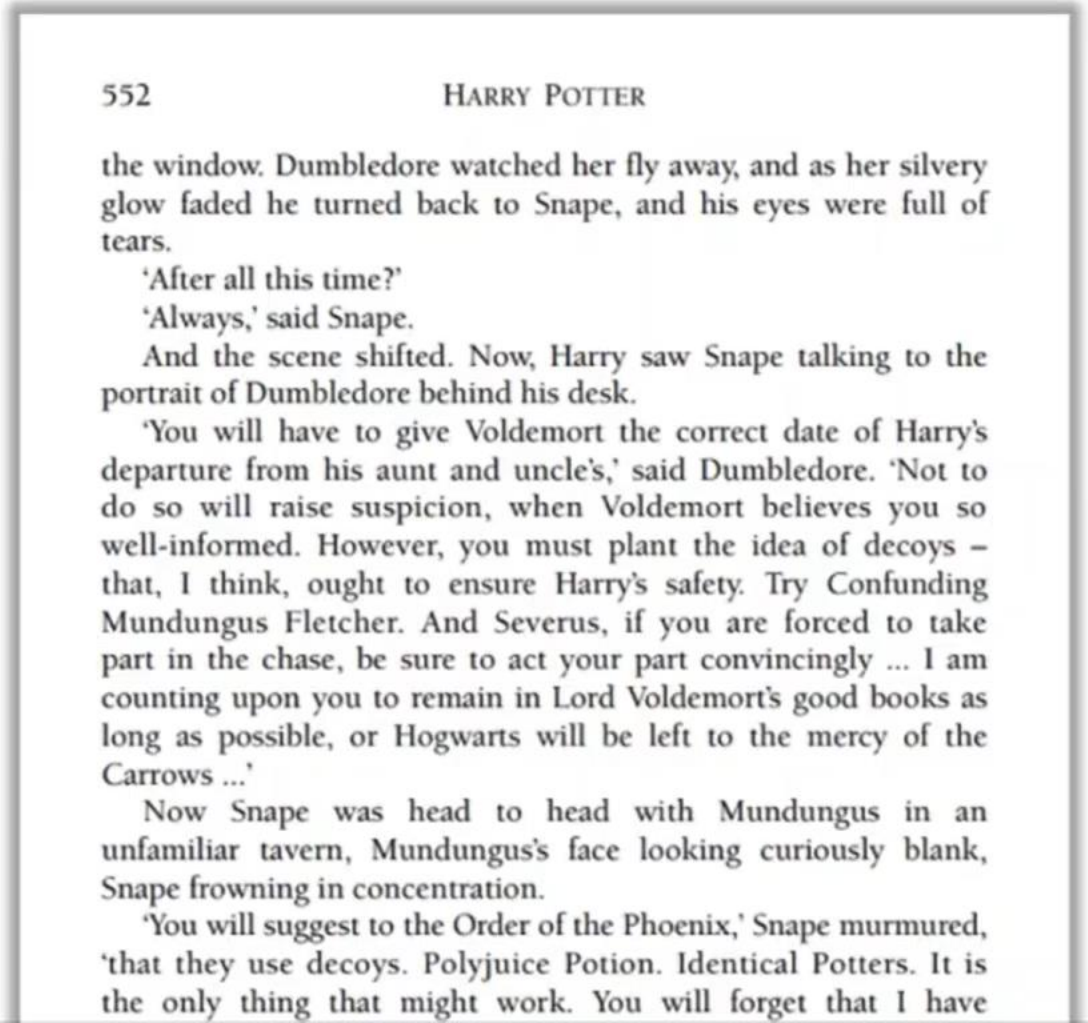
Attention Mechanism

CSE 4237

Soft Computing

Introduction

Attention in Human Visual Processing System



552 HARRY POTTER

the window. Dumbledore watched her fly away, and as her silvery glow faded he turned back to Snape, and his eyes were full of tears.

'After all this time?'

'Always,' said Snape.

And the scene shifted. Now, Harry saw Snape talking to the portrait of Dumbledore behind his desk.

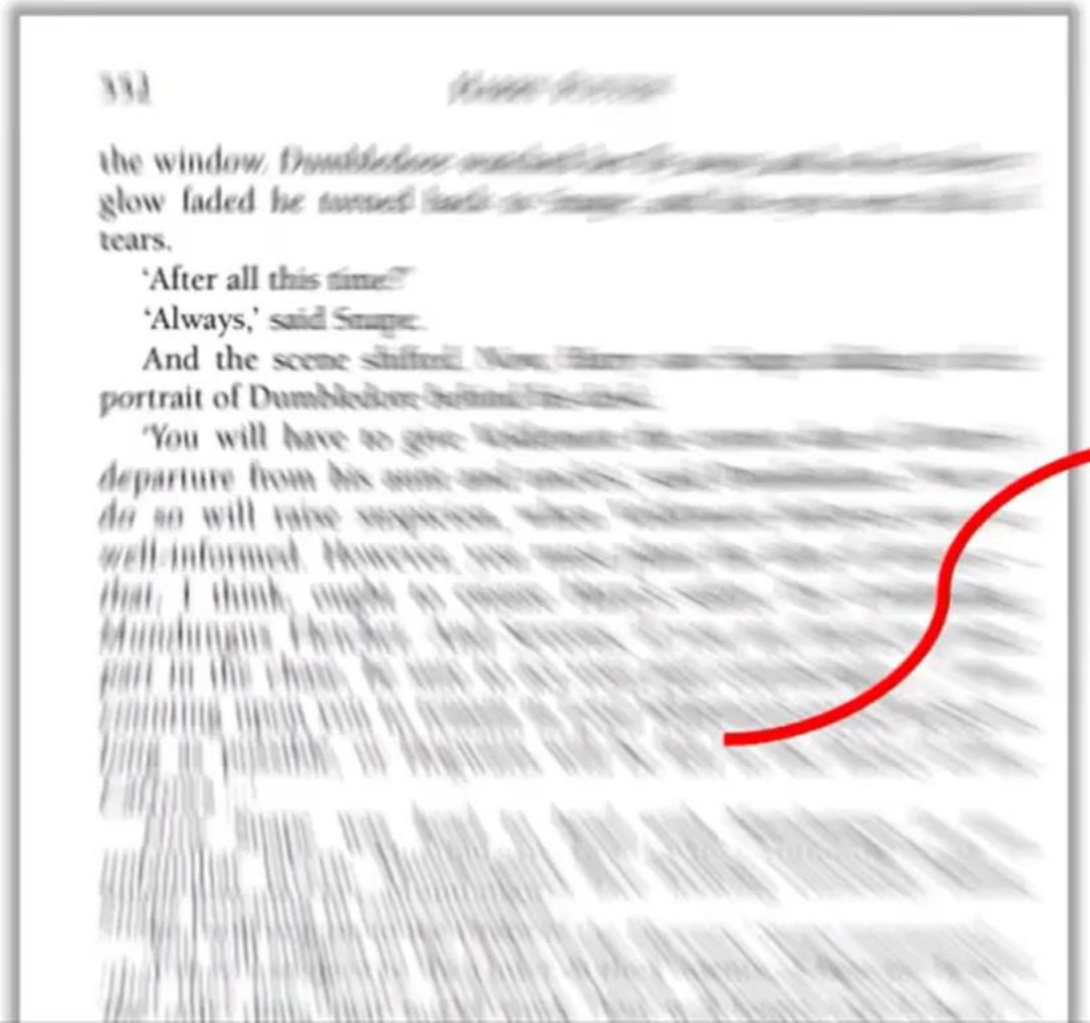
'You will have to give Voldemort the correct date of Harry's departure from his aunt and uncle's,' said Dumbledore. 'Not to do so will raise suspicion, when Voldemort believes you so well-informed. However, you must plant the idea of decoys – that, I think, ought to ensure Harry's safety. Try Confunding Mundungus Fletcher. And Severus, if you are forced to take part in the chase, be sure to act your part convincingly ... I am counting upon you to remain in Lord Voldemort's good books as long as possible, or Hogwarts will be left to the mercy of the Carrows ...'

Now Snape was head to head with Mundungus in an unfamiliar tavern, Mundungus's face looking curiously blank, Snape frowning in concentration.

'You will suggest to the Order of the Phoenix,' Snape murmured, 'that they use decoys. Polyjuice Potion. Identical Potters. It is the only thing that might work. You will forget that I have

Introduction

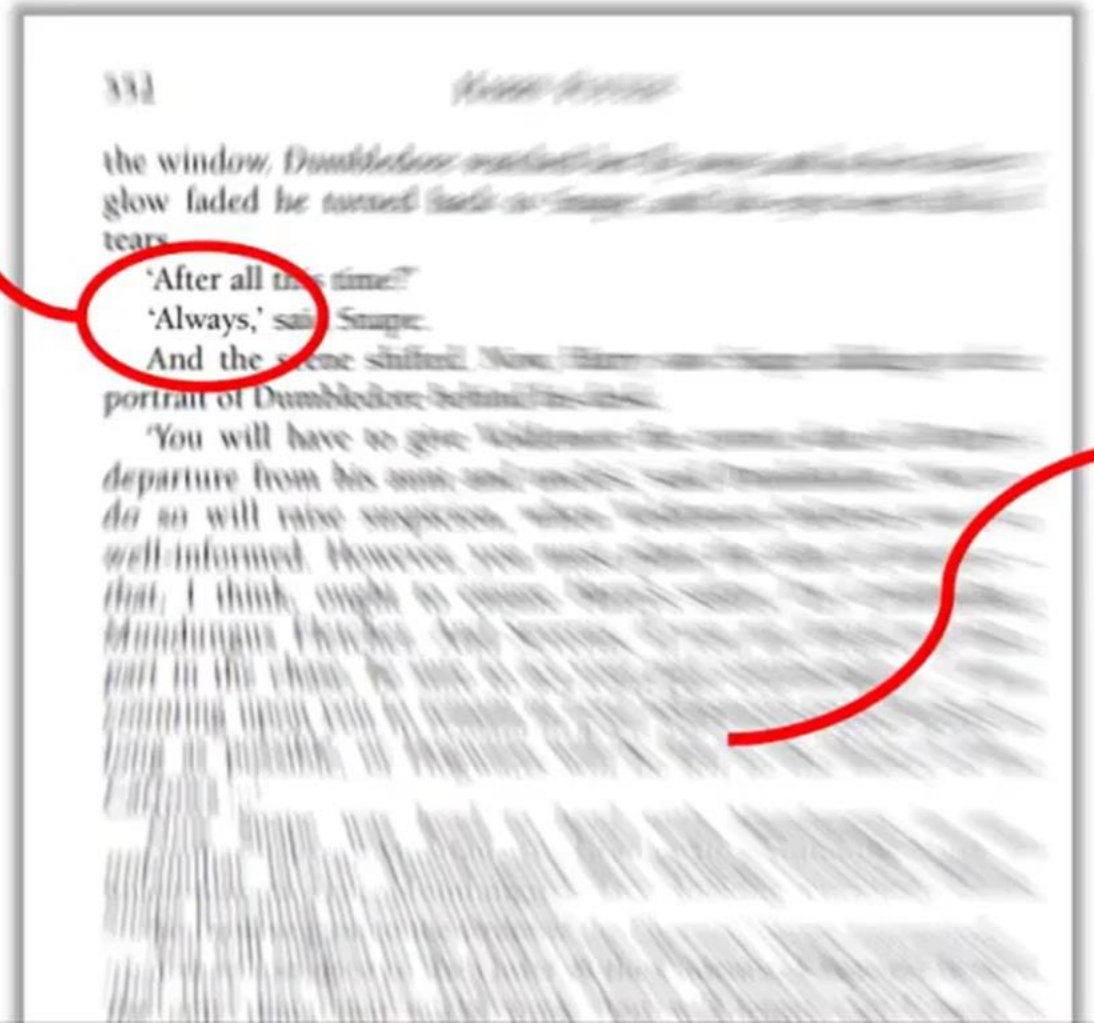
Attention in Human Visual Processing System



Introduction

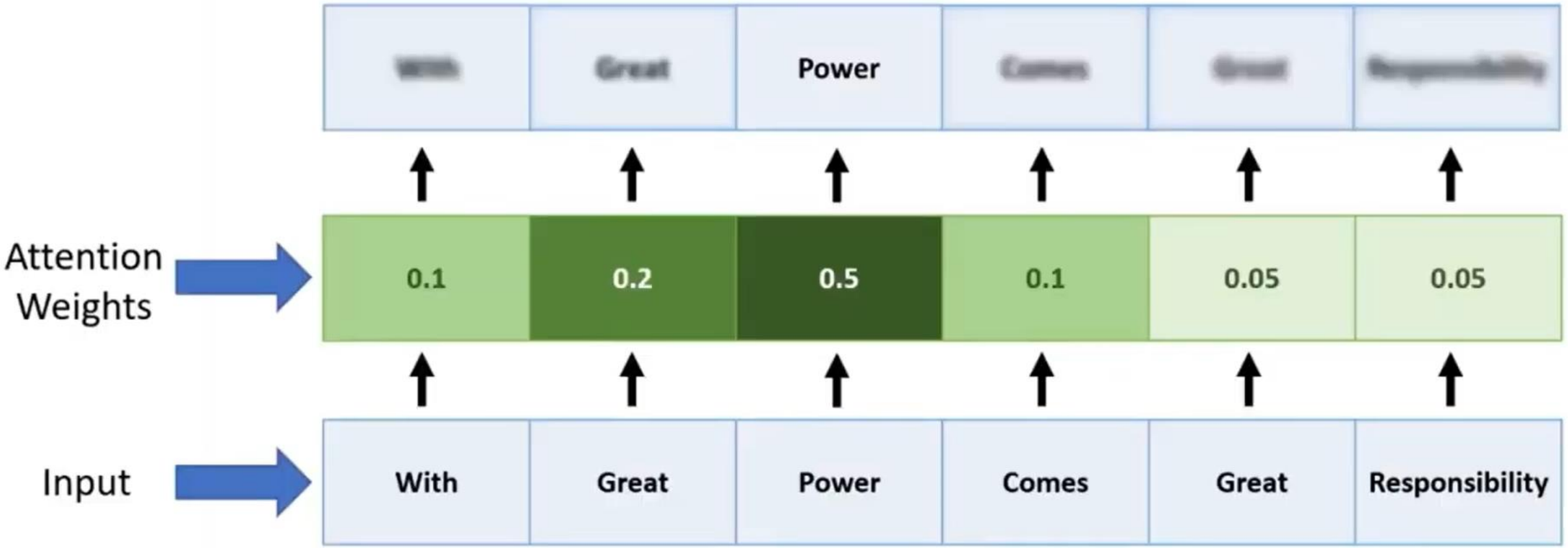
Attention in Human Visual Processing System

Brain puts More focus on the word you are currently reading.

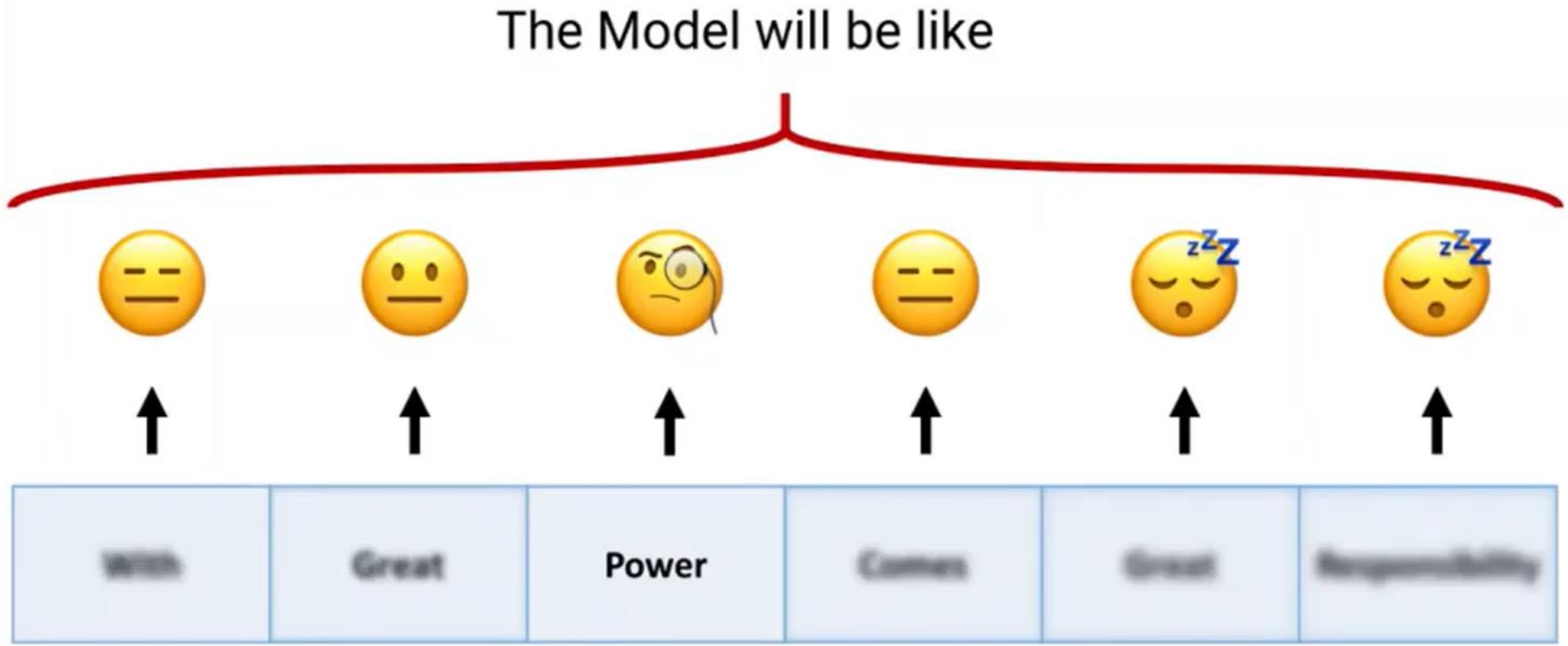


Your Brain doesn't care about most of the text.

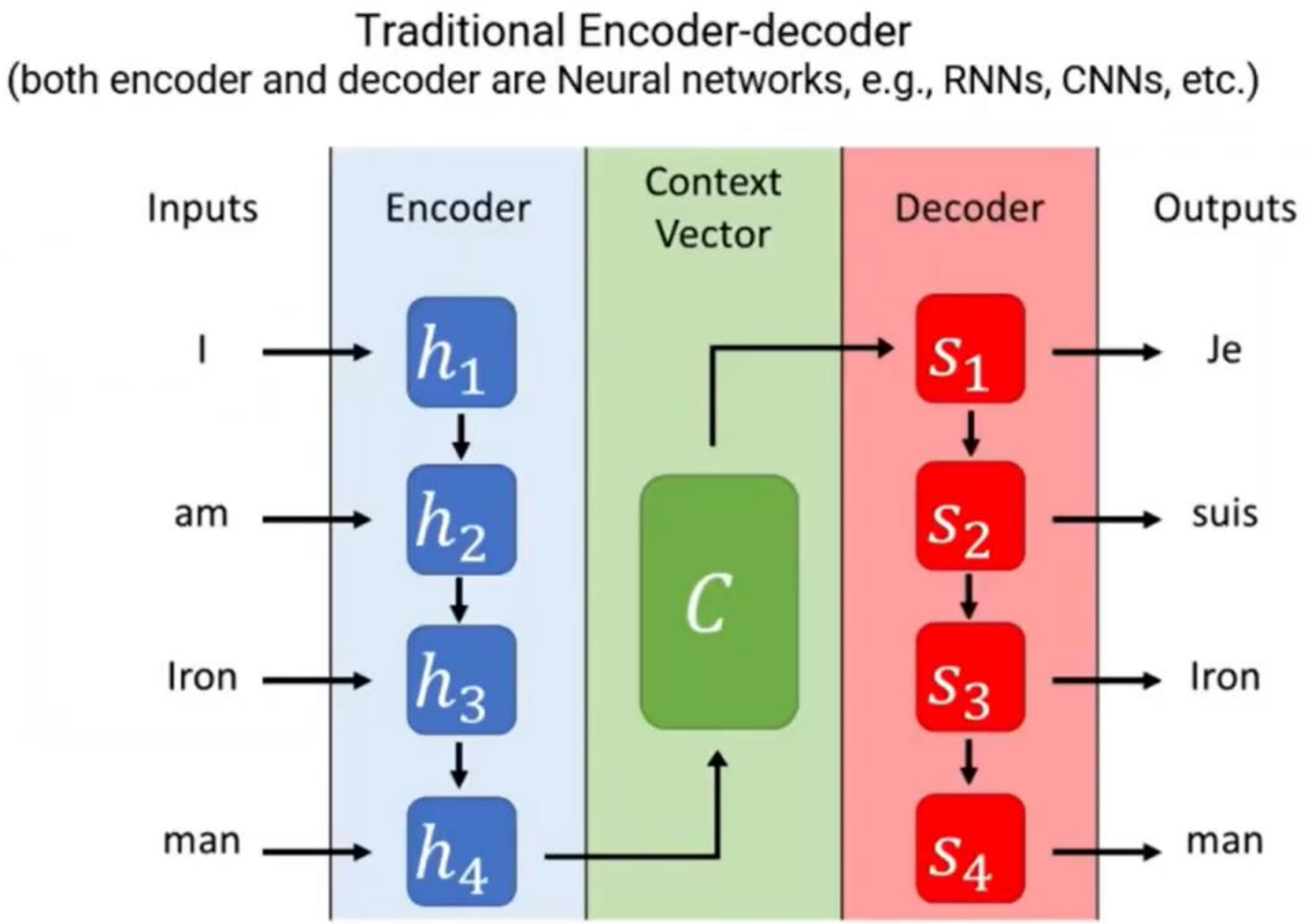
Introduction



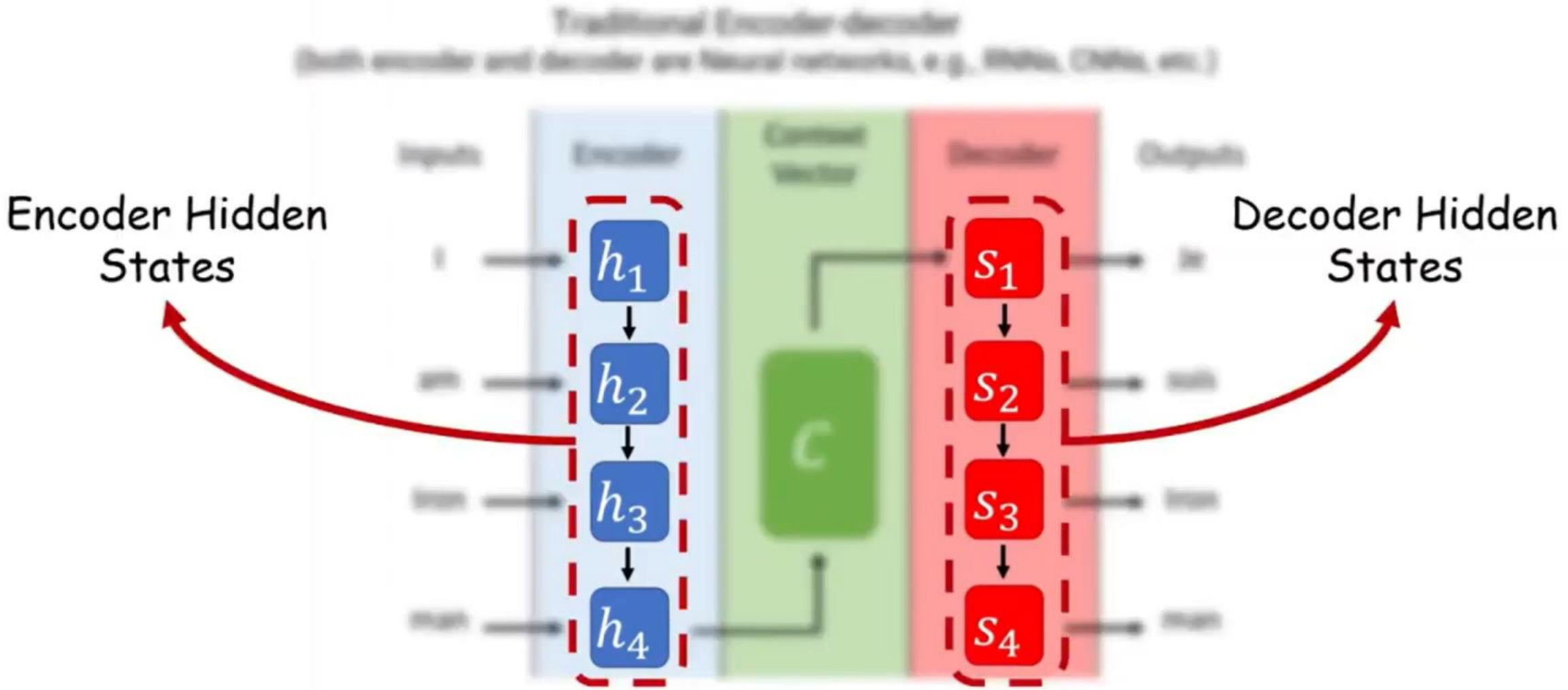
Introduction



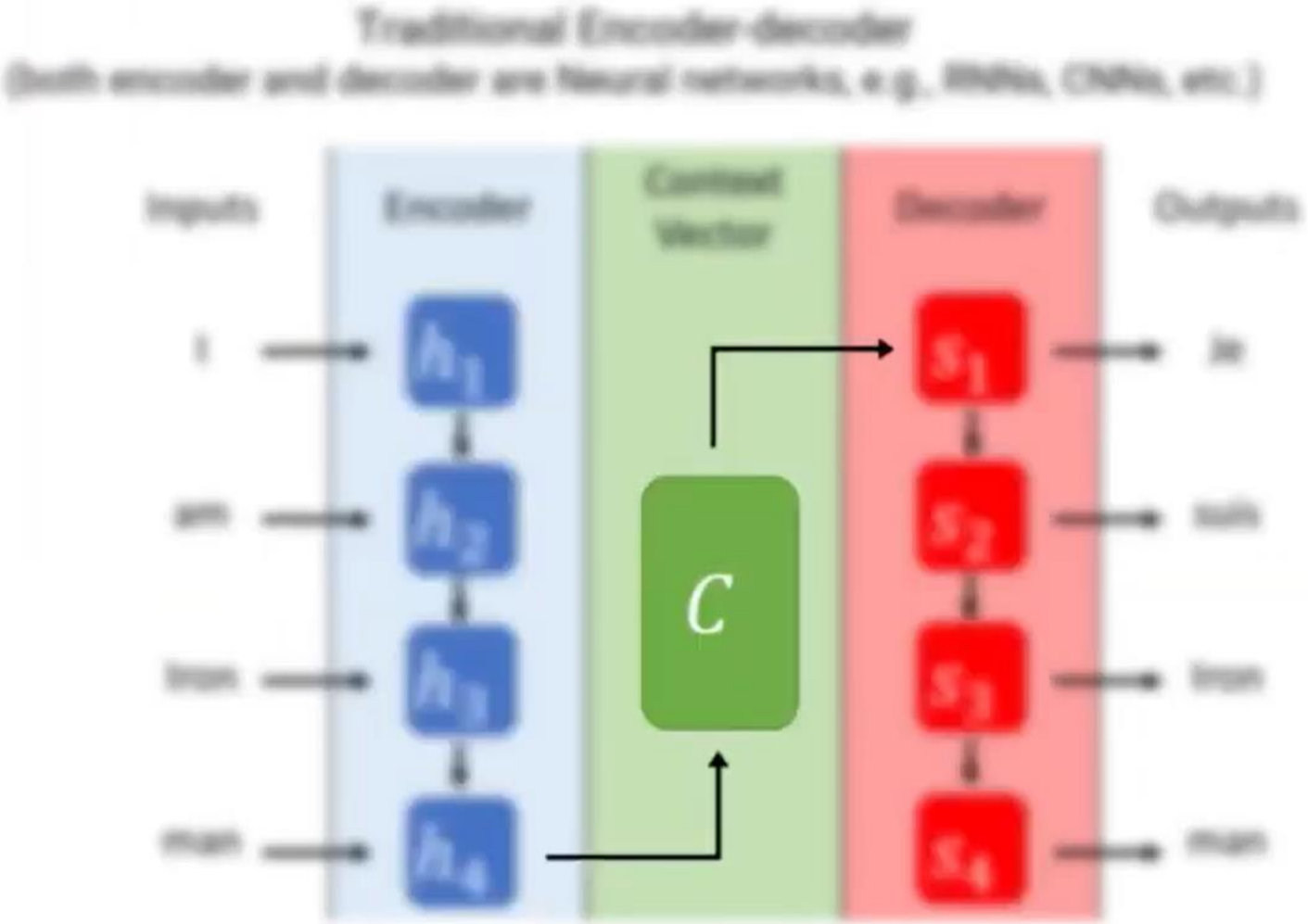
Attention Mechanism



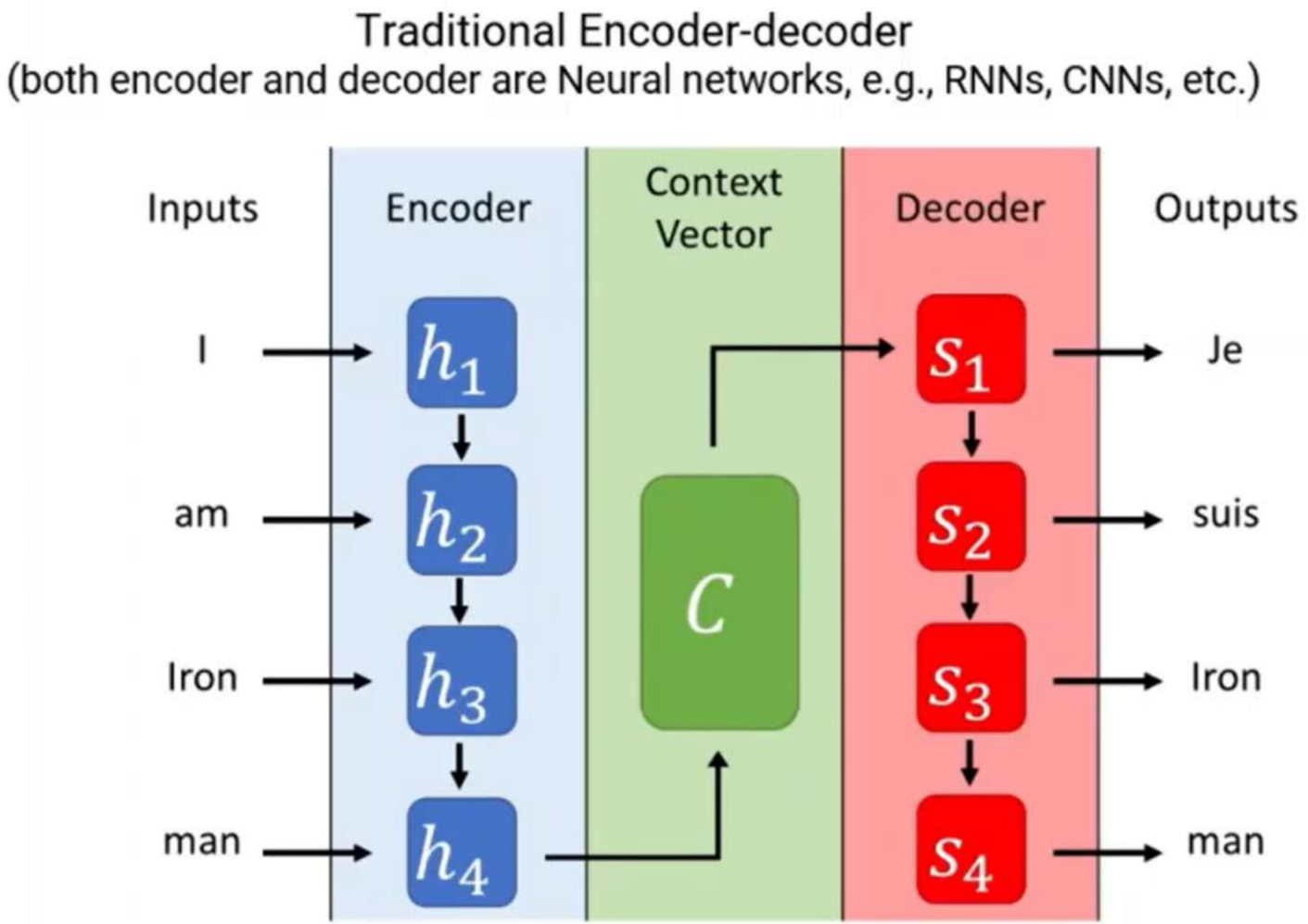
Attention Mechanism



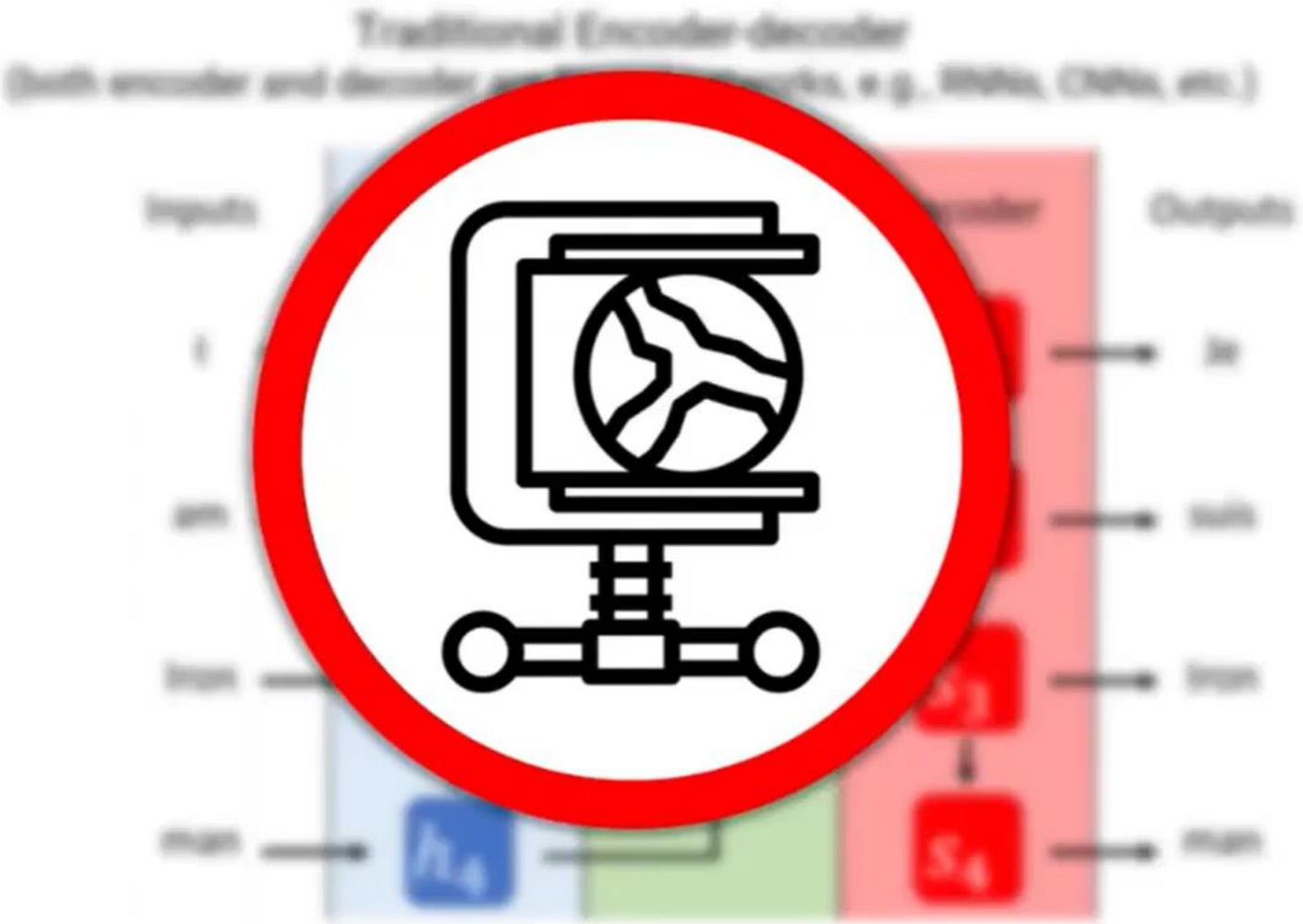
Attention Mechanism



Attention Mechanism

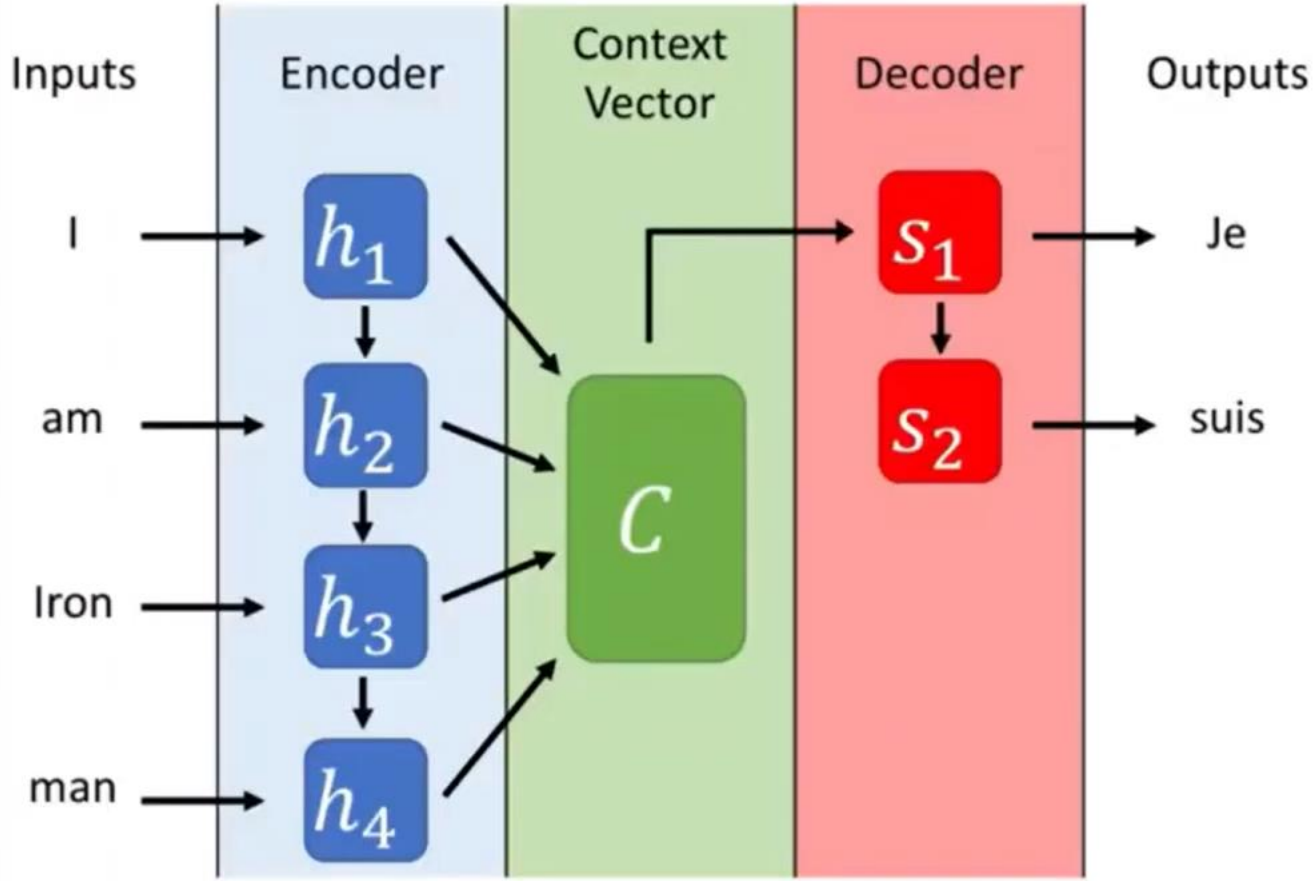


Attention Mechanism

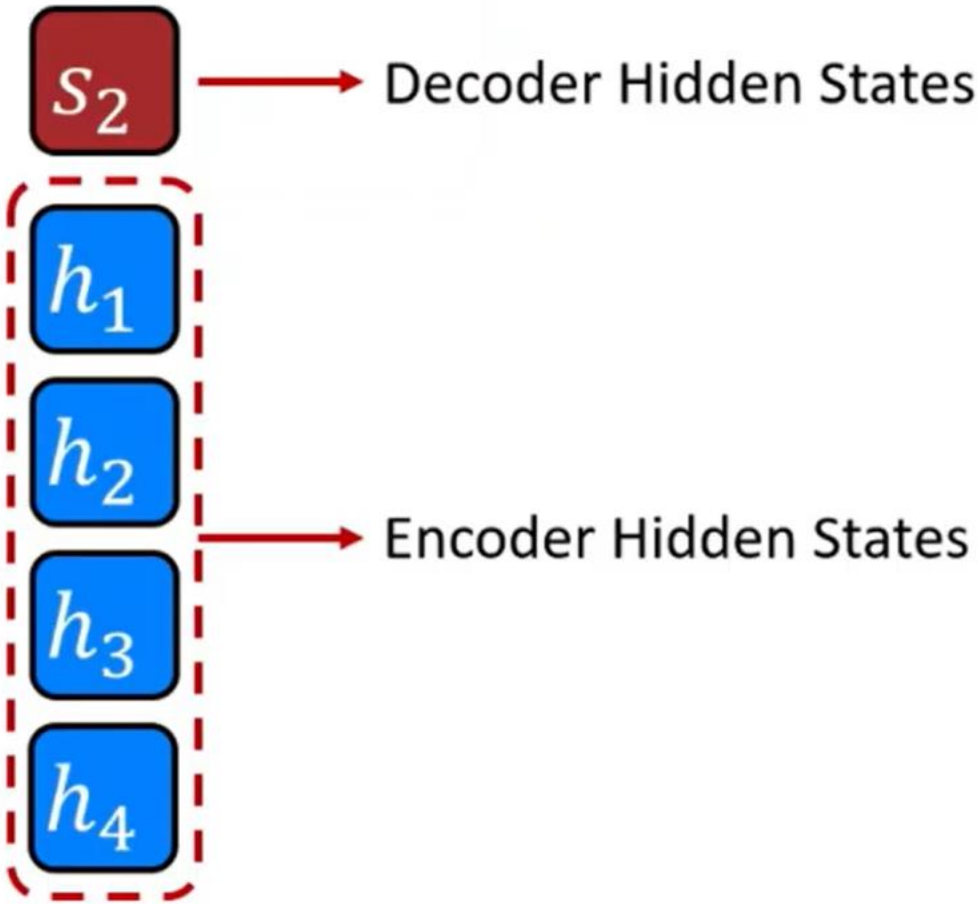


Attention Mechanism

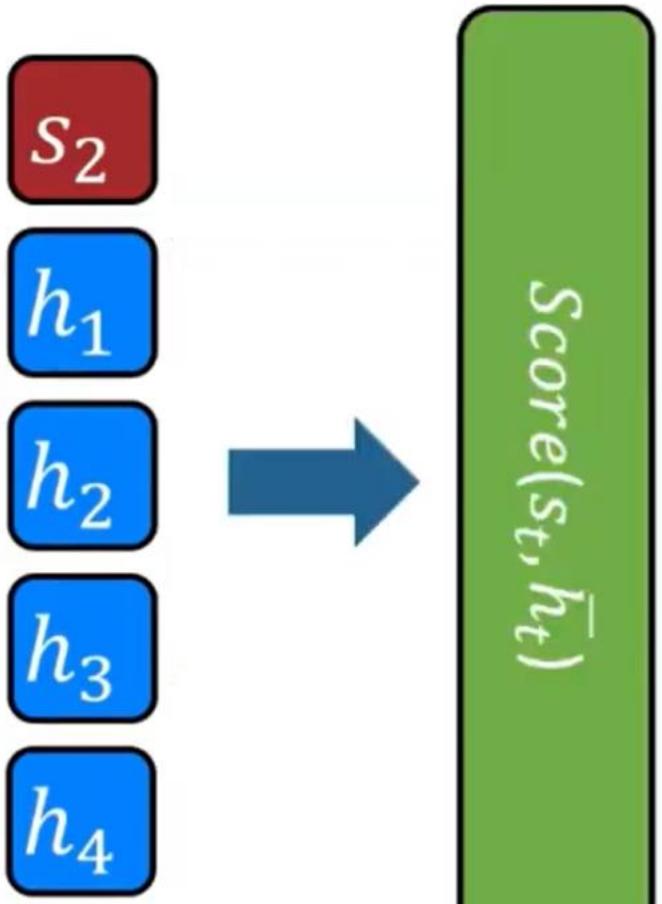
Key Idea: Let's give Context vector access to every input!



Attention Mechanism



Attention Mechanism



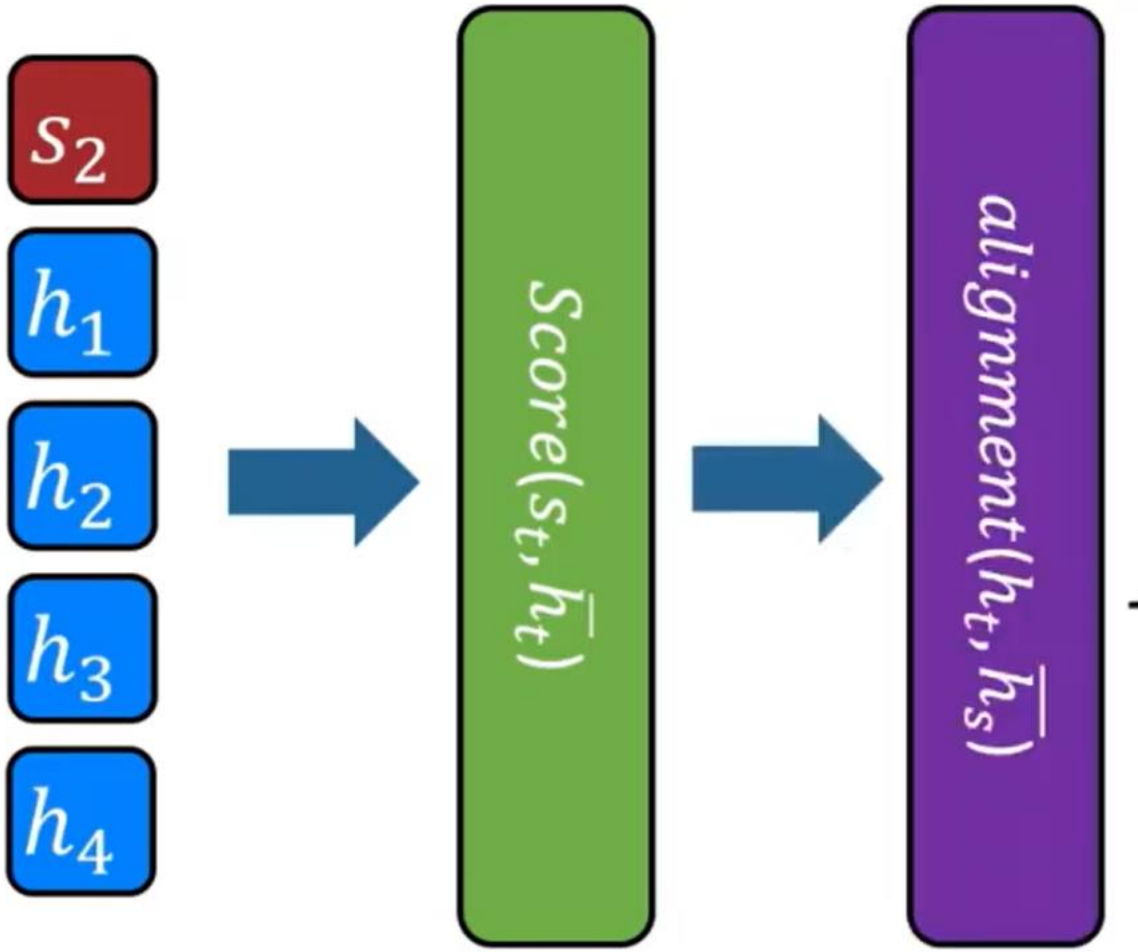
$s_t^T W_a \bar{h}_t$ General

$s_t^T \bar{h}_t$ Dot

$v_a^T \tanh(W_a[s_t; \bar{h}_t])$ Concat

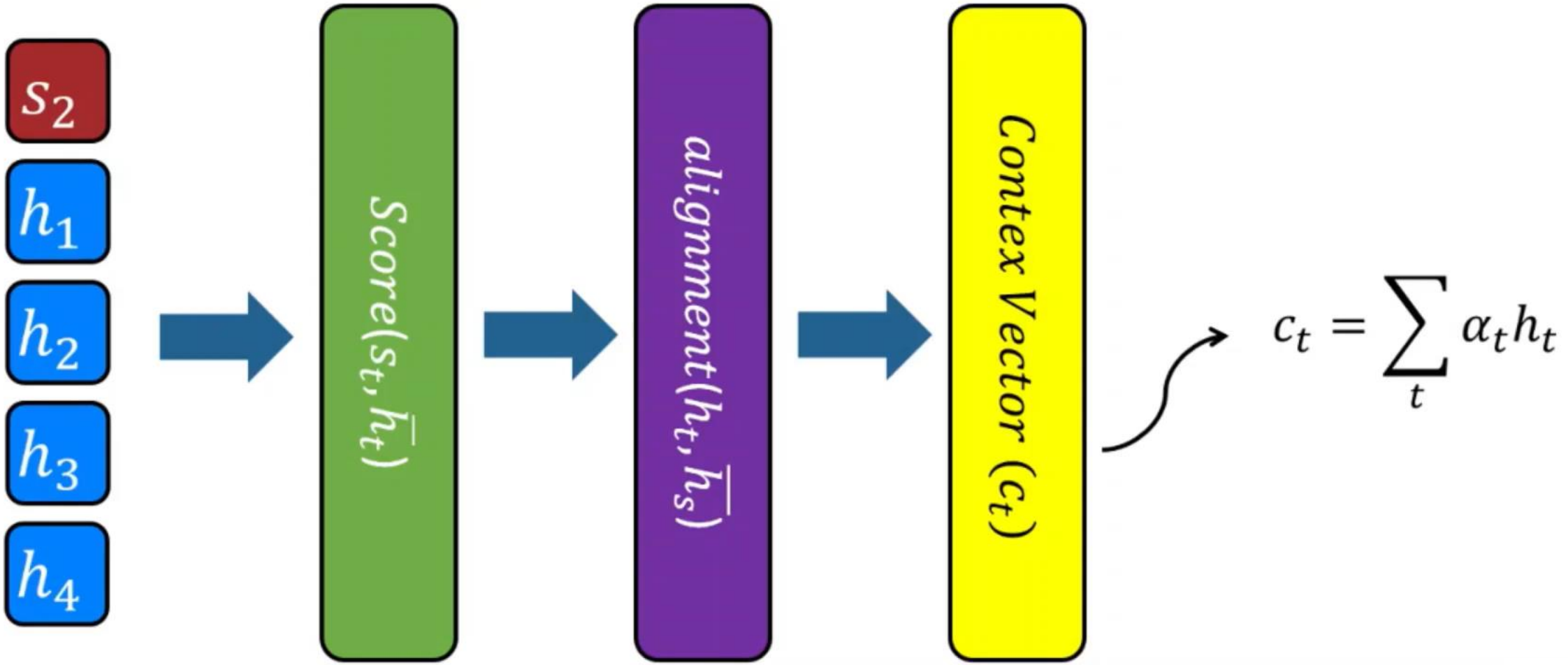
$a_t = softmax(W_a s_t)$ Location

Attention Mechanism



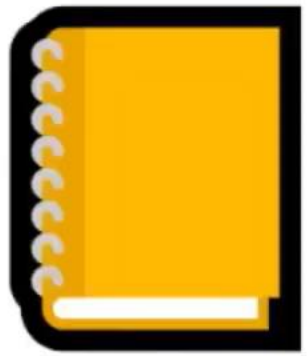
$$a_t(s) = \frac{\exp(Score(h_t, \overline{h_s}))}{\sum_{s'} \exp(Score(h_t, \overline{h_{s'}}))}$$

Attention Mechanism



Attention Mechanism

Attention Model works like an accounting notebook!

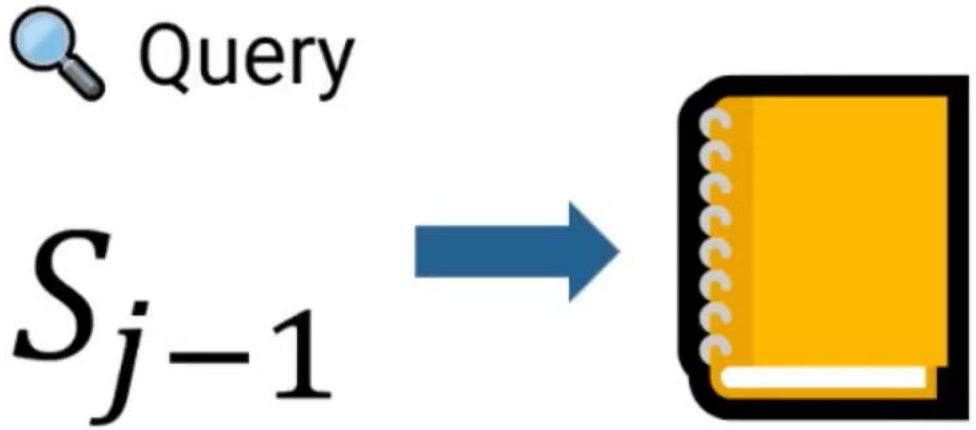


Attention Mechanism

Attention Model works like an accounting notebook!

🔍 Query

S_{j-1}

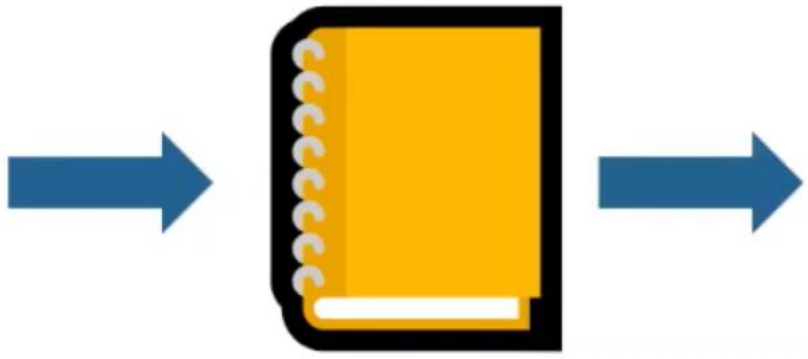


Attention Mechanism

Attention Model works like an accounting notebook!

🔍 Query

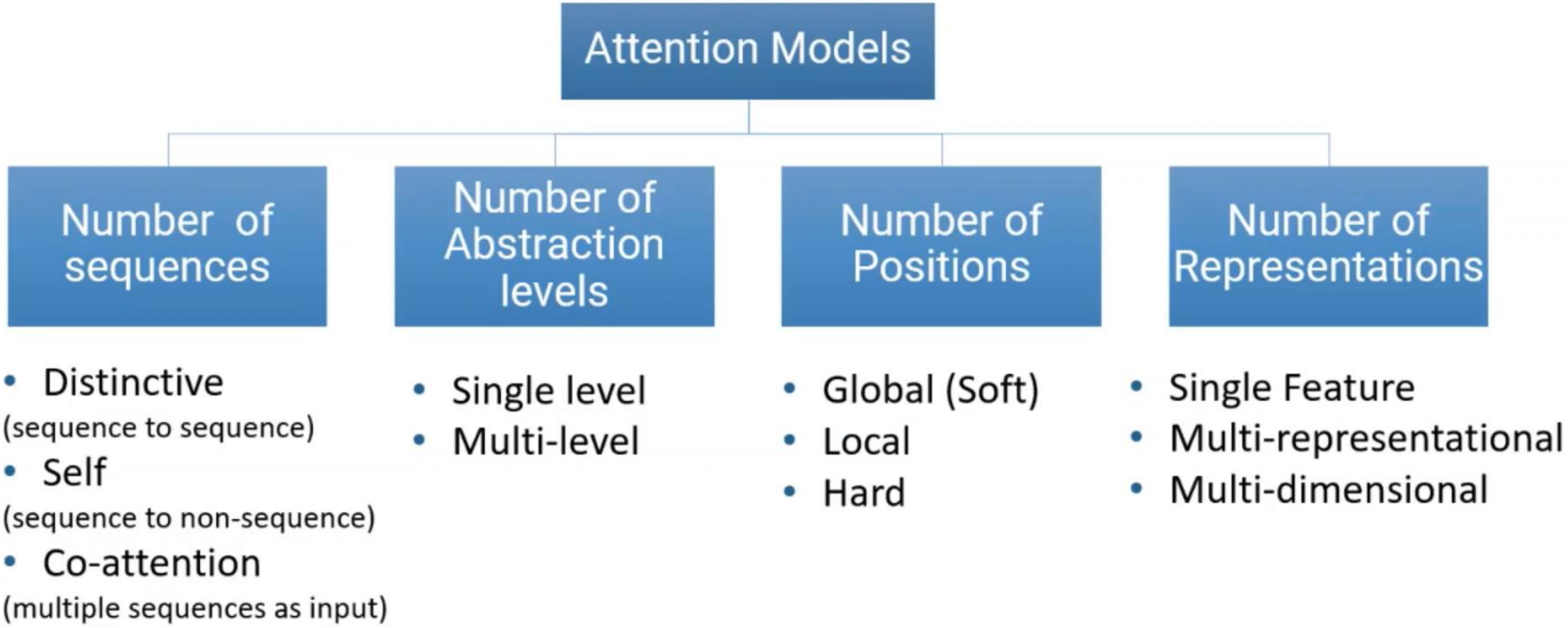
S_{j-1}



📋 Table

keys	attentions
h_1	0.3
h_2	0.6
h_3	0.1

Taxonomy of Attention



BERT

- BERT (Bidirectional Encoder Representations from Transformers) is a recent paper published by researchers at Google AI Language.
- BERT's key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling.
- This is in contrast to previous efforts which looked at a text sequence either from left to right or combined left-to-right and right-to-left training.
- The paper's results show that a language model which is bidirectionally trained can have a deeper sense of language context and flow than single-direction language models.
- In the paper, the researchers detail a novel technique named Masked LM (MLM) which allows bidirectional training in models in which it was previously impossible.

BERT

- BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text.
- In its vanilla form, Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task.
- Since BERT's goal is to generate a language model, only the encoder mechanism is necessary.
- The detailed workings of Transformer are described in a paper by Google.

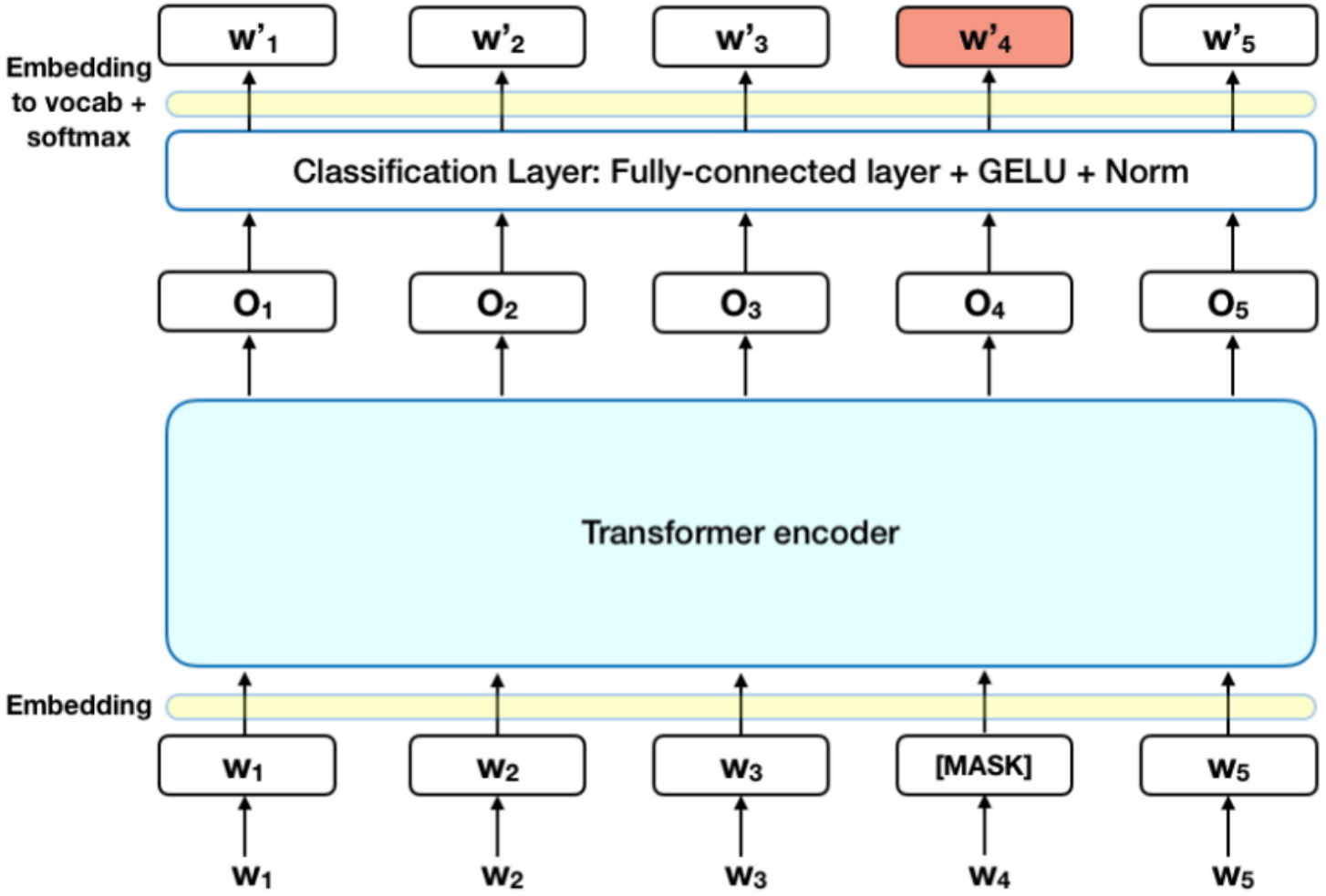
BERT

- As opposed to directional models, which read the text input sequentially (left-to-right or right-to-left), the Transformer encoder reads the entire sequence of words at once.
- Therefore it is considered bidirectional, though it would be more accurate to say that it's non-directional.
- This characteristic allows the model to learn the context of a word based on all of its surroundings (left and right of the word).

BERT - Masked LM (MLM)

- Before feeding word sequences into BERT, 15% of the words in each sequence are replaced with a [MASK] token.
- The model then attempts to predict the original value of the masked words, based on the context provided by the other, non-masked, words in the sequence.
- In technical terms, the prediction of the output words requires:
    1. Adding a classification layer on top of the encoder output.
    2. Multiplying the output vectors by the embedding matrix, transforming them into the vocabulary dimension.
    3. Calculating the probability of each word in the vocabulary with softmax.

BERT - Masked LM (MLM)



GELU - GAUSSIAN ERROR LINEAR UNIT

BERT

- The BERT loss function takes into consideration only the prediction of the masked values and ignores the prediction of the non-masked words.
- As a consequence, the model converges slower than directional models, a characteristic which is offset by its increased context awareness.

BERT

BERT can be used for a wide variety of language tasks, while only adding a small layer to the core model:

- **Classification tasks** such as sentiment analysis are done similarly to Next Sentence classification, by adding a classification layer on top of the Transformer output for the [CLS] token.
- In **Question Answering tasks** (e.g. SQuAD v1.1), the software receives a question regarding a text sequence and is required to mark the answer in the sequence. Using BERT, a Q&A model can be trained by learning two extra vectors that mark the beginning and the end of the answer.
- In **Named Entity Recognition (NER),** the software receives a text sequence and is required to mark the various types of entities (Person, Organization, Date, etc) that appear in the text. Using BERT, a NER model can be trained by feeding the output vector of each token into a classification layer that predicts the NER label.

Referencs

- https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270
- Attention Mechanism In a nutshell (Halfling Wizard)