Deep Learning School

Механизм внимания (Attention)

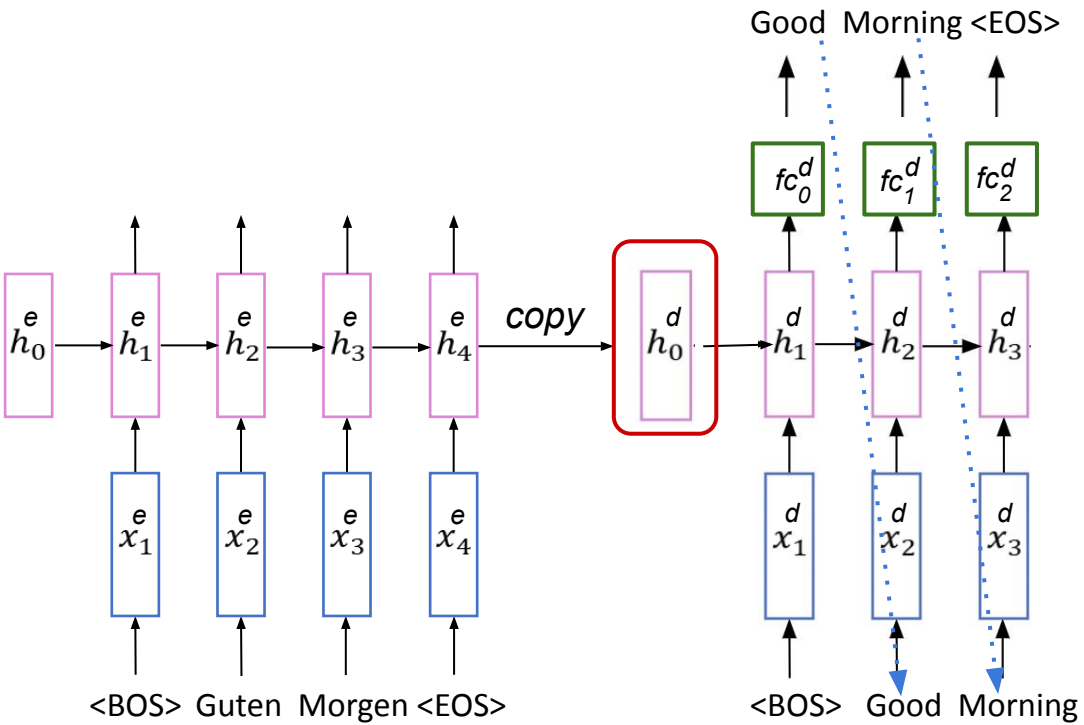
В этом видео

- Задача машинного перевода;
- Архитектура Seq2Seq для решения задачи перевода и ее обучение;
- Недостатки RNN-модели для решения задачи перевода
- **Механизм внимания (Attention)**

В этом видео

- **Механизм внимания (Attention):**
  - Механизм attention: идея;
  - Применение Attention для RNN-модели Seq2Seq для задачи машинного перевода;
  - Разные варианты Attention

RNN для MT: напоминание



Недостатки:

- Encoder RNN может забывать информацию из начала предложения;
- Вся информация о входном предложении содержится в одном векторе h_d^0

Attention: идея

Допустим, нам надо перевести следующее предложение на английский язык:

Высокий статный мужчина в шляпе с прекрасным ярким попугаем на плече только что зашел в здание на углу Старого Косого переулка

Attention: идея

Допустим, нам надо перевести следующее предложение на английский язык:

Высокий статный мужчина в шляпе с прекрасным ярким попугаем на плече только что зашел в здание на углу Старого Косого переулка

Как работает наша RNN-модель:

- Читает это предложение один раз и “запоминает” его в вектор скрытого состояния;
- Генерирует предложение-перевод

Attention: идея

Допустим, нам надо перевести следующее предложение на английский язык:

Высокий статный мужчина в шляпе с прекрасным ярким попугаем на плече только что зашел в здание на углу Старого Косого переулка

Что бы делали мы (люди):

- Во время генерации перевода мы бы несколько раз смотрели на разные части исходного предложения, чтобы найти те части, на которые стоит обратить внимание в данный момент времени, и лучше их запомнить

Attention: идея

Допустим, нам надо перевести следующее предложение на английский язык:

Высокий статный мужчина в шляпе с прекрасным ярким попугаем на плече только что зашел в здание на углу Старого Косого переулка

Механизм Attention — это то, что поможет декодеру модели “подглядывать” в нужные части исходного предложения во время генерации.

Attention

Идея Attention была впервые предложена в 2014-2015 году в этой статье

Published as a conference paper at ICLR 2015

NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

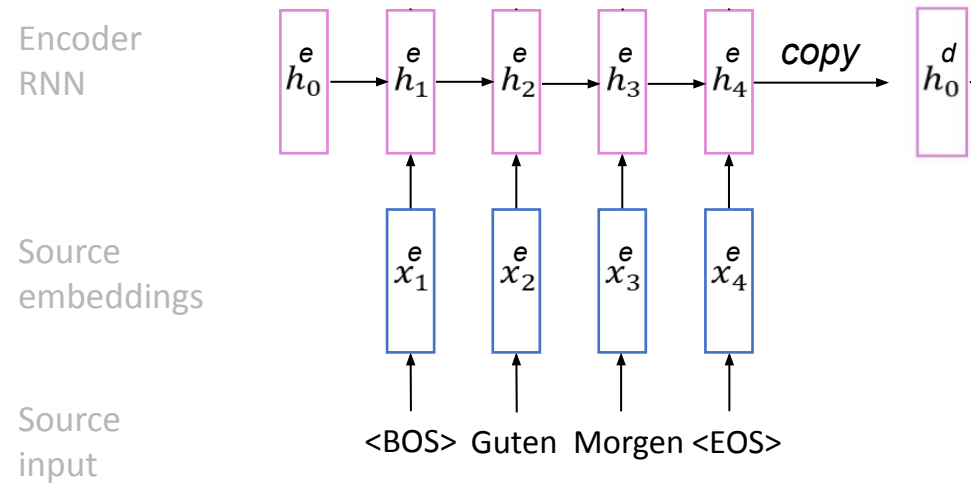
Dzmitry Bahdanau
Jacobs University Bremen, Germany

KyungHyun Cho **Yoshua Bengio***
Université de Montréal

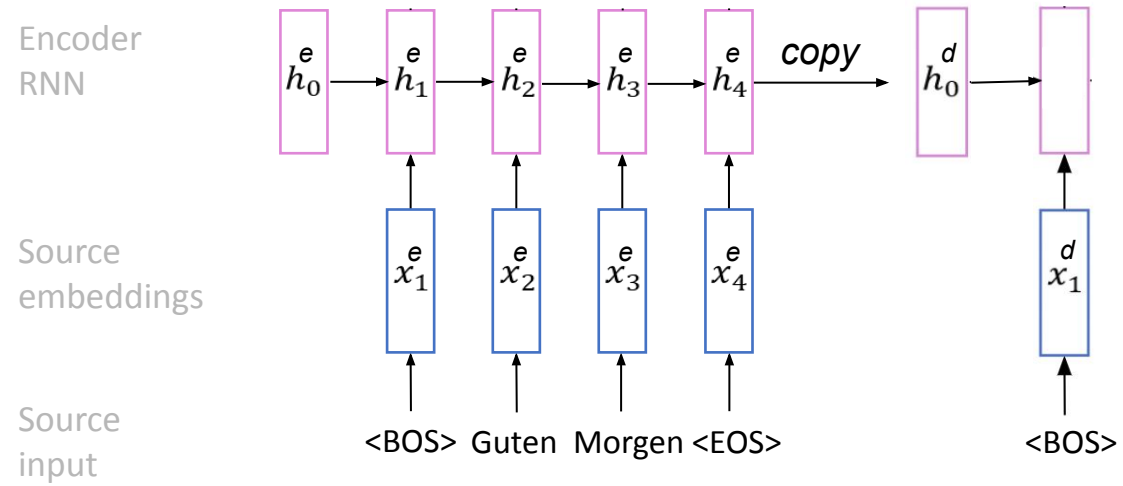
ABSTRACT

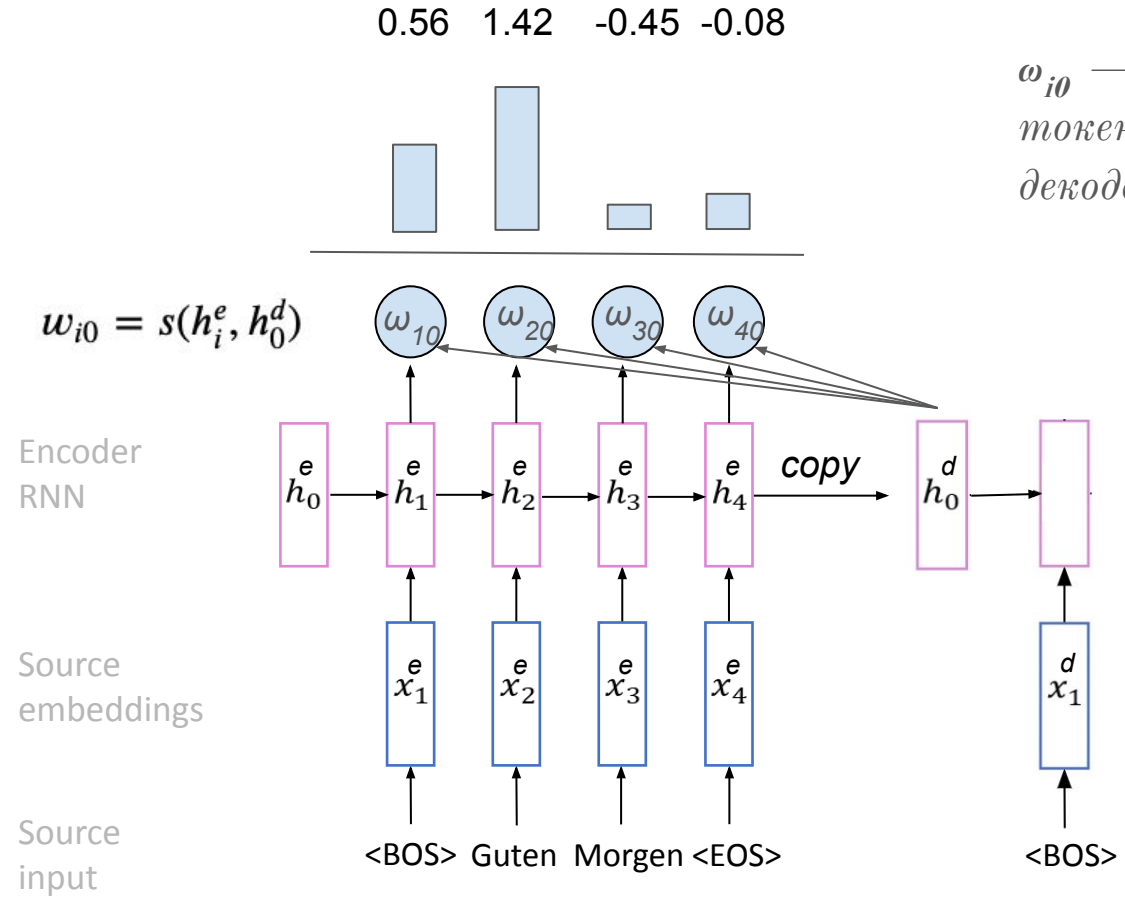
Neural machine translation is a recently proposed approach to machine translation. Unlike the traditional statistical machine translation, the neural machine translation aims at building a single neural network that can be jointly tuned to maximize the translation performance. The models proposed recently for neural machine translation often belong to a family of encoder–decoders and encode a source sentence into a fixed-length vector from which a decoder generates a translation. In this paper, we conjecture that the use of a fixed-length vector is a bottleneck in improving the performance of this basic encoder–decoder architecture, and propose to extend this by allowing a model to automatically (soft-)search for parts of a source sentence that are relevant to predicting a target word, without having to form these parts as a hard segment explicitly. With this new approach, we achieve a translation performance comparable to the existing state-of-the-art phrase-based system on the task of English-to-French translation. Furthermore, qualitative analysis reveals that the (soft-)alignments found by the model agree well with our intuition.

Энкодер будет работать точно так же,
как и раньше

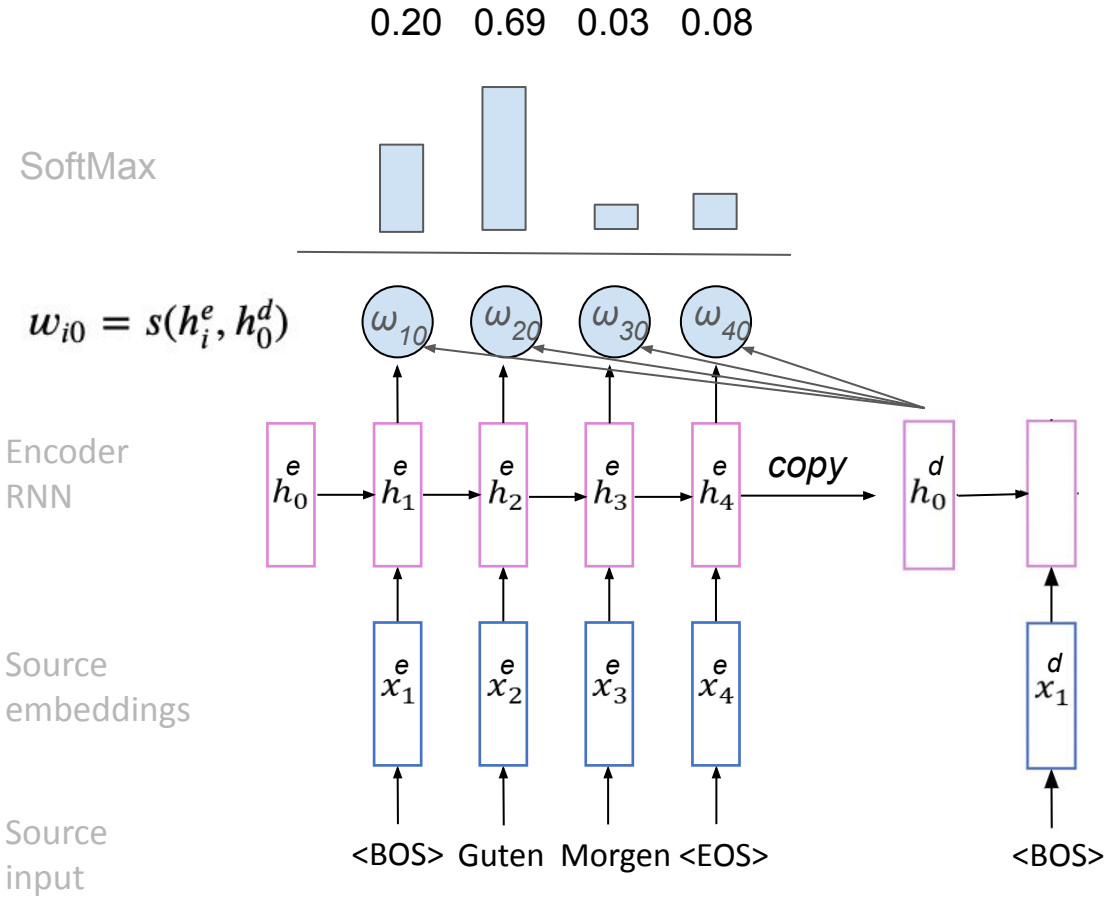


Перед тем, как обновить скрытое состояние декодера, нам нужно “обратить внимание” на входную последовательность

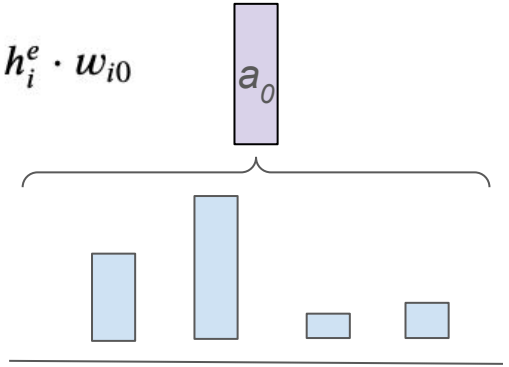




ω_{i0} — *это вес (важность) i-го входного токена для текущего состояния декодера*


0.20
0.69
0.03
0.08
SoftMax
$w_{i0} = s(h_i^e, h_0^d)$
ω_{10}
ω_{20}
ω_{30}
ω_{40}
Encoder RNN
h_0^e
h_1^e
h_2^e
h_3^e
h_4^e
copy
h_0^d
Source embeddings
x_1^e
x_2^e
x_3^e
x_4^e
x_1^d
Source input
<BOS>
Guten
Morgen
<EOS>
<BOS>

$$a_0 = \sum_{i=1}^{4} h_i^e \cdot w_{i0}$$



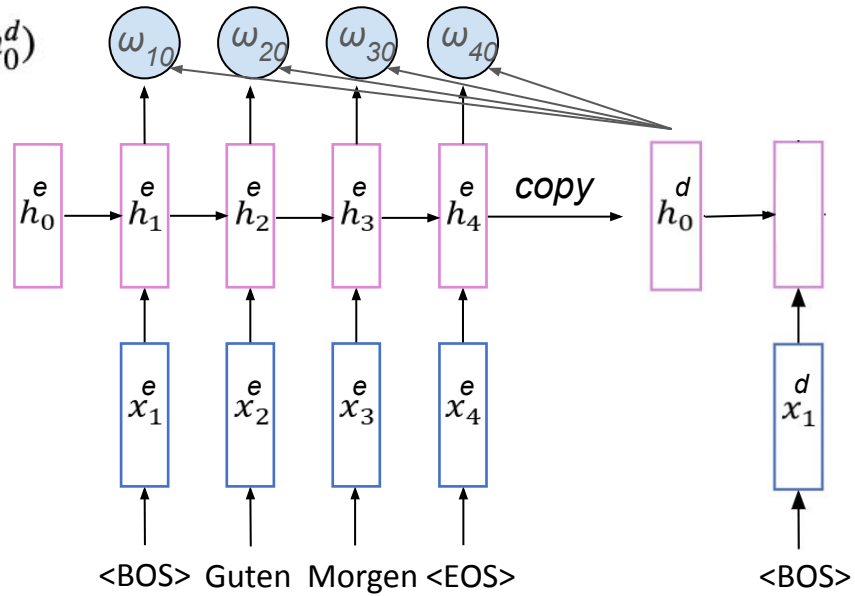
SoftMax

$$w_{i0} = s(h_i^e, h_0^d)$$

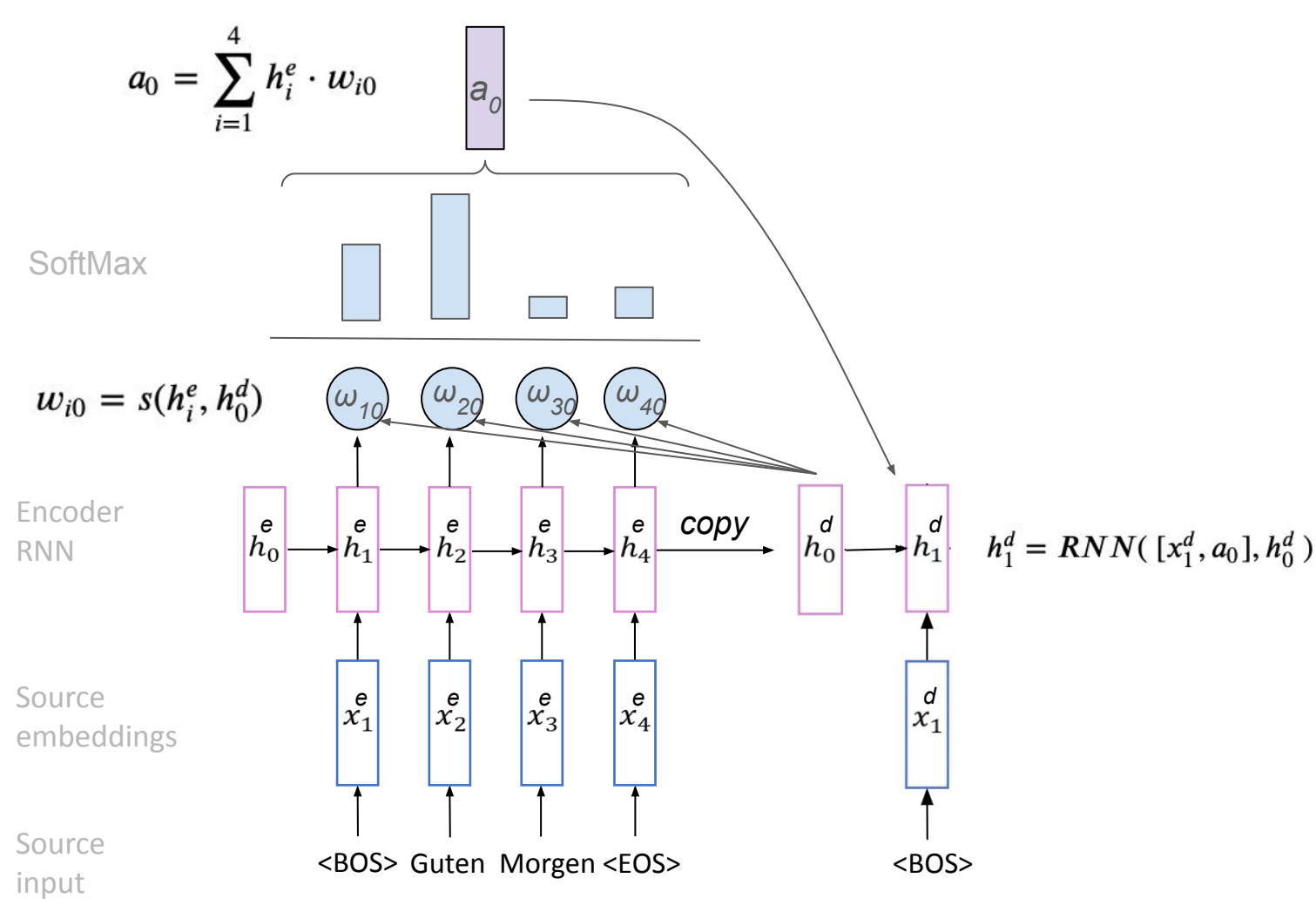
Encoder RNN

Source embeddings

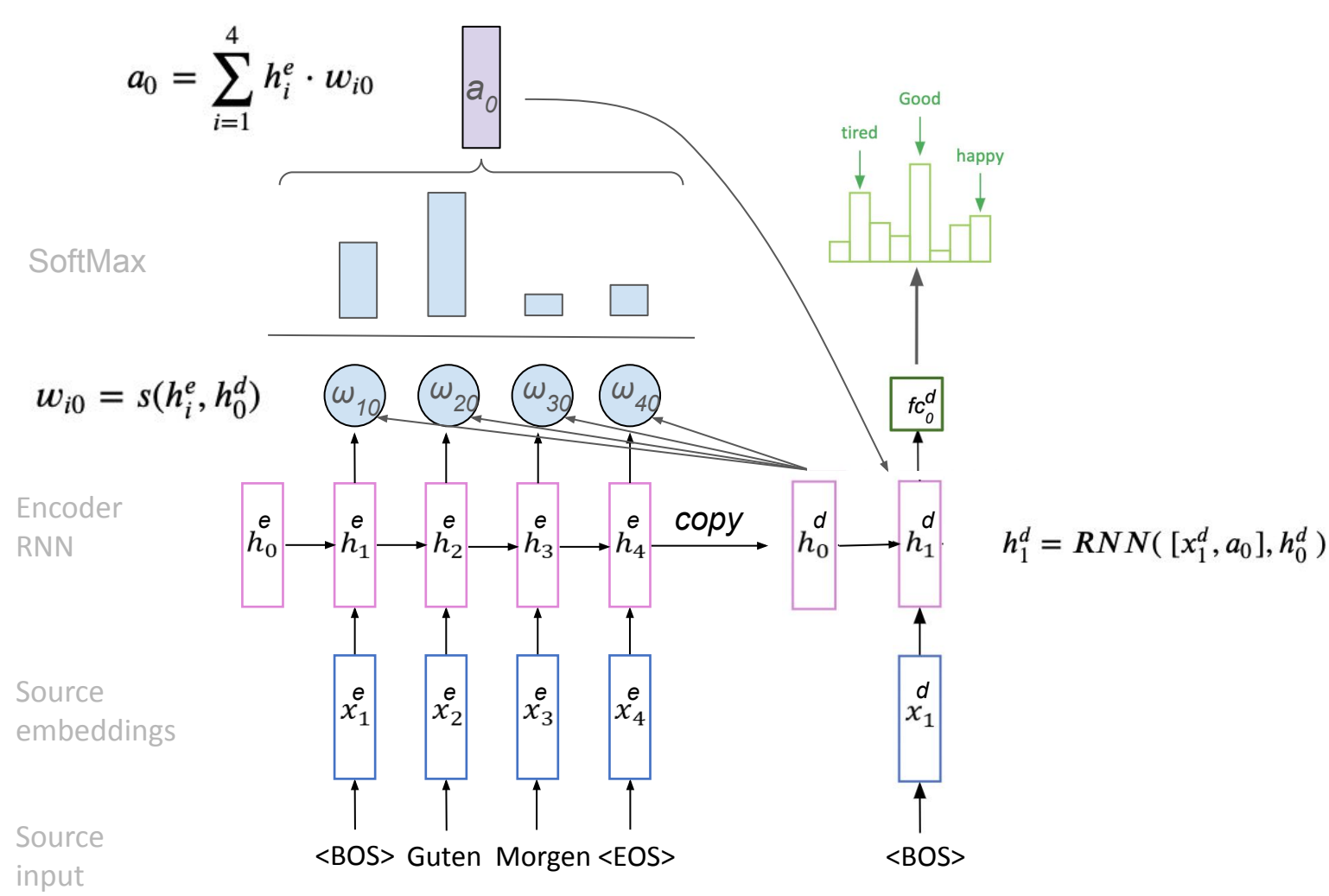
Source input



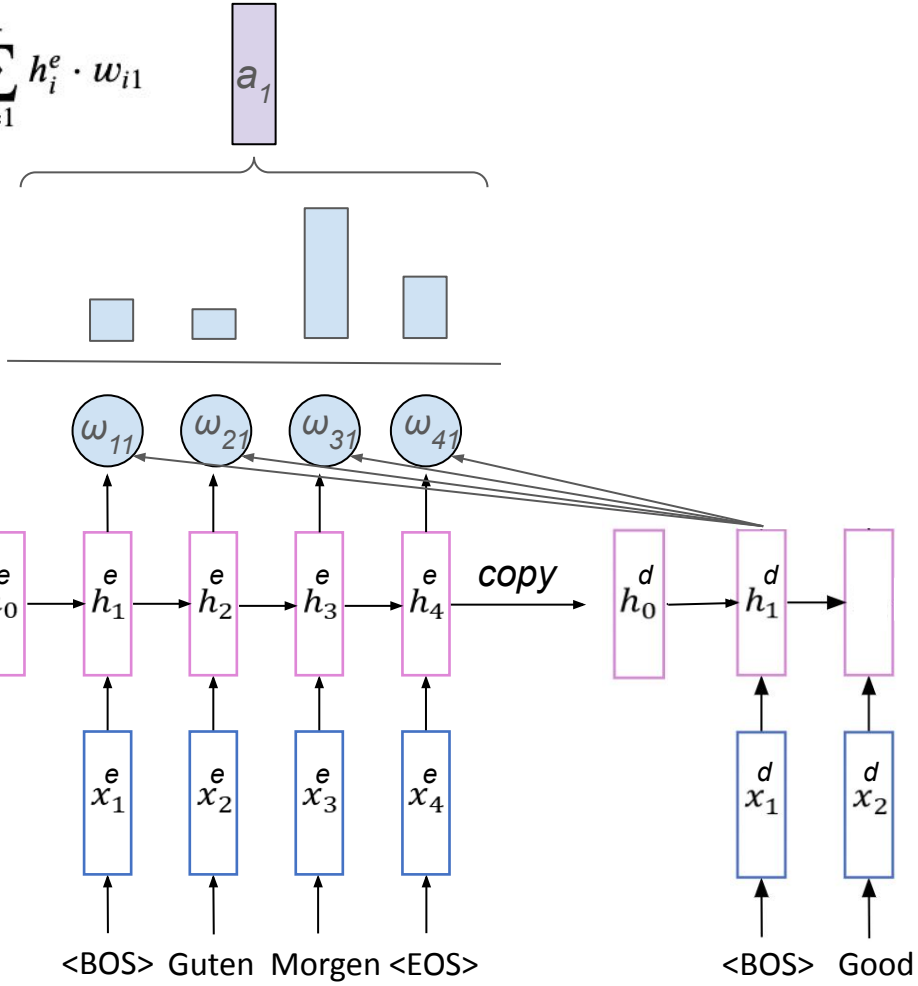
$$a_0 = \sum_{i=1}^{4} h_i^e \cdot w_{i0}$$



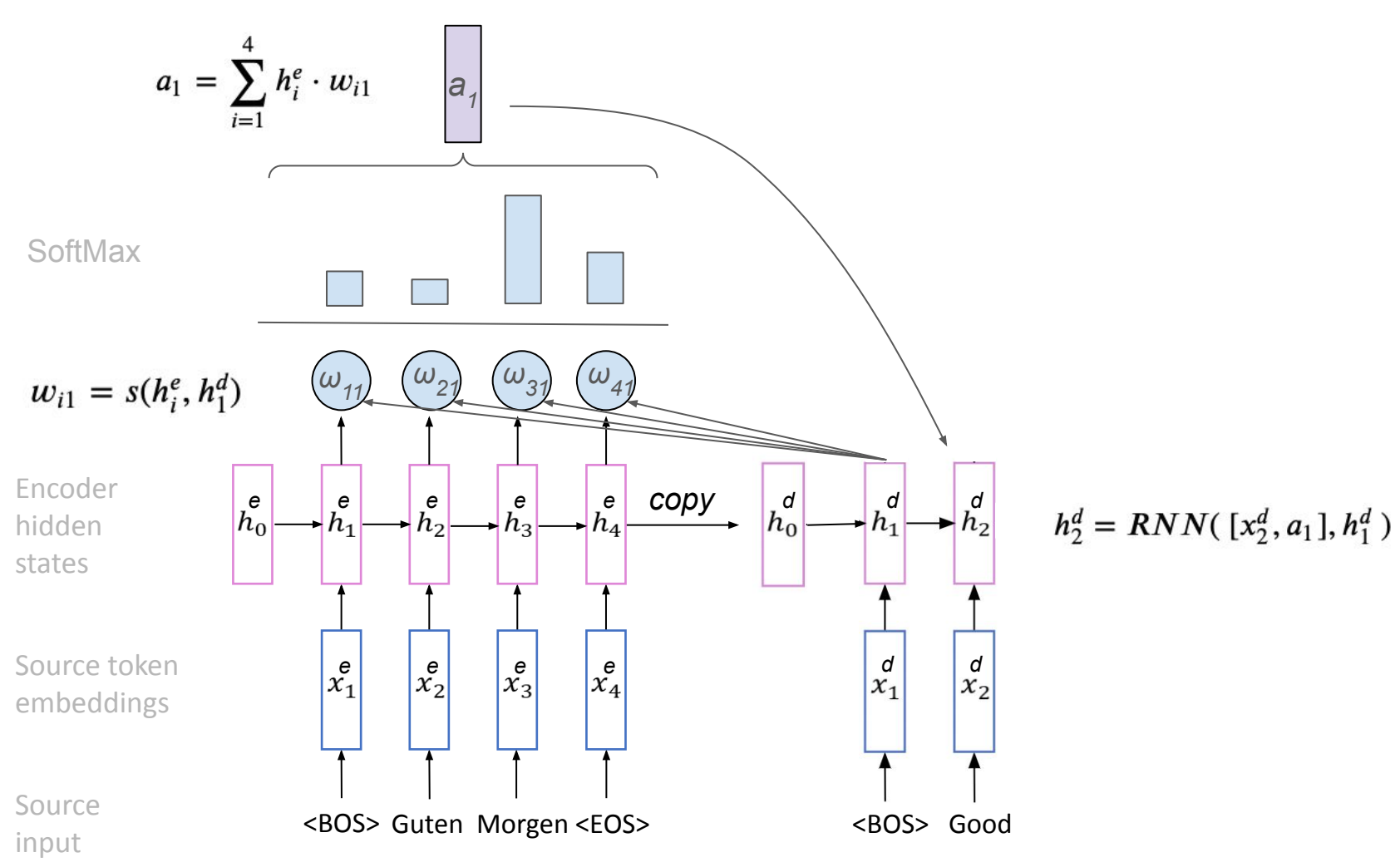
$$a_0 = \sum_{i=1}^{4} h_i^e \cdot w_{i0}$$

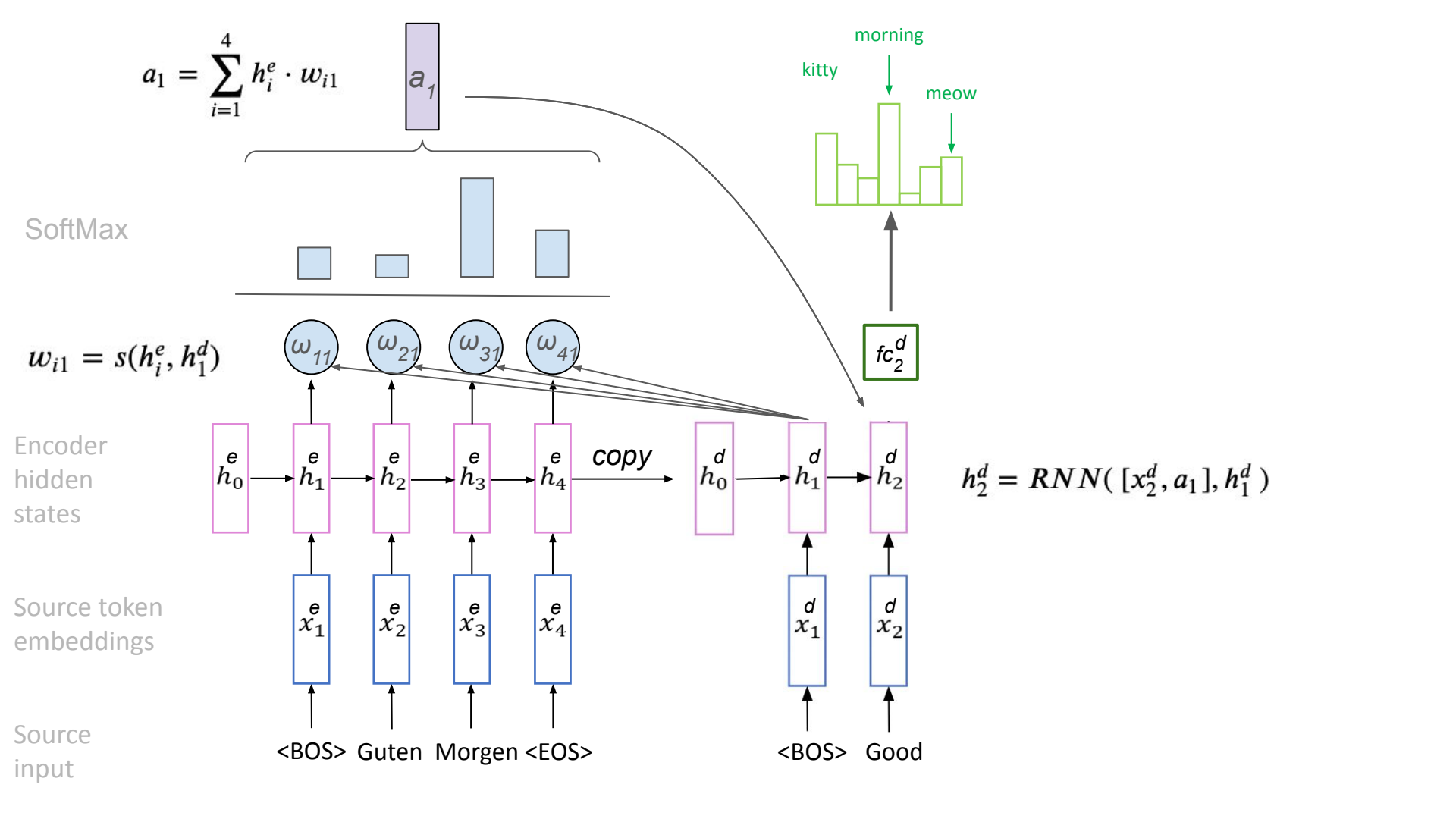


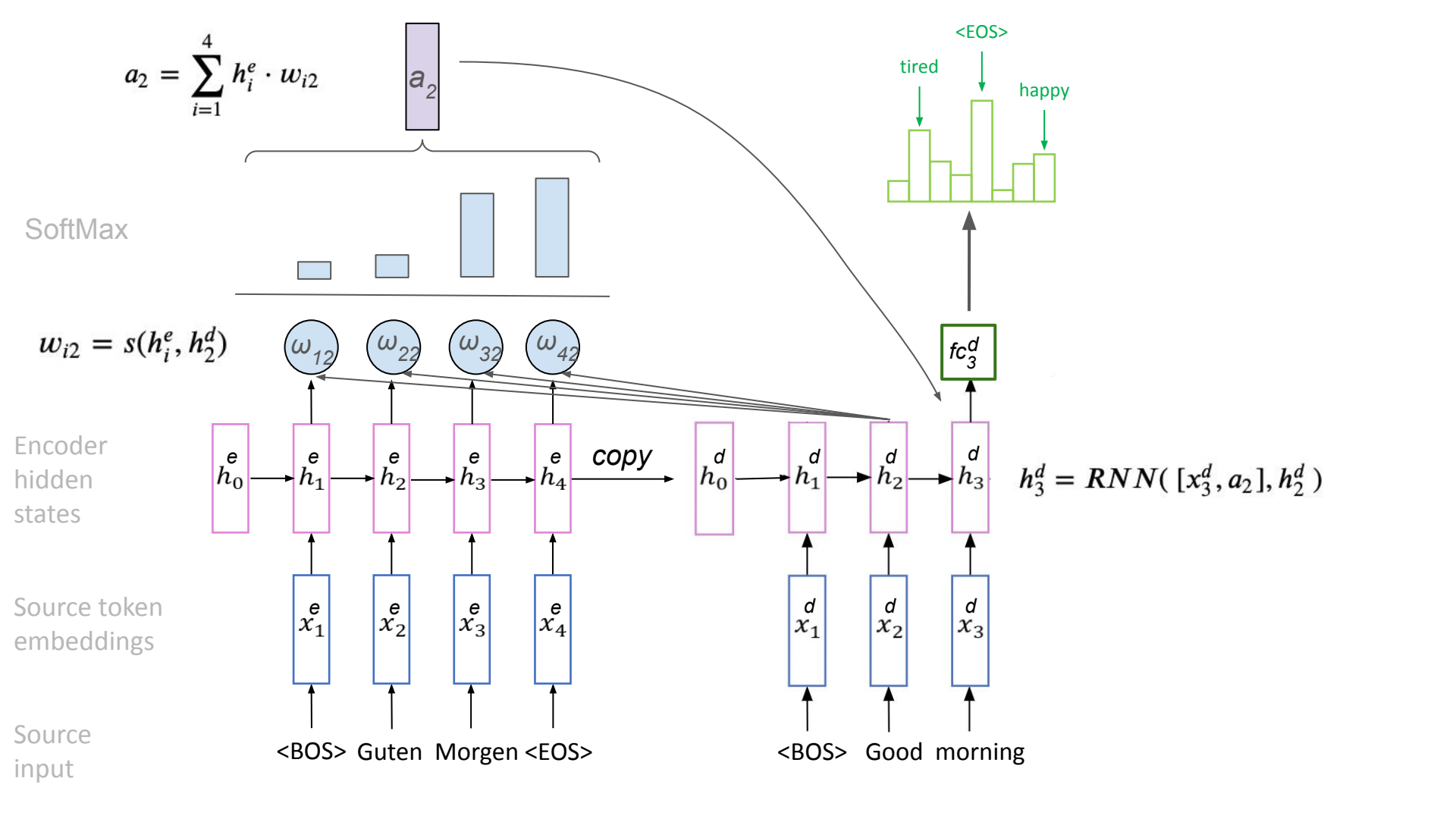
$$a_1 = \sum_{i=1}^{4} h_i^e \cdot w_{i1}$$

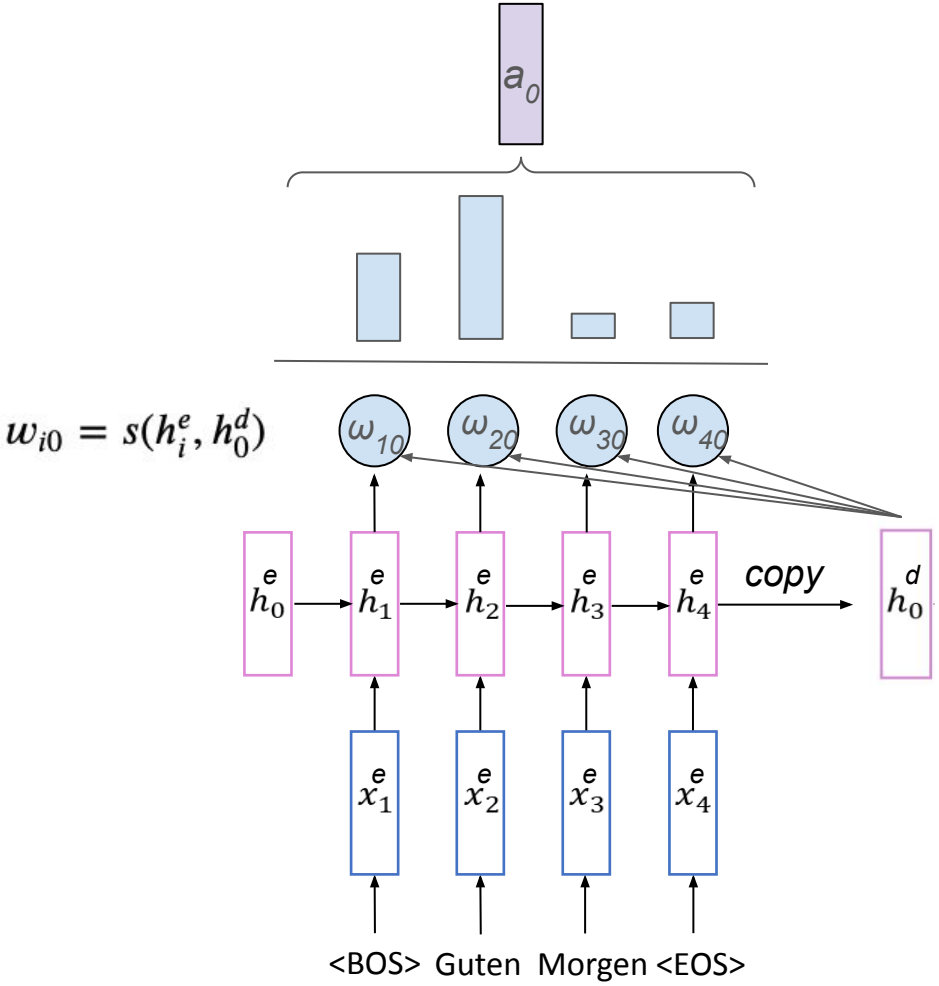


$$a_1 = \sum_{i=1}^{4} h_i^e \cdot w_{i1}$$




$a_1 = \sum_{i=1}^{4} h_i^e \cdot w_{i1}$
a_1
morning
kitty
meow
SoftMax
$w_{i1} = s(h_i^e, h_1^d)$
ω_{11}
ω_{21}
ω_{31}
ω_{41}
fc_2^d
Encoder hidden states
h_0^e
h_1^e
h_2^e
h_3^e
h_4^e
copy
h_0^d
h_1^d
h_2^d
$h_2^d = RNN([x_2^d, a_1], h_1^d)$
Source token embeddings
x_1^e
x_2^e
x_3^e
x_4^e
x_1^d
x_2^d
Source input
<BOS> Guten Morgen <EOS>
<BOS> Good


$a_2 = \sum_{i=1}^{4} h_i^e \cdot w_{i2}$
a_2
<EOS>
tired
happy
SoftMax
$w_{i2} = s(h_i^e, h_2^d)$
ω_{12}
ω_{22}
ω_{32}
ω_{42}
fc_3^d
Encoder hidden states
h_0^e
h_1^e
h_2^e
h_3^e
h_4^e
copy
h_0^d
h_1^d
h_2^d
h_3^d
$h_3^d = RNN([x_3^d, a_2], h_2^d)$
Source token embeddings
x_1^e
x_2^e
x_3^e
x_4^e
x_1^d
x_2^d
x_3^d
Source input
<BOS> Guten Morgen <EOS>
<BOS> Good morning



Функция s может быть обучаемой или не обучаемой.

Если она обучаемая, то ее веса обучаются совместно со всеми параметрами сети.

Варианты функций Attention

$$w = s(h_1, h_2)$$

- Dot product attention:
  (без обучаемых параметров)

  $$s(h_1, h_2) = h_1^T h_2$$

- Однослойная нейросеть
  (с обучаемыми параметрами)

  $$s(h_1, h_2) = \sigma(W[h_1, h_2])$$

- Многие другие

Эффект применения Attention

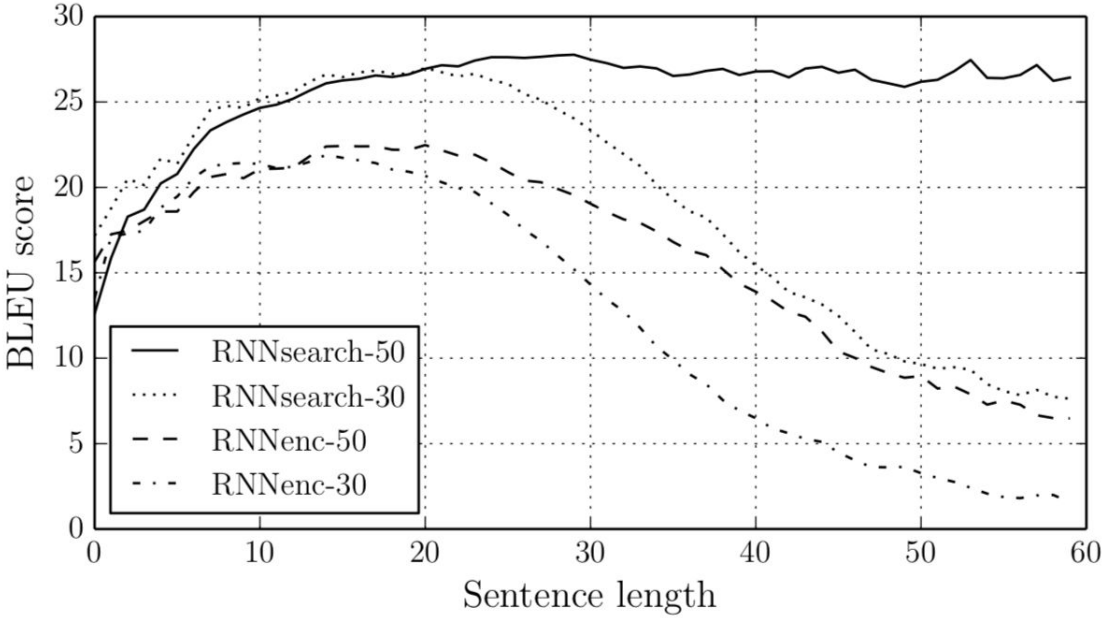
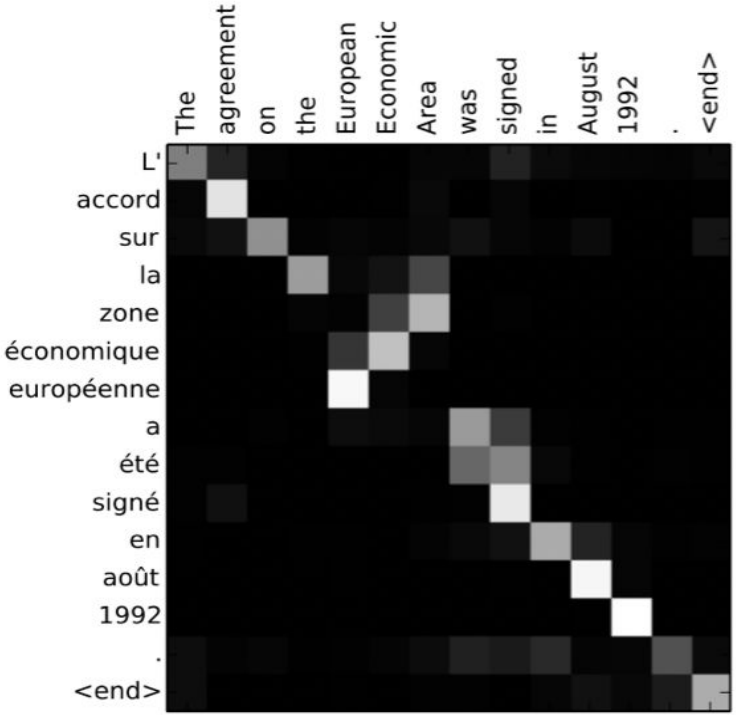


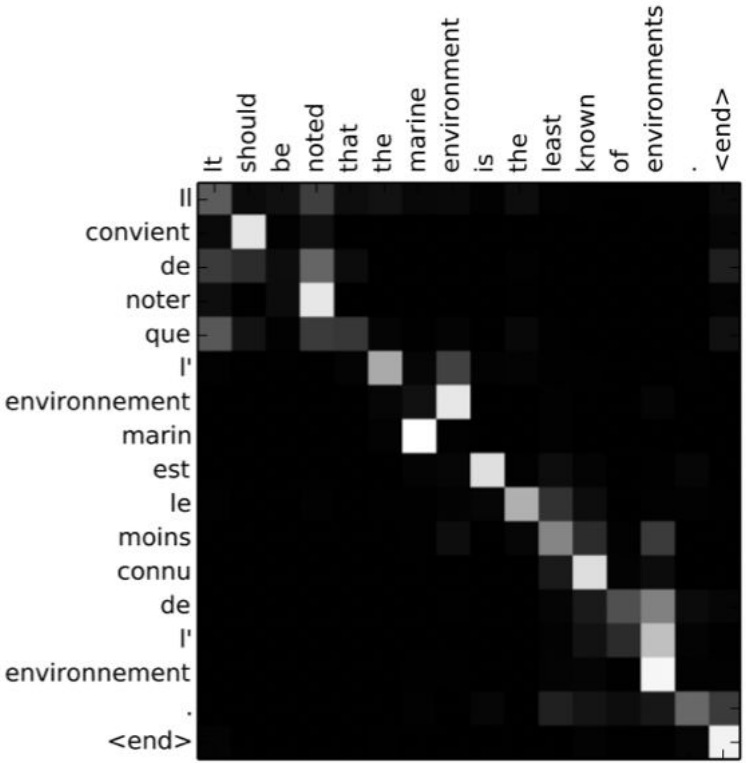
Figure 2: The BLEU scores of the generated translations on the test set with respect to the lengths of the sentences. The results are on the full test set which includes sentences having unknown words to the models.

https://arxiv.org/pdf/1409.0473.pdf

Attention visualization



Attention visualization



Революция после Attention

Изобретение Attention было поворотным моментов в задаче машинного перевода и всей области NLP

В 2017 году Google представили новую архитектуру нейросети, которую назвали Transformer. Основой ее был механизм Attention. Transformer сотворили революцию в NLP, и потом были успешно адаптированы и к другим доменам.

Итоги видео

В этом видео:

- Идея механизма Attention;
- Attention для RNN-модели Seq2Seq;
- Разные варианты Attention;
- Attention может быть интерпретируемым.