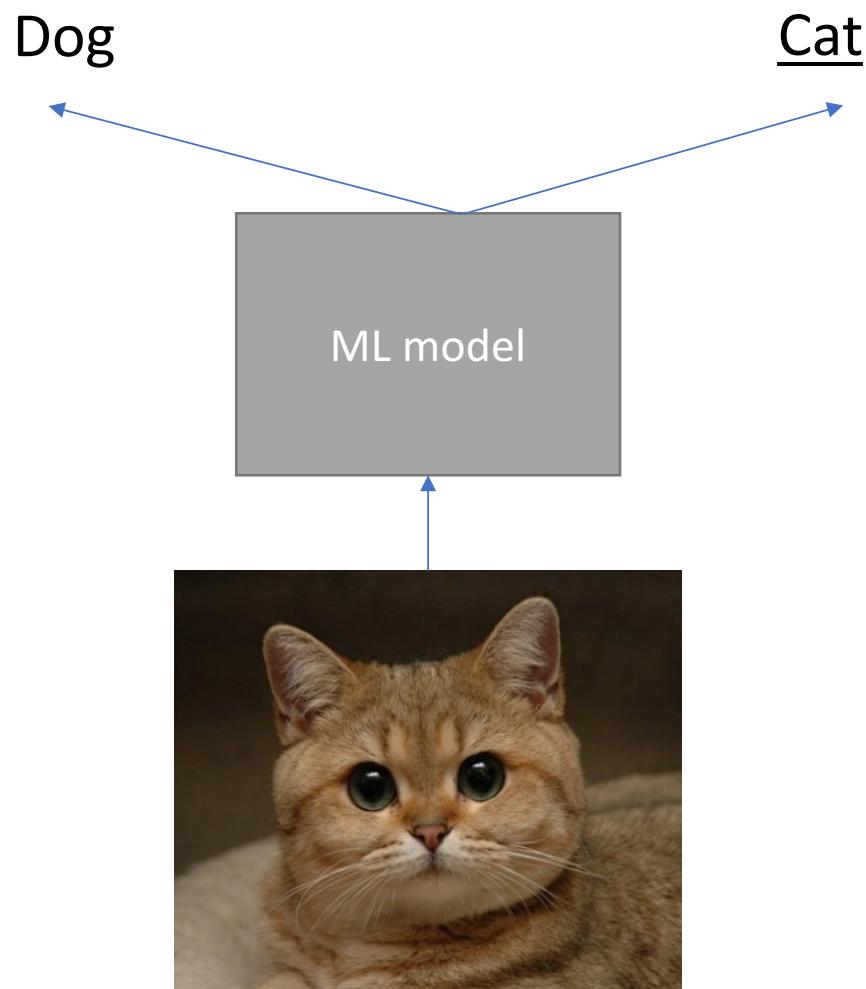


NLP

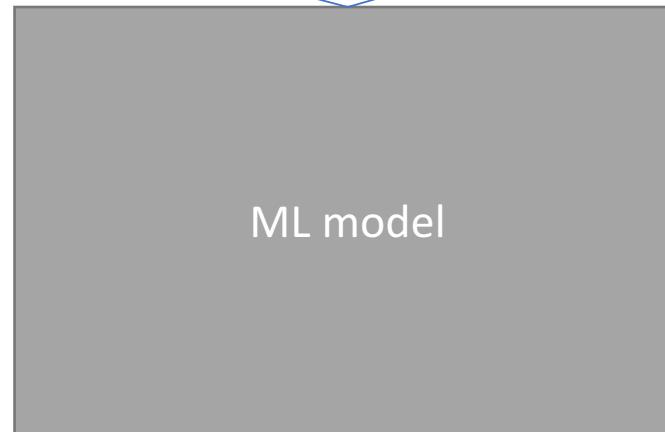
Text, speech, last lecture

Classification



Abusive content

Non-abusive content



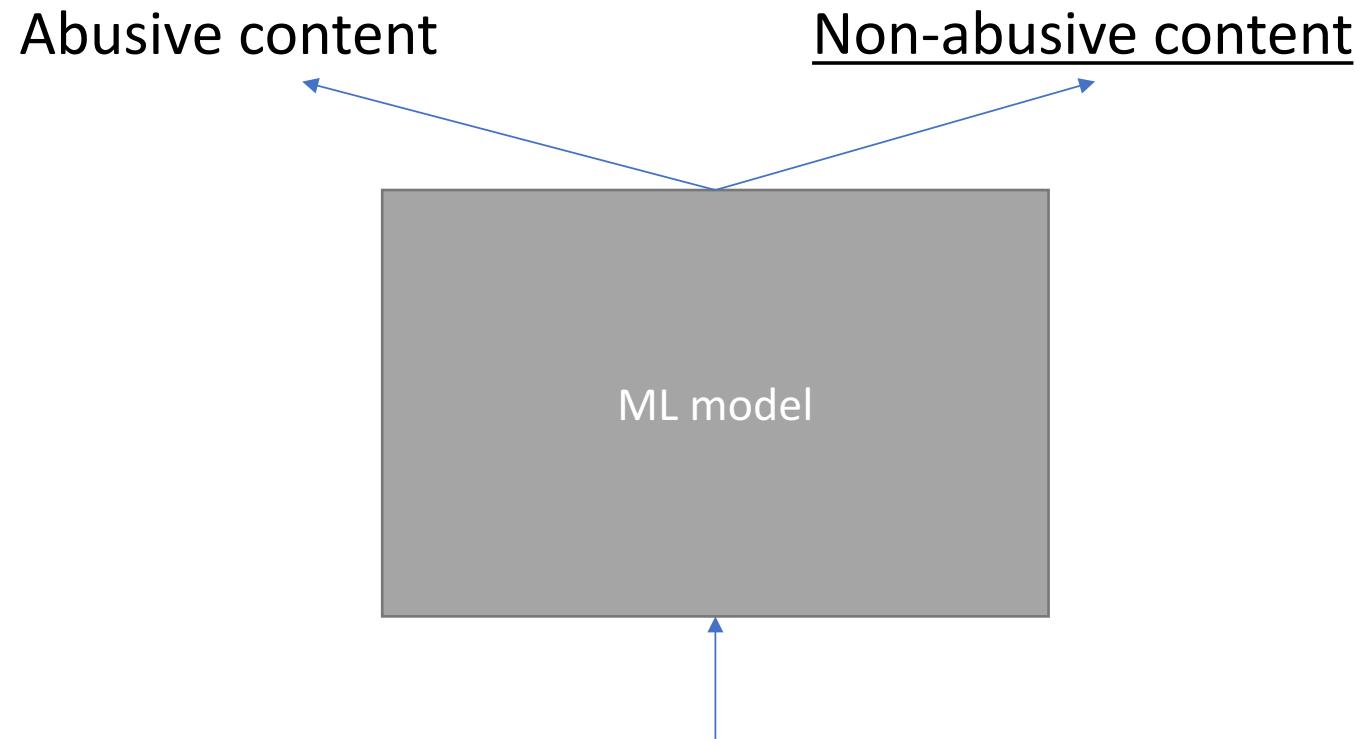
А ты чистил когда-нибудь клавиатуру свою?

NLP tasks

- Automatic speech recognition
- CCG supertagging
- Common sense
- Constituency parsing
- Coreference resolution
- Dependency parsing
- Dialogue
- Domain adaptation
- Entity linking
- Grammatical error correction
- Information extraction
- Language modeling
- Lexical normalization
- Machine translation
- Multi-task learning
- Multi-modal
- Named entity recognition
- Natural language inference

- Part-of-speech tagging
- Question answering
- Relation prediction
- Relationship extraction
- Semantic textual similarity
- Semantic parsing
- Semantic role labeling
- Sentiment analysis
- Shallow syntax
- Simplification
- Stance detection
- Summarization
- Taxonomy learning
- Temporal processing
- Text classification
- Word sense disambiguation

Какие здесь могут быть сложности?



А ты чистил когда-нибудь клавиатуру свою?

New problems

- Как представить текст в виде вектора?
- Текст – это последовательность слов
- Слова разной длины
- Предложения разной длины
- Нужно учитывать схожесть слов по смыслу и различные варианты написания

NLP без deep learning

- Bag of words, Tf-idf
- Word2vec, FastText, ELMO

Bag of words

- Мешок слов – способ представления текста в виде вектора
- Текст в виде счетчиков вхождения слов
- Размер вектора - число уникальных слов в текстах

Текст 1: “текст состоит из слов”

Текст 2: “вхождения данного слова среди всех слов”

	текст	состоит	слово	вхождение	данный	все
текст 1	0.33	0.33	0.33	0	0	0
текст 2	0	0	0.4	0.2	0.2	0.2

Недостатки Bag of words

- Не учитывает порядок слов и связи между словами
- Зависит от препроцессинга

Препроцессинг

- Токенизация
- Удаление редких и стоп-слов
- Стемминг и лемматизация

Токенизация

- Разбиение текста на токены
- Удаление символов пунктуации и спец-символов
- Добавление новых токенов
- Lowercase

Текст: Узнаю ли я хоть ЧТО-НИБУДЬ новое?!!!

Токены: ['узнаю', 'ли', 'я', 'хоть', 't_up', 'что-нибудь',
'новое', '?', 't_rep 3', '!']

Удаление редких и стоп-слов

- Редкие слова - встречаются всегда несколько раз на большой текст (не оказывают достаточного влияния)
- Стоп слова - союзы, предлоги, очень часто встречающиеся слова

Стемминг и Лемматизация

- Стемминг - от каждого слова отрезается его окончание
 - + Ходила - ходил, ходили - ходил
 - - Был, есть, будет
 - + Просто и быстро работает
- Лемматизация - слова приводятся к начальной форме по словарю
 - + Ходила -ходить, ходила -ходить
 - + Был - есть, есть - есть, будет - есть
 - - Медленно работает

Как учитывать связи и порядок слов?

- N-gram - последовательность из n идущих подряд слов в тексте
 - униграммы (n=1), биграммы (n=2), триграммы (n=3)
- K-skip-N-gram - последовательность из n идущих подряд слов в тексте, причём расстояние между соседними должно составлять не более k токенов
- **Пример: "Набор подряд идущих токенов"**
- 2-gram: набор подряд, подряд идущих, идущих токенов
- 1-skip-2-gram: набор подряд, подряд идущих, идущих токенов, набор идущих, подряд токенов

N-grams

Source Text

The quick brown fox jumps over the lazy dog. →

The quick brown fox jumps over the lazy dog. →

The quick brown fox jumps over the lazy dog. →

The quick brown fox jumps over the lazy dog. →

Training Samples

(the, quick)
(the, brown)

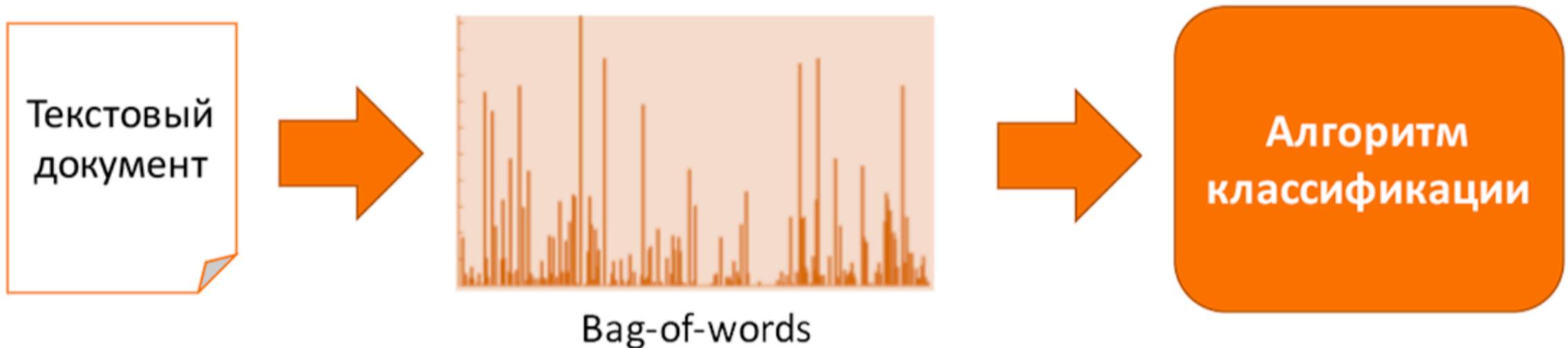
(quick, the)
(quick, brown)
(quick, fox)

(brown, the)
(brown, quick)
(brown, fox)
(brown, jumps)

(fox, quick)
(fox, brown)
(fox, jumps)
(fox, over)

Bag of words pipeline

- Препроцессинг текста
- Bag-of-words на словах и N-grammax как векторное представление текста
- Linear models, SVD, Random forest и т.д. для классификации



Word2Vec

- Другой способ векторного представления
- То, что называют "pretrained word embeddings"
- Статья издана 7 сентября 2013 (Google research)
- Существуют разновидности идей Word2Vec
 - FastText (9 августа 2016, Facebook research)
 - ELMO (22 марта 2018, Allen Institute for Artificial Intelligence)

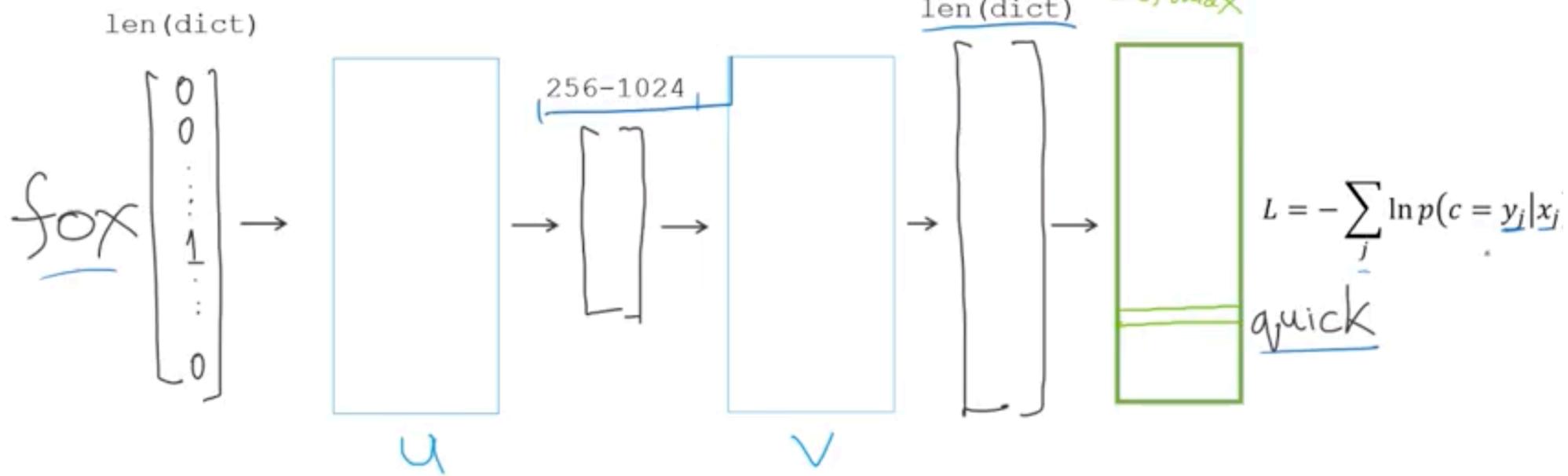
Simon Says about Word2Vec



word2vec

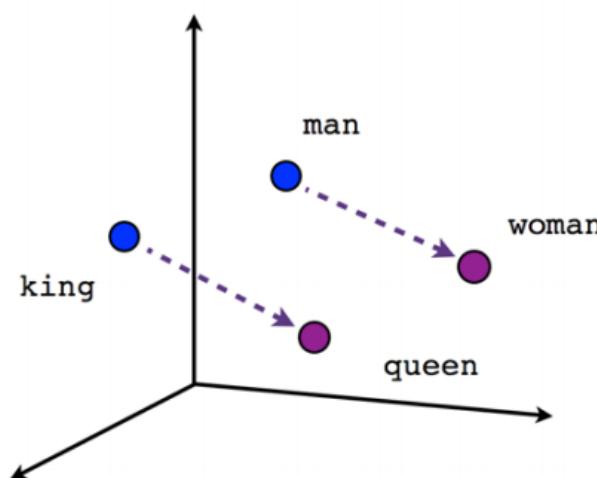
The quick brown fox jumps over the lazy dog

fox -> quick
fox -> brown
fox -> jumps
fox -> over

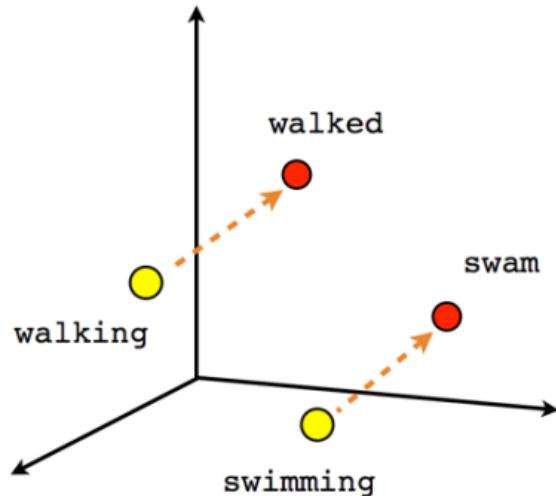


Свойства Word2Vec

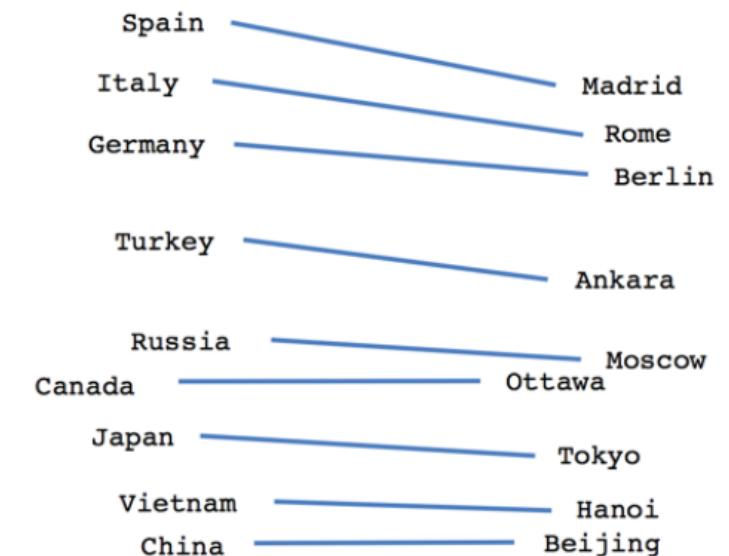
- King - Man + Woman = Queen
- Paris - France + Italy = Rome



Male-Female



Verb tense



Country-Capital

TSNE on Word2Vec vectors



Минусы эмбеддингов

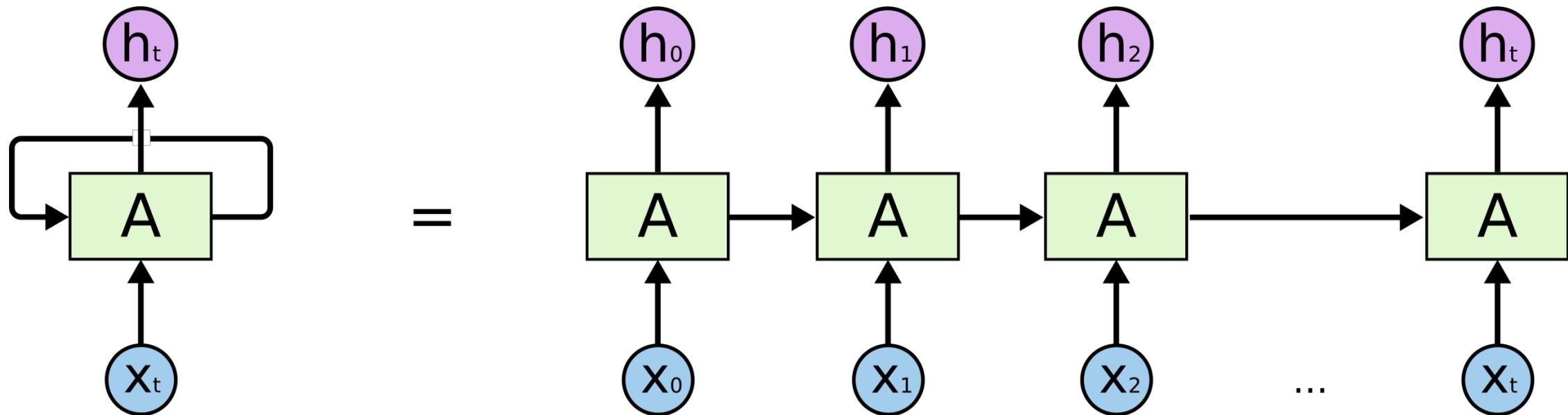
- Эмбеддинги рандомно инициализируют слова не из словаря
- Эмбеддинги нельзя использовать с разными языками сразу

Neural networks

- Recurrent NN
- Convolution NN
- Transformer

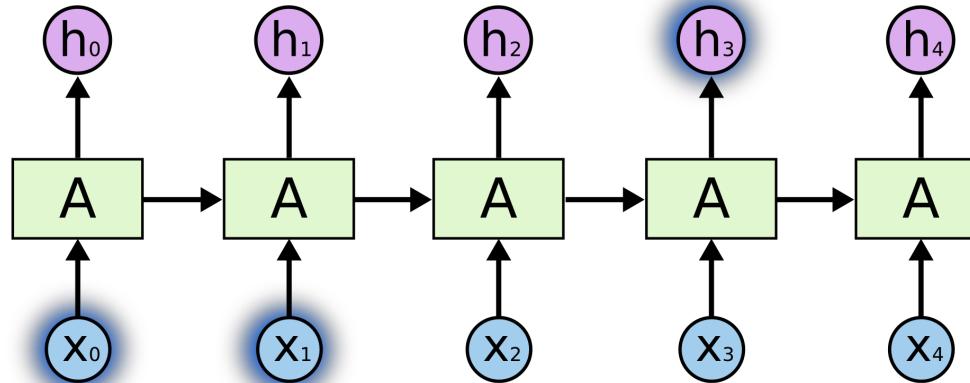
Recurrent Neural Networks (RNN)

- Выход каждого слоя подаем на вход предыдущему
- Количество пробросов ограничено (обычно около 80)
- Градиент прорасывается по всей цепочке (BPTT)

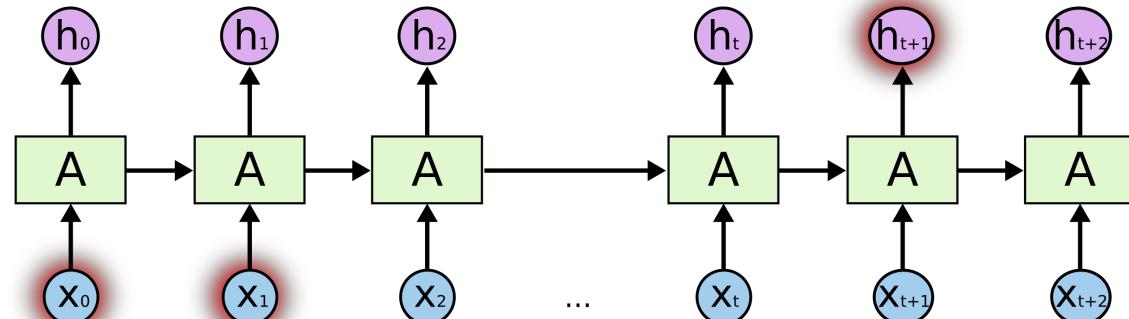


The Problem of Long-Term Dependencies

- The clouds are in the *sky*

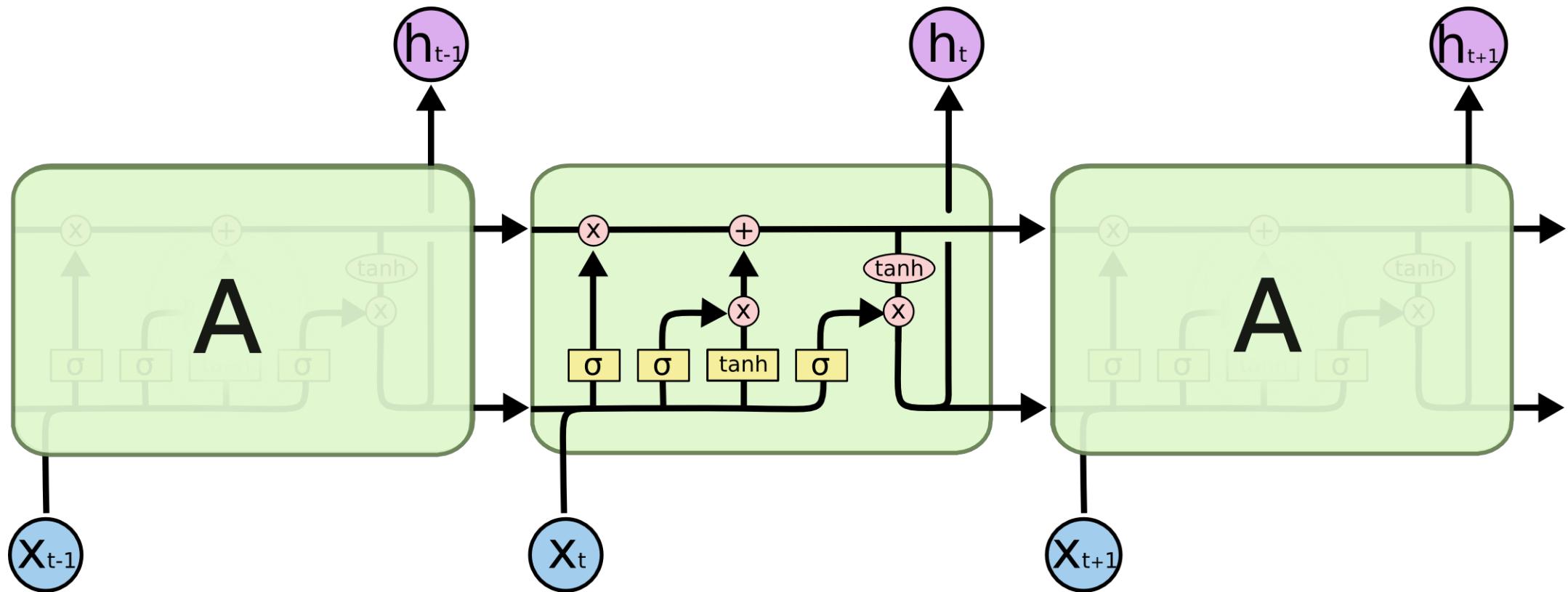


- I grew up in France... I speak fluent *French*.



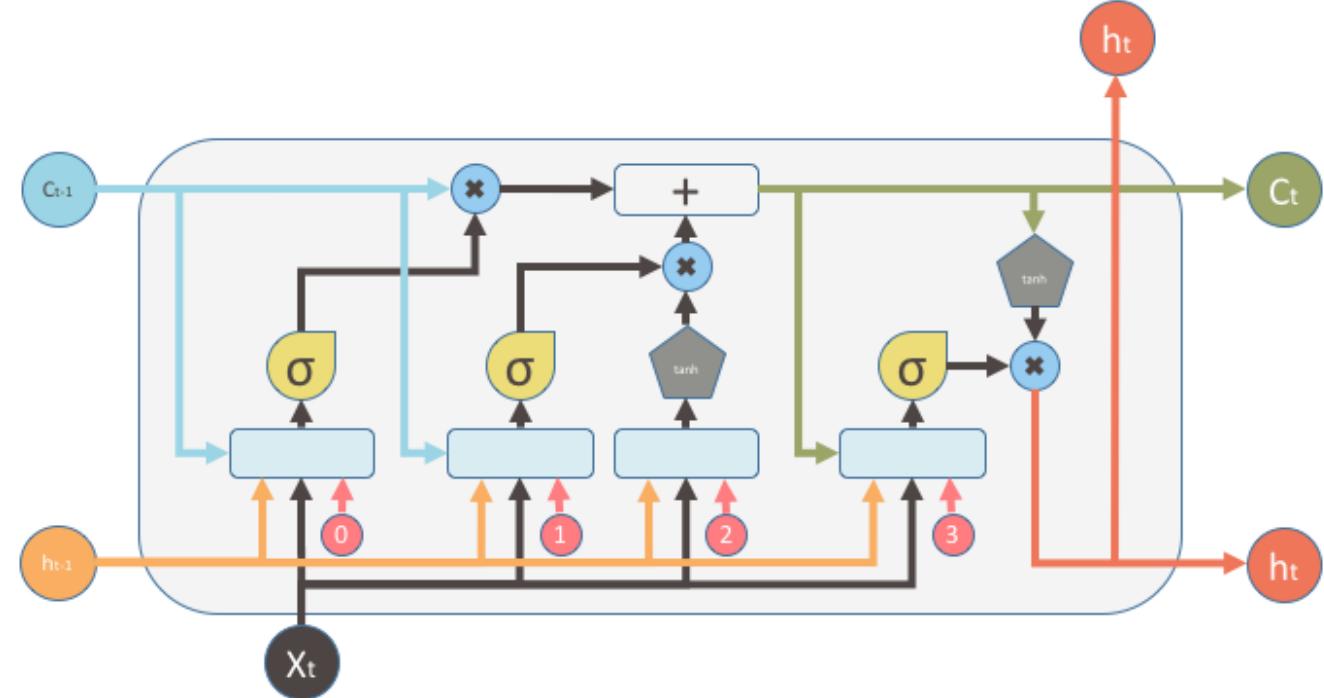
Solution: Gated Recurrent Neural Networks

- LSTM (на картинке) и GRU



LSTM cell in detail

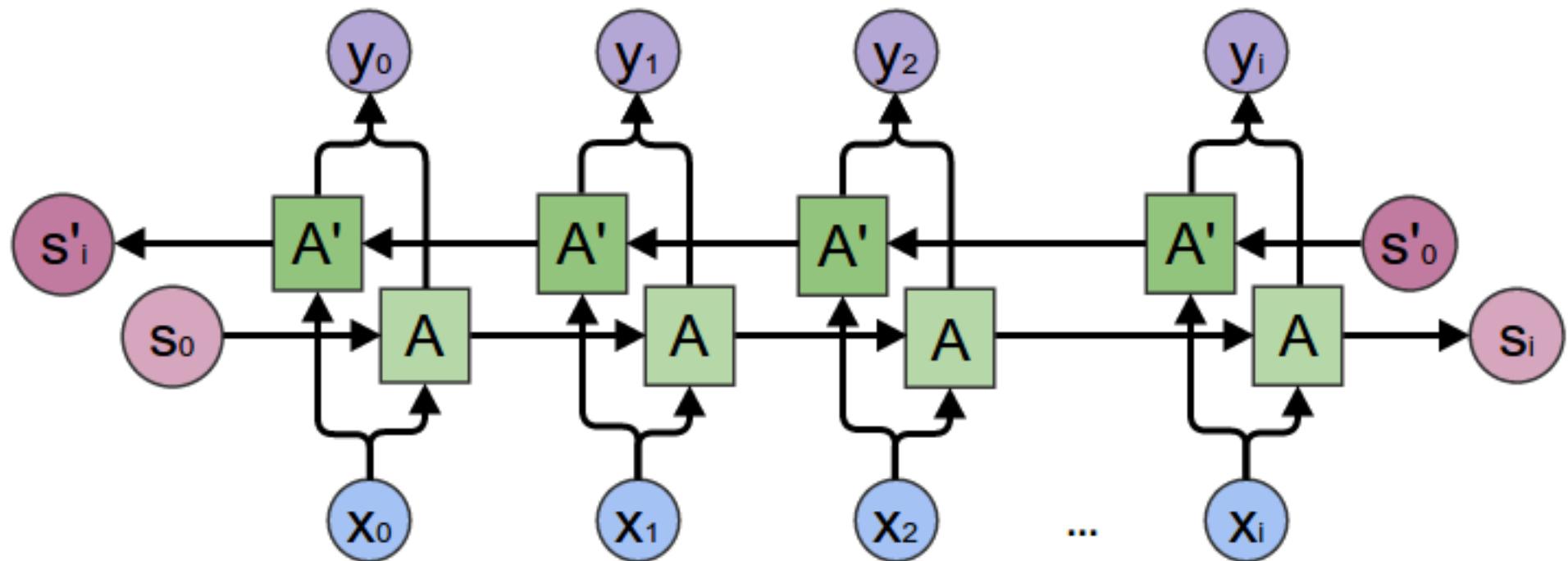
- Forget gate layer
 - Забыть либо запомнить
- Input gate layer
- Output gate layer



Inputs:	outputs:	Nonlinearities:	Vector operations:
X_t	C_t Memory from current block	σ Sigmoid	\times Element-wise multiplication
C_{t-1} Memory from previous block	h_t Output of current block	\tanh Hyperbolic tangent	$+$ Vector addition
h_{t-1} Output of previous block	Bias: 0		
	Linear (dense) NN layer for concatenated inputs Equivalent to: matrix-weighted sum of each input		

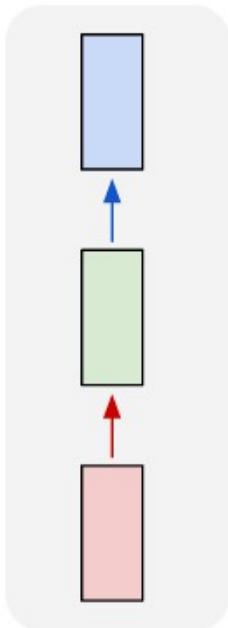
Bidirectionality (BiLSTM)

- В конце предложения забываем что было вначале
- Поэтому проходим в обе стороны и объединяем

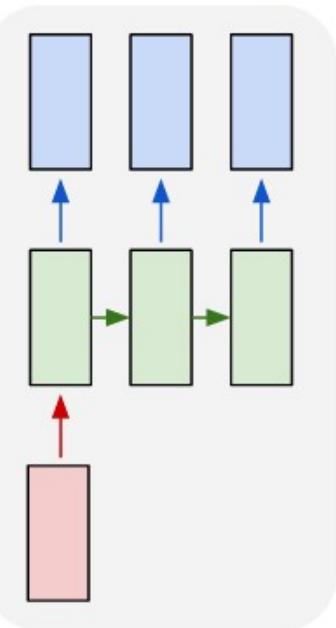


Recurrent architectures

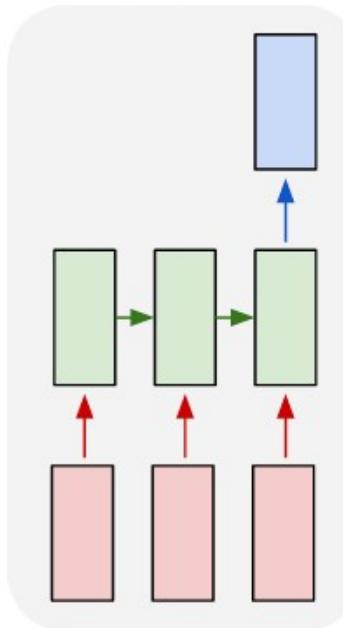
one to one



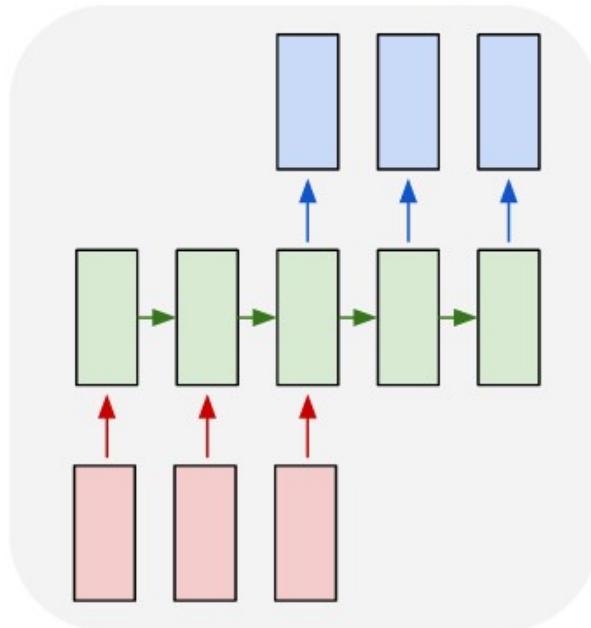
one to many



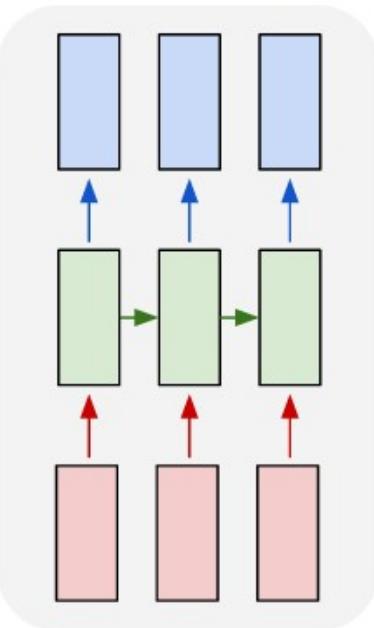
many to one



many to many

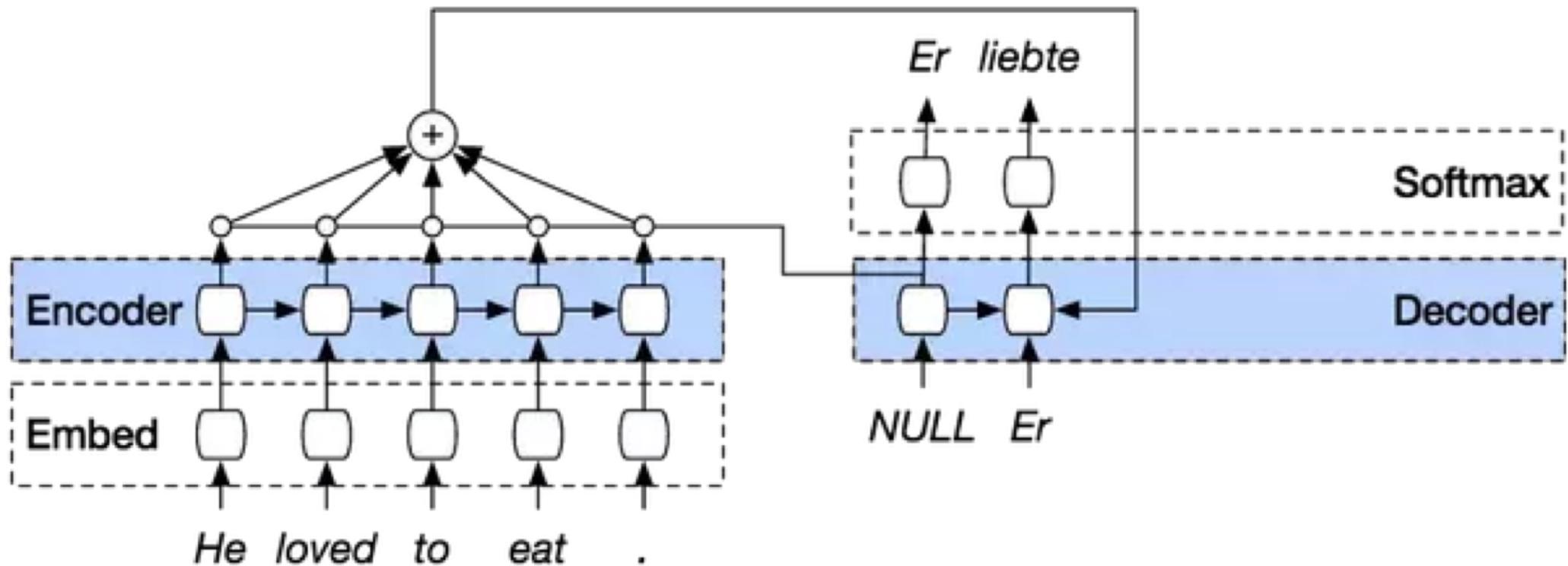


many to many



Я - король 2017

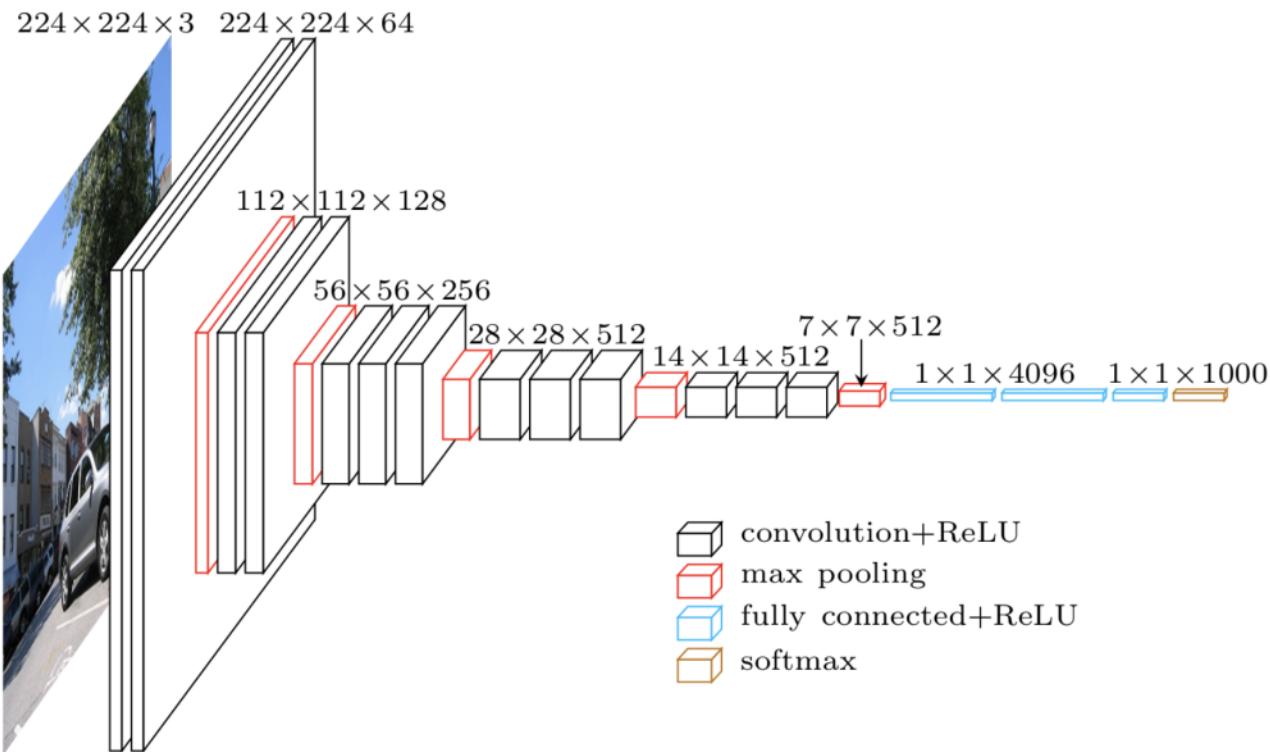
- Seq-to-seq BiLSTM + Word2Vec + Attention



Минусы

- Почти все учится с нуля
- Мало размеченных данных, но много неразмеченных
- Некоторые считают, что подавать Word2Vec на вход LSTM это давать границы изображения на вход CNN
- Нет взаимозаменяемых частей
- В целом – нет Transfer learning

Transfer learning в CV



- Imagenet
- Меньше данных
- Меньше ресурсов
- Лучше качество
- Fine-tuning
- Мультизадачность

Домашняя CV практика

```
model = resnet34(pretrained=True)
for param in model.parameters():
    param.requires_grad=False

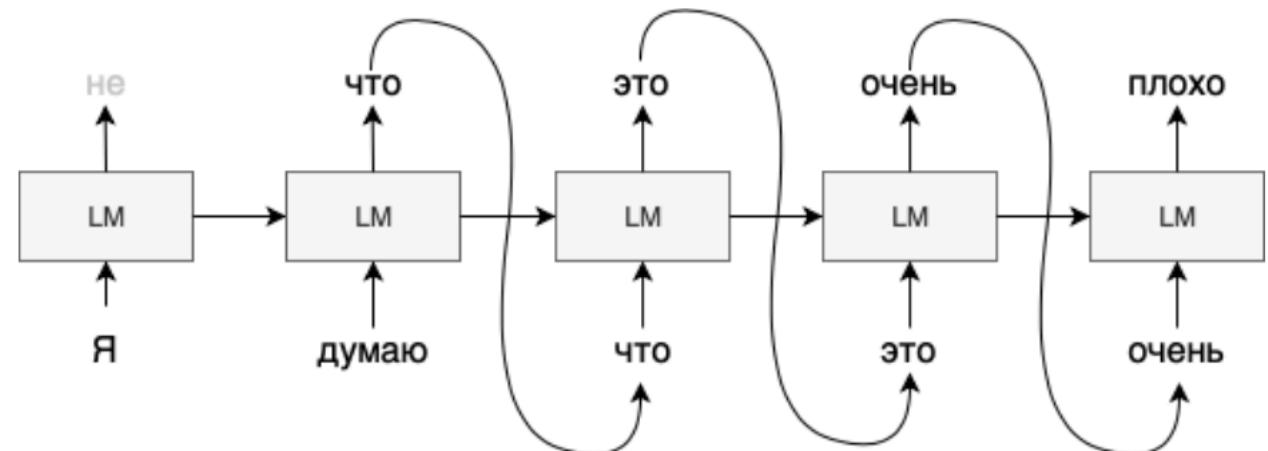
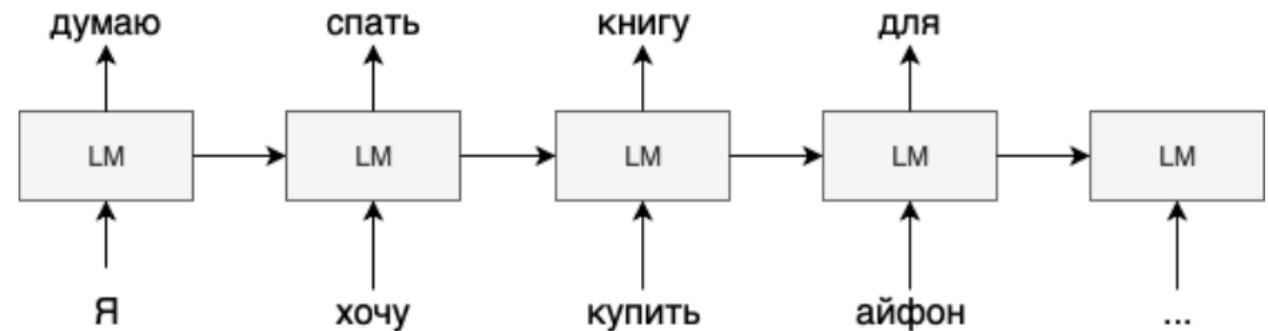
num_ftrs = model.fc.in_features
model.fc = torch.nn.Linear(num_ftrs, 4)
```

Transfer learning в NLP

- Наконец появился в 2018!
- Основан на обучении и использовании Language Model

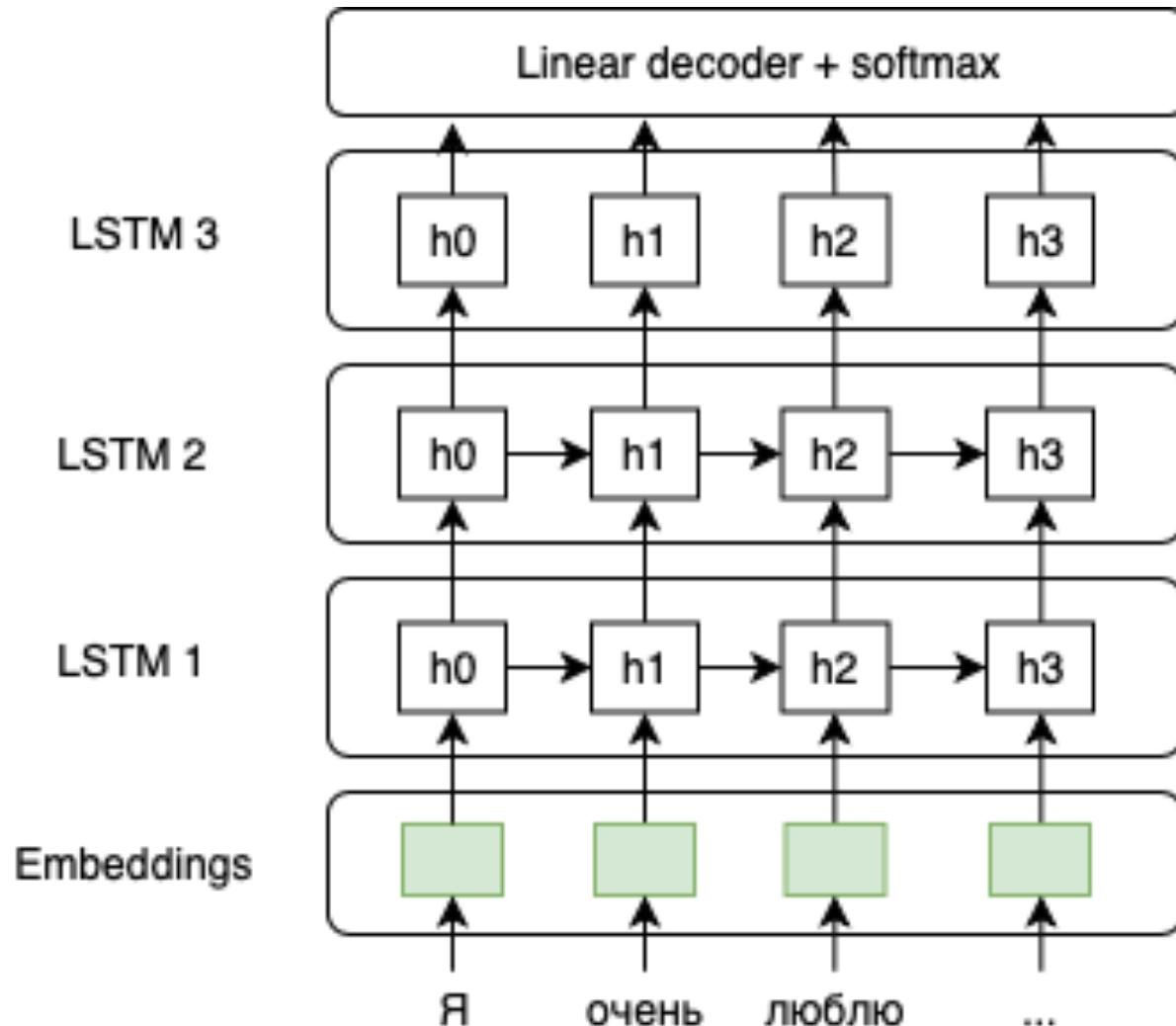
Language model

- Предсказание следующего слова в предложении
- Генерация текста



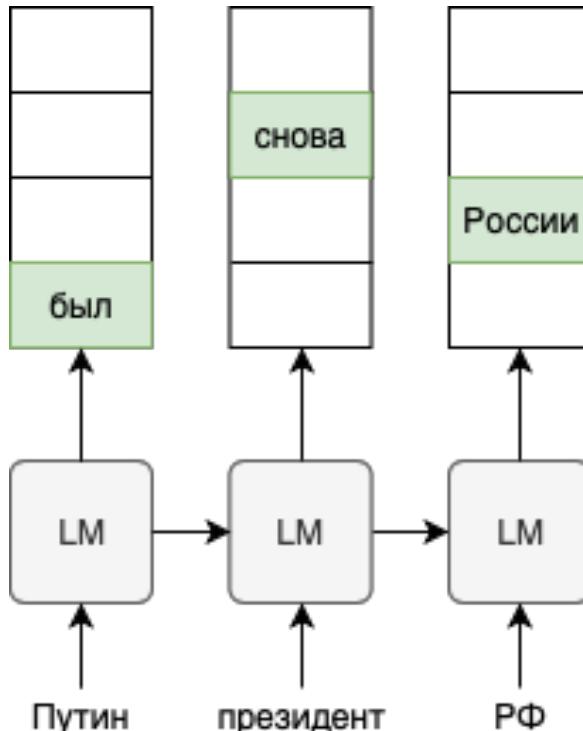
AWD-LSTM (ASGD Weight-Dropped LSTM)

- Embedding size 400
- 3 lstm layers
- Lstm hidden size 1150
- Linear decoder + softmax

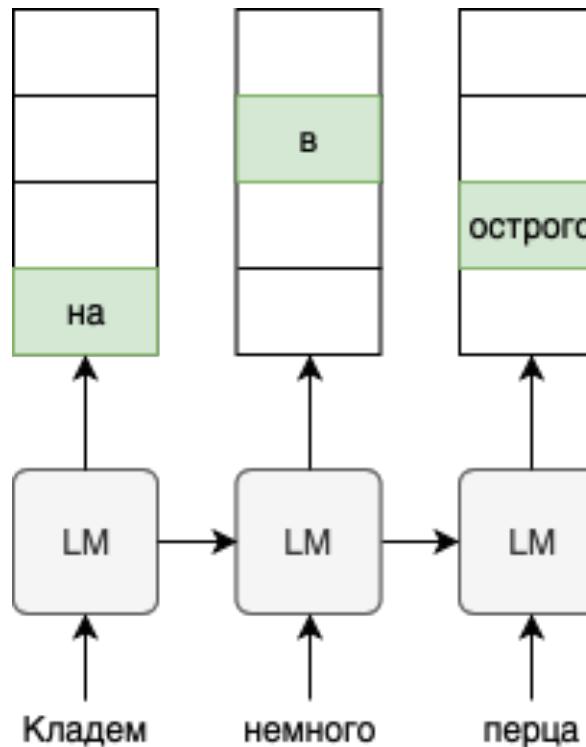


Основная идея

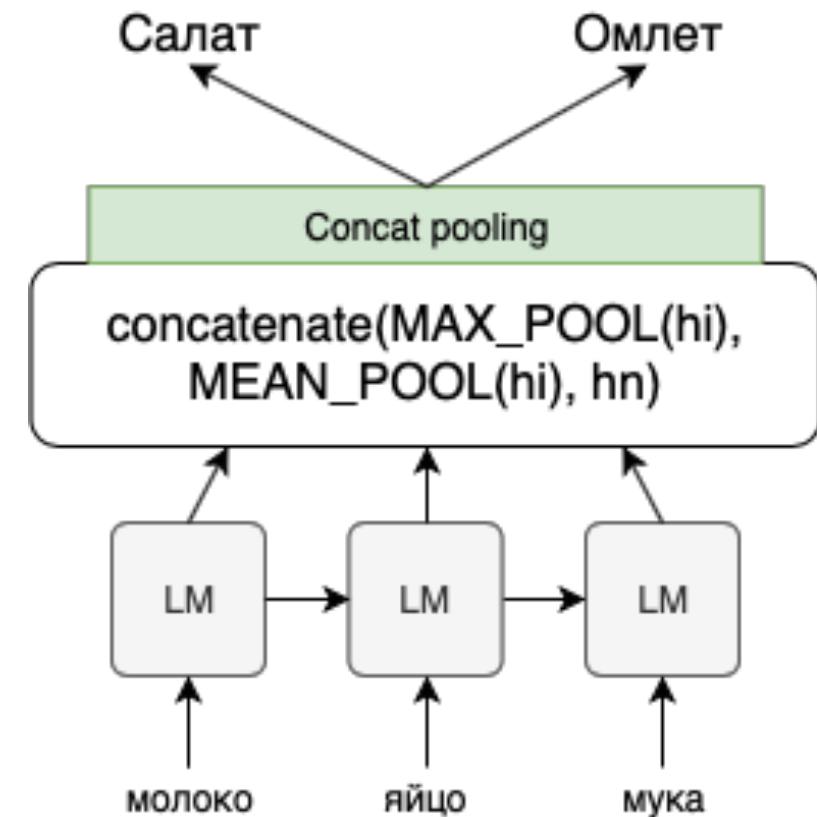
1. Претренировать LM
на wikipedia



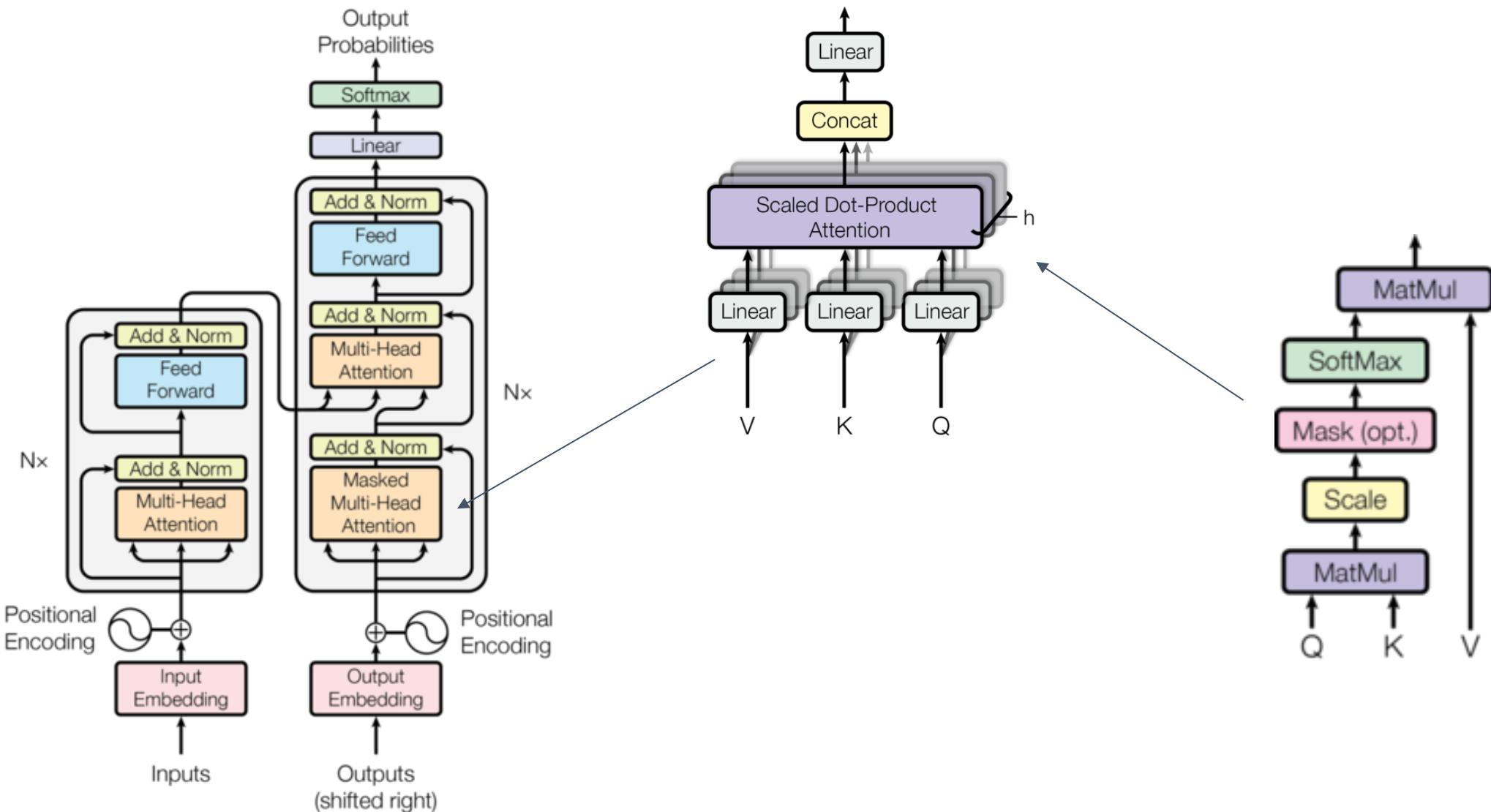
2. Дотренировать LM
на target dataset



3. Заменить decoder на
pooling и учить как
классификатор



Attention Is All You Need

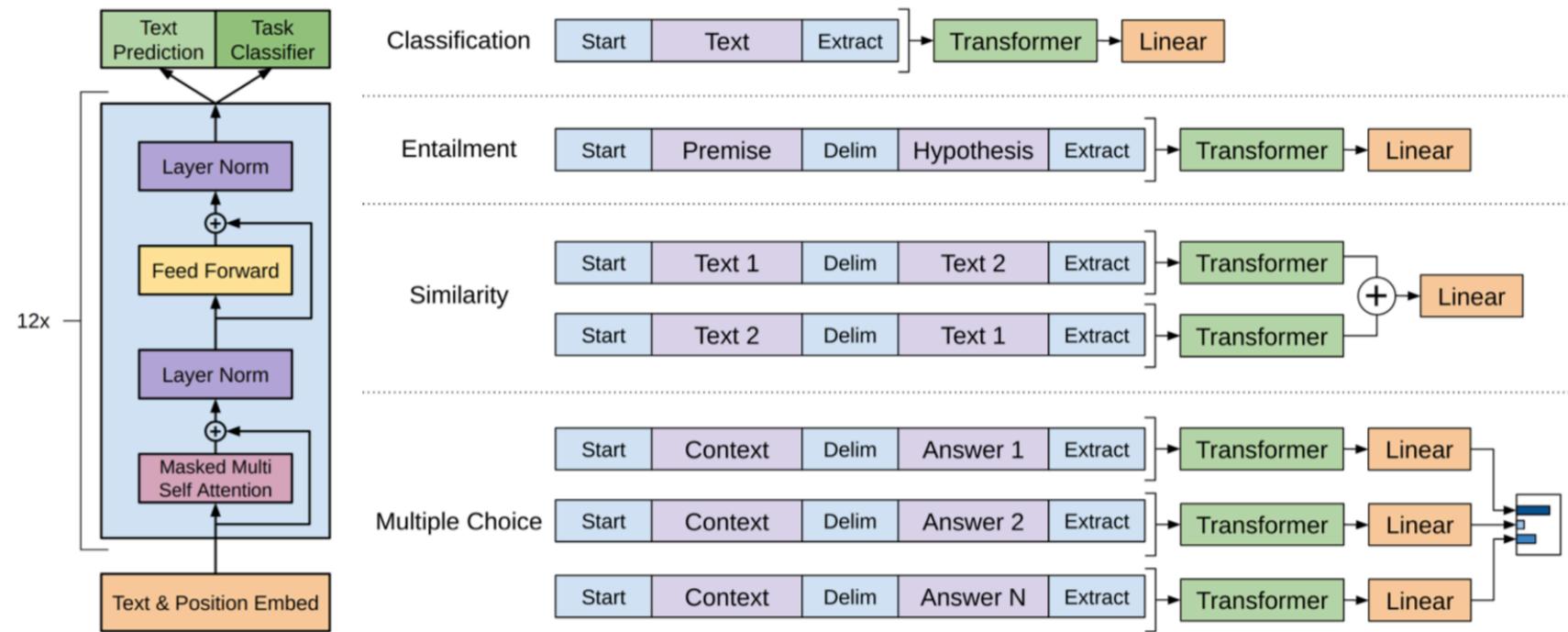


Transformer

- <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

State of the art

- BERT and GPT-2 (2018 and 2019)
- Transfer learning on Transformer Language Model



GPT-2

- **FEBRUARY 14, 2019**
- <https://openai.com/blog/better-language-models/>

DATASET	METRIC	OUR RESULT	PREVIOUS RECORD	HUMAN
Winograd Schema Challenge	accuracy (+)	70.70%	63.7%	92%+
LAMBADA	accuracy (+)	63.24%	59.23%	95%+
LAMBADA	perplexity (-)	8.6	99	~1-2
Children's Book Test Common Nouns (validation accuracy)	accuracy (+)	93.30%	85.7%	96%
Children's Book Test Named Entities (validation accuracy)	accuracy (+)	89.05%	82.3%	92%
Penn Tree Bank	perplexity (-)	35.76	46.54	unknown
WikiText-2	perplexity (-)	18.34	39.14	unknown
enwik8	bits per character (-)	0.93	0.99	unknown
text8	bits per character (-)	0.98	1.08	unknown
WikiText-103	perplexity (-)	17.48	18.3	unknown

GG

