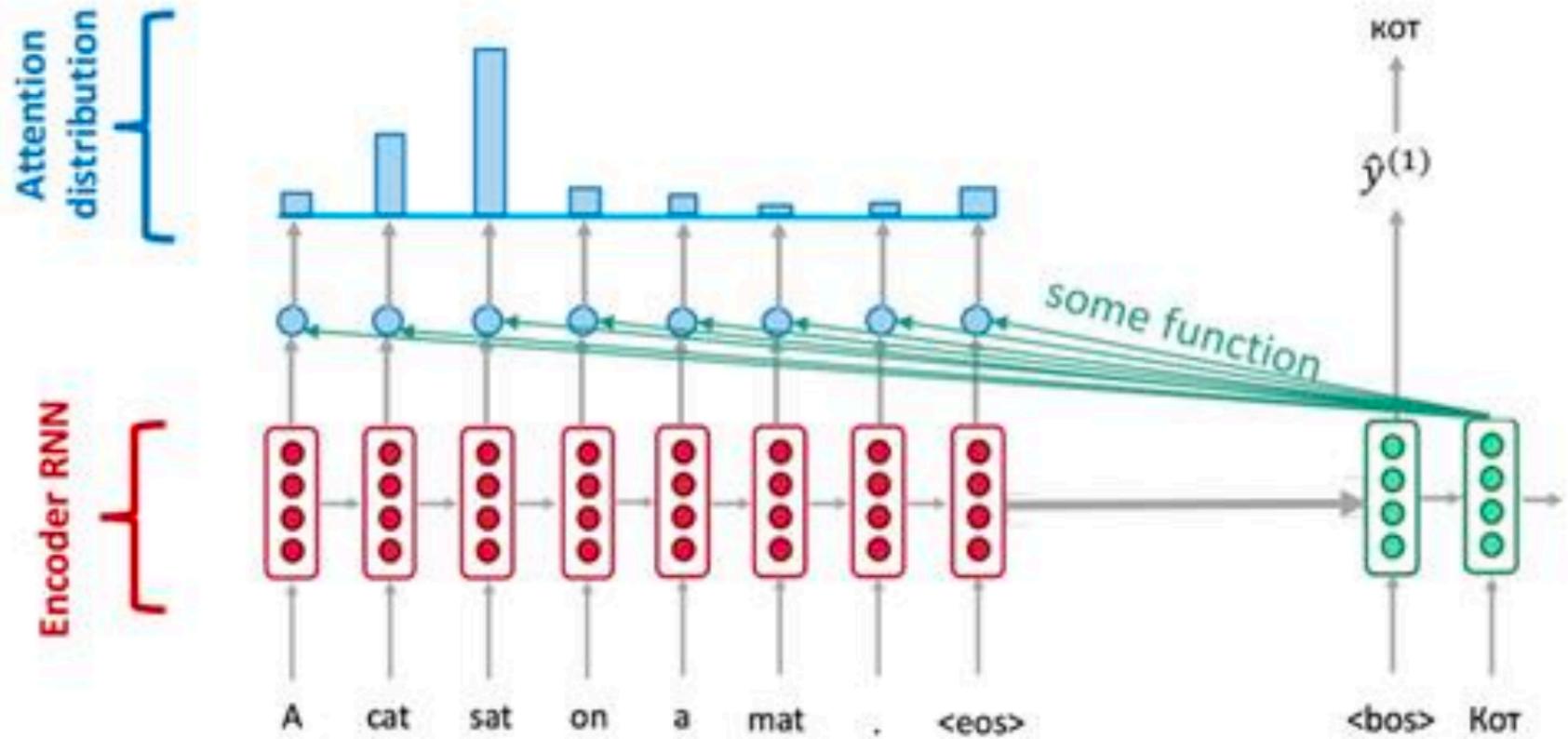


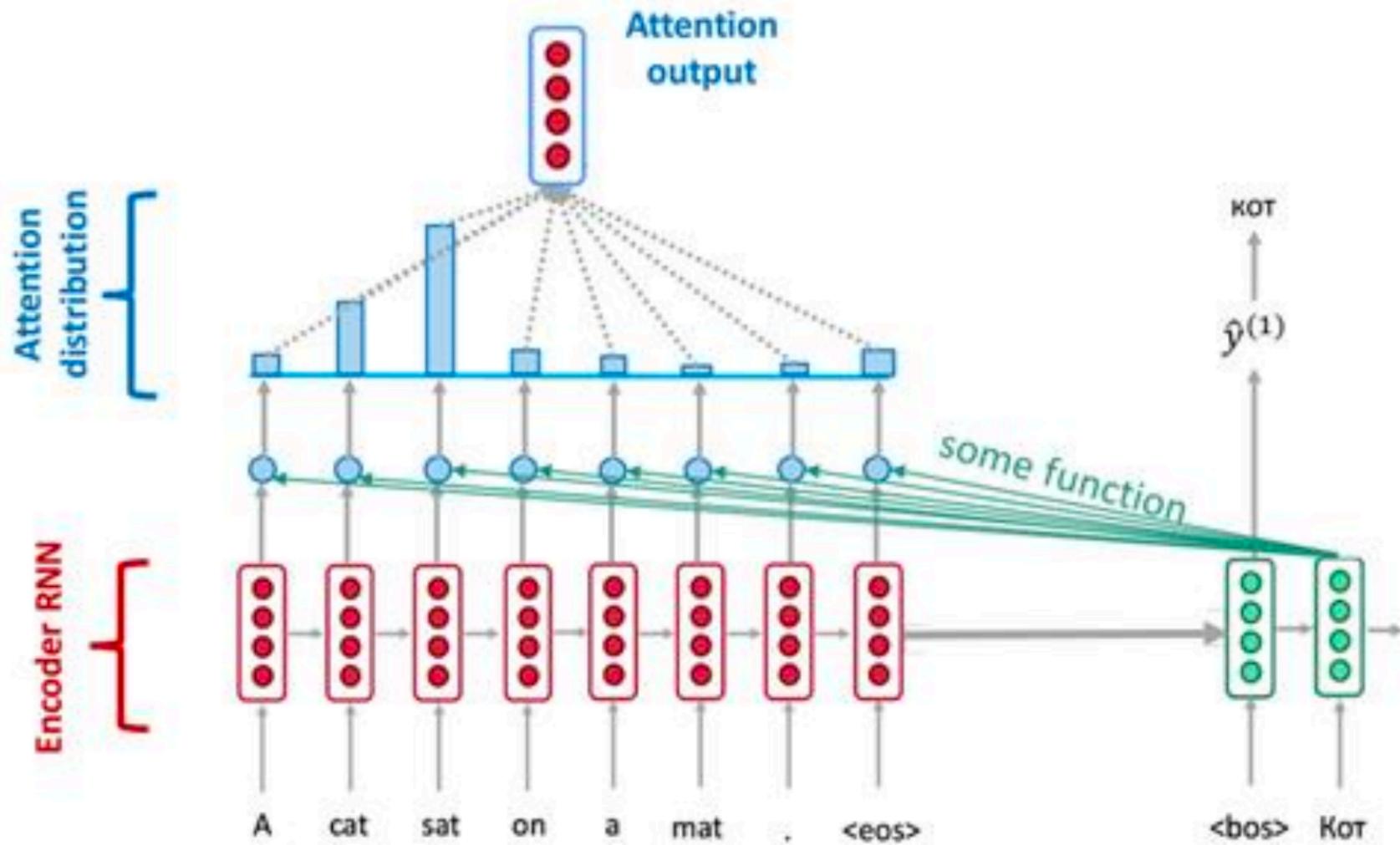


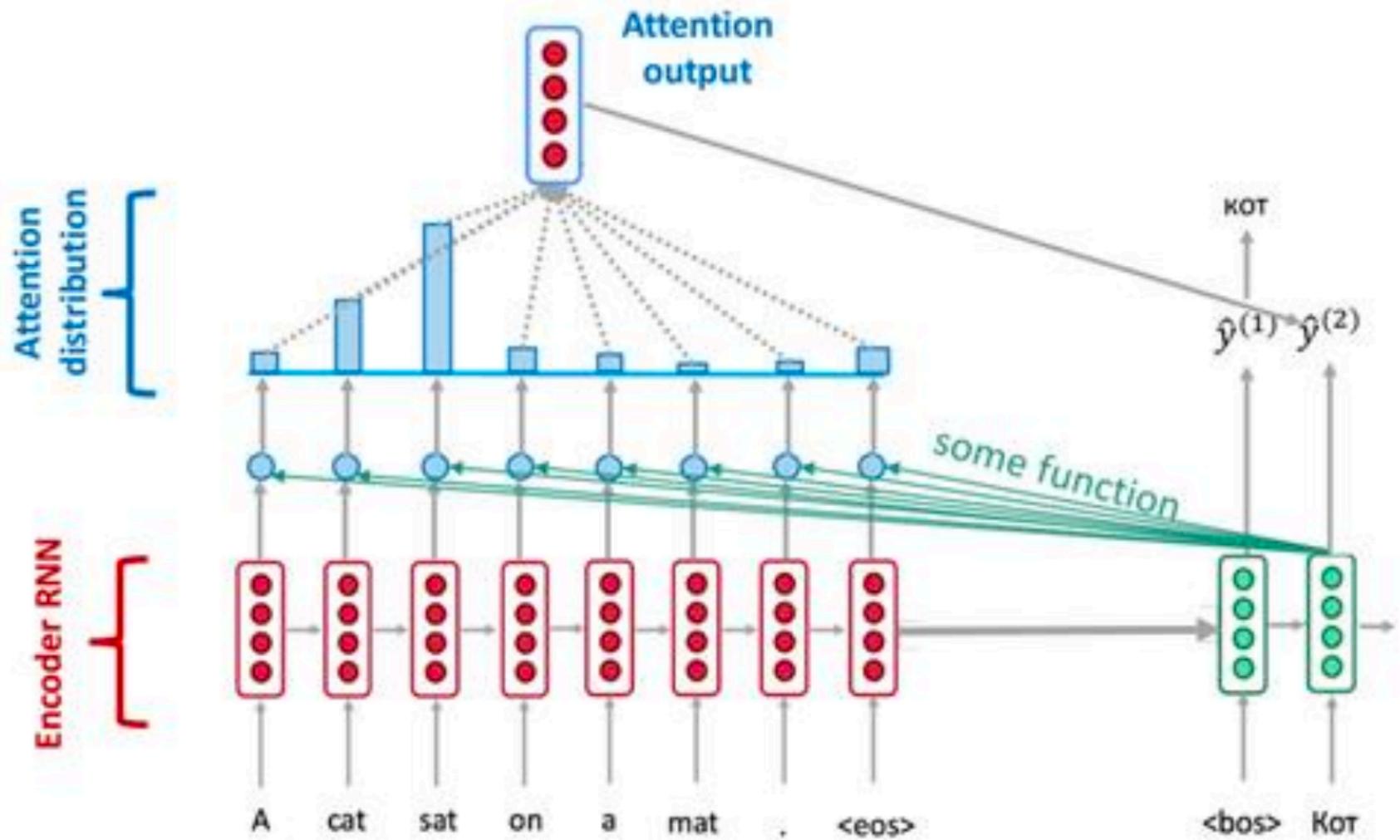


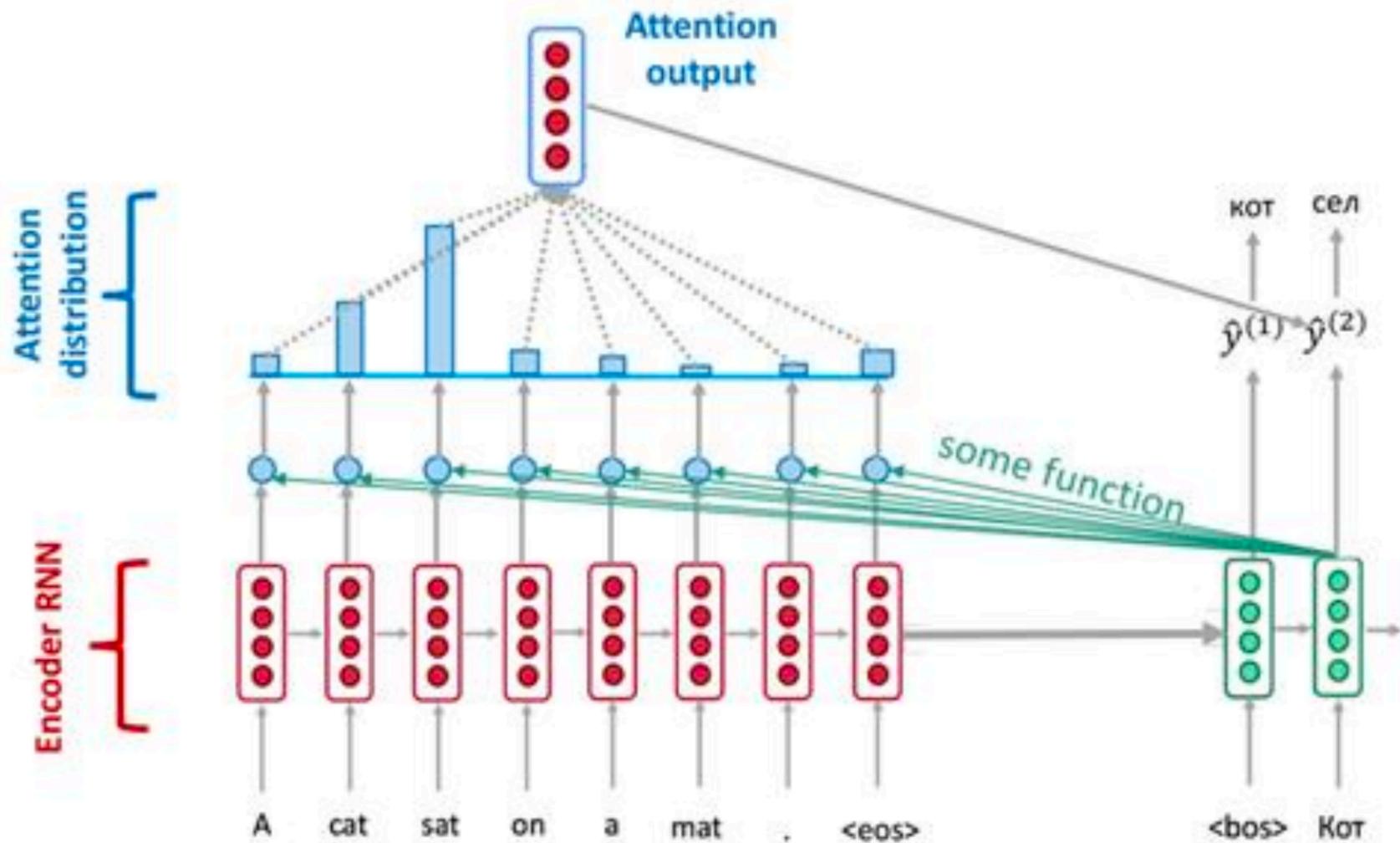


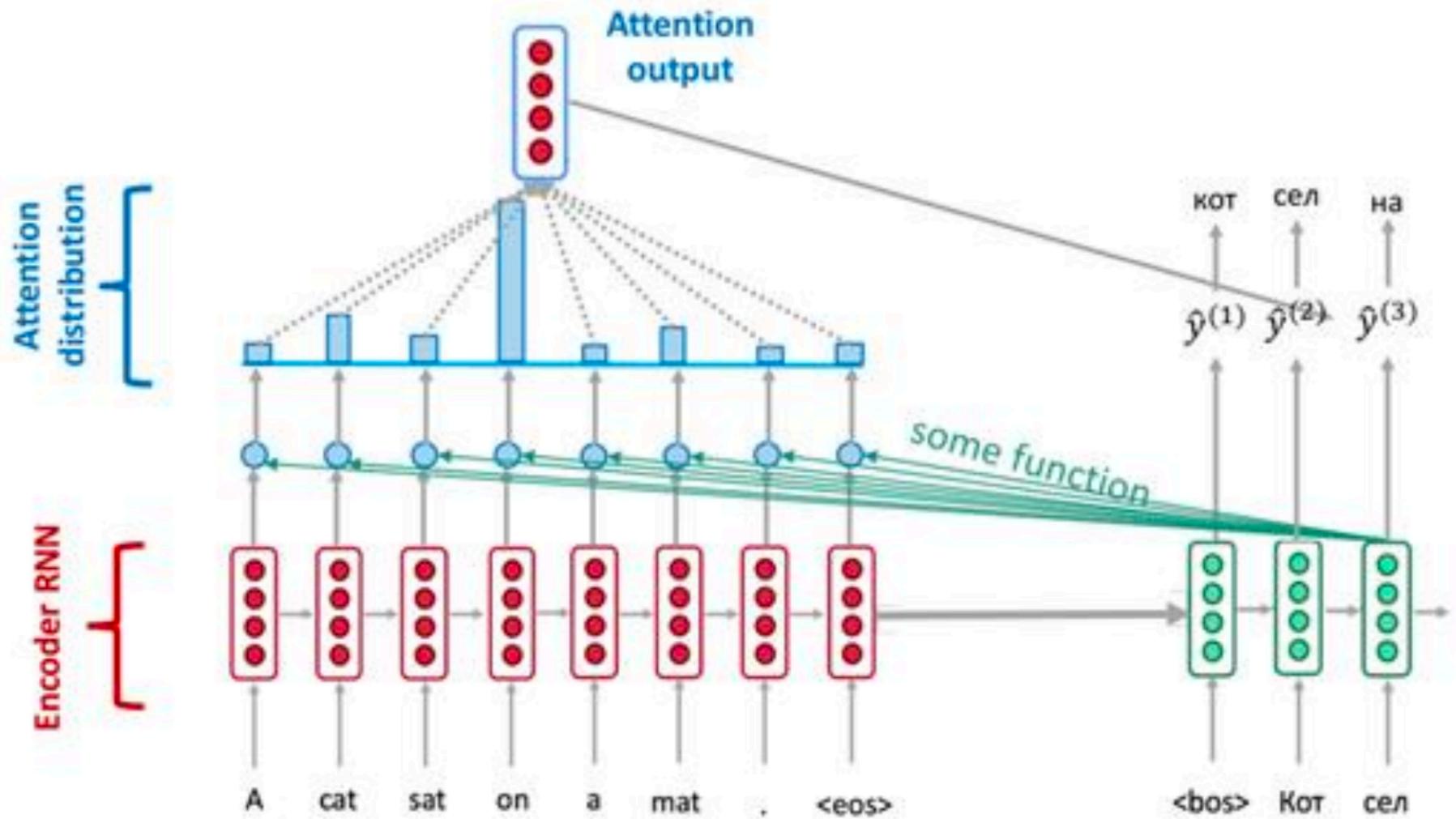


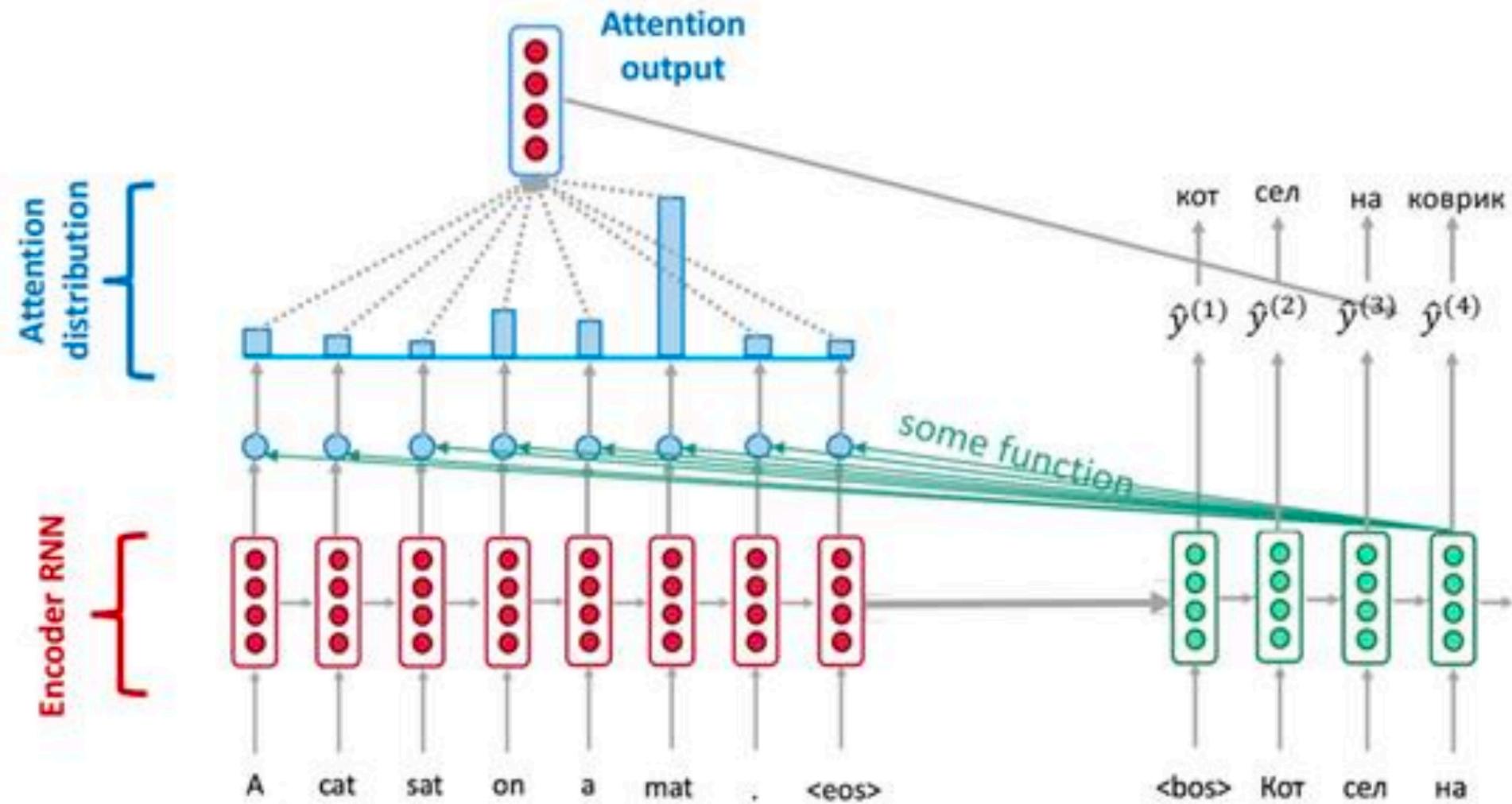


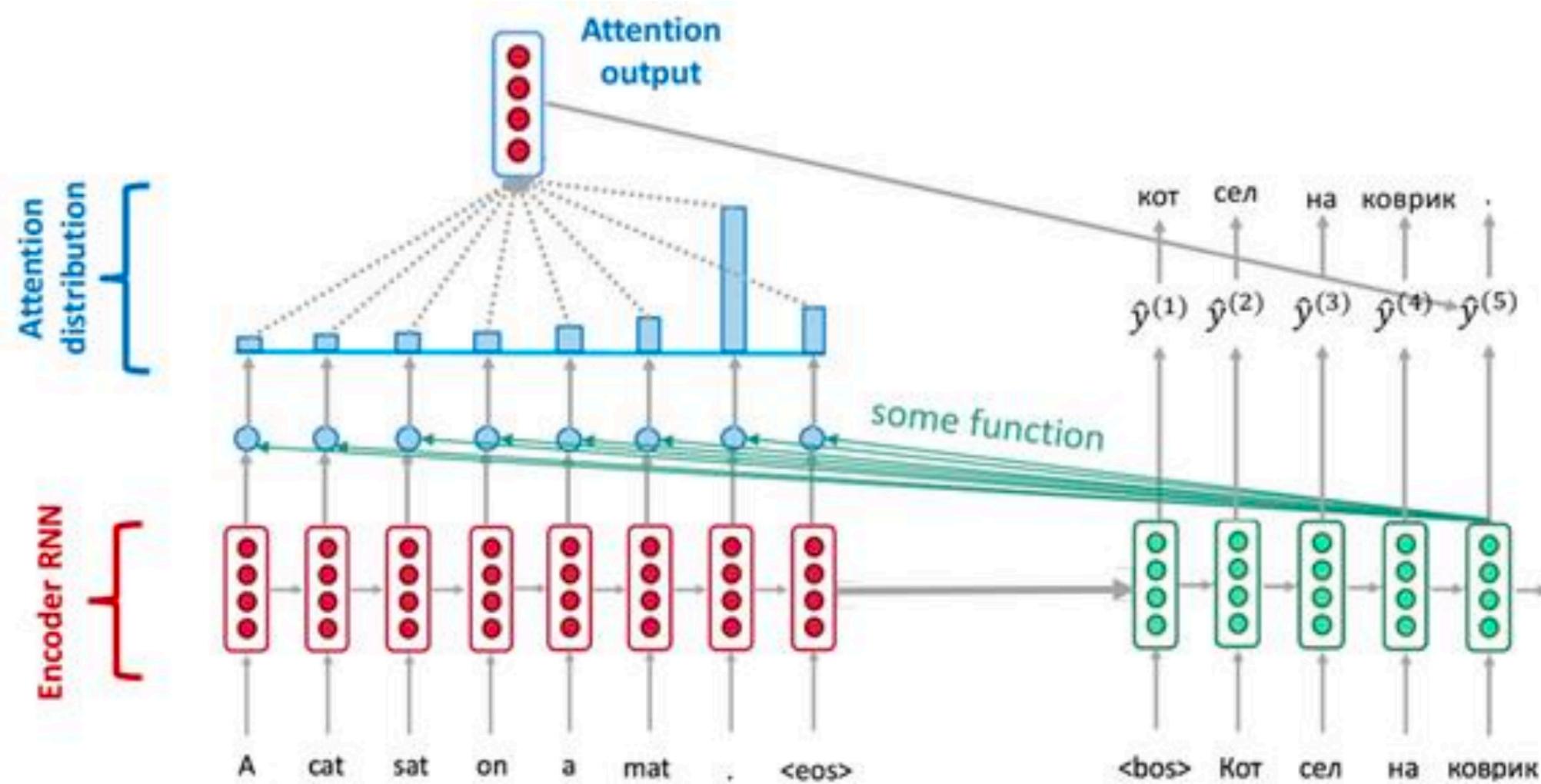


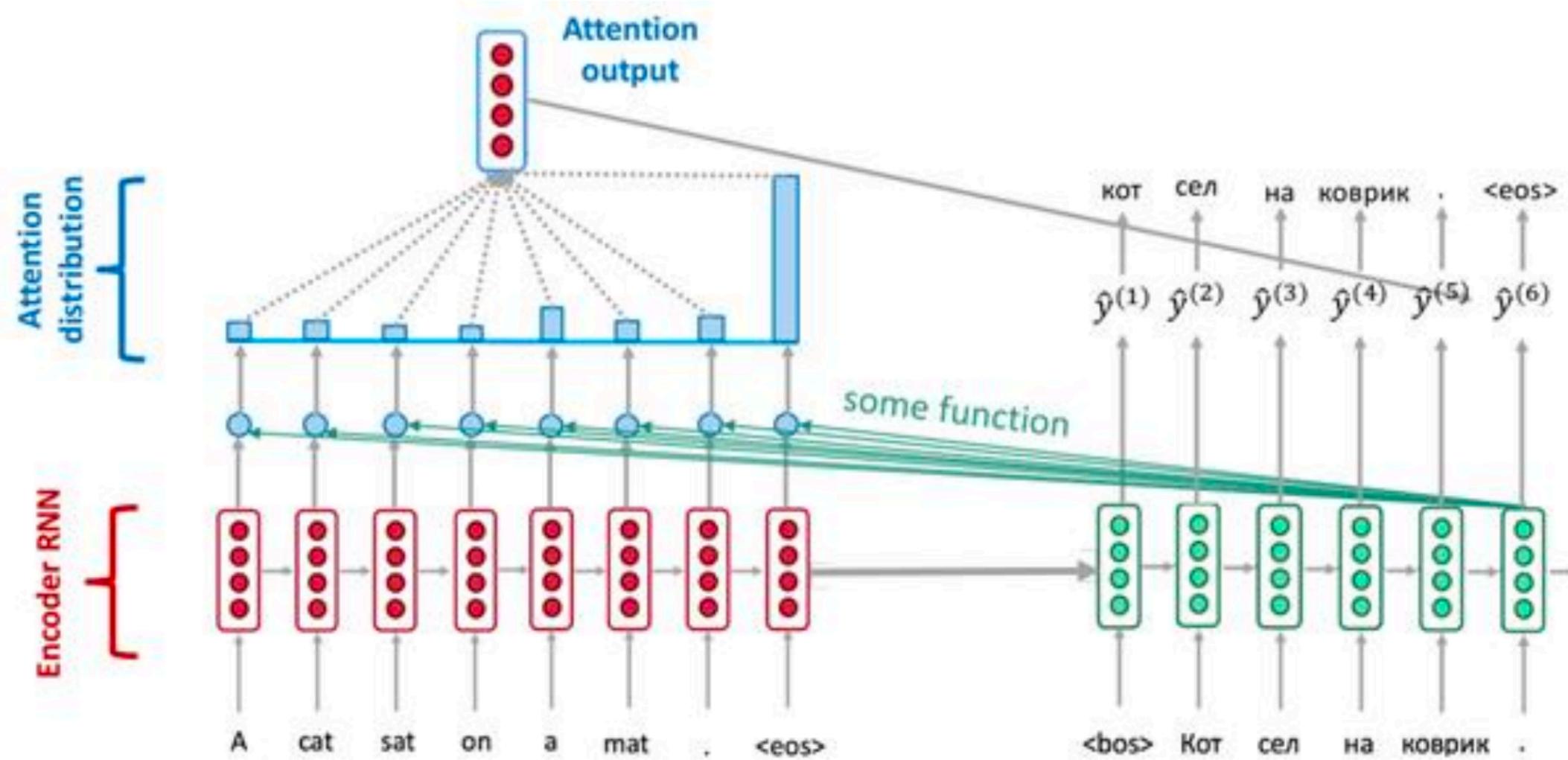




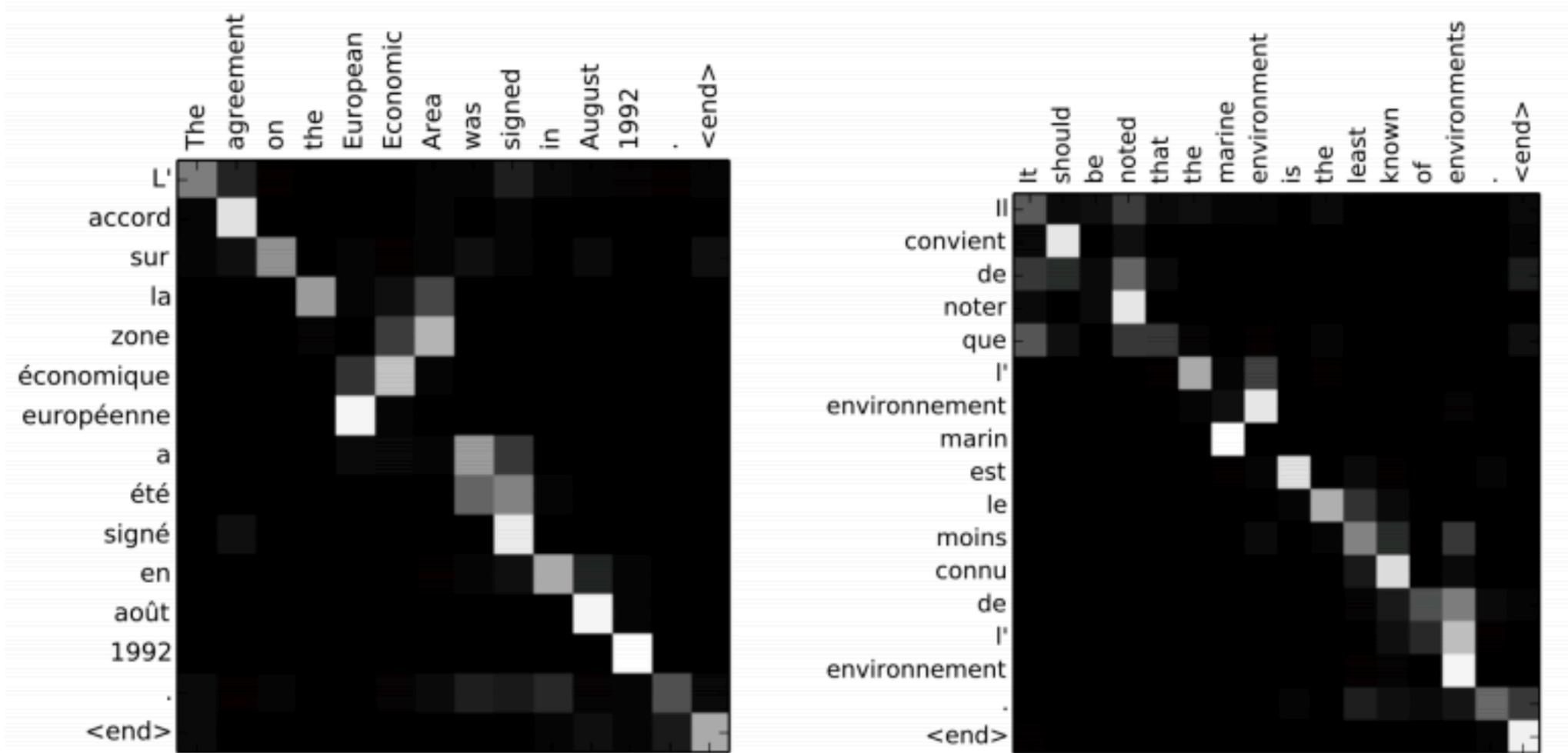




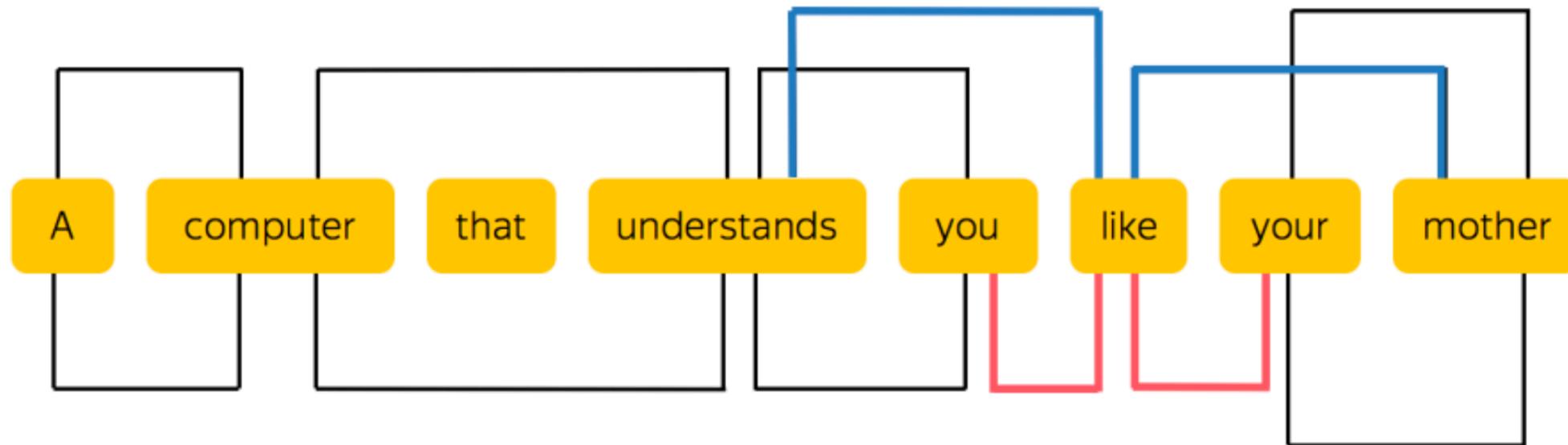




Визуализация Attention



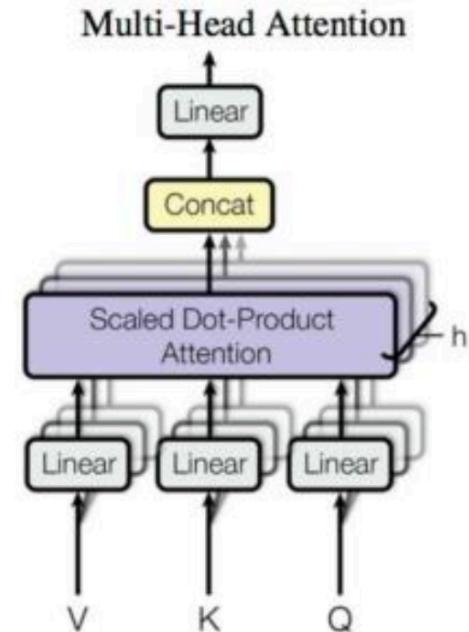
Self-attention



Multi-head Attention

Она руководит **новым** проектом

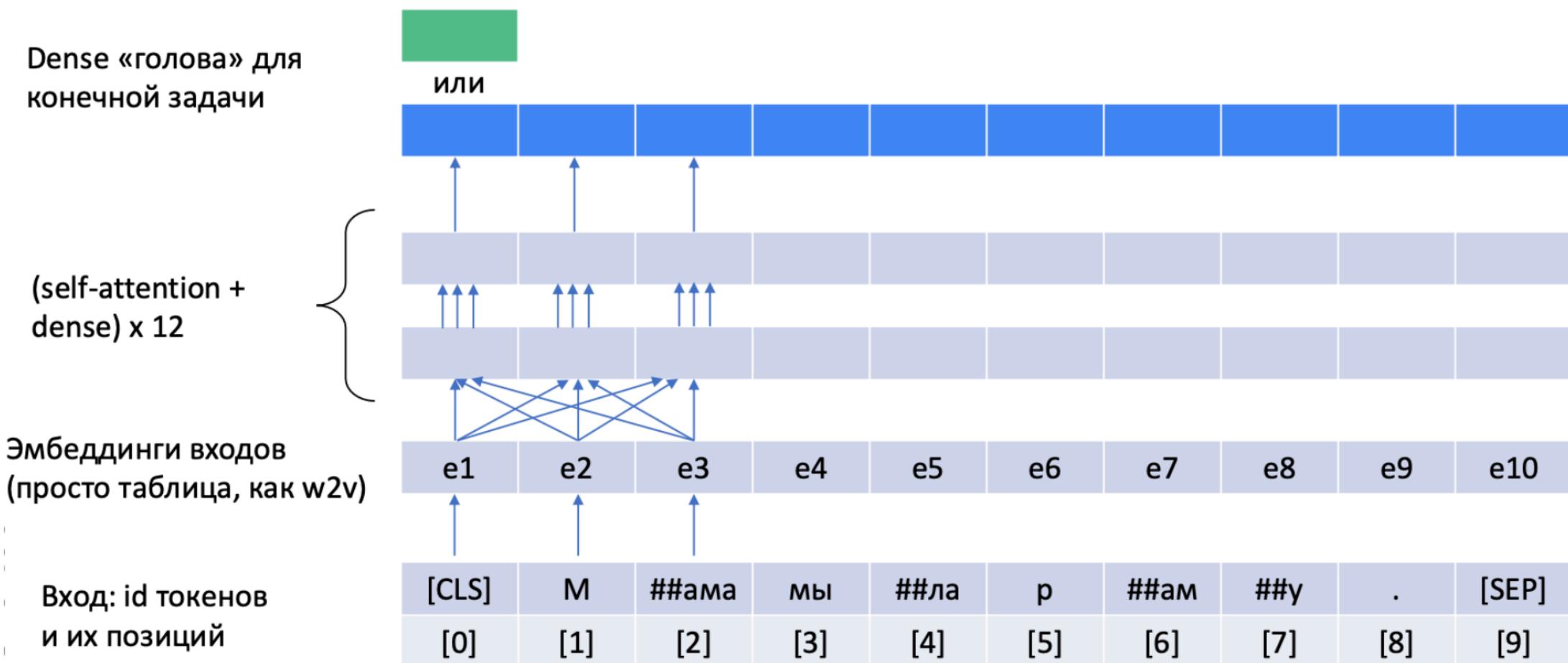
- Gender agreement
- Case government
- Lexical preferences
- ...



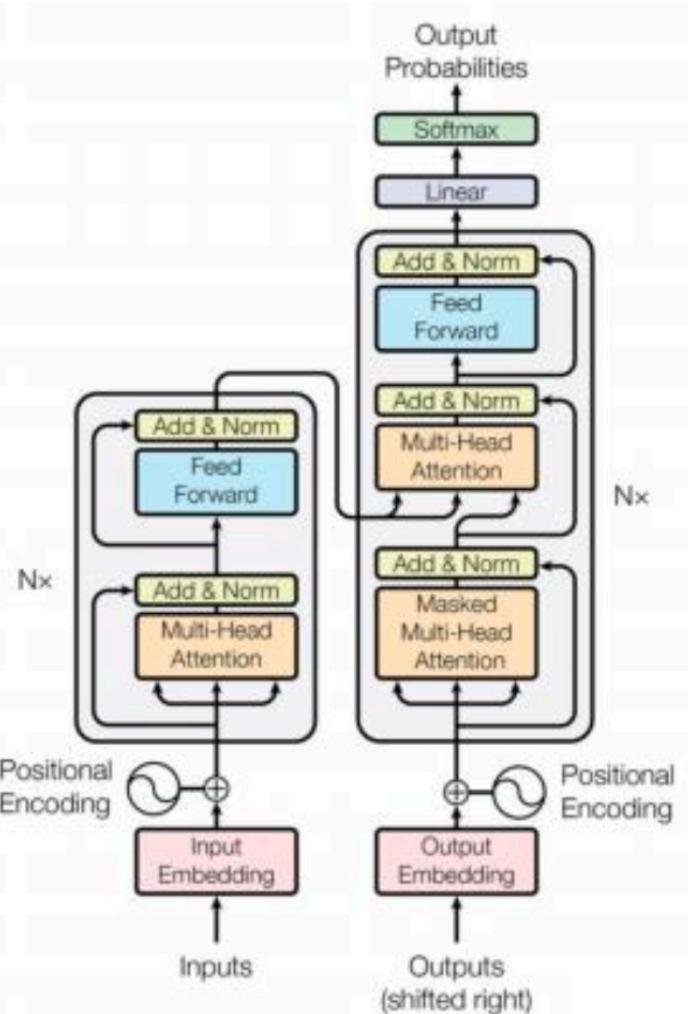
$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$

Устройство модели типа Transformer



Устройство модели типа Transformer



Преимущества:

- Нет рекуррентности, параллельное кодирование
- Быстрое обучение: encoder и decoder могут быть параллельны
- Нет длинных связей.
- Multi-head attention позволяет обращать внимание на разные аспекты

GPT

- Generative pre-trained transformer
- Семейство нейросетей от компании OpenAI
 - Есть русскоязычные аналоги: ruGPT-3 от Сбера
- Обучались исключительно на задаче продолжения текста
- Огромные
 - GPT-2 содержит 1.5 млрд параметров, обучалась на 40 ГБ текста
 - Можно скачать и запустить/дообучить локально
 - GPT-3 содержит 175 млрд параметров, обучалась на 500 ГБ текста
 - Доступна только по API
- Генерируют очень правдоподобные тексты

Пример генерации

RuGPT3 demo

Ведите текст:

Самое удивительное животное - это

RuGPT3: Самое удивительное животное - это кальмар. Чуть больше теннисного мяча.

А этих работяг вообще выковыривают из воды за ноги.

И это все - единственный вид тропических рыб, водящихся у нас. Встал рано, пошел смотреть на кальмаров.

Дополнить

<https://russiannlp.github.io/rugpt-demo>

Модели типа BERT: только энкодер

- Классификация текстов
- Named entity recognition
- Question answering

Модели типа BERT: только энкодер

- Классификация текстов
- Named entity recognition
- Question answering

Модели типа GPT: только декодер

- Продолжение текста по образцу
- Диалоги (болталка)
- Оценка вероятности текстов
-

Модели типа BERT: только энкодер

- Классификация текстов
- Named entity recognition
- Question answering

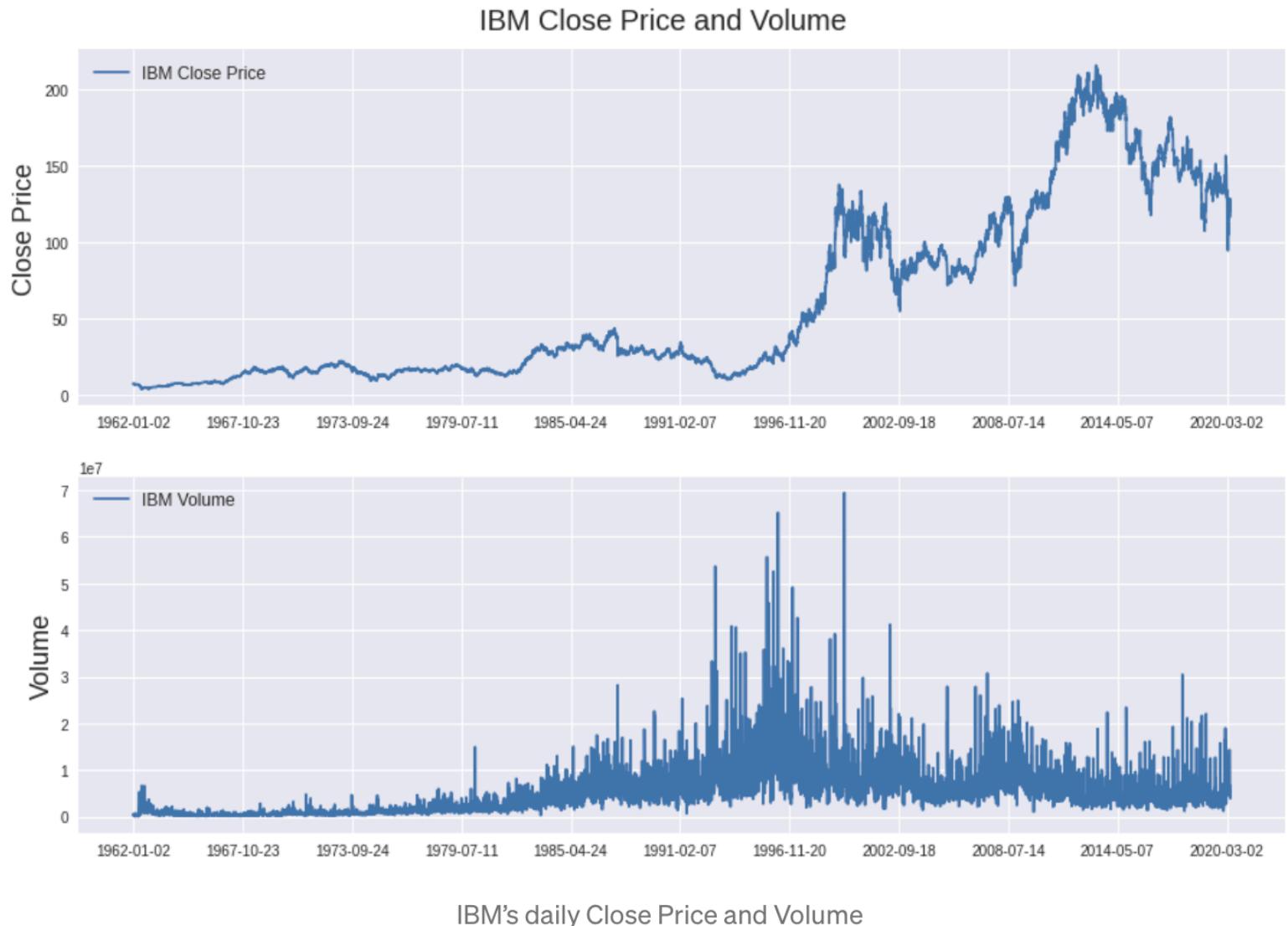
Классический трансформер

- Перевод
- Суммаризация
- Перефразирование

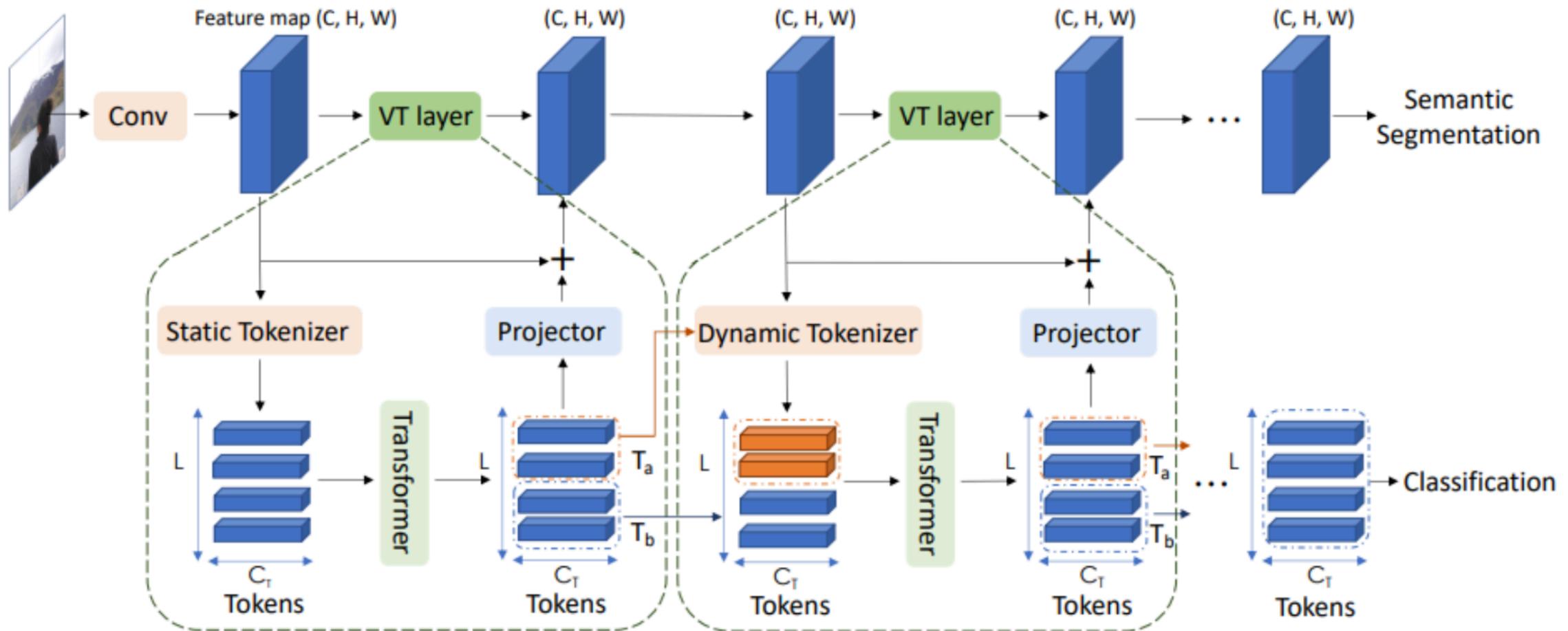
Модели типа GPT: только декодер

- Продолжение текста по образцу
- Диалоги (болталка)
- Оценка вероятности текстов
-

Трансформеры для анализа временных рядов



Трансформеры для Computer Vision



Заключение

Трансформер — наиболее мощная современная архитектура

- Хорошо подходит для параллельного обучения
- Может использоваться в разных сферах AI
- Есть множество готовых моделей для решения разных задач

Недостатки

- Требуется очень много данных для обучения
- И вычислительных ресурсов тоже
- Большая стоимость обучения новых моделей
- Очень объемные и тяжелые модели, сложно использовать локально
- Особенно на мобильных платформах
- С трудом поддаются интерпретации

Модели типа BERT: только энкодер

- Классификация текстов
- Named entity recognition
- Question answering

Классический трансформер

- Перевод
- Суммаризация
- Перефразирование

Модели типа GPT: только декодер

- Продолжение текста по образцу
- Диалоги (болталка)
- Оценка вероятности текстов
-

Как оценивать модели

- Классификация текстов и токенов: классические метрики
 - Accuracy, precision, recall, F score, ROC AUC, ...
- Генерация текстов: всё сложно
 - Сравнение с образцом в лоб даёт низкую точность
 - Ручная оценка – долгая, дорогая и субъективная
 - Особенно если делать разметку с перекрытием
 - Автоматические метрики: что проверять будем?
 - Вероятность генерации моделью образца (*перплексия*)
 - Нечёткое совпадение с образцом
 - Естественность
 - Наличие желаемых атрибутов (например, стиль)
 - Для этого обычно нужны кастомные классификаторы

Метрики качества

- Сравнение текстов
 - BLEU, ROUGE, METEOR – доля совпавших последовательностей слов
 - Разница в том, как учитываются совпадения и как нормируется доля
 - Word mover distance, BERTScore – близость эмбеддингов пар слов
 - При вычислении пары вычисляются оптимальным образом
 - Обучаемые метрики (в основном – трансформеры)
 - BLEURT, NUBIA, SimCSE, и множество других
 - Высокое качество на целевой задаче, но не всегда хорошо обобщаются
- Естественность
 - Оценка правдоподобия текста генеративной моделью (типа GPT)
 - Обучаемые классификаторы естественности (например, CoLA)