Neural Network Architectures

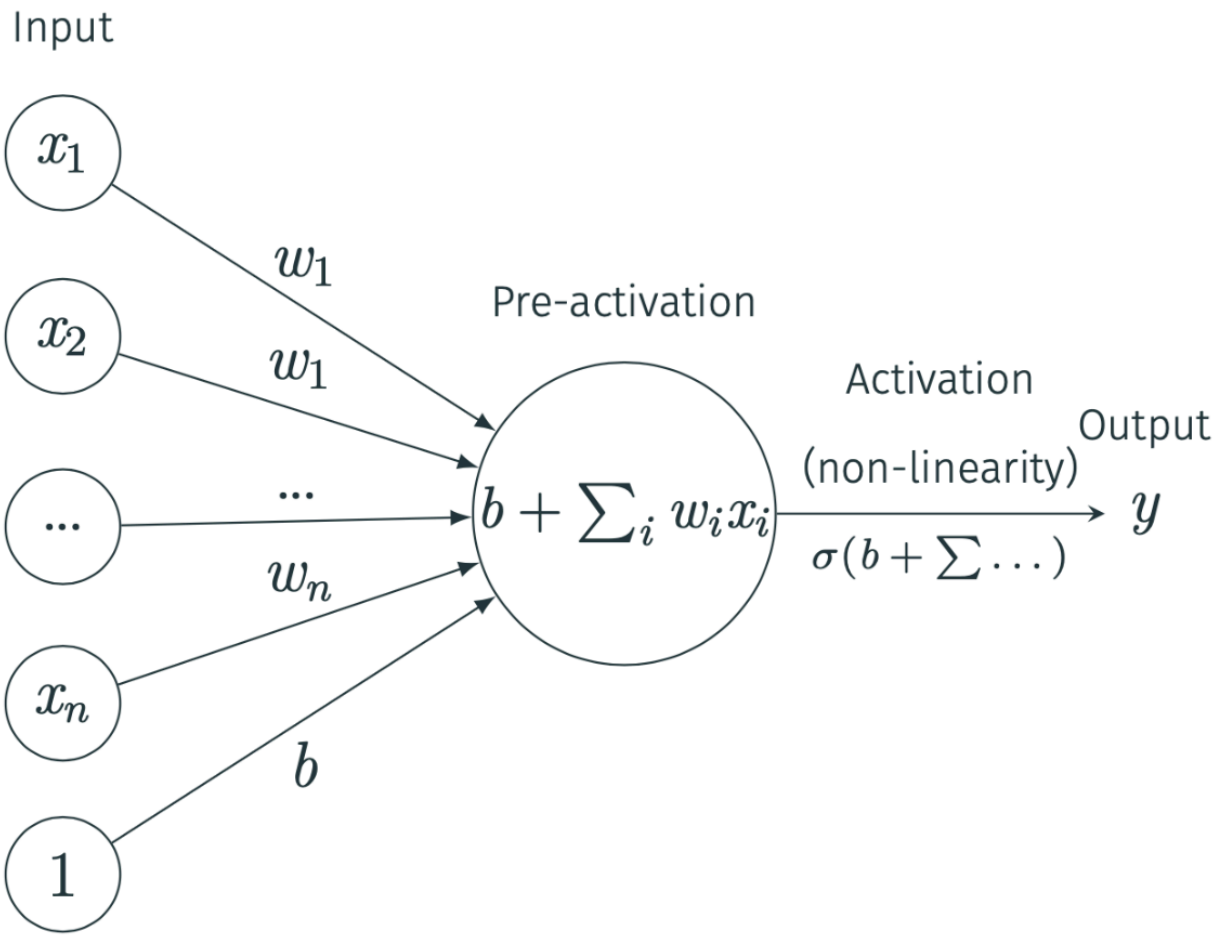
Prof. Dr. Mohsen Mesgar

Universität Duisburg-Essen

Basics …

UNIVERSITÄT DUISBURG ESSEN

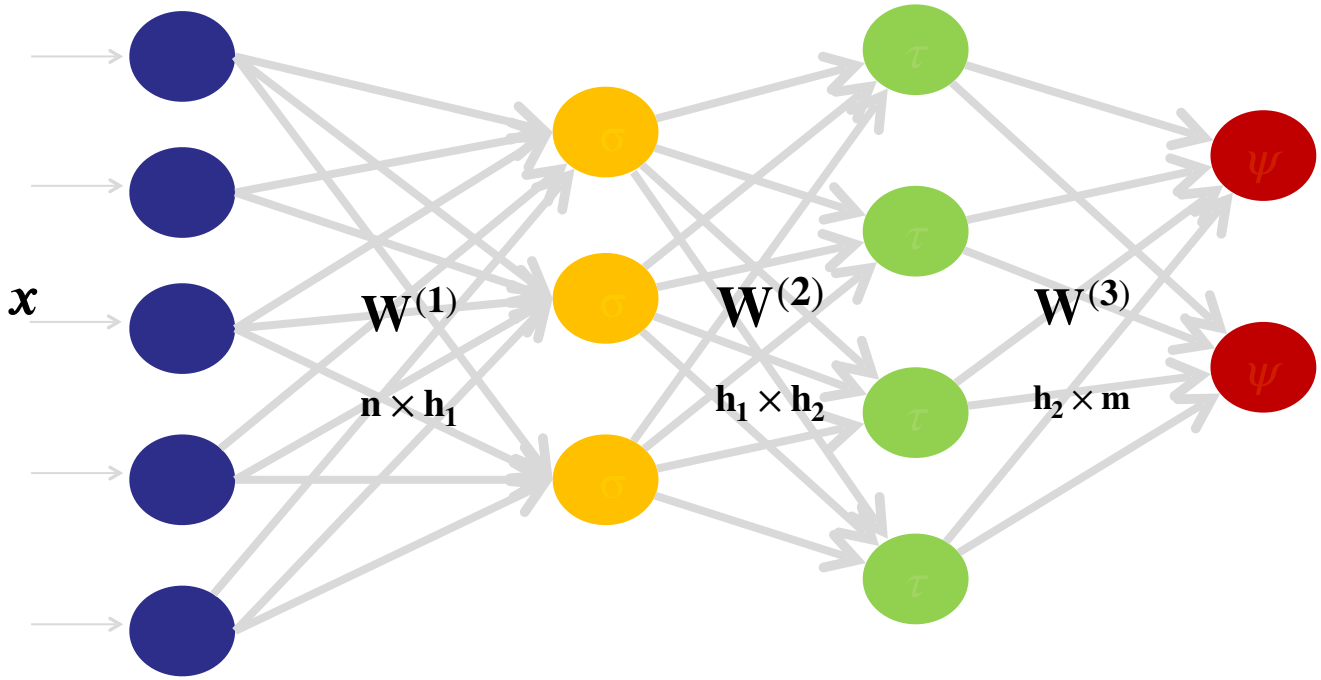
Offen im Denken



Basics …

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken



Basics …

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Algorithm 2.1 Online stochastic gradient descent training.

Input:

- Function $f(\boldsymbol{x}; \Theta)$ parameterized with parameters Θ.
- Training set of inputs $\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}$ and desired outputs $\boldsymbol{y_1}, \ldots, \boldsymbol{y_n}$.
- Loss function L.

1: **while** stopping criteria not met **do**
2: Sample a training example $\boldsymbol{x_i}, \boldsymbol{y_i}$
3: Compute the loss $L(f(\boldsymbol{x_i}; \Theta), \boldsymbol{y_i})$
4: $\hat{\boldsymbol{g}} \leftarrow$ gradients of $L(f(\boldsymbol{x_i}; \Theta), \boldsymbol{y_i})$ w.r.t Θ
5: $\Theta \leftarrow \Theta - \eta_t \hat{\boldsymbol{g}}$
6: **return** Θ

Basics …

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Cross-entropy loss

- $\ell(\mathbf{t}, \mathbf{y}) = -\sum_{j} \mathbf{t}_j \log(\mathbf{y}_j)$

Example: $\mathbf{t} = (0, 1, 0, 0),\ \mathbf{y} = (0.25, 0.3, 0.4, 0.05)$

Square loss:

- $0.25^2 + 0.7^2 + 0.4^2 + 0.05^2$

Cross-entropy loss:

- $-\log(0.3)$

Deep Learning Architectures

Deep learning architectures

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Processing images

- Convolutional neural networks

Processing sequences (e.g., text)

- Recurrent neural networks
- Transformers

Generating data

- Generative adversarial networks

Convolutional Neural Networks

Variable input in image recognition

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

The 1d convolution operation

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

$$(f * g)[i] = \sum_{m=-M}^{M} f[i-m]\,g[m]$$

- ◊ $*$ is the convolution operator
- ◊ $f(x)$ is the input representation
- ◊ i is the current position in the input representation
- ◊ M is the window size
- ◊ $g[m]$ is the weight for an input element with distance m to the current input
- ◊ → it is also often referred to as the filter.

- ◊ Careful! You will find many terminological alternatives in the literature: w (for weights) instead of g, n or t (for time) instead of i, a (for age) instead of m, …

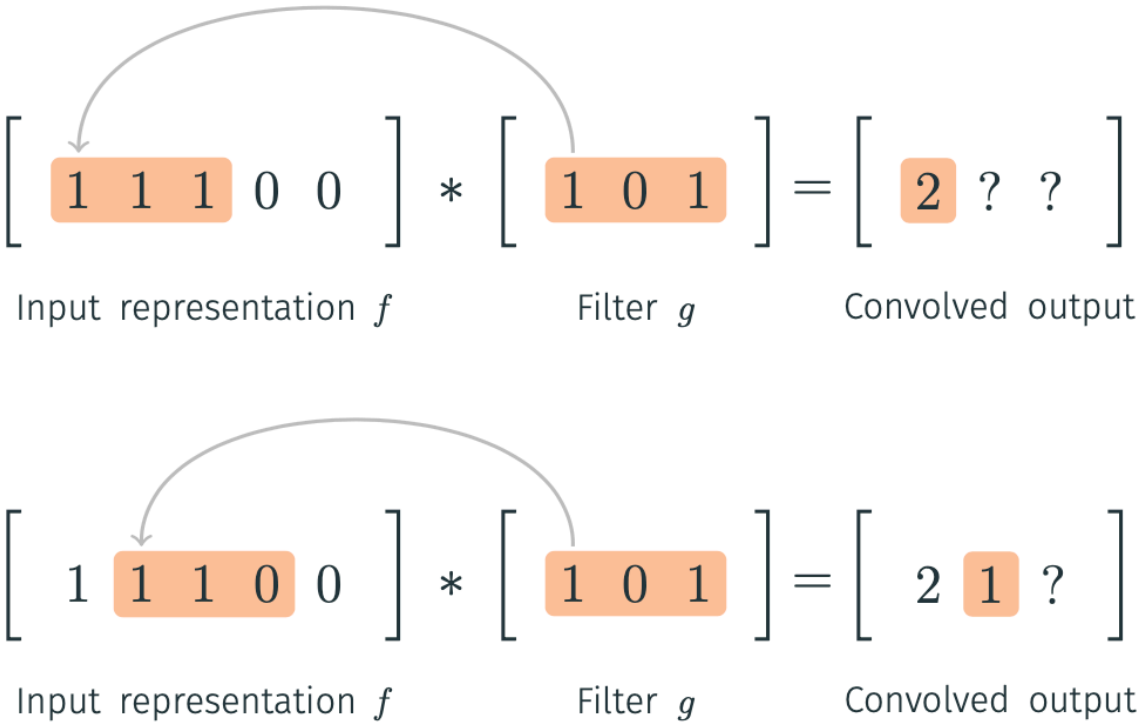
1-D Convolution

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} ? & ? & ? \end{bmatrix}$$

Input representation f | Filter g (also 'kernel') | Convolved output

(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

1-D Convolution



(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

2-D Convolution

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} ? & ? & ? \\ ? & ? & ? \\ ? & ? & ? \\ ? & ? & ? \end{bmatrix}$$

Input representation f

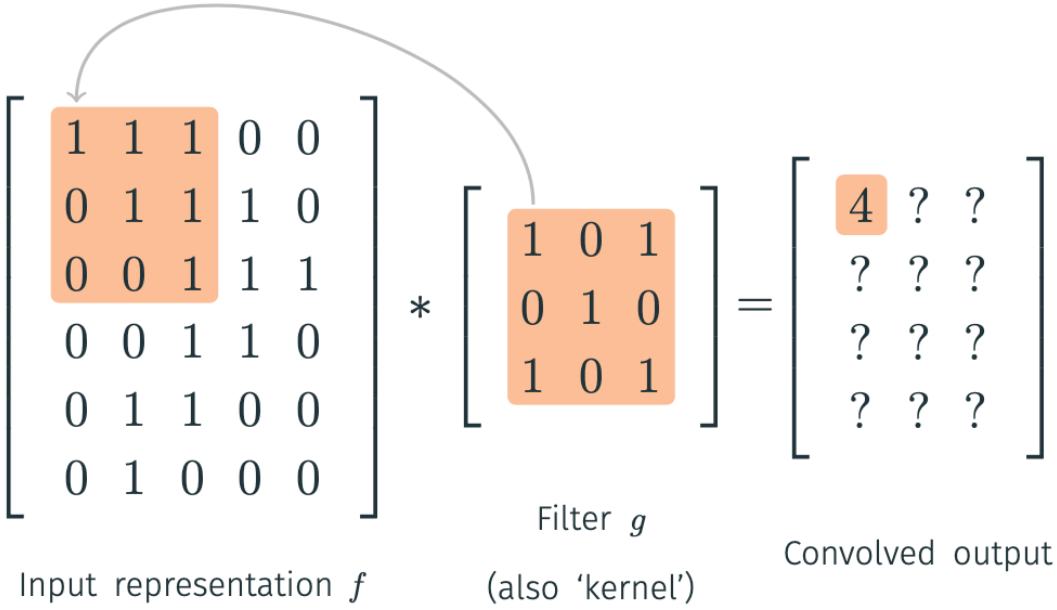
Filter g

(also 'kernel')

Convolved output

(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

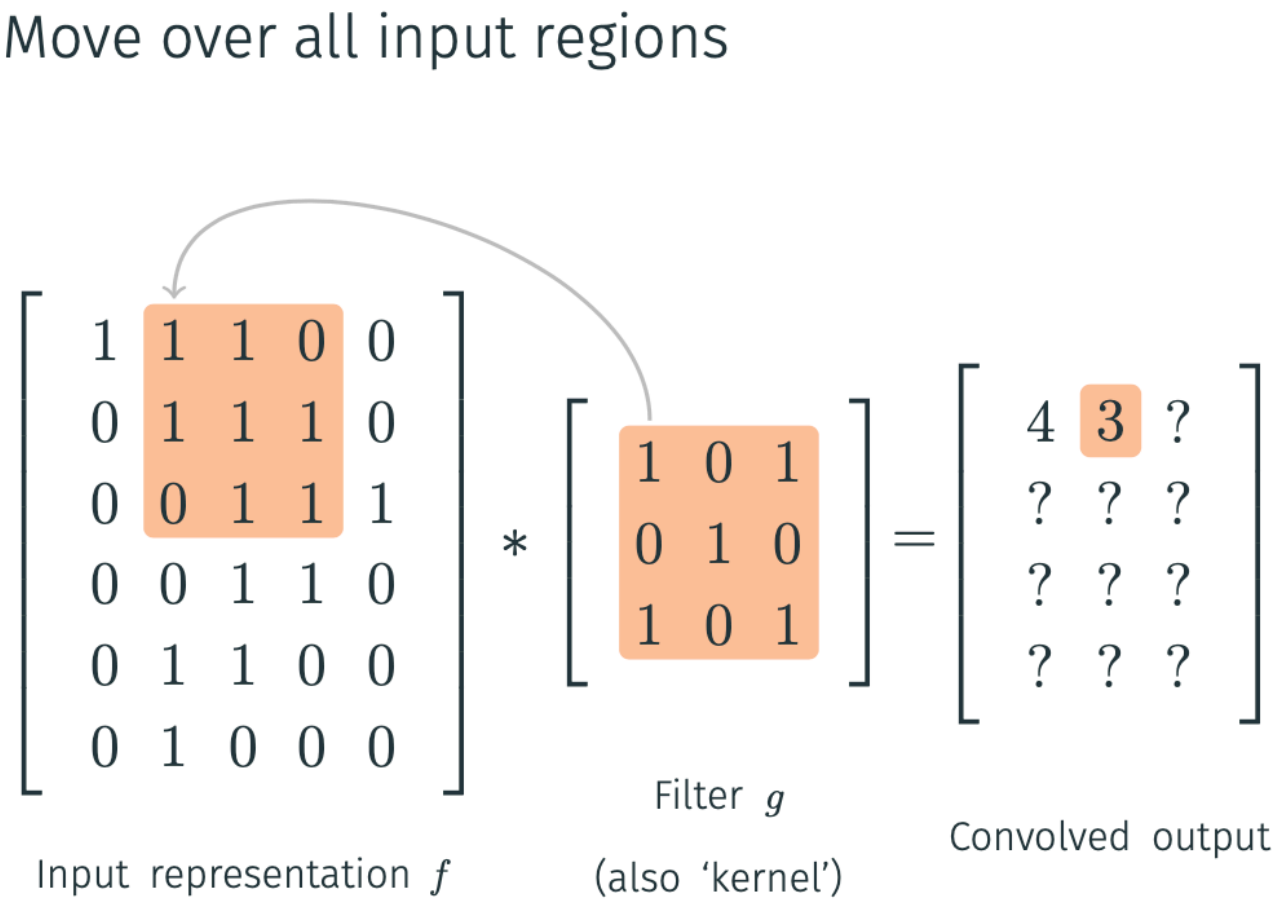
2-D Convolution



$$\underbrace{1\cdot 1+1\cdot 0+1\cdot 1}_{\text{1st row}}+\underbrace{0\cdot 0+1\cdot 1+1\cdot 0}_{\text{2nd row}}+\underbrace{0\cdot 1+0\cdot 0+1\cdot 1}_{\text{3rd row}}=4$$

(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

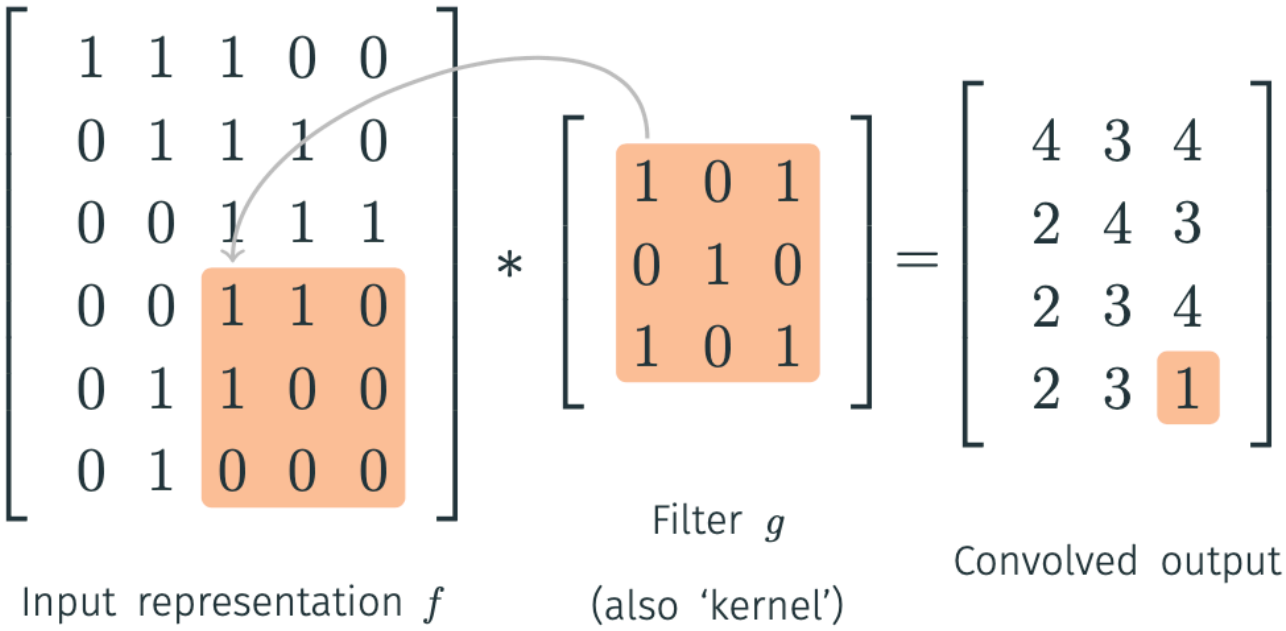
2-D Convolution



(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

2-D Convolution

The goal is to **learn** good kernels (filter parameters)



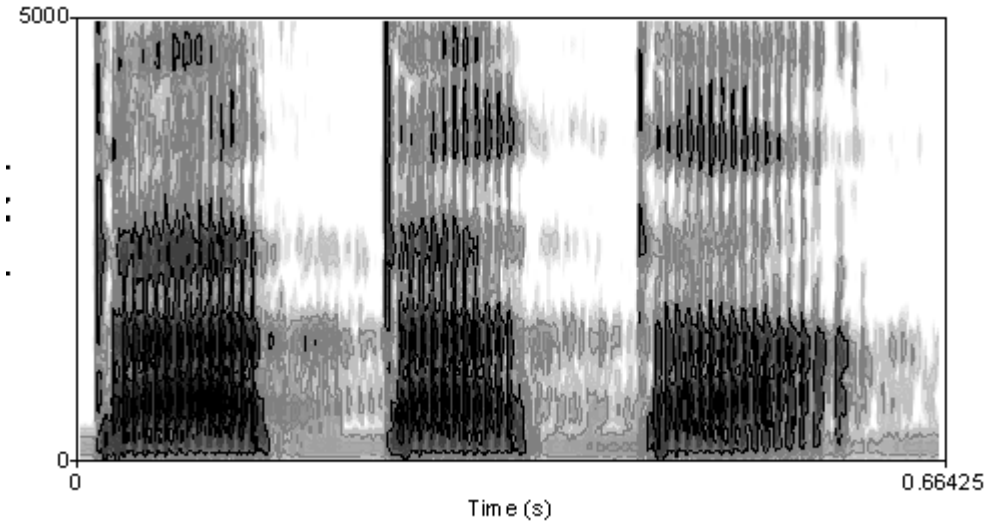
(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

Convolutional networks in speech recognition

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Convolution over frequency to account for variations in the vocal tracts of different speakers



And in texts?

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Sentiment classification
  - The movie was **really good**.
  - We saw this **really good** movie.
  - The movie, which we saw yesterday with all the colleagues in this tiny movie theatre next to the bridge, was (despite my expectations) **really good**.

- For this task, position information is not very relevant.

Dimensionality

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Advantages of text flow:
  - Usually only one dimension
    as opposed to two dimensions in images

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean …

Convolution operation

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Input sentence $\mathbf{x}_{i:i+n}$ are stacked word embeddings of dimensionality d

- Convolutional filter $\mathbf{w} \in \mathbb{R}^{h \times d}$

where h is the filter size (how many neighboring words are convoluted)

- Convolution operation $\mathbf{w} * \mathbf{x}_{i:i+n}$

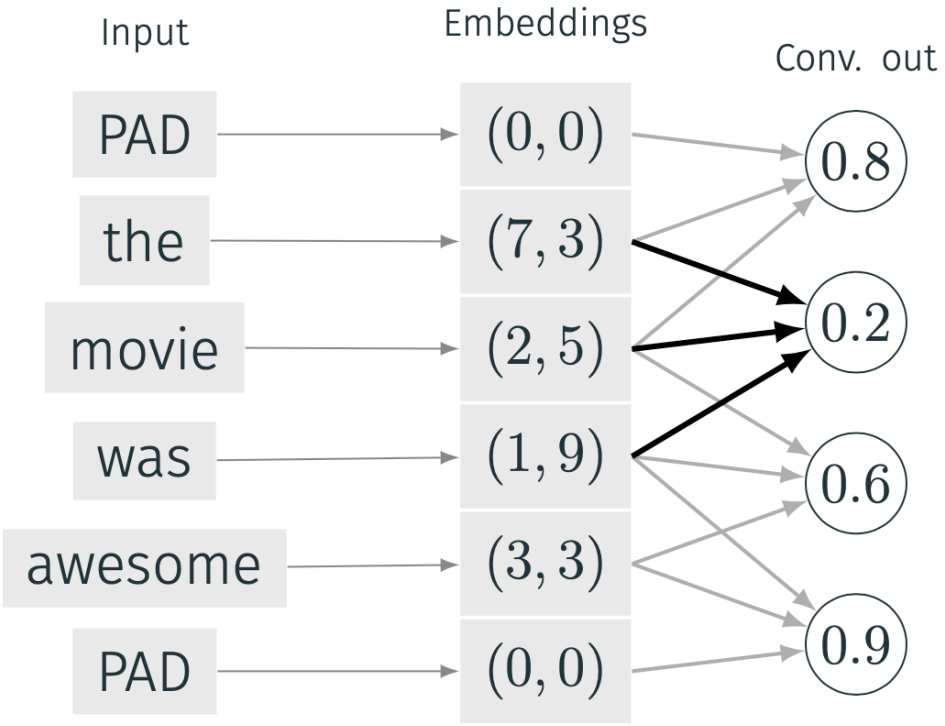
- Non-linear operation

$$c_i = \mathrm{ReLU}(\mathbf{w} * \mathbf{x}_{i:i+n} + b)$$

(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

Example

$d = 2 \qquad h = 3$



(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

CNN vs FeedForward

UNIVERSITÄT DUISBURG ESSEN

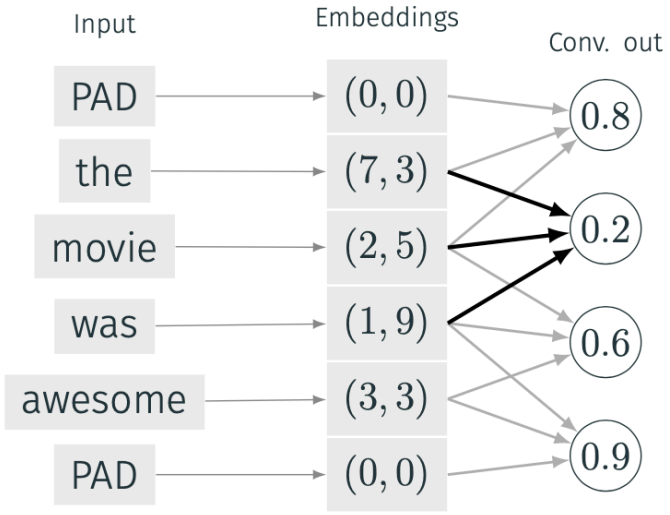
Offen im Denken

Not every input is connected to every output in the following layer

- **sparse connectivity** (vs fully-connected/dense layers)

For each window, we use the same weights and bias values

- **parameter sharing**



(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

Stride

Stride: the step size for moving over the sentence

- Stride 1 common in NLP; other values in comp. vision

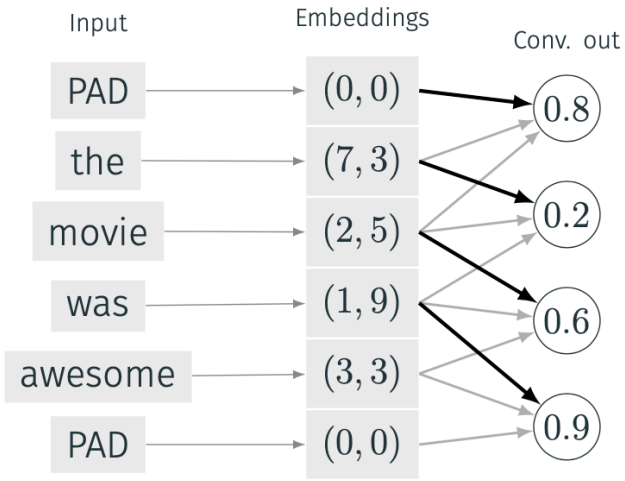


Figure 1: Stride = 1

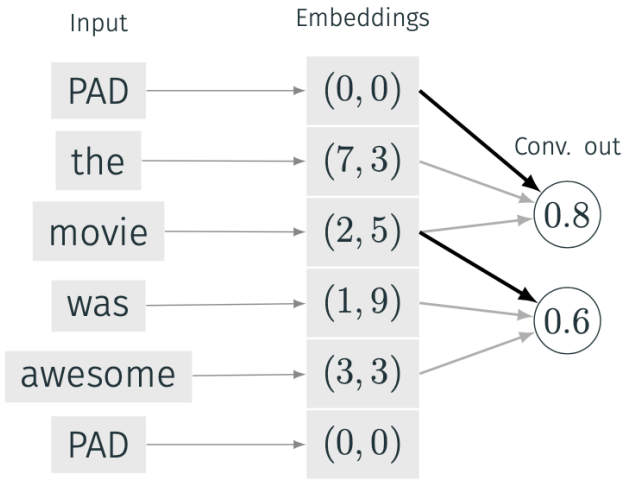


Figure 2: Stride = 2

(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

Pooling

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Another new building block: pooling layer

- Idea: capture the most important activation

Let $c_1, c_2, \ldots, \in \mathbb{R}$ denote the output values of the convolutional filter

Compute output o for a **max-over-time pooling** layer as

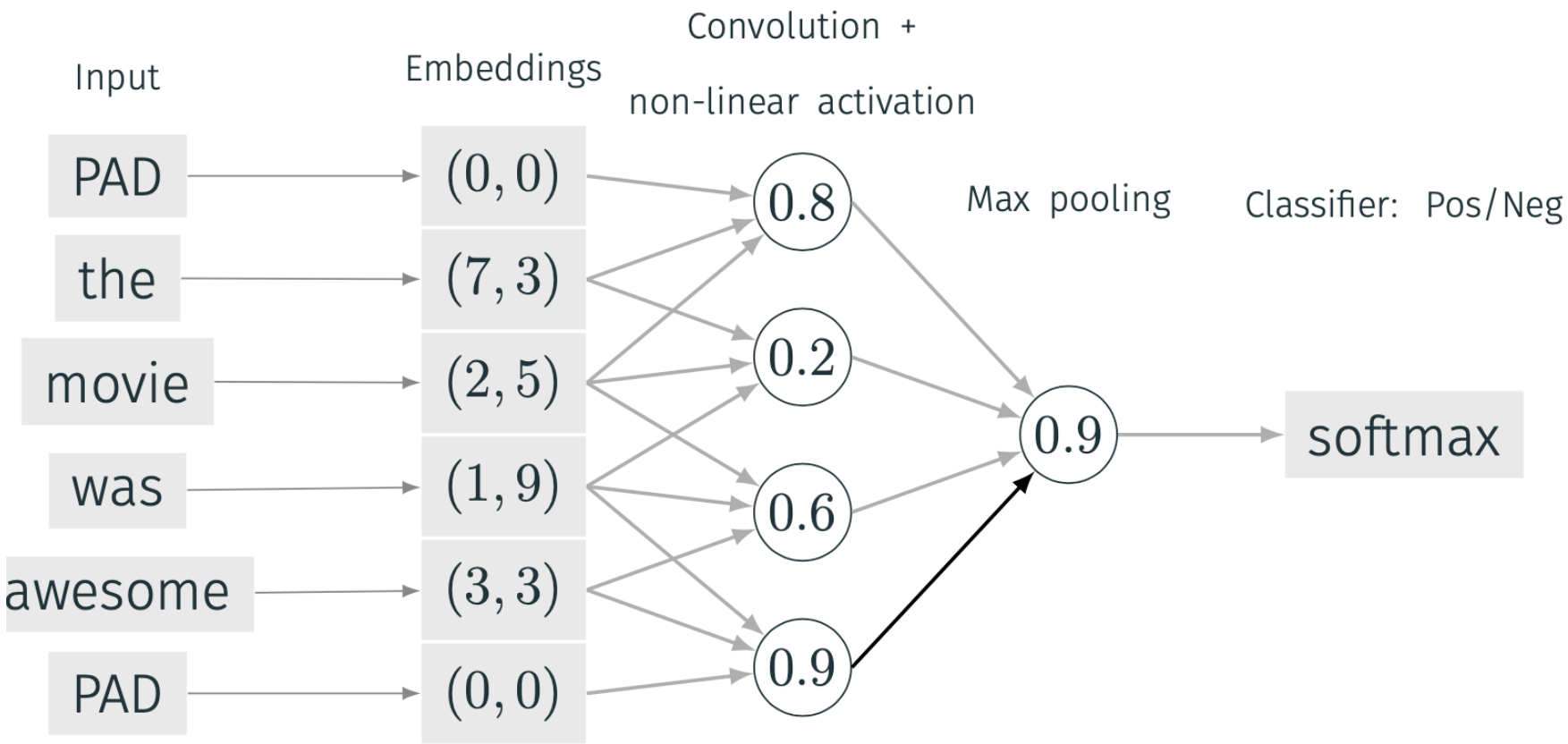
$$o = \max_i c_i$$

Max-over-time pooling is most common in NLP. You can also find min-pooling and mean-pooling in other areas. Could also use some other averaging

Note that there are no associated weights

(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

Pooling



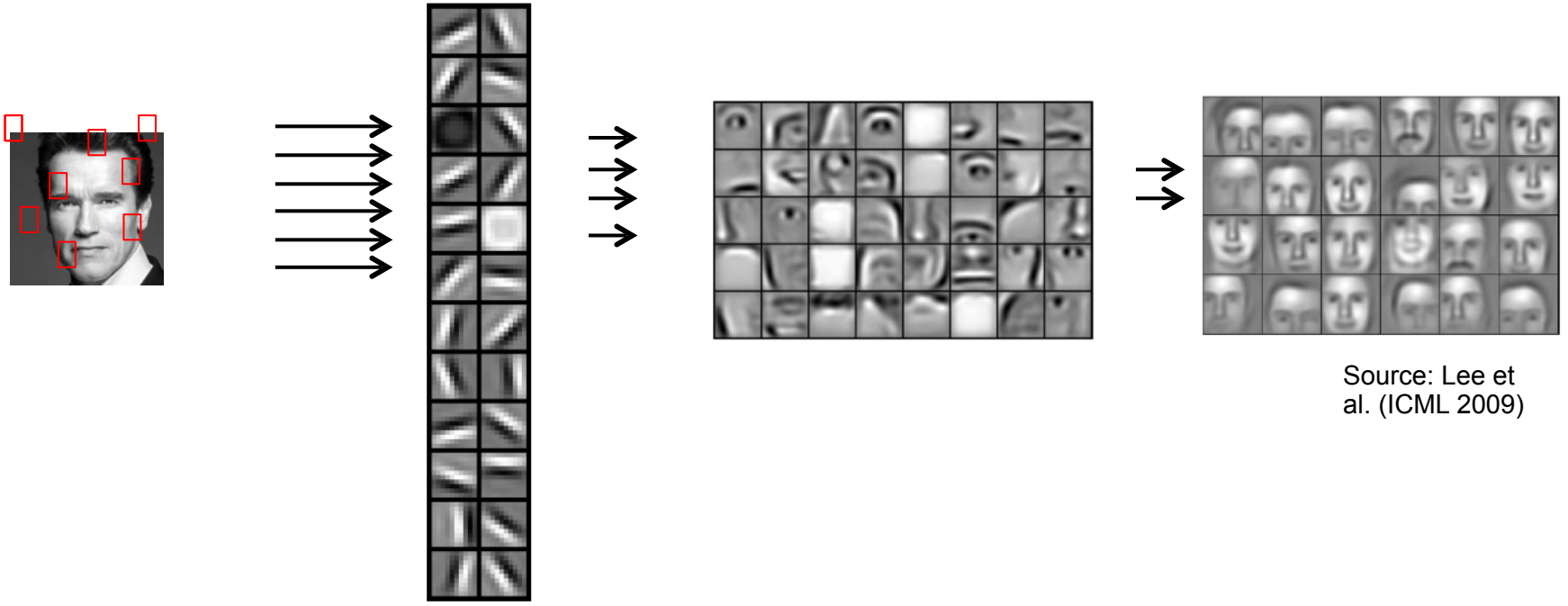
(Taken from the DL4NLP course 2021 (Ivan Habernal and Mohsen Mesgar)

Stacked convolutional layers

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Convolutional layers can be stacked to derive high-level representations from simple representations
- In images: edge detection → identifying facial features (eyes, nose, lips) → face recognition



Source: Lee et al. (ICML 2009)

Stacked convolutional layers

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Convolutional layers can be stacked to derive high-level representations from simple representations
- In images: edge detection → identifying facial features (eyes, nose, lips) → face recognition
- In NLP?
- Polarity of words → Identifying polarity of phrase (negated positive phrase) → sentiment classification

- This is only a very rough approximation, usually features in neural networks are hard to interpret (if at all)

Recurrent Neural Networks

Representation of Language

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

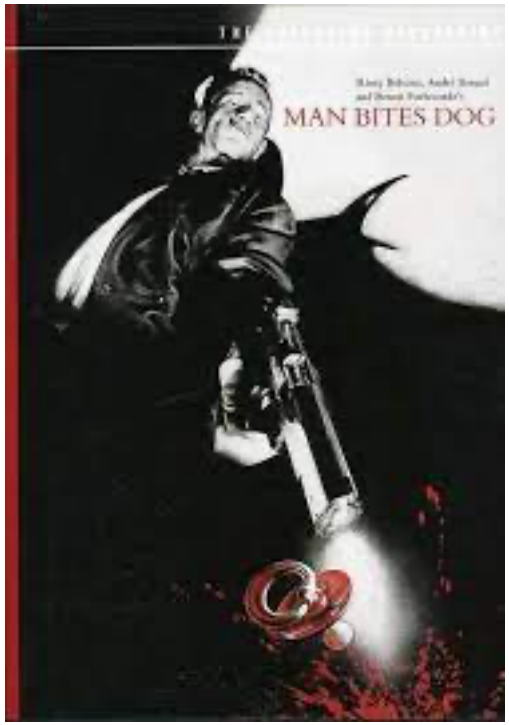
Neural networks need vectorized/numerical input

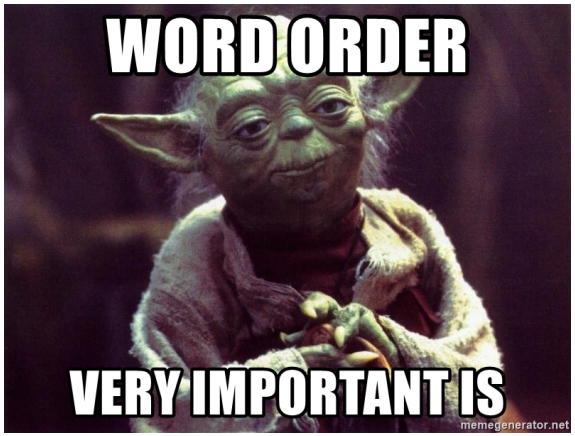
- `This is an example sentence .`
- `[ 35, 5, 3, 134, 96, 10 ]`

Fixed number of input neurons → fixed size

- `[ 0 0 0 0 15 16 2 19 178 32]`
- `[ 52 154 462 33 89 78 285 16 145 95] 108 7 4 2`
- `[ 0 0 624 35 534 6 227 7 129 113]`
- `[ 26 49 2 15 566 30 579 21 64 2]`
- `[ 19 14 5 2 6 226 251 7 61 113] 4 226 65 12 10 10`

Word order matters ...





„Man bites dog“
„Dog bites man“

Long-range Dependencies

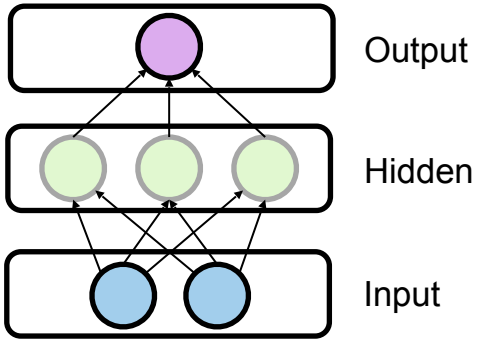
e.g. Sentiment classification

- `I like the new iPhone but it doesn‘t have any new features and it is much too expensive.`

Long-range Dependencies

e.g. Sentiment classification

- **[I like the new iPhone but it doesn't have any new features and it is]** ~~much too expensive.~~



How to model long input sequences?

Long-range Dependencies

UNIVERSITÄT DUISBURG ESSEN

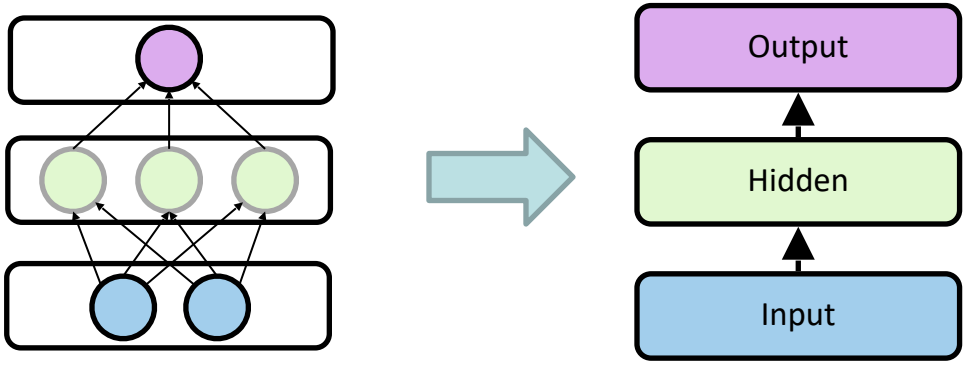
Offen im Denken

e.g. Sentiment classification

- **[I like the new iPhone but it doesn't have any new features and it is]** ~~much too expensive.~~

How to model long input sequences?

Collapsed View

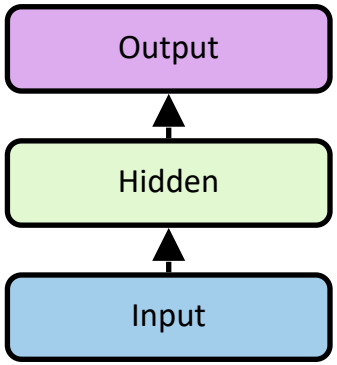


From Feed-forward to Recurrent Nets

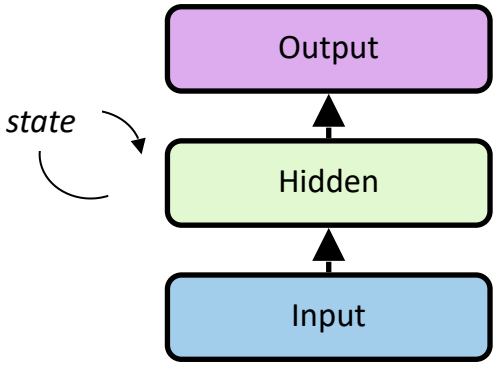
UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

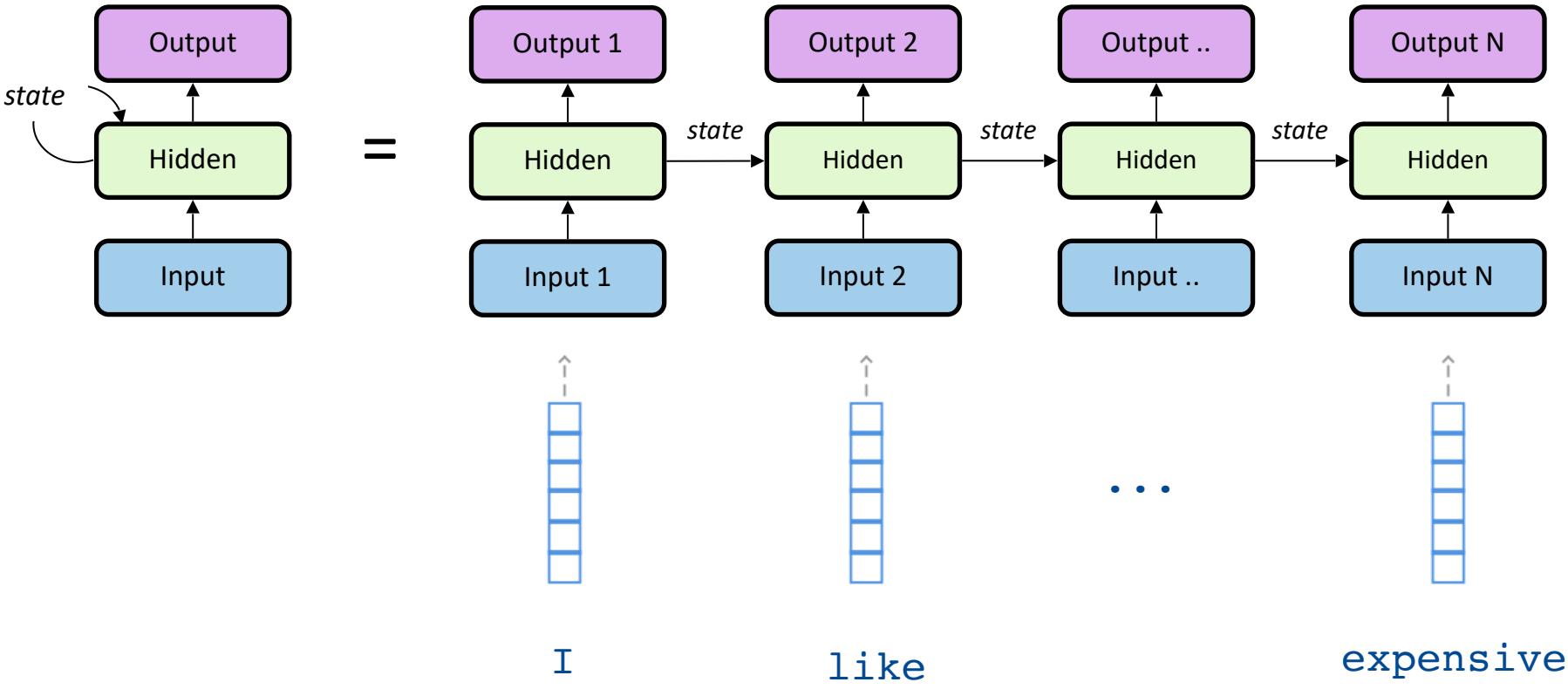
Feed-forward



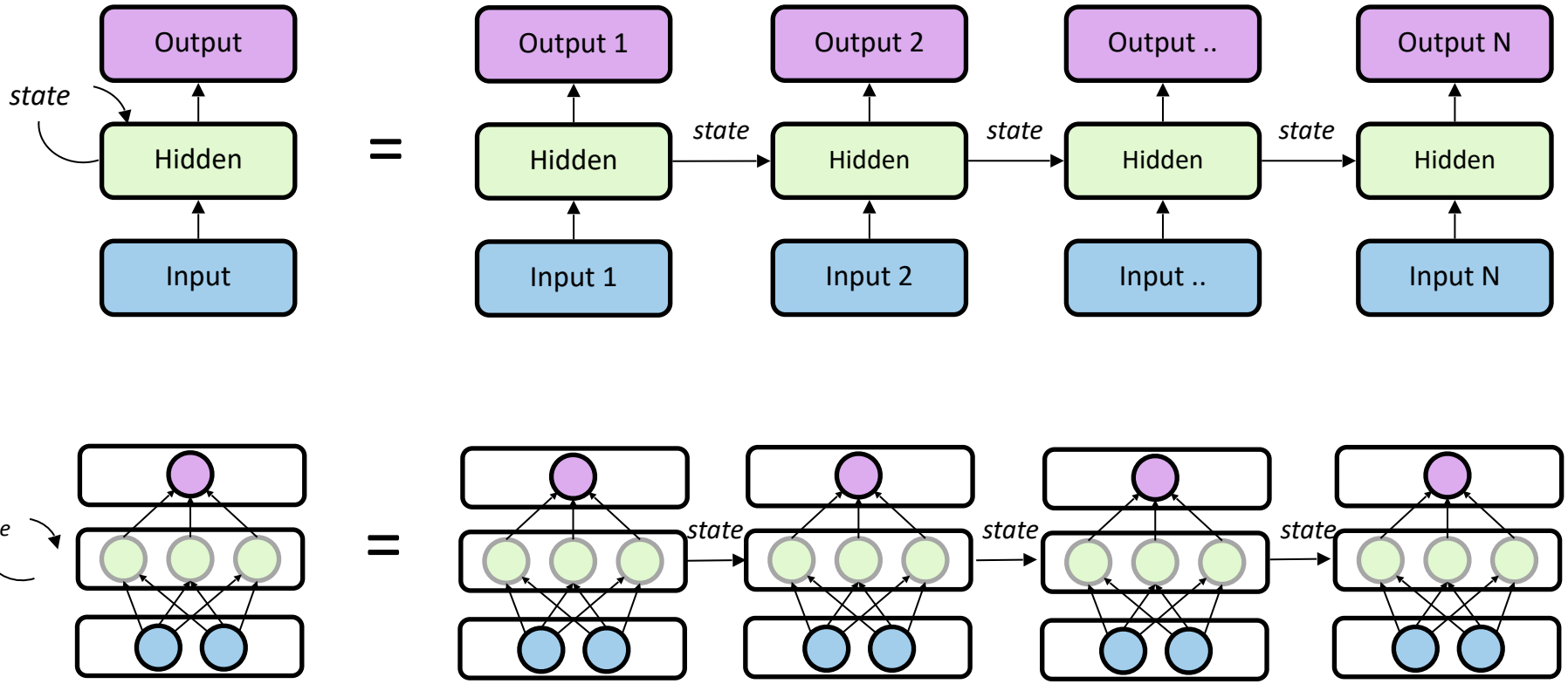
Recurrent



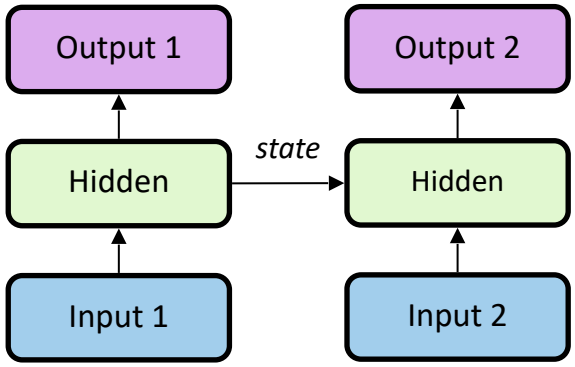
Unrolling in time

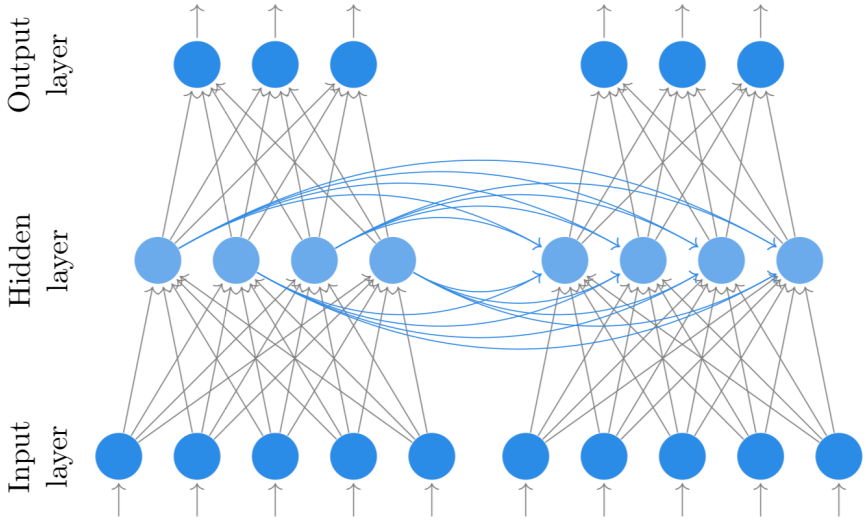


Unrolling in time



Zoomed-in View



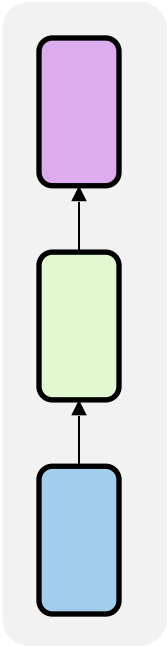


Variants

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

one to one



Vanilla Feed-Forward

http://karpathy.github.io/2015/05/21/rnn-effectiveness/ & http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture10.pdf

Variants

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

one to one

one to many

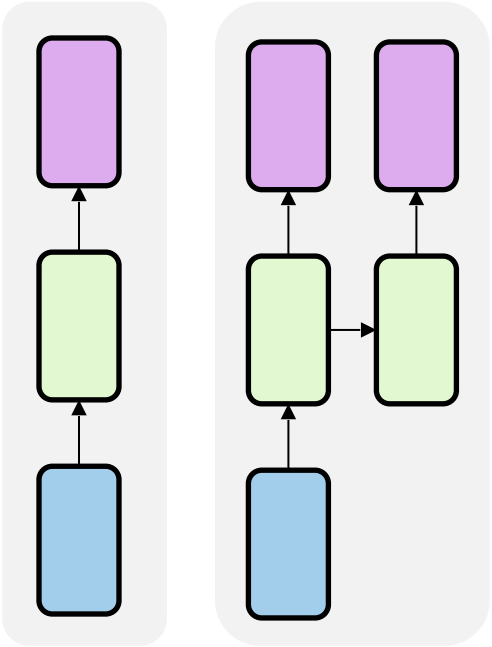
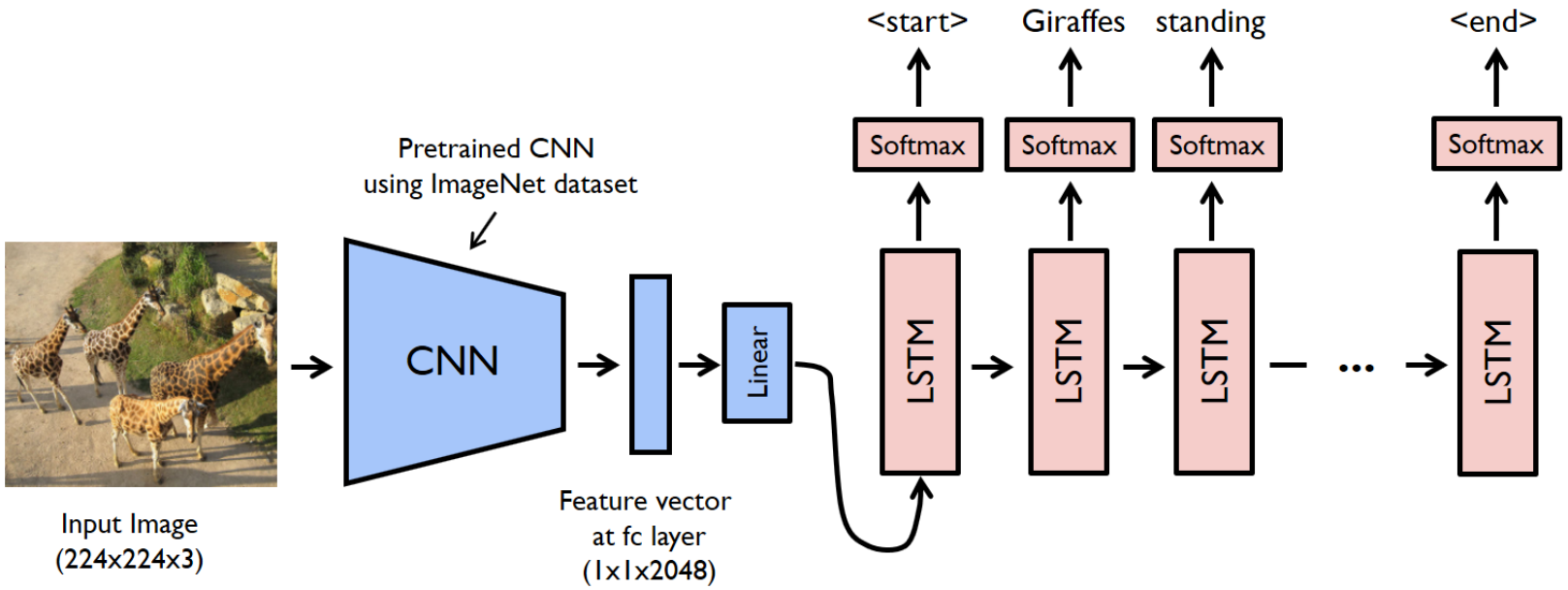


Image Captioning

- *one* image to *many* words

http://karpathy.github.io/2015/05/21/rnn-effectiveness/ & http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture10.pdf

Image Captioning



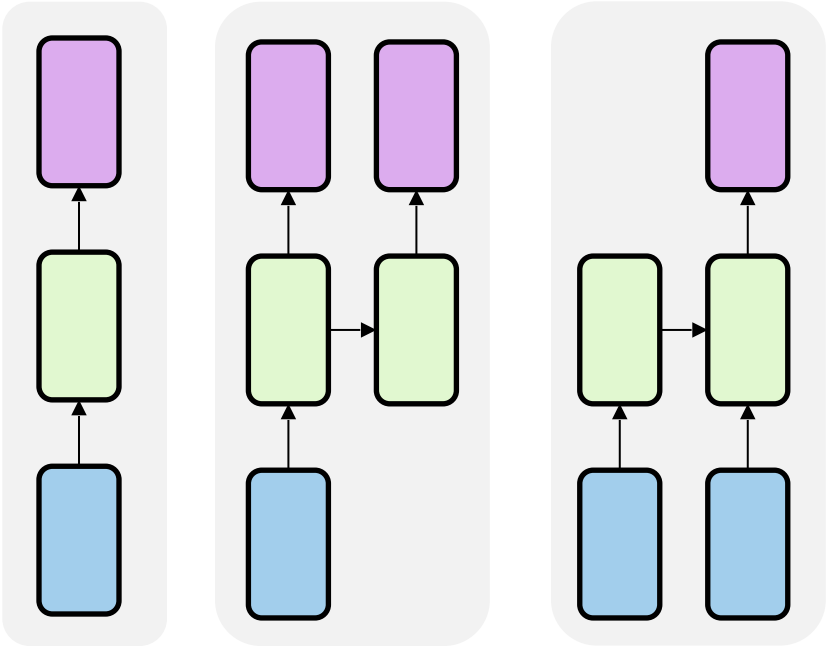
https://www.analyticsvidhya.com/blog/2018/04/solving-an-image-captioning-task-using-deep-learning/

Variants

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

one to one | one to many | many to one

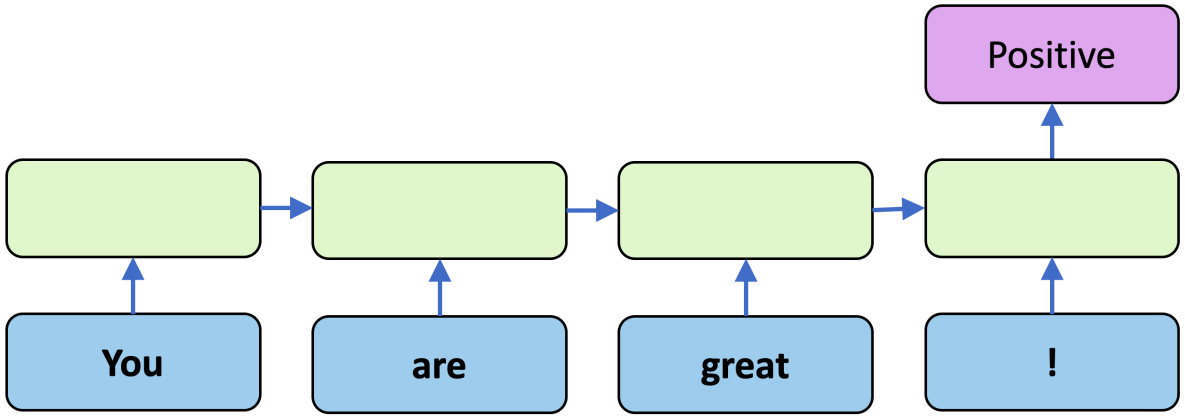


Sentiment Classification

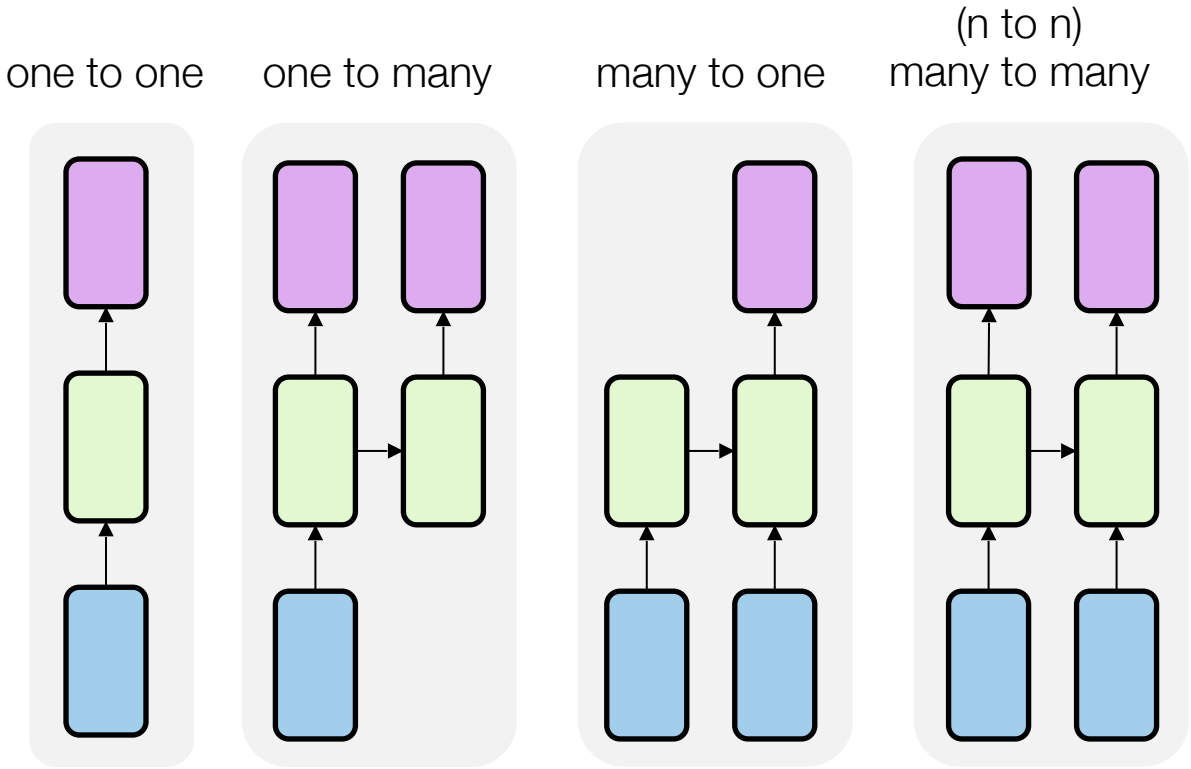
- *many* words to *one* category

http://karpathy.github.io/2015/05/21/rnn-effectiveness/ & http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture10.pdf

Sentiment Classification



Variants



POS Tagging

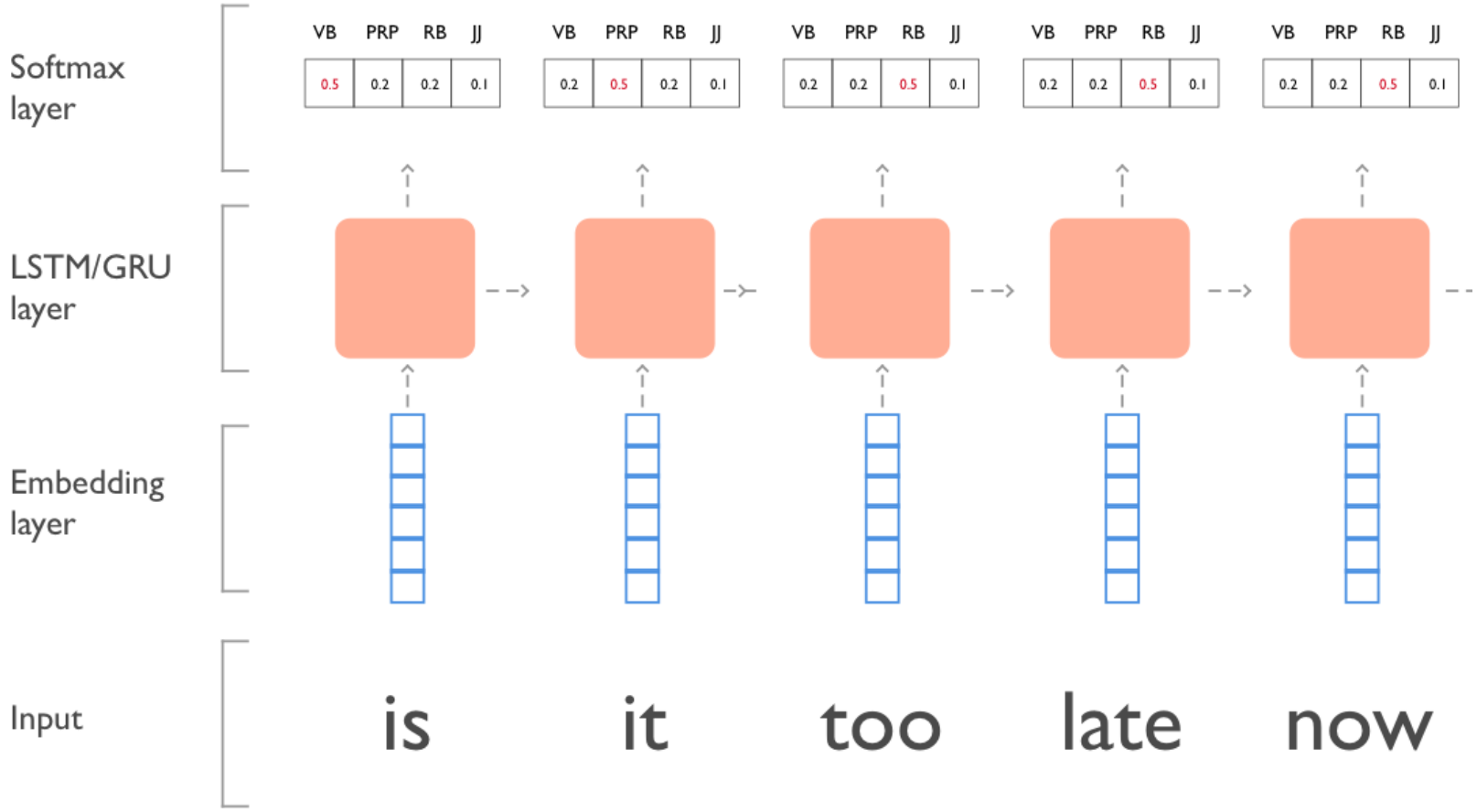
- *many* words to *many tags*

http://karpathy.github.io/2015/05/21/rnn-effectiveness/ & http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture10.pdf

POS Tagging

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

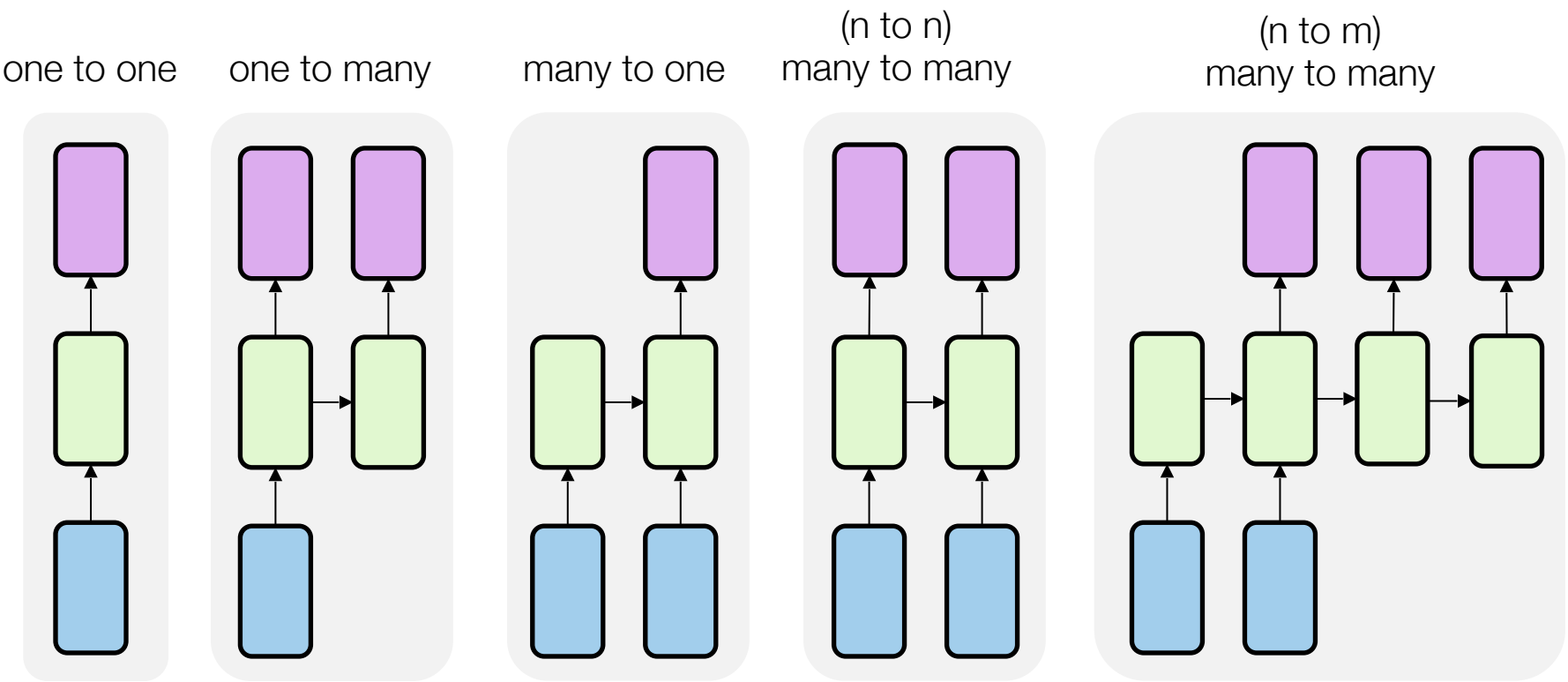


https://towardsdatascience.com/taming-lstms-variable-sized-mini-batches-and-why-pytorch-is-good-for-your-health-61d35642972e

Variants

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

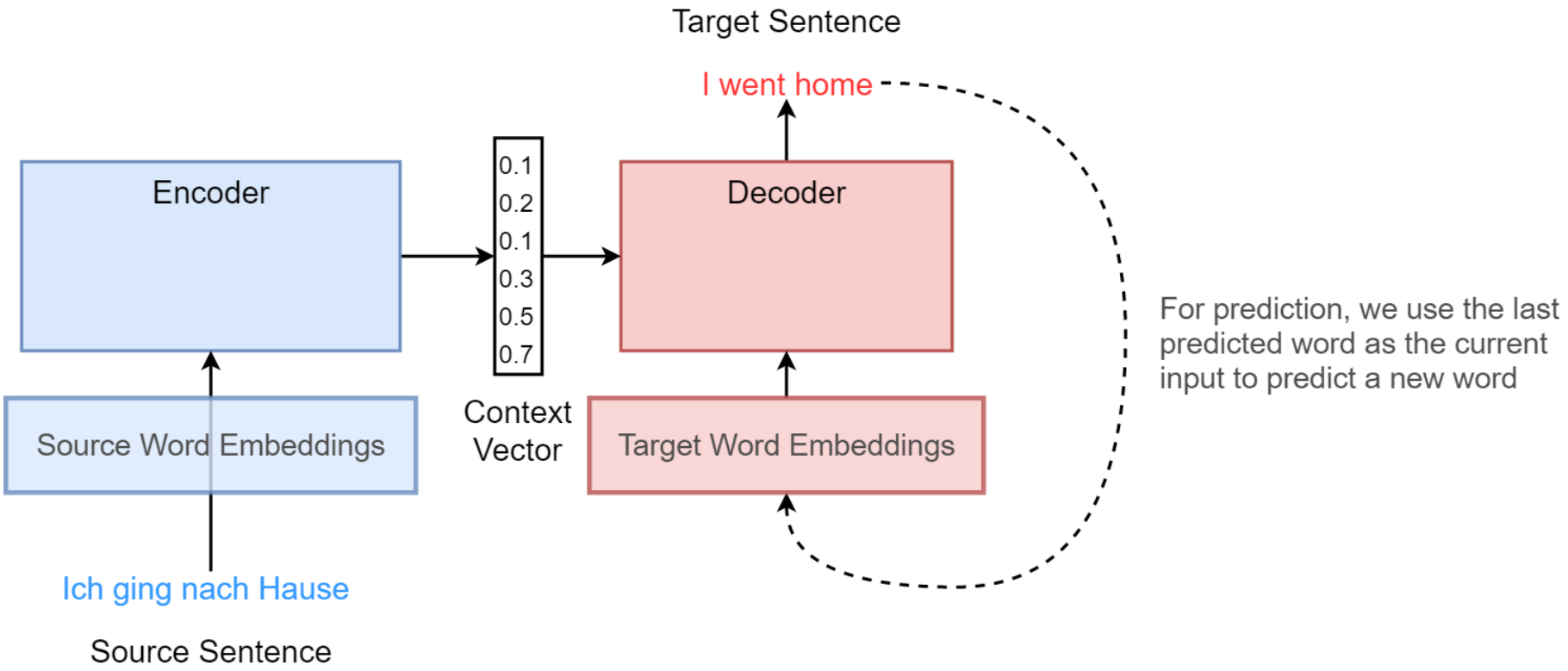


Machine Translation

- *many* words to *many words*

http://karpathy.github.io/2015/05/21/rnn-effectiveness/ & http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture10.pdf

Machine Translation



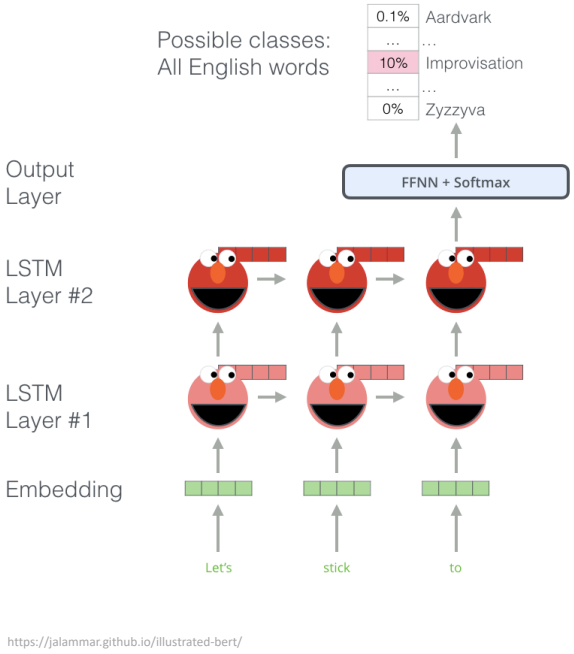
ELMo

ELMo

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Stands for Embeddings from Language Model (Peters, et al, 2018)
- Idea: predict the next word in a sequence of word



https://jalammar.github.io/illustrated-bert/

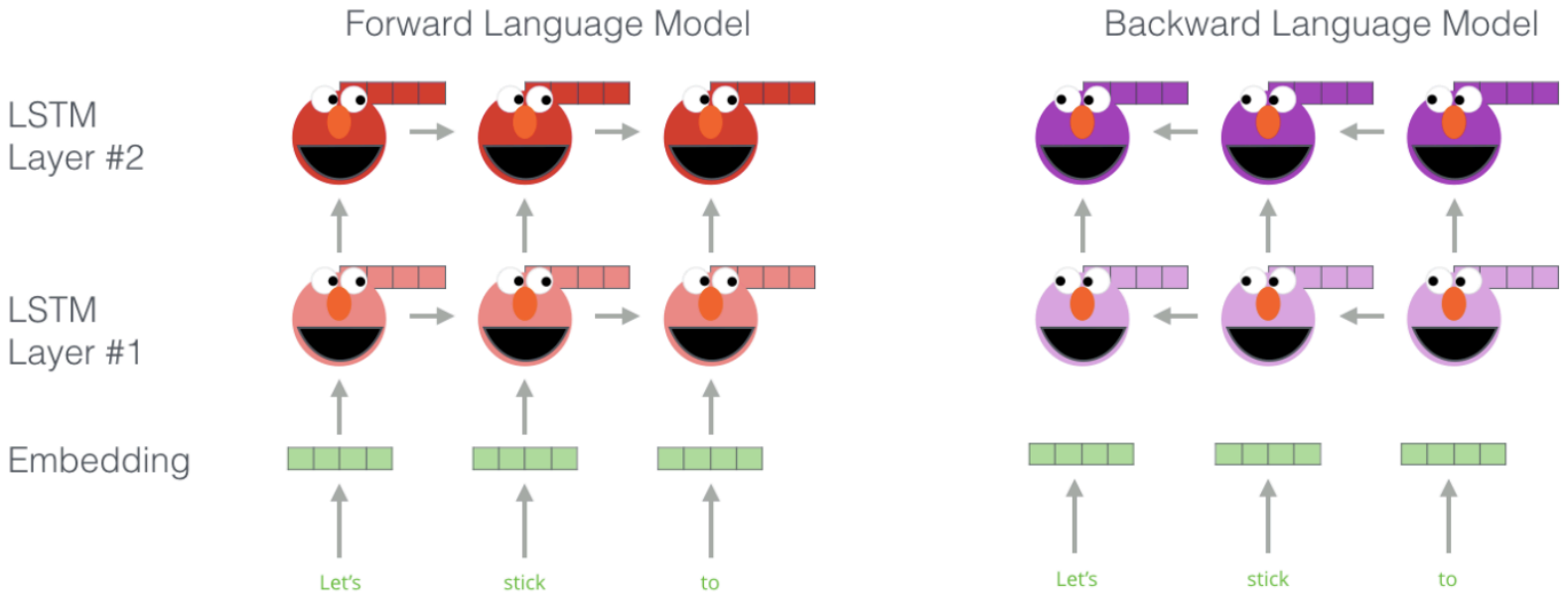
- Data: 1B Word Benchmark

ELMo

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Idea: predict the target word using its left and right context



https://jalammar.github.io/illustrated-bert/

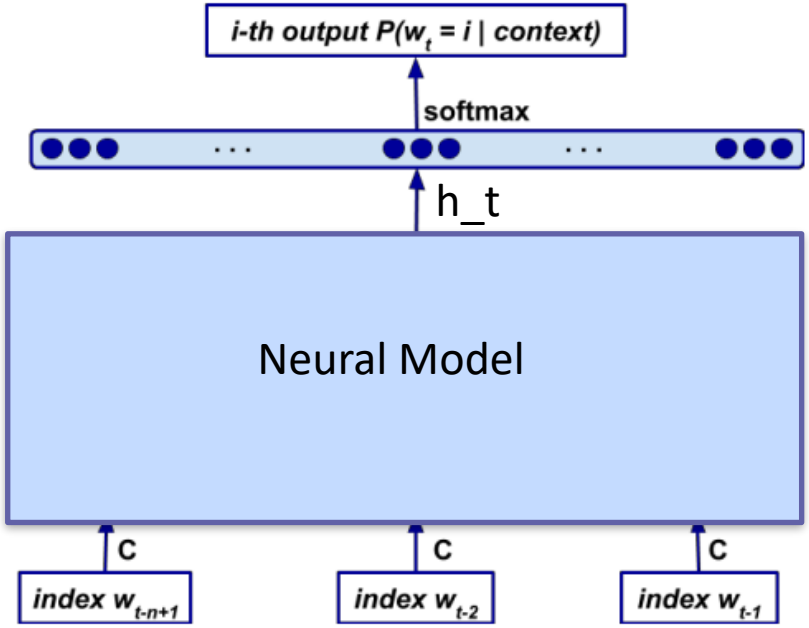
-

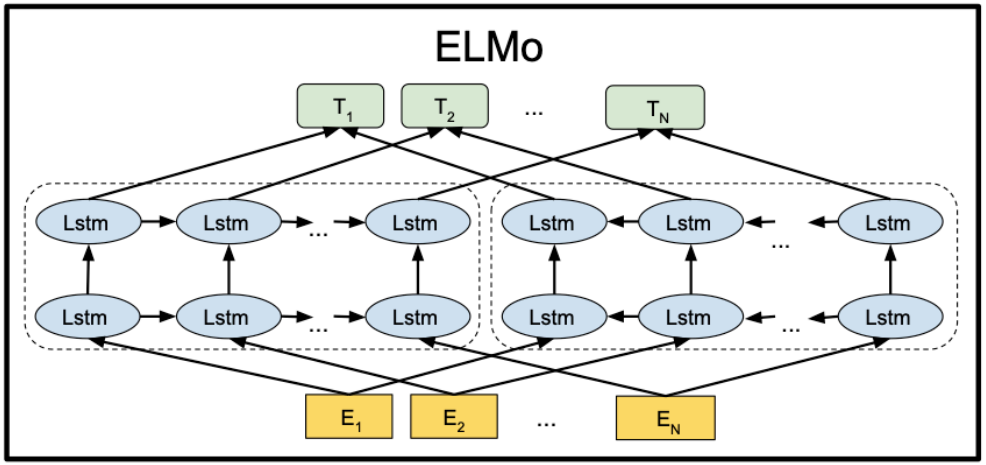
ELMo

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

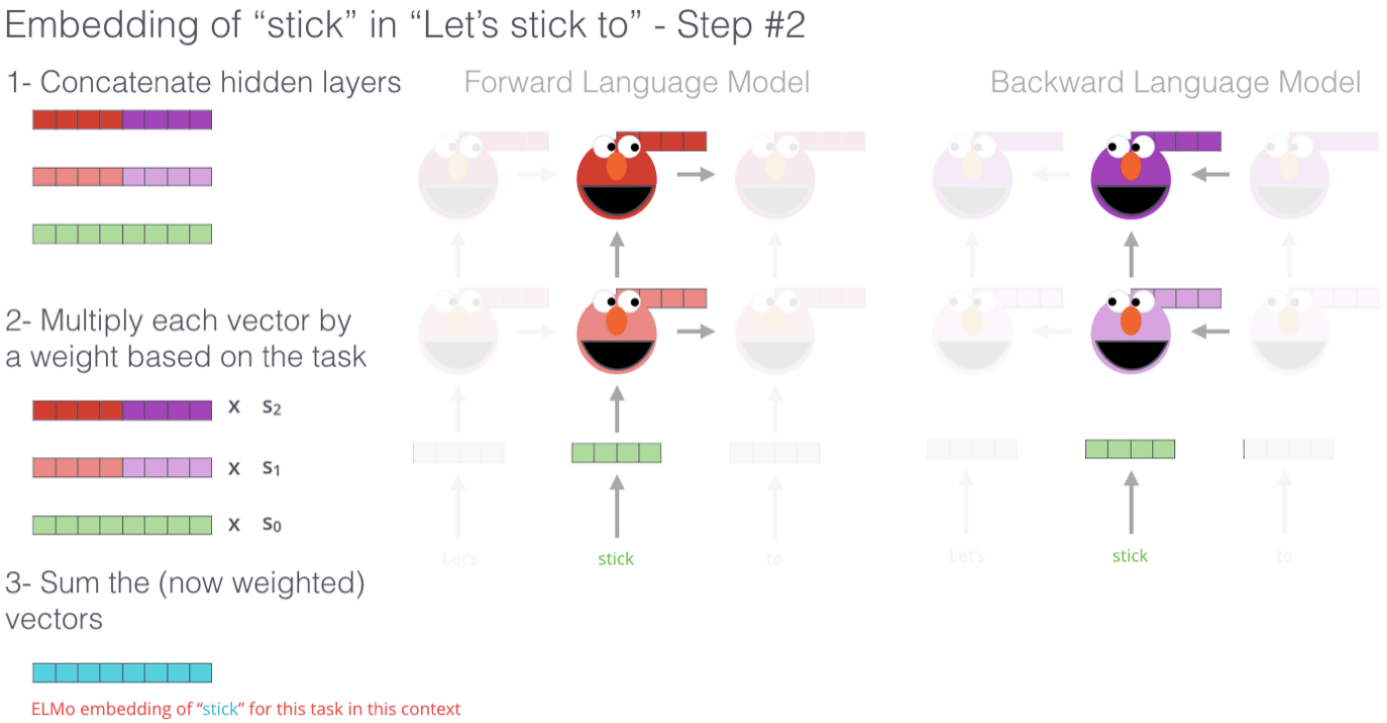
- Idea: predict the target word using its left and right context





https://arxiv.org/pdf/1810.04805.pdf

- Each word can be represented by a vector



https://jalammar.github.io/illustrated-bert/

ELMo

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Pre-trained ELMo Models

Model	Link (Weights/Options File)	# Parameters (Millions)	LSTM Hidden Size/Output size
Small	weights | options	13.6	1024/128
Medium	weights | options	28.0	2048/256
Original	weights | options	93.6	4096/512
Original (5.5B)	weights | options	93.6	4096/512

https://allenai.org/allennlp/software/elmo

- All models except for the 5.5B model were trained on the **1 Billion Word Benchmark,** approximately **800M tokens of news crawl data**.
- The ELMo 5.5B model was trained on a dataset of **5.5B tokens consisting of Wikipedia (1.9B) and all of the monolingual news crawl data** from WMT 2008-2012 (3.6B).
- The 5.5B model has slightly higher performance than the original ELMo model, so they recommend it as a default model.

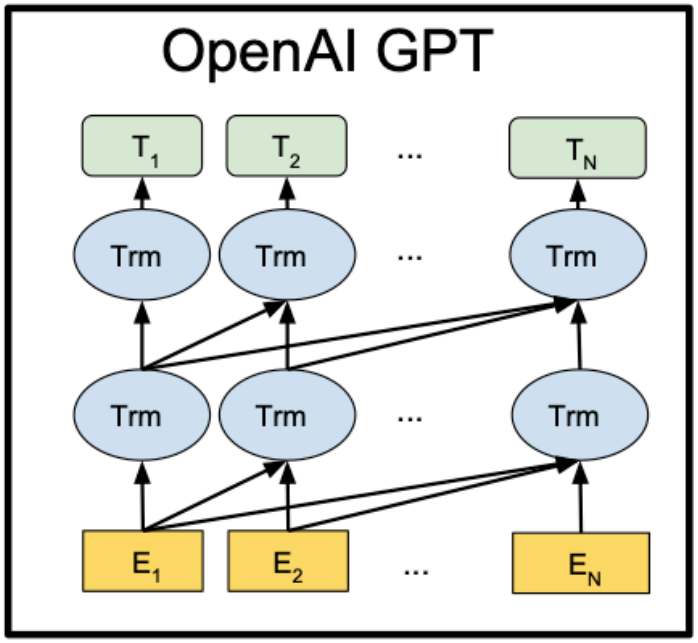
Today

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

GPT-3

GPT-3

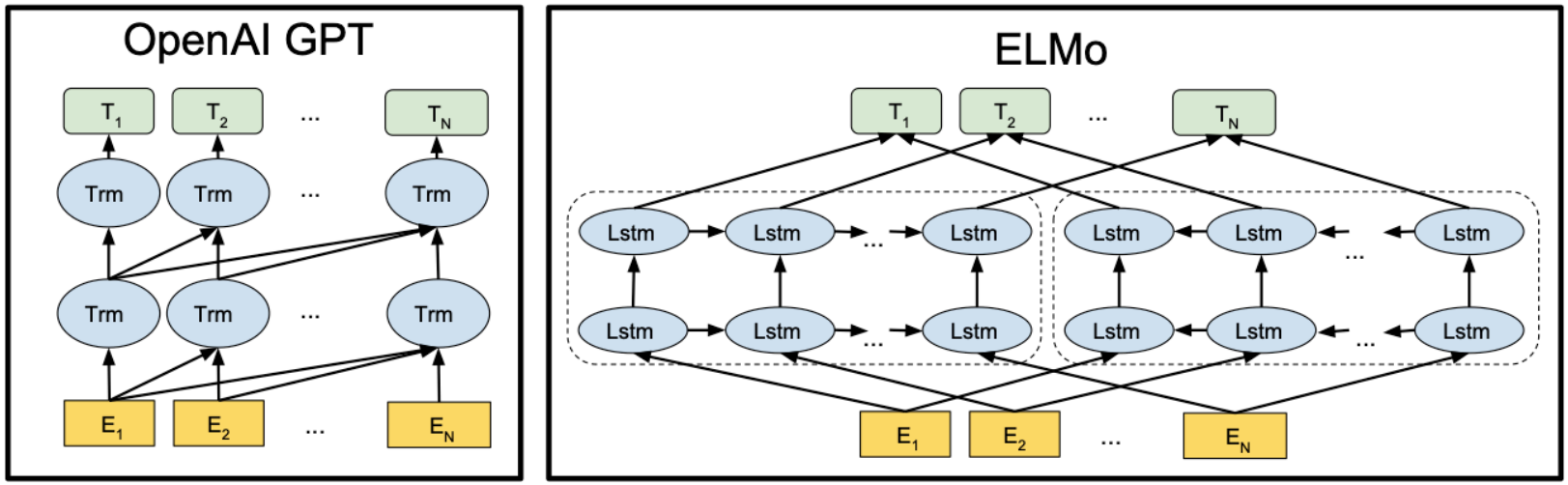


GPT-3 (a.k.a OpenAI GPT)

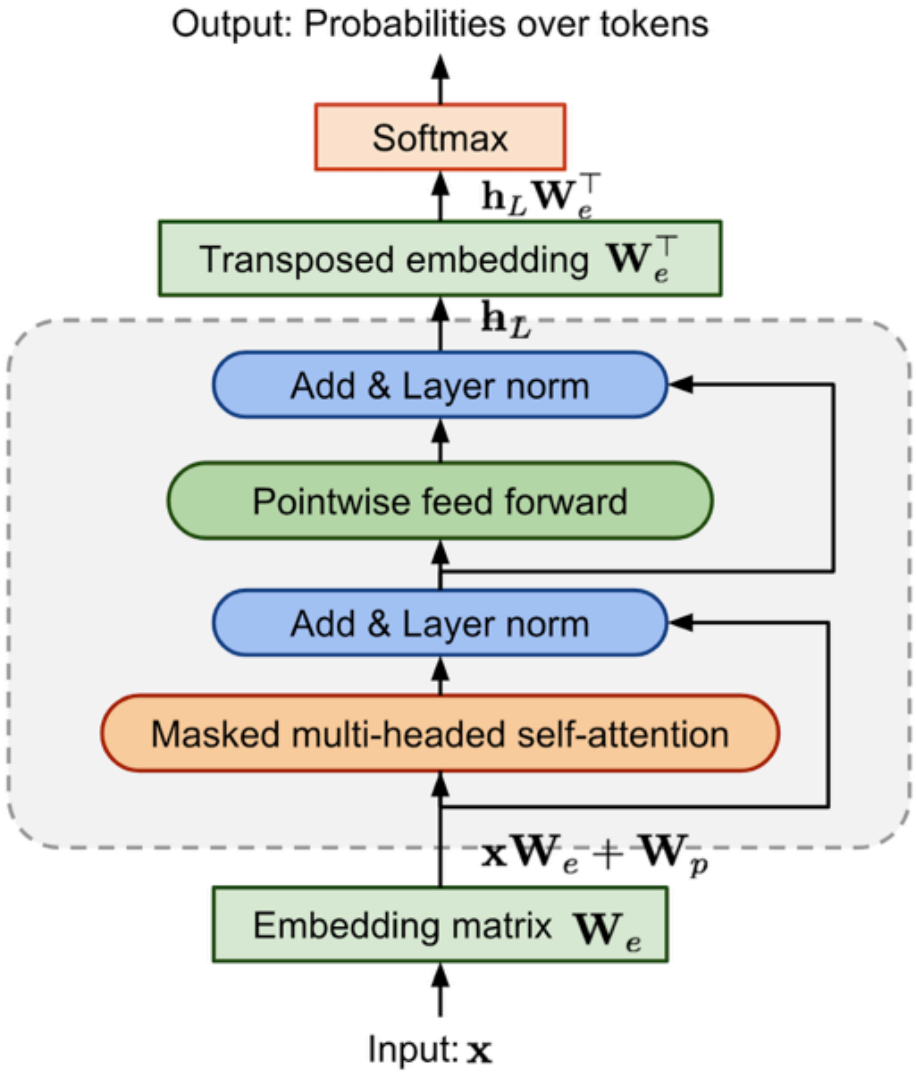
UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Comparing with ELMo: The model architectures are different. ELMo uses a shallow concatenation of independently trained left-to-right and right-to-left multi-layer LSTMs, while GPT is a multi-layer transformer.



GPT-3 (a.k.a OpenAI GPT)



Transformer Block
Repeat x L=12

$$\mathbf{h}_\ell = \text{transformer\_block}(\mathbf{h}_{\ell-1})$$
$$\ell = 1, \ldots, L$$

GPT-3

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- **Size:** GPT-3 model with **175 billion parameters**
  - 100 times more parameters than GPT-2
- **Data:** GPT-3 was trained on a mix of five different corpora:
  - Common Crawl, WebText2, Books1, Books2 and Wikipedia
  - Each having certain weight: High quality datasets were sampled more often, and model was trained more often on those datasets.

- **In-Context Learning:** With in-context learning, the text given to the model is a written description (optional) plus some examples. The last example is left unfinished for the model to complete.
- In-context learning is flexible. We can use this scheme to describe many possible tasks, from translating between languages to improving grammar to coming up with joke punch-lines

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1  Translate English to French:          ← task description
2  sea otter => loutre de mer            ← examples
3  peppermint => menthe poivrée          ←
4  plush girafe => girafe peluche        ←
5  cheese =>                             ← prompt
```

GPT-3

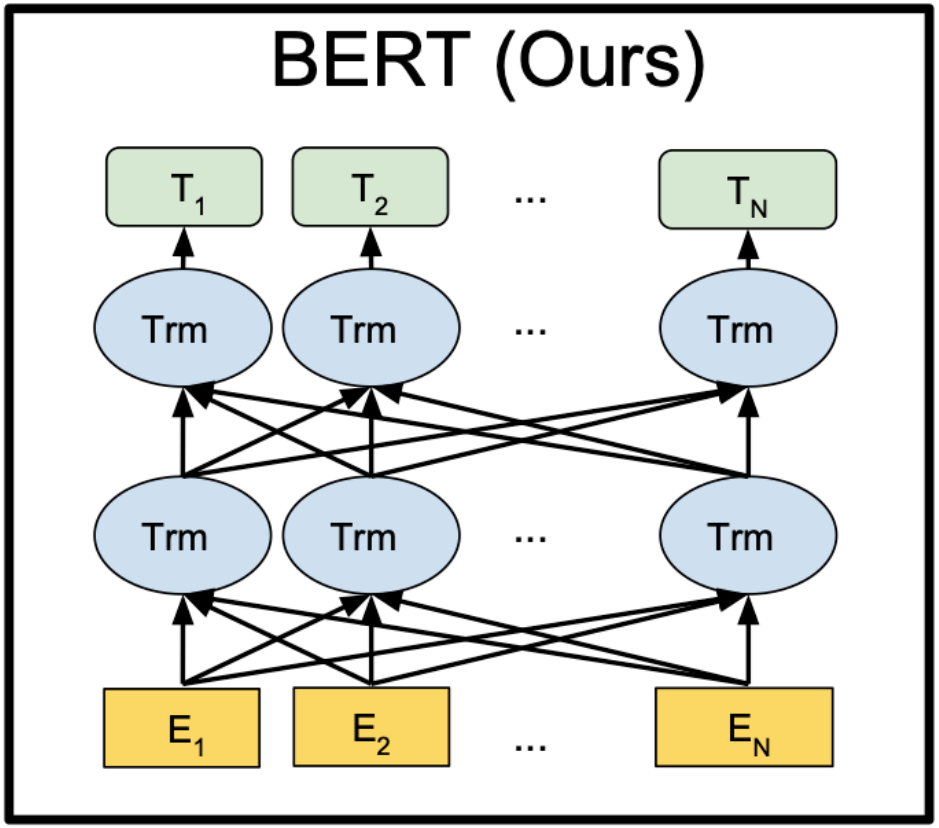
UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- GPT-3 is good in few-shot and zero-shot transfer
- **GPT-3 is not publicly available for free**

BERT

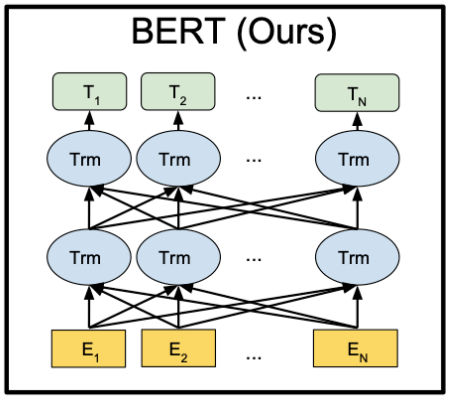
- Stands for Bidirectional Encoder Representations from Transformers

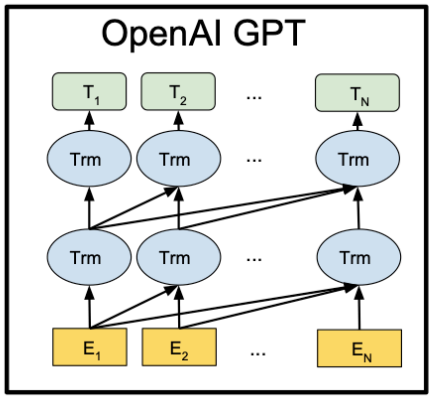


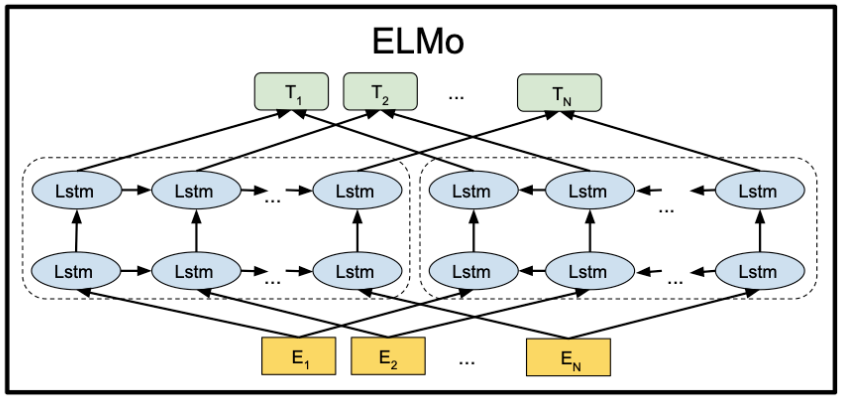
BERT

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken





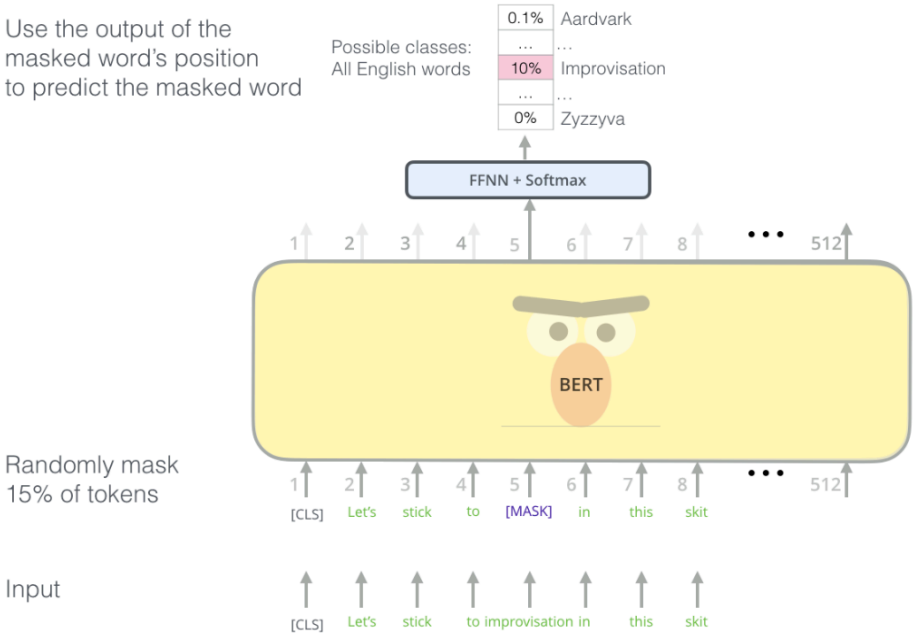


BERT

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- BERT relies on Transformers as well
- BERT is bidirectional
- BERT is a **Masked Language Model**
-



BERT's clever language modeling task masks 15% of words in the input and asks the model to predict the missing word.

https://jalammar.github.io/illustrated-bert/

BERT

UNIVERSITÄT DUISBURG ESSEN

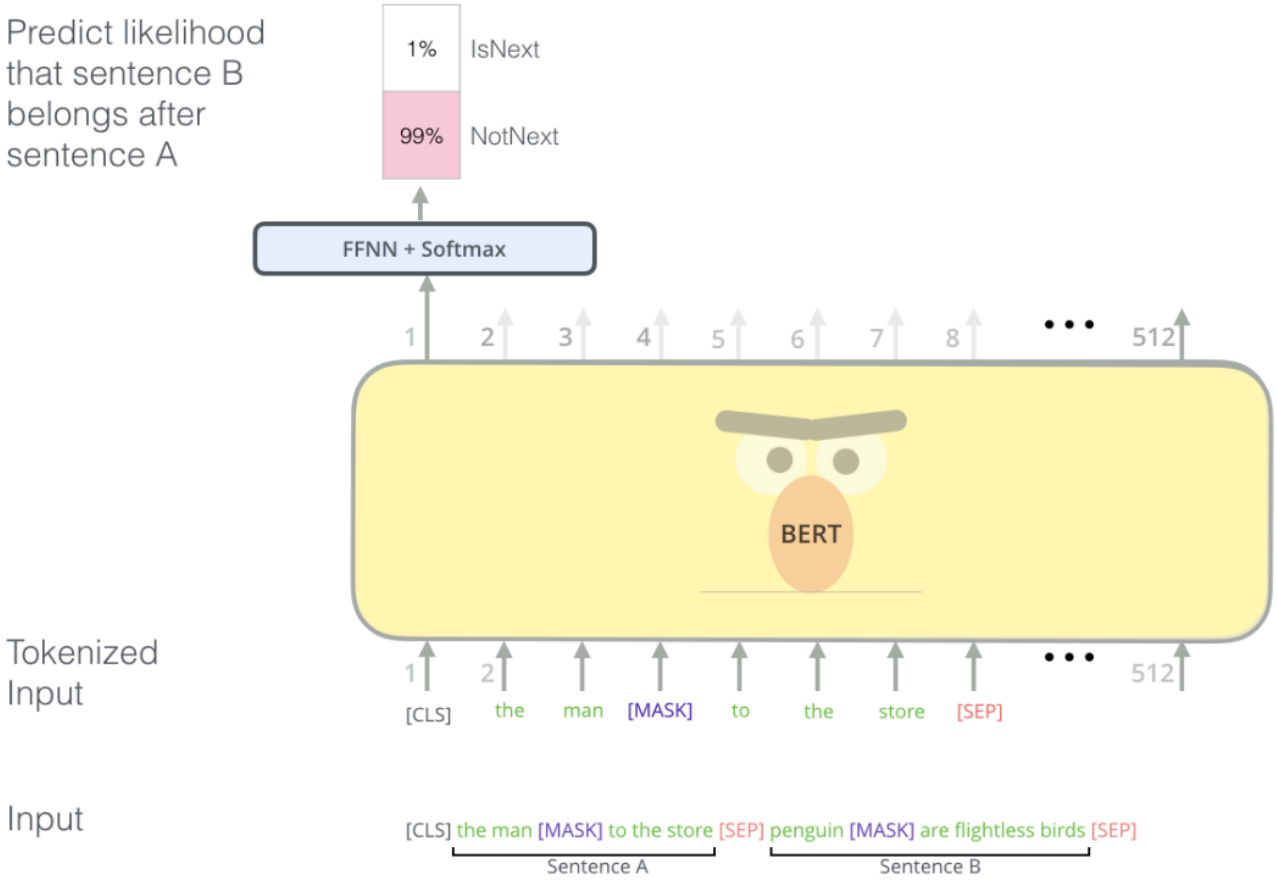
Offen im Denken

- Corrupts the inputs by using random masking,
- during pretraining, a given percentage of tokens (usually 15%) is masked by:
  - a special mask token with probability 0.8
  - a random token different from the one masked with probability 0.1
  - the same token with probability 0.1
- The model is trained for two objectives:
  - It must predict the original sentence
  - Inputs are two sentences A and B (with a separation token in between). With probability 50%, the sentences are consecutive in the corpus, in the remaining 50% they are not related. The model must predict if the sentences are consecutive or not.

BERT

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken



The second task BERT is pre-trained on is a two-sentence classification task. The tokenization is oversimplified in this graphic as BERT actually uses WordPieces as tokens rather than words --- so some words are broken down into smaller chunks.

BERT

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- BERT has two versions:
  - $\text{BERT}_{\text{BASE}}$: 110 million parameters
  - $\text{BERT}_{\text{LARGE}}$: 340 million parameters

- Data: 16GB unlabeled text data extracted from
  - BooksCorpus with 800M words
  - English Wikipedia with 2,500M words.
- BERT was trained on over 100 languages, it wasn't optimized for multi-lingual models — most of the vocabulary isn't shared between languages and therefore the shared knowledge is limited.

Generative Adversarial Networks (GANs)

Generative networks

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- The networks we have seen so far, read an input matrix of data and modify it by using weights and activation functions.
- In the final layer, we run a softmax over the modified matrix to receive classification probabilities.

- The input matrix represents the input data.
- Could we modify the matrix such that it represents new data?
  → generative model

- For many tasks, we encounter the problem that we do not have enough training data.
  Couldn't we just generate it?

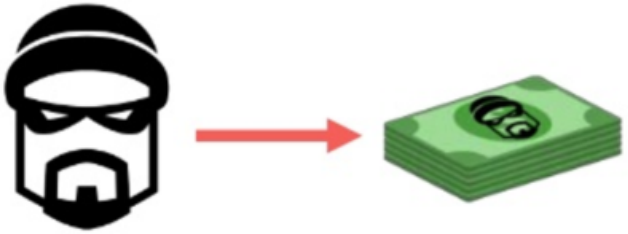
Generator-discriminator competition

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

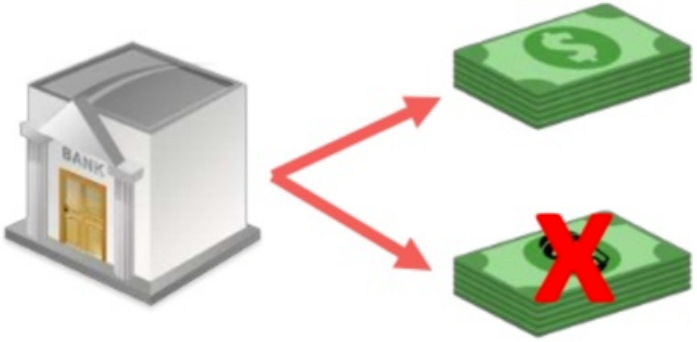
- Generator: try to generate authentic looking instances.
- Discriminator: distinguish between the real and generated instances.

Generator



Goal: produce counterfeit money that is as similar as real money.

Discriminator (Classifier)



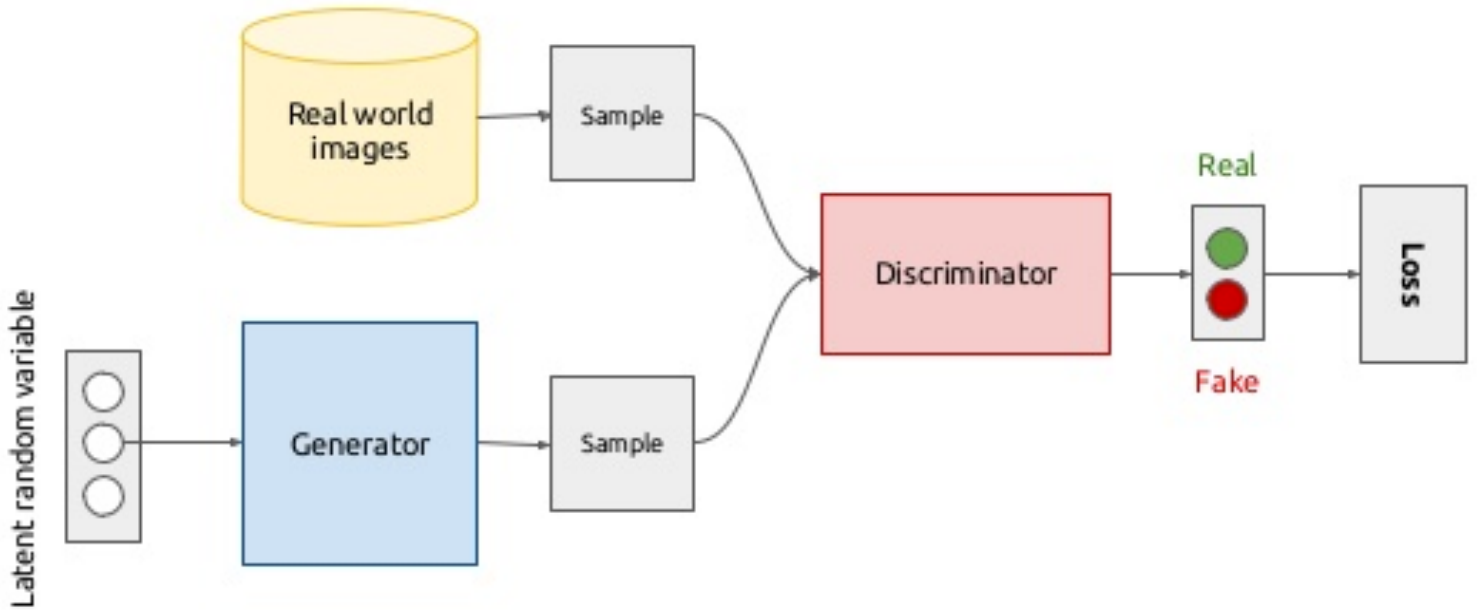
Goal: distinguish between real and counterfeit money.

Generative Adversarial Networks (GAN)

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- During training, both the discriminator and the generator improve.



5

Objective

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Generator:

Input

100 numbers

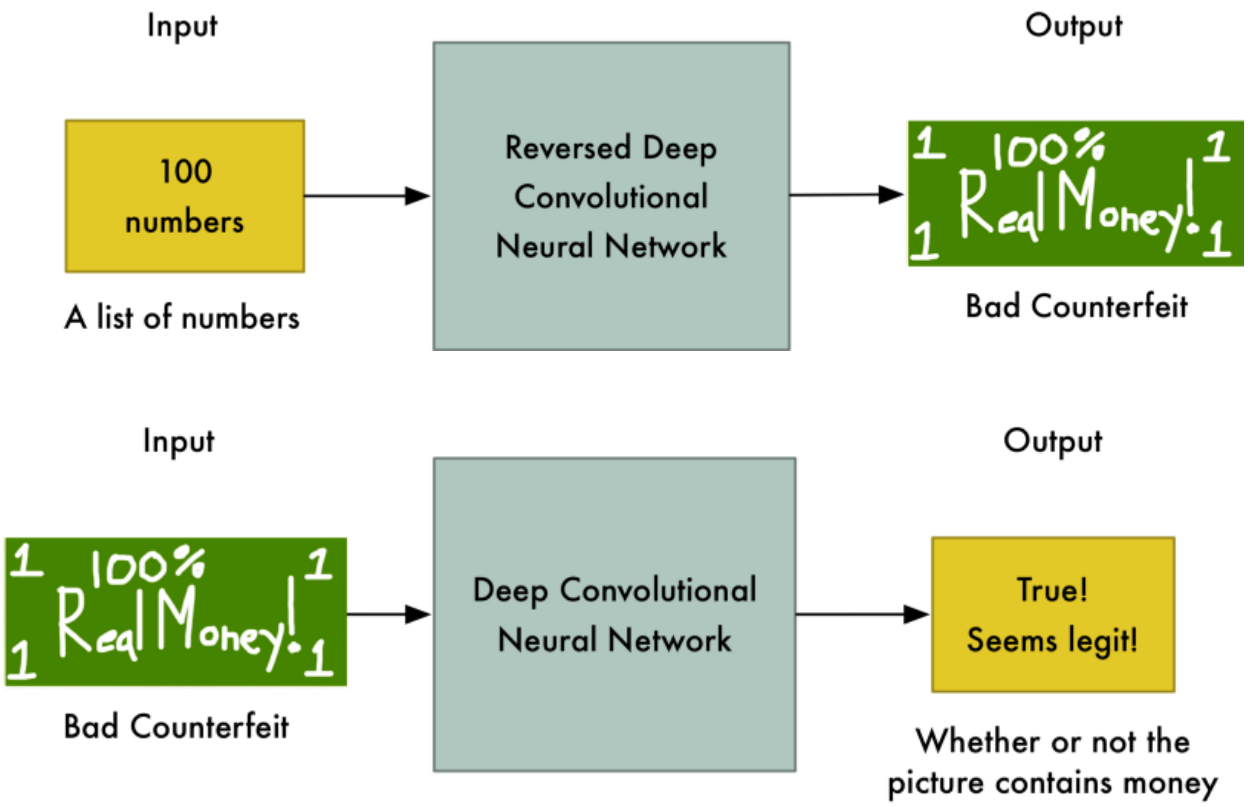
A list of numbers

Reversed Deep Convolutional Neural Network

Discriminator:

Deep Convolutional Neural Network

GAN: Start

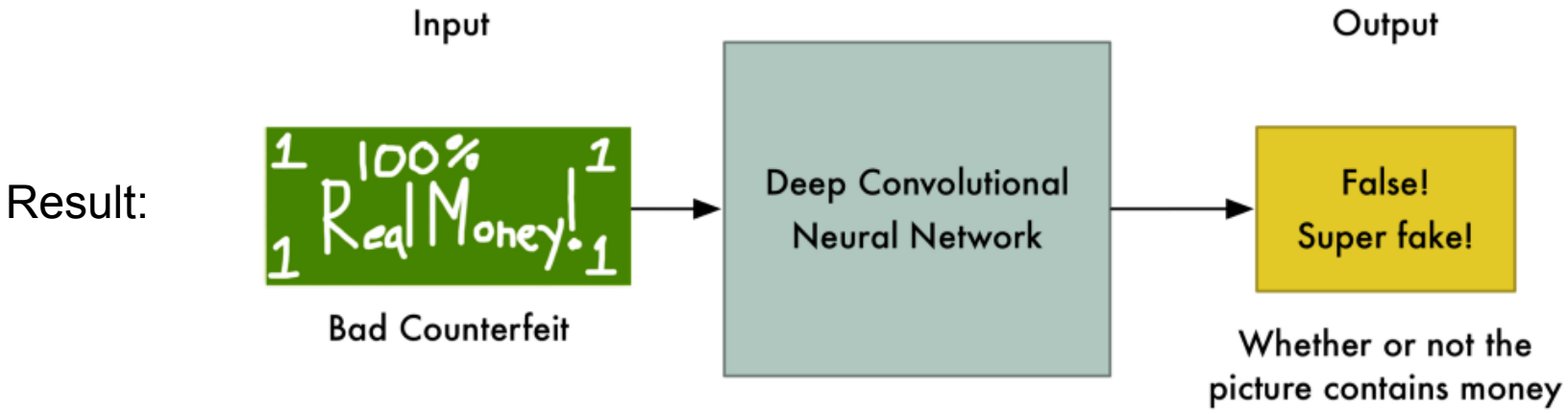

Input
100 numbers
A list of numbers
Reversed Deep Convolutional Neural Network
Output
100% Real Money!
Bad Counterfeit
Input
100% Real Money!
Bad Counterfeit
Deep Convolutional Neural Network
Output
True! Seems legit!
Whether or not the picture contains money

GAN: Improve classification

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- The discriminator learns from sample data to distinguish between real and fake money.
  - e.g. real money has a picture of a person on it



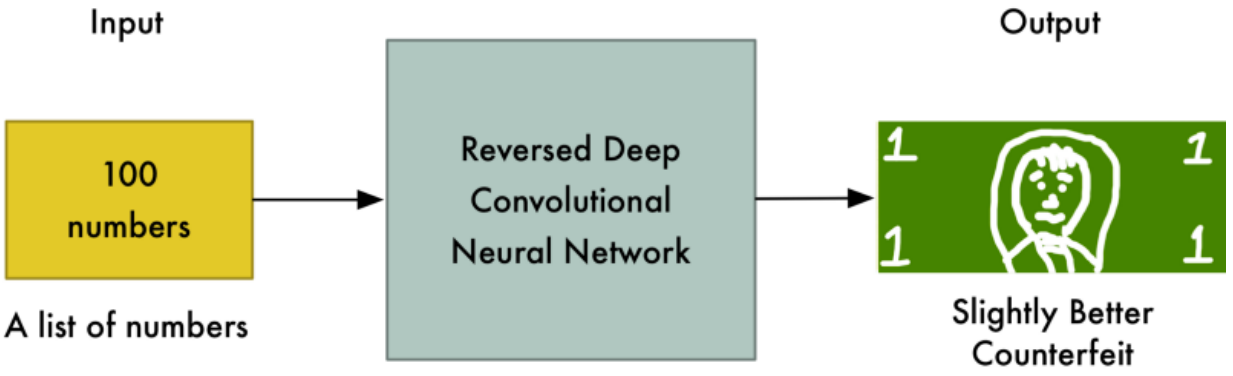
GAN: Improve generation

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- The generator learns that the generated samples are getting rejected.
- It learns which features the discriminator is looking for (pictures of people) and improves the generation.

Result:



- Now: iterate!
- An improved generator leads to an improved discriminator leads to an improved generator...

Summary

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Different architectures serve different purposes:

- Image processing, document classification
  - Convolutional neural networks

- Sequence processing
  - Recurrent neural network

- Text processing
  - Transformers

- Generative adversarial neural networks
  - Artificial data generation

References

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

- Ian Goodfellow, Yoshua Bengio, Aaron Courville:
- Deep Learning, *Chapter 9*, 9.1 to 9.3.
- Deep Learning, *Chapter 10*, 10.2 and 10.4
  http://www.deeplearningbook.org/
- http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/