UPPSALA UNIVERSITET

Machine learning – Block 5

Måns Magnusson
Department of Statistics, Uppsala University

Autumn 2023

This week's lectures

- Word embeddings
- Recurrent Neural Networks
- Attention and Transformers
- BERT models

Practicalities

- One lecture later next week on word embeddings (Väinö Yrjänäinen)
- Transformers are new... but now there is finally a book chapter!

Assignment 4: Evaluation

- Validation or test set when evaluating the models not clear
- VGG could be expanded more to work more with fine-tuning

Section 2

Word embeddings

How do we represent words?

- One-hot encoding
  - A vector of length V (vocabulary size)

$$\text{Uppsala} = [0, ..., 1, ..., 0] = \mathbf{1}_i$$

UPPSALA UNIVERSITET

How do we represent words?

- One-hot encoding
  - A vector of length V (vocabulary size)

$$\text{Uppsala} = [0, \ldots, 1, \ldots, 0] = \mathbf{1}_i$$

- Word embeddings
  - A vector of length D (embedding dimension)

$$\text{Uppsala} = [-0.1231, \ldots, 1.9001, \ldots, 0.012]$$

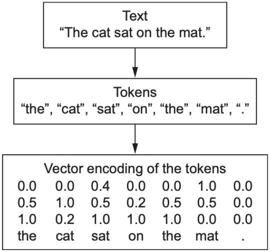


Figure: Representing words as word emnbeddings (Chollet and Allair, 2018, Fig. 6.1)

Word embeddings vs. One-Hot

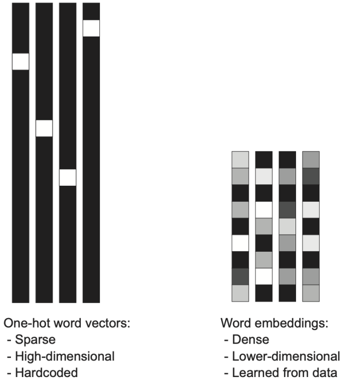


Figure: One-Hot vs. Word embeddings (Chollet and Allair, 2018, Fig. 6.2)

UPPSALA UNIVERSITET

Word embeddings

The quick brown fox jumps over the lazy dog.

- A word type represent meaning in a low-dimensional semantic space
- The distributional hypothesis:
  - Harris (1954) and Firth (1957):
    "A word is characterized by the company it keeps"
  - Semantics (broadly defined) is captured by context
- Lots of different embeddings:
  `word2vec`, GloVe, Probabilistic Embeddings

Word embeddings

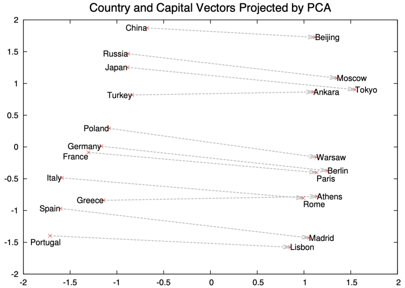


Figure: Word embedding properties (Mikolov et al, 2013)

$$\text{king} - \text{man} + \text{woman} \approx \text{queen}$$

Word embeddings

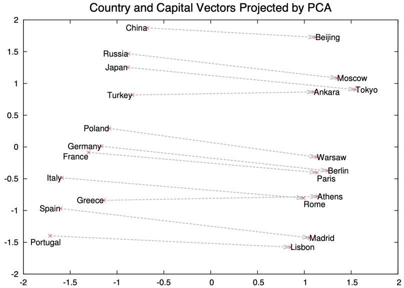


Figure: Word embedding properties (Mikolov et al, 2013)

$$\text{king} - \text{man} + \text{woman} \approx \text{queen}$$

But also (Bolukbasi et al., 2016):

$$\text{computer programmer} - \text{man} + \text{woman} \approx \text{homemaker}$$

Context Matters!

Figure: Context matters (Alammar, 2020)

UPPSALA UNIVERSITET

Section 3

Recurrent Neural Networks

Recurrent Neural Networks

- Recurrent Neural Networks, Recurrent Nets, RNN, ...
- Modeling of temporal data structures, such as
  - Time series data
  - Sequences of words (language models)

Recurrent Neural Networks

- Recurrent Neural Networks, Recurrent Nets, RNN, ...
- Modeling of temporal data structures, such as
  - Time series data
  - Sequences of words (language models)
- Examples of applications:
  - Text classification
  - Sequence / word classification
  - Time series predictions
  - Audio data

Recurrent Neural Networks

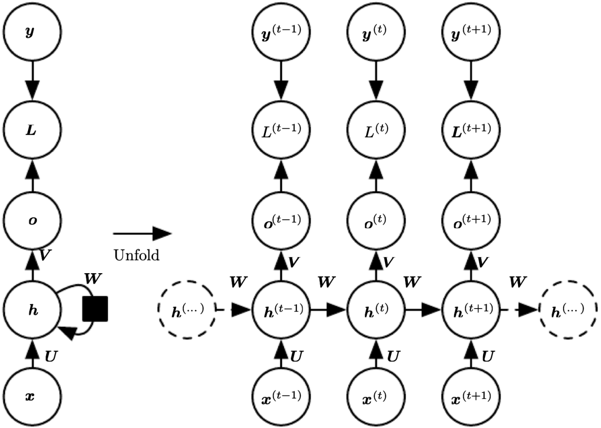


Figure: Recurrent Neural Network (Goodfellow et al, 2017, Fig. 10.3)

Recurrent Neural Networks

$$
\begin{aligned}
a_t &= b + Wh_{t-1} + Ux_t \\
h_t &= \sigma_1(a_t) \\
o_t &= c + Vh_t \\
\hat{y}_t &= \sigma_{\text{output}}(o_t) = \text{softmax}(o_t)
\end{aligned}
$$

Think of h_t as the "state" at timepoint t

Recurrent network with one output

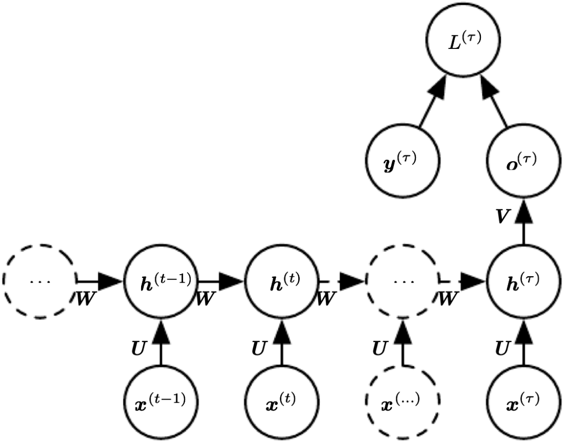


Figure: Recurrent Neural Network with one output (Goodfellow et al, 2017, Fig. 10.5)

Sequence to Sequence: Encoder-Decoder

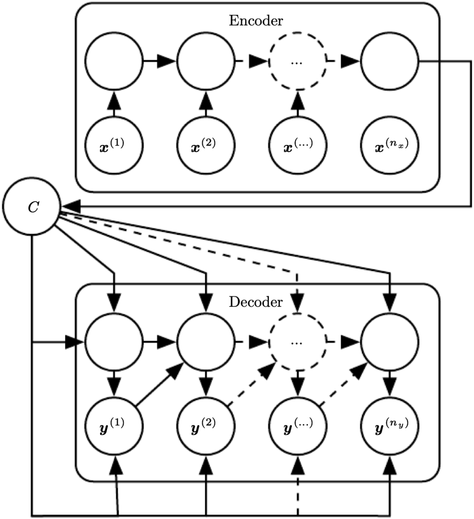


Figure: Encoder-Decoder Recurrent Networks (Goodfellow et al, 2017, Fig. 10.12)

Problems with RNN

- Exploding and vanishing gradients
- Long-term dependencies

Long Short-Term Memory (LSTM)

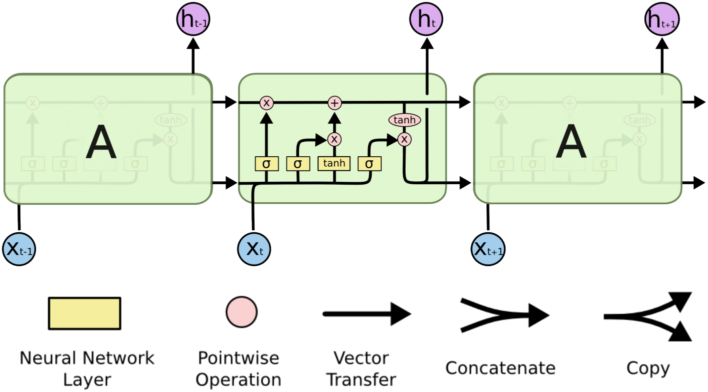


Figure: The LSTM (Olah, 2015)

LSTM cell state

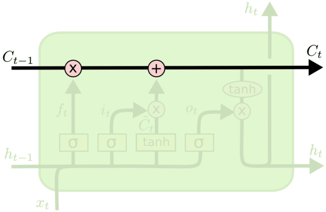
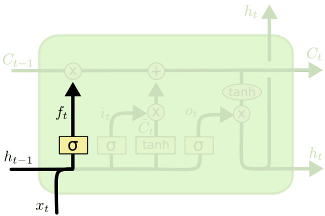


Figure: LSTM cell state, i.e. "carrybelt" (Olah, 2015)

LSTM forget gate

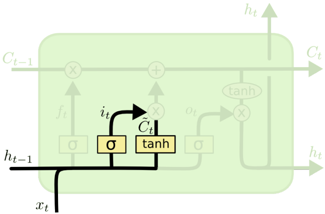


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Figure: LSTM forget gate (Olah, 2015)

UPPSALA UNIVERSITET

LSTM input gate



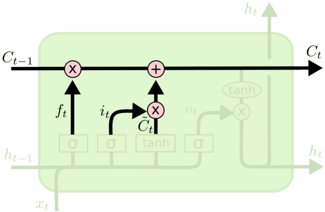
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

Figure: LSTM input gate (Olah, 2015)

UPPSALA UNIVERSITET

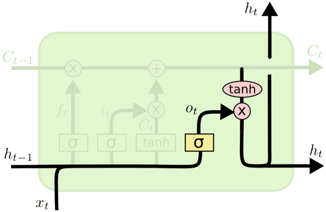
LSTM cell state update



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figure: Update cell state (Olah, 2015)

LSTM output gate



$$o_t = \sigma\left(W_o \left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Figure: LSTM output gate (Olah, 2015)

UPPSALA UNIVERSITET

Problems

- Still a recurrent structure, (vanishing and exploding gradients)
- Long-term dependencies still difficult
- Hard to do transfer learning

UPPSALA UNIVERSITET

Section 4

Transformers

The Transformer

- Introduced in 2017 in Vaswani et al. (2017)
- Behind the recent progress in NLP: BERT, GPT-2, GPT-3, etc.

The Transformer

- Introduced in 2017 in Vaswani et al. (2017)
- Behind the recent progress in NLP: BERT, GPT-2, GPT-3, etc.
- Becoming de-facto standard in industry and academia

The Transformer

- Introduced in 2017 in Vaswani et al. (2017)
- Behind the recent progress in NLP: BERT, GPT-2, GPT-3, etc.
- Becoming de-facto standard in industry and academia
- Four benefits:
  - Enables more parallelism
  - Better handling of long-range dependencies
  - Brings transfer learning to text data
  - Enables deeper networks

A Sequence-to-Sequence Model

Figure: Attention (Allamar, 2018)

Stacked Encoder-Decoder Structure

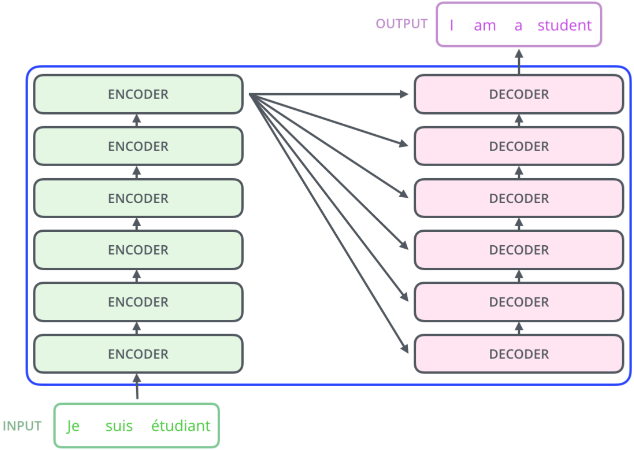


Figure: Attention (Allamar, 2018)

Transformer

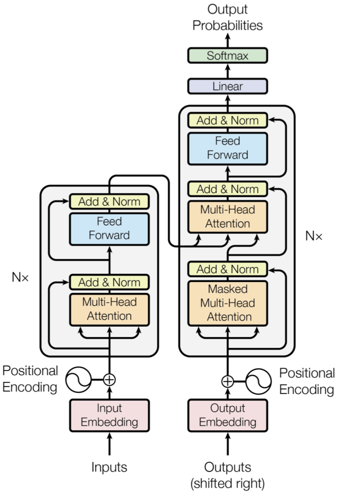


Figure: The Transformer Architecture (Vaswani et al., 2017)

UPPSALA UNIVERSITET

The encoder vs. the decoder

- Encoder:
  - Input: words (embeddings)
  - Output: contextualized embeddings
- Decoder:
  - Input: contextualized embeddings and previous words (embeddings)
  - Output: words (embeddings)

The Transformer Layer (Encoder layer)

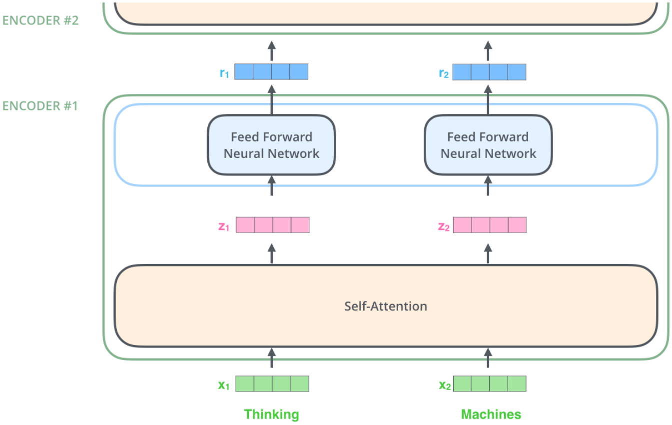


Figure: The Encoder Layer (Alammar, 2018b)

Scaled Dot-Product Attention

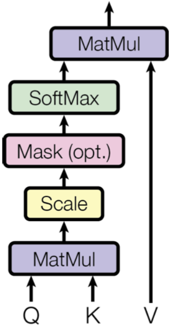


Figure: Scaled Dot-Product Attention (Vaswani et al., 2017)

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

Attention components

- (Q)uery: Word *i* query other words
- (K)ey: The other words return their key to *i*
- (V)alue: The value of the other words to *i*

Computing Q, V and K

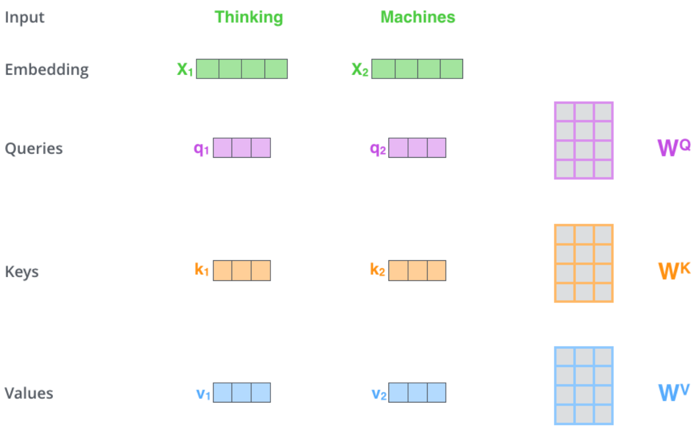


Figure: Attention heads (Alammar, 2018b)

Computing Self-Attention

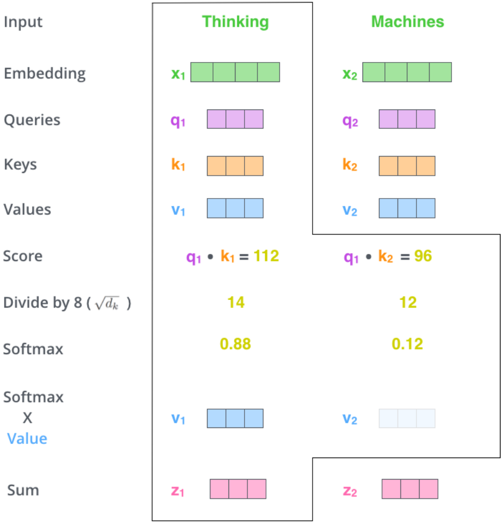


Figure: Attention (Alammar, 2018b)

Multi-Head Attention

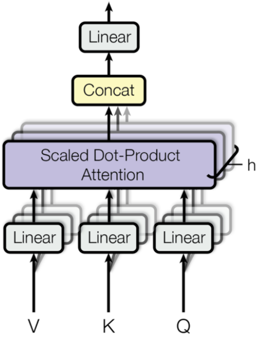


Figure: Scaled Dot-Product Attention (Vaswani et al., 2017)

Attentions Heads

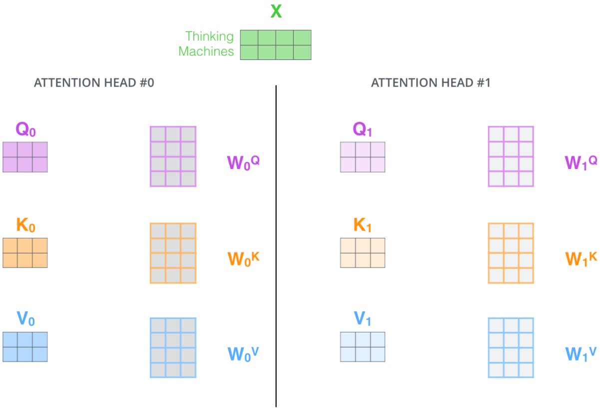


Figure: Attention heads (Alammar, 2018b)

Multi-head attention

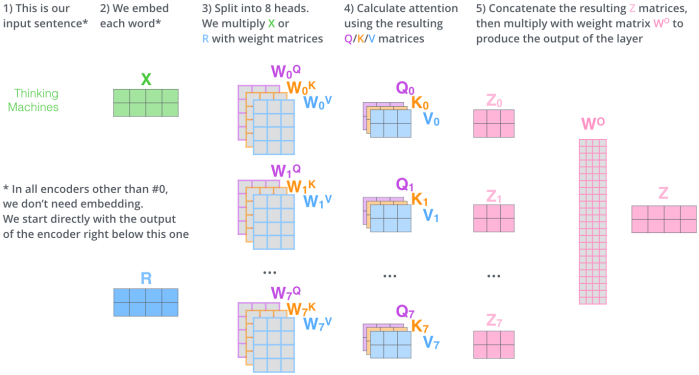


Figure: Attention heads (Alammar, 2018b)

UPPSALA
UNIVERSITET

Multi-Head Attention example

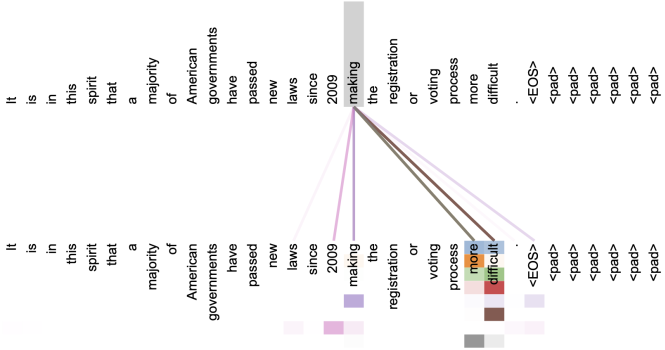


Figure 3: An example of the attention mechanism following long-distance dependencies in the encoder self-attention in layer 5 of 6. Many of the attention heads attend to a distant dependency of the verb 'making', completing the phrase 'making...more difficult'. Attentions here shown only for the word 'making'. Different colors represent different heads. Best viewed in color.

Figure: Attention (Vaswani et al., 2017)

Positional Encoding

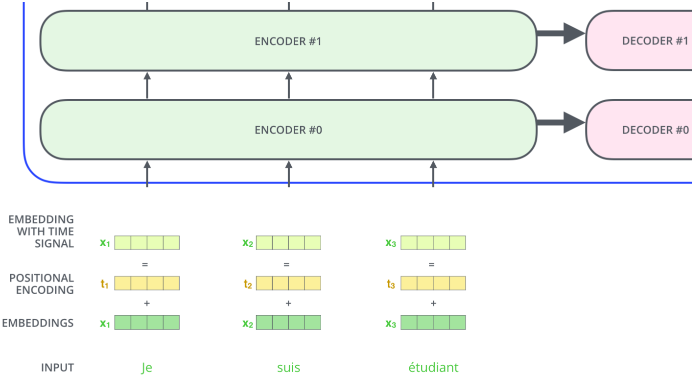


Figure: Attention heads (Alammar, 2018b)

Positional Encoding

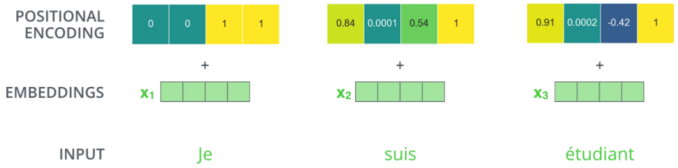


Figure: Adding positional encodings to embeddings (Alammar, 2018b)

UPPSALA UNIVERSITET

- Practicalities
- Word embeddings
- Recurrent Neural Networks
  — LSTM
- Transformers
  — Attention
  — Multi-Head Attention
  — Positional encoding
  — Add and Normalize
- BERT
  — Training BERT
  — Using BERT

Add and Normalize

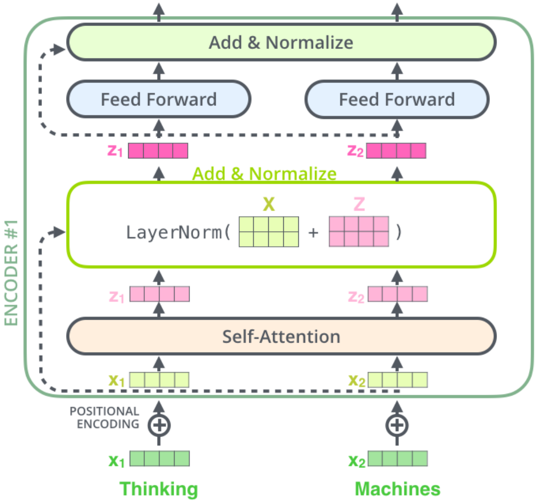


Figure: Add and Normalize (Alammar, 2018b)

Transformer

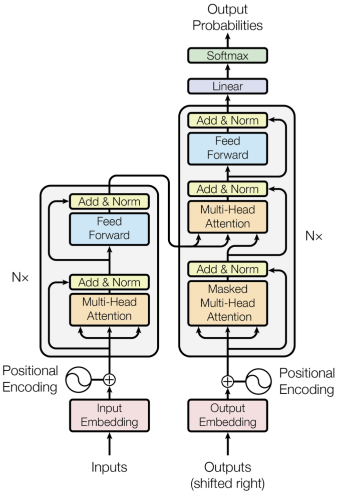


Figure: The Transformer Architecture (Vaswani et al., 2017)

Section 5

BERT

BERT

- Bidirectional Encoder Representations from Transformers (BERT)
- Introduced in 2018/2019 in Devlin et al. (2018)

BERT

- Bidirectional Encoder Representations from Transformers (BERT)
- Introduced in 2018/2019 in Devlin et al. (2018)
- State-of-the-Art in many text prediction tasks, such as
  - Question-Answering
  - Named-Entity Recognition
  - Text Classification

BERT

- Bidirectional Encoder Representations from Transformers (BERT)
- Introduced in 2018/2019 in Devlin et al. (2018)
- State-of-the-Art in many text prediction tasks, such as
  - Question-Answering
  - Named-Entity Recognition
  - Text Classification
- Many new versions has come out RoBERTa, DistillBERT, AlBERT etc.

BERT

UPPSALA UNIVERSITET

- Bidirectional Encoder Representations from Transformers (BERT)
- Introduced in 2018/2019 in Devlin et al. (2018)
- State-of-the-Art in many text prediction tasks, such as
  - Question-Answering
  - Named-Entity Recognition
  - Text Classification
- Many new versions has come out RoBERTa, DistillBERT, AlBERT etc.
- Pre-trained on a large corpus

BERT

- Bidirectional Encoder Representations from Transformers (BERT)
- Introduced in 2018/2019 in Devlin et al. (2018)
- State-of-the-Art in many text prediction tasks, such as
  - Question-Answering
  - Named-Entity Recognition
  - Text Classification
- Many new versions has come out RoBERTa, DistillBERT, AlBERT etc.
- Pre-trained on a large corpus
- Available both in English, Swedish and many other languages (The National Library)

BERT and transfer learning

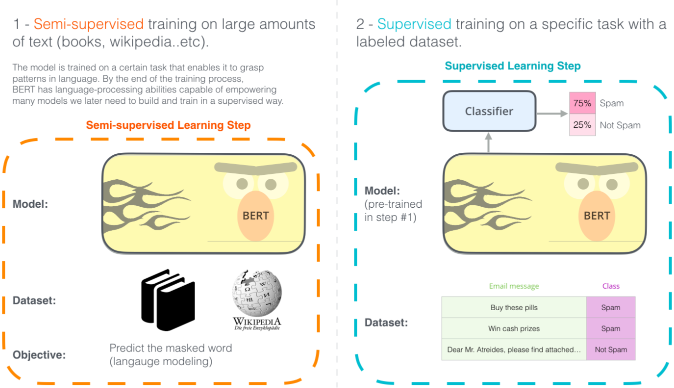


Figure: Using BERT for Transfer Learning (Alammar, 2018c)

The BERT model

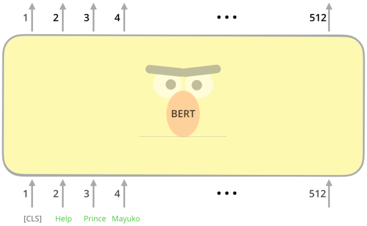


Figure: The BERT model (Alammar, 2018c)

BERT Architecture

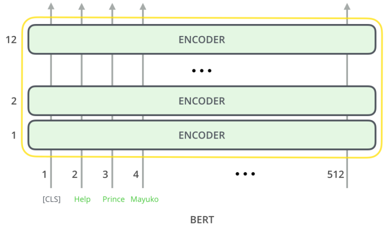


Figure: Opening up BERT (Vaswani et al., 2017)

Task 1: Masked Language Model

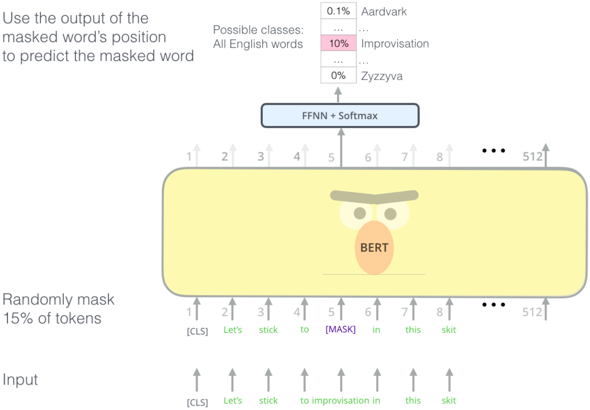


Figure: Masked Language Modeling (Alammar, 2018c)

Next Sentence Prediction

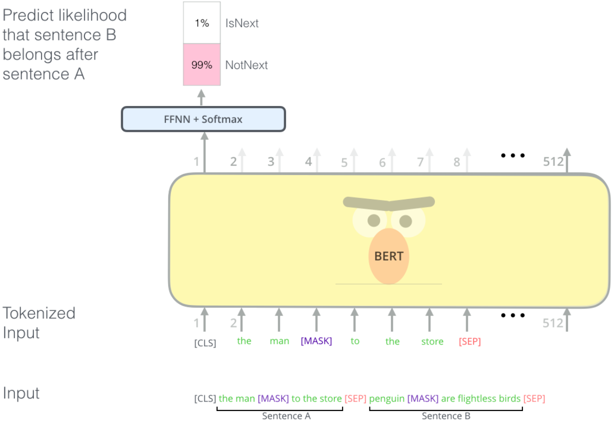


Figure: Next Sentence Prediction (Alammar, 2018c)

UPPSALA UNIVERSITET

Using BERT for Classification

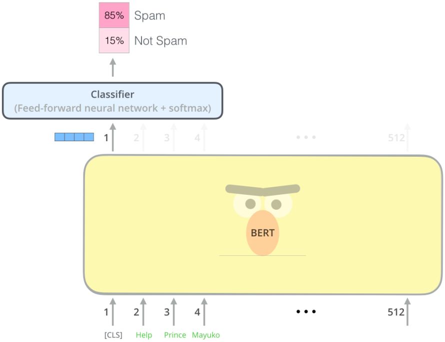


Figure: Using BERT for classification (Alammar, 2018c)

BERT and Contextualized embeddings

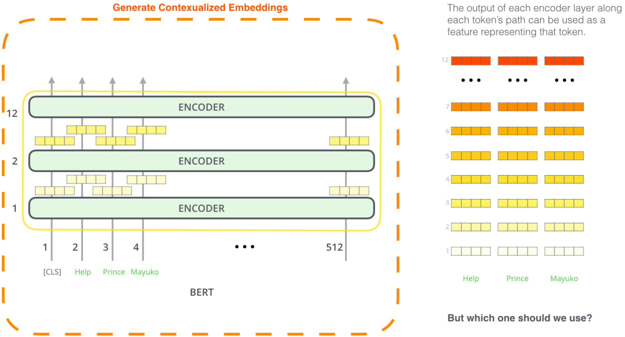


Figure: Contextualized Embeddings (Alammar, 2018c)

Using Contextualized Embeddings

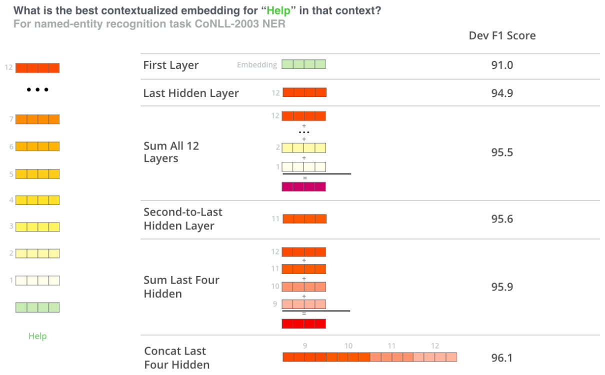


Figure: Using Contextualized Embeddings (Alammar, 2018c)