Neural Networks: A Theory Lab

Attention!

Hadi Daneshmand

University of Virginia

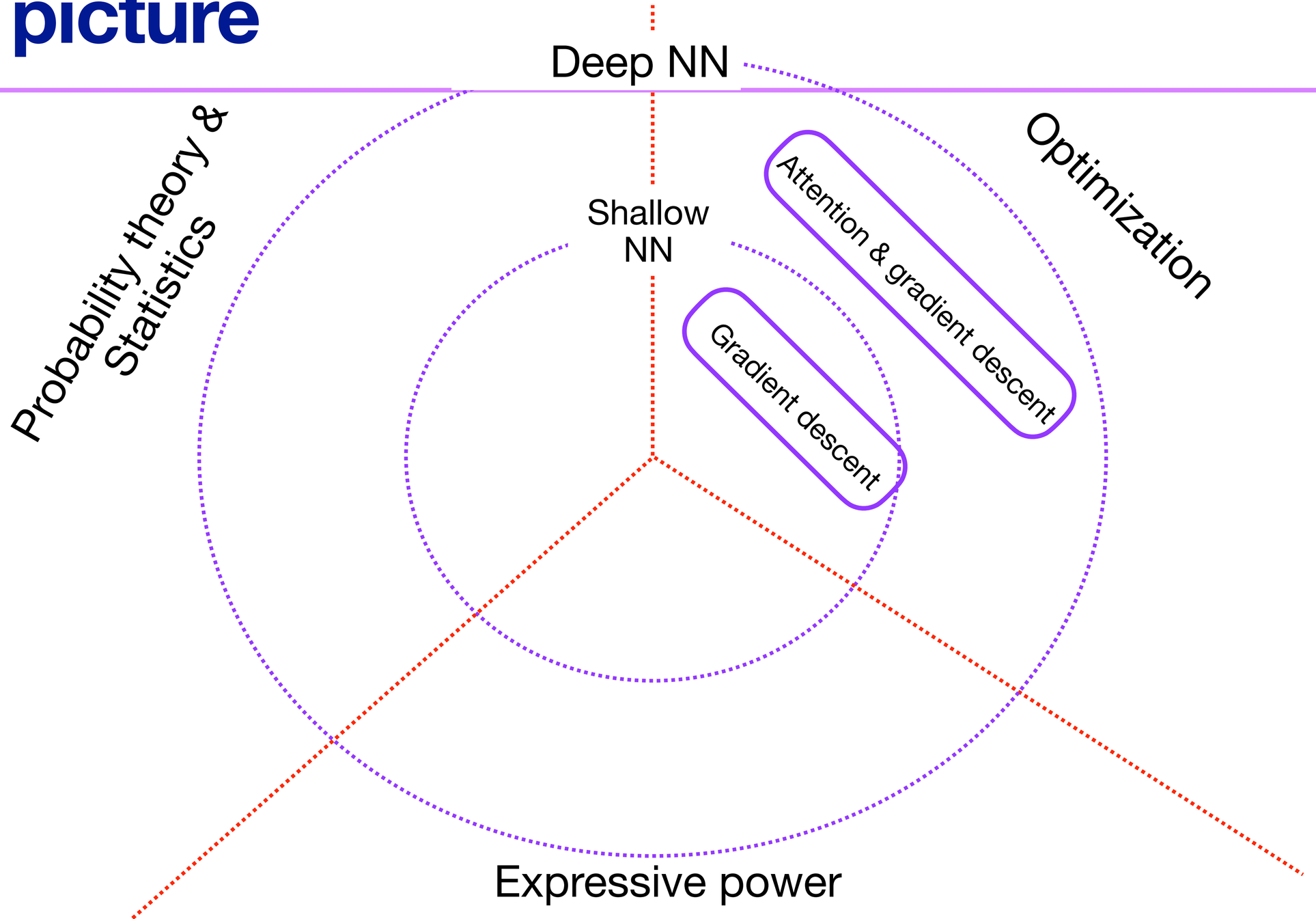
Outline

- Introduction to LLMs and self-supervised learning
- Attention layers in LLMs
- A group activity

Question about accessibility of slides

- Are slides easy to read in .pdf?

Big picture


Deep NN
Shallow NN
Probability theory & Statistics
Optimization
Attention & gradient descent
Gradient descent
Expressive power

Large Language Models (LLMs)

Natural Language Processing before LLMs

Sentiment Analysis

Negative. I do not like this product 

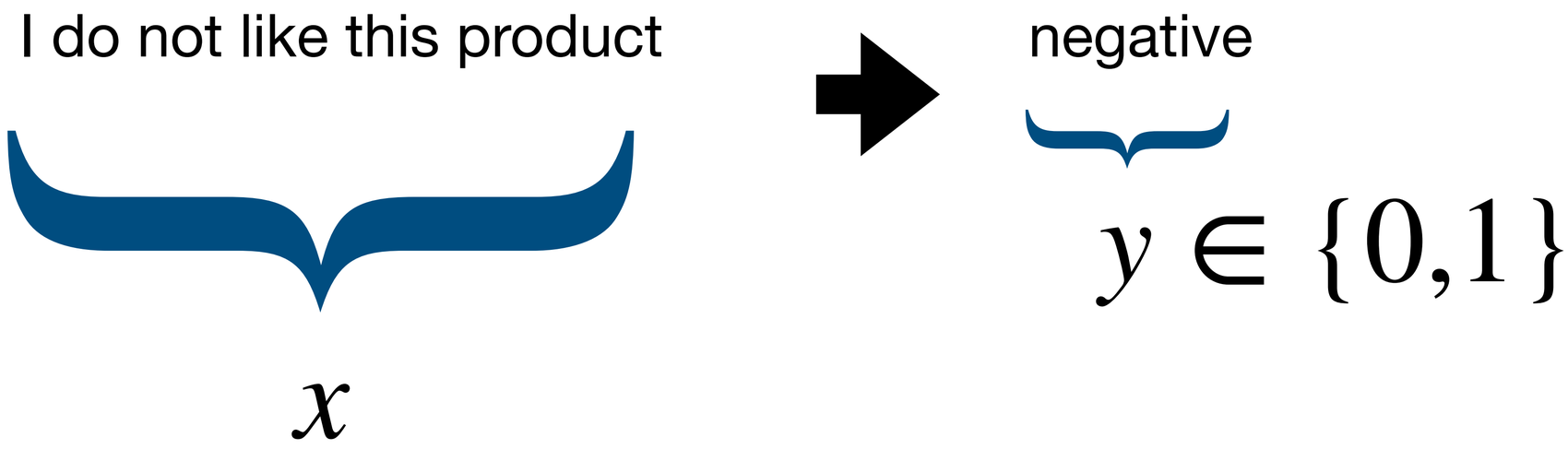
Positive. I like this product 

Translation

English: This is a book

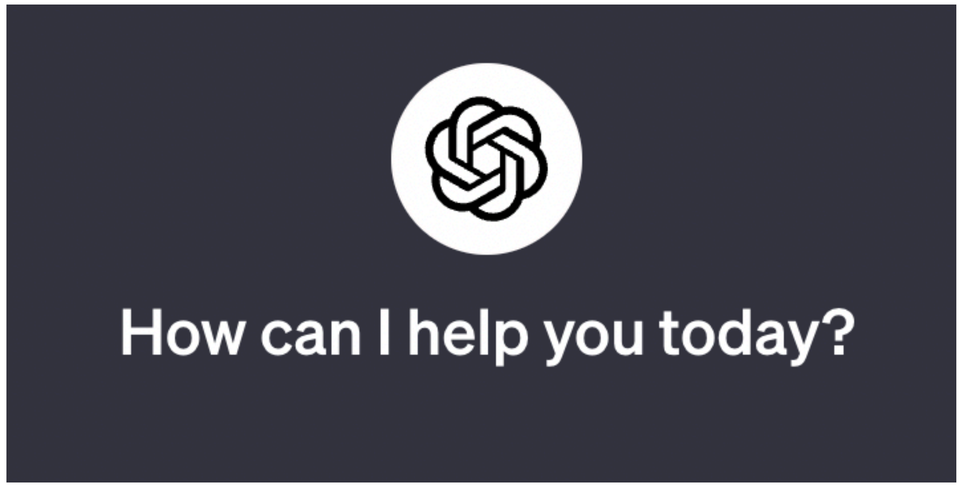
Dutch: Dit is een boek

Supervised learning formulation



- Two challenges:
  - How to represent x as a vector?
  - Labeling data is labor-intensive and time consuming

LLMs do not need human supervision



Self-supervised learning

- Self-supervised learning aims to generate labels from data itself.

- Input x

- Output y

The power of self-supervised learning

Sentiment Analysis

Prompt. “I do not like this product” is positive?

Output. No

Translation

Prompt. Translate “this is a book” to Dutch

Output: Dit is een boek

Traditional word embedding

Traditional NLP: Word Embedding

- How to represent a sentence/word?
  - Example: I do not like this product
- One solution:
  - construct a vector of the size of dictionary
  - $$this = [0,\ldots,0,\underbrace{1}_{\text{index of "this" in dictionary}},0,\ldots,0]$$
- Question: What is the issue with the above binary representation?
- Solution: Does not capture semantic

Traditional word embedding

- w is a word with vector representations $v_w \in \mathbb{R}^d$
- All text in world can be presented as a sequence of words

$$w_1, w_2, \ldots, w_i, \ldots,$$

- $p(w_i \mid w_j) = \begin{bmatrix} 1 & w_i = \arg\max_w \langle v_w, v_{w_j} \rangle \\ 0 & \text{otherwise} \end{bmatrix}$
- $\max_{v_{w_i}} \sum_{j-c \le k \le j+c} \log p(w_{j+k} \mid w_j)$

Reference: Distributed Representations of Words and Phrases and their Compositionality: https://arxiv.org/abs/1310.4546

Softmax

- Consider model $p(w_i \mid w_j) = \begin{bmatrix} 1 & w_i = \arg\max_w \langle v_w, v_{w_j} \rangle \\ 0 & \text{otherwise} \end{bmatrix}$
- Challenge: The above model is not differentiable
- Softmax: $p(w_i \mid w_j) = \frac{e^{\langle v_{w_i}, v_{w_j} \rangle}}{\sum_k e^{\langle v_{w_k}, v_{w_j} \rangle}}$

Semantic of embeddings

- $v_{\text{madrid}} - v_{\text{spain}} + v_{\text{france}} = ?$
  - $\approx v_{\text{paris}}$
- $v_{\text{czech}} + v_{\text{currency}} = ?$
  - $\approx v_{\text{koruna}}$
- $v_{\text{german}} + v_{\text{airline}} = ?$
  - $\approx v_{\text{Lufthansa}}$

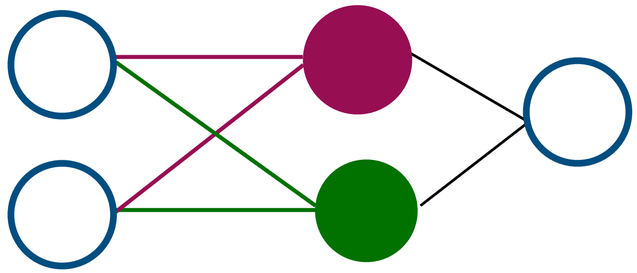
Reference: Distributed Representations of Words and Phrases and their Compositionality: https://arxiv.org/abs/1310.4546

Traditional NLP: Inference using word embeddings

I do not like this product

preprocessing

$v_{\text{I}}, v_{\text{do}}, v_{\text{not}}, v_{\text{like}}, \ldots,$



negative or positive

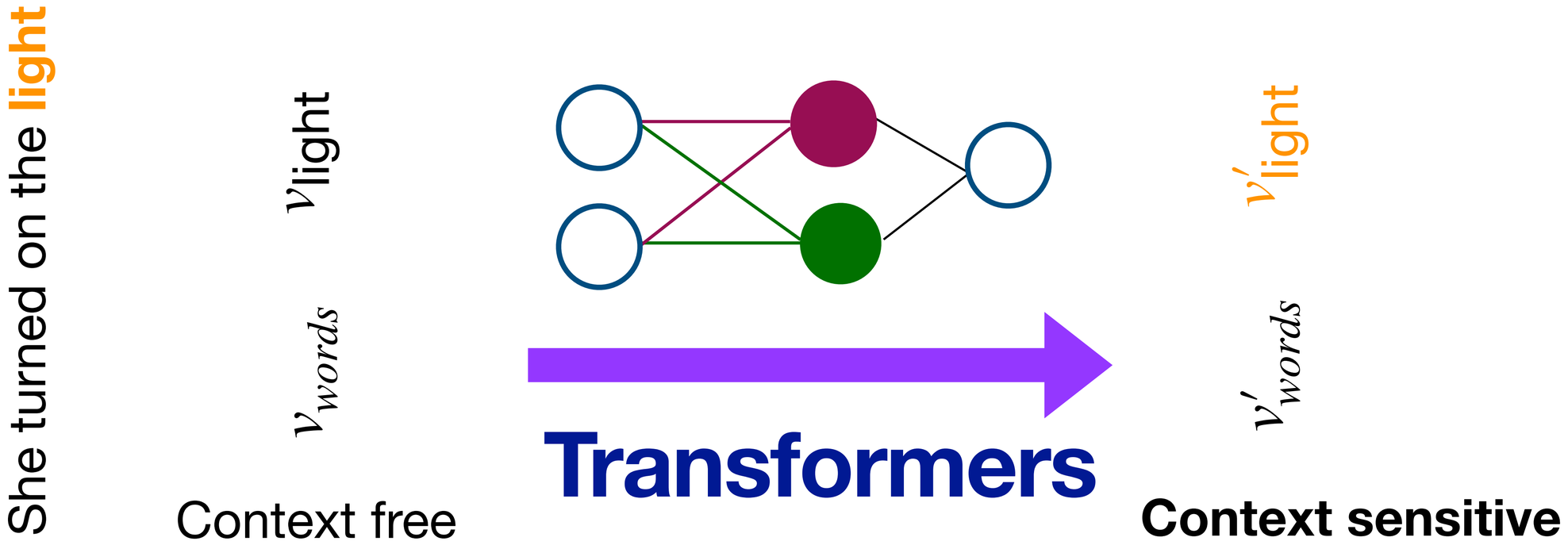
$$y \in \{0,1\}$$

$$\arg\min_{\theta} \mathbb{E}_{v_i,y} \| NN_{\theta}(v_1, \ldots) - y \|^2$$

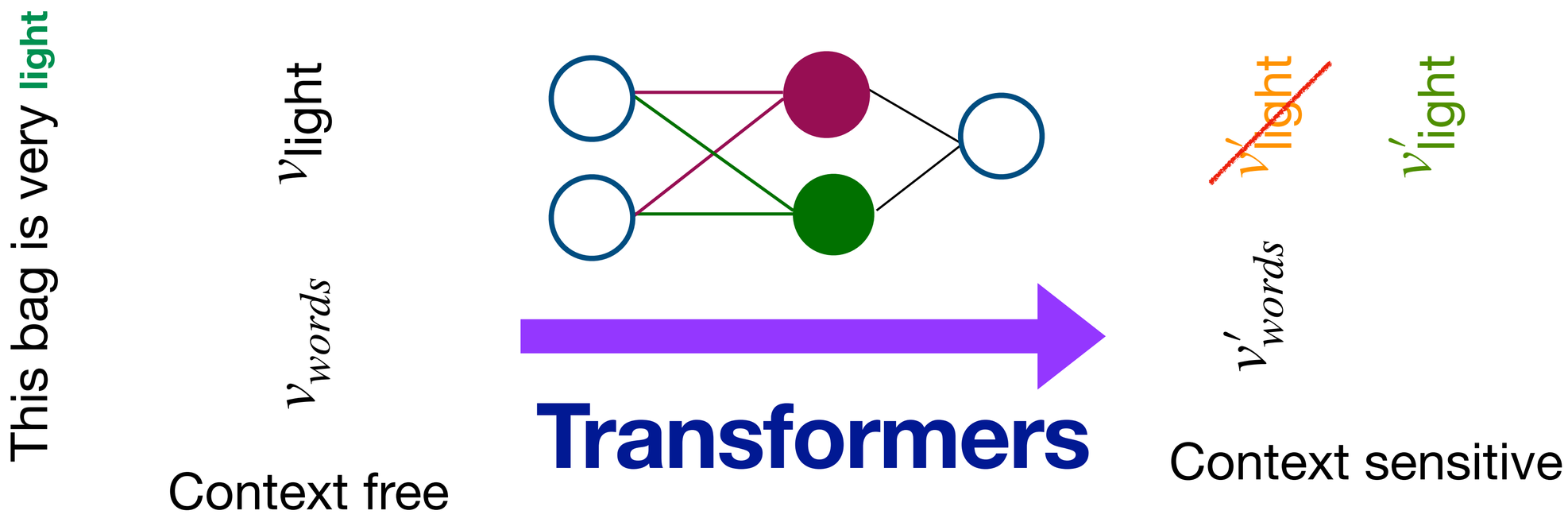
The issue with word embedding

- Question: Word embedding is context free, why this is an issue?
- Consider word “light”
  - This bag is very (light: not heavy)
  - She turned on the (**light**: brightness)
- Word embedding for light is a fixed vector v_{light}
- Conclusion: while semantic is context sensitive, traditional word embedding is context free

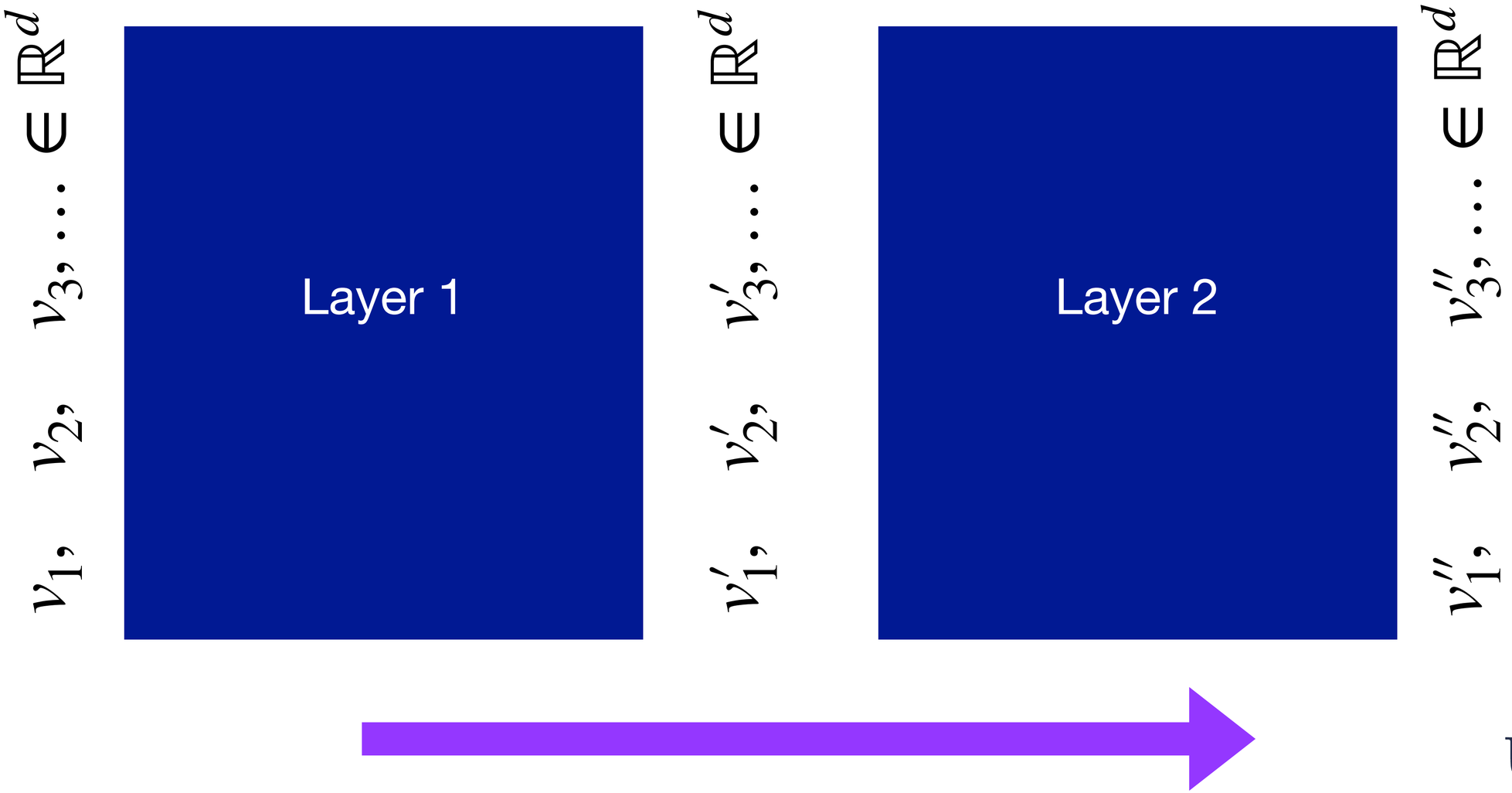
LLMs: context-sensitive word embedding



LLMs: context-sensitive word embedding

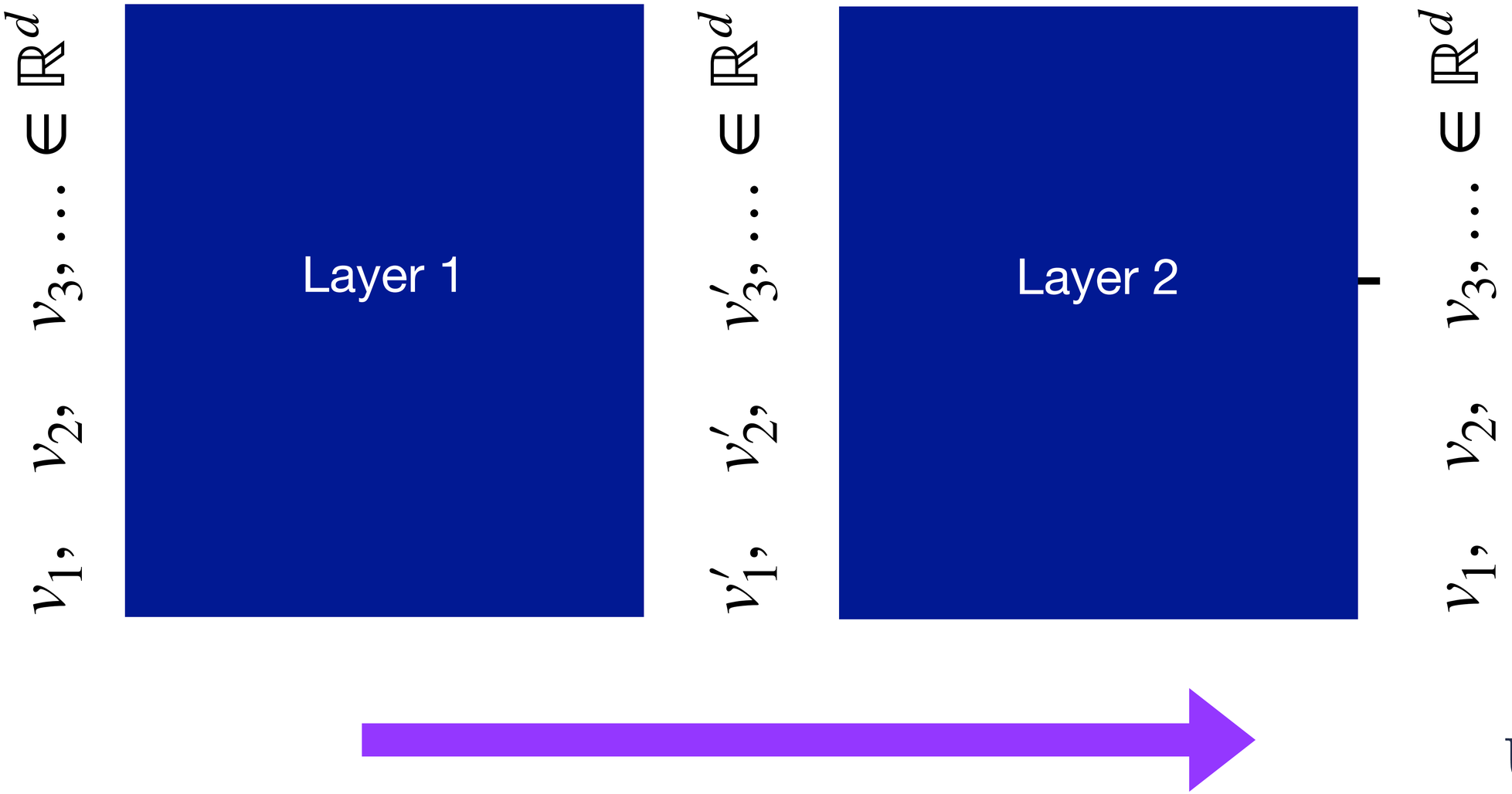


Context sensitive transformers



How to generate context-sensitive embeddings?

Context sensitive transformers



Outline

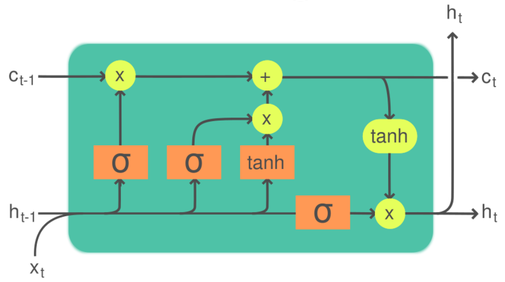
- Introduction to LLMs and self-supervised learning
- Attention component in LLMs
- A group activity

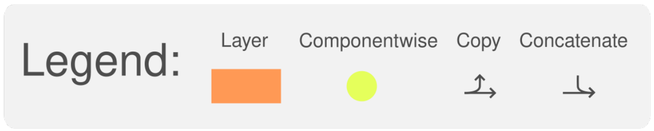
Development of attention over time

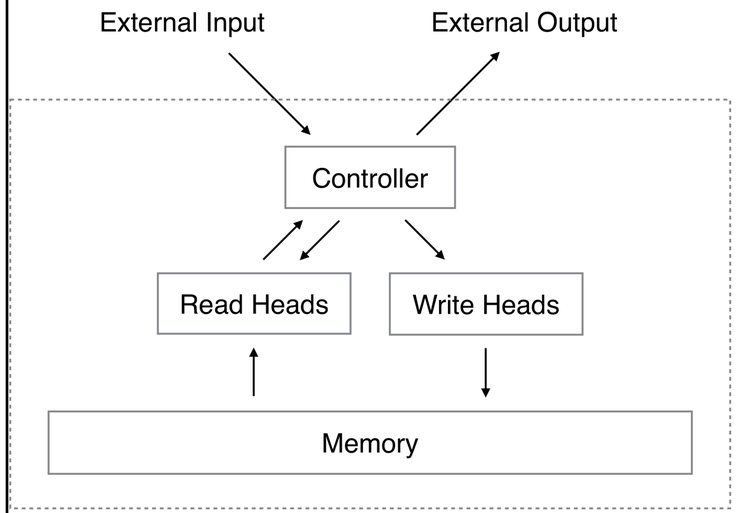
Model:

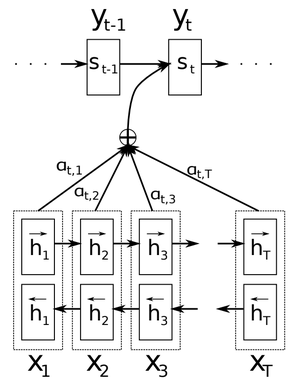
Goal:

LSTM	Neural Turing Machine	Neural Translation	Transformers
Long short-term memory			
Talking signal vanishing	Designing memory for NNs	Translation	next token pred.
1993 +120K citations	2014	2015	2019
Used in Google translate		+30K citations	









[Bahdanau, Cho, Bengio]

UNIVERSITY of VIRGINIA

Language model and attention mechanism

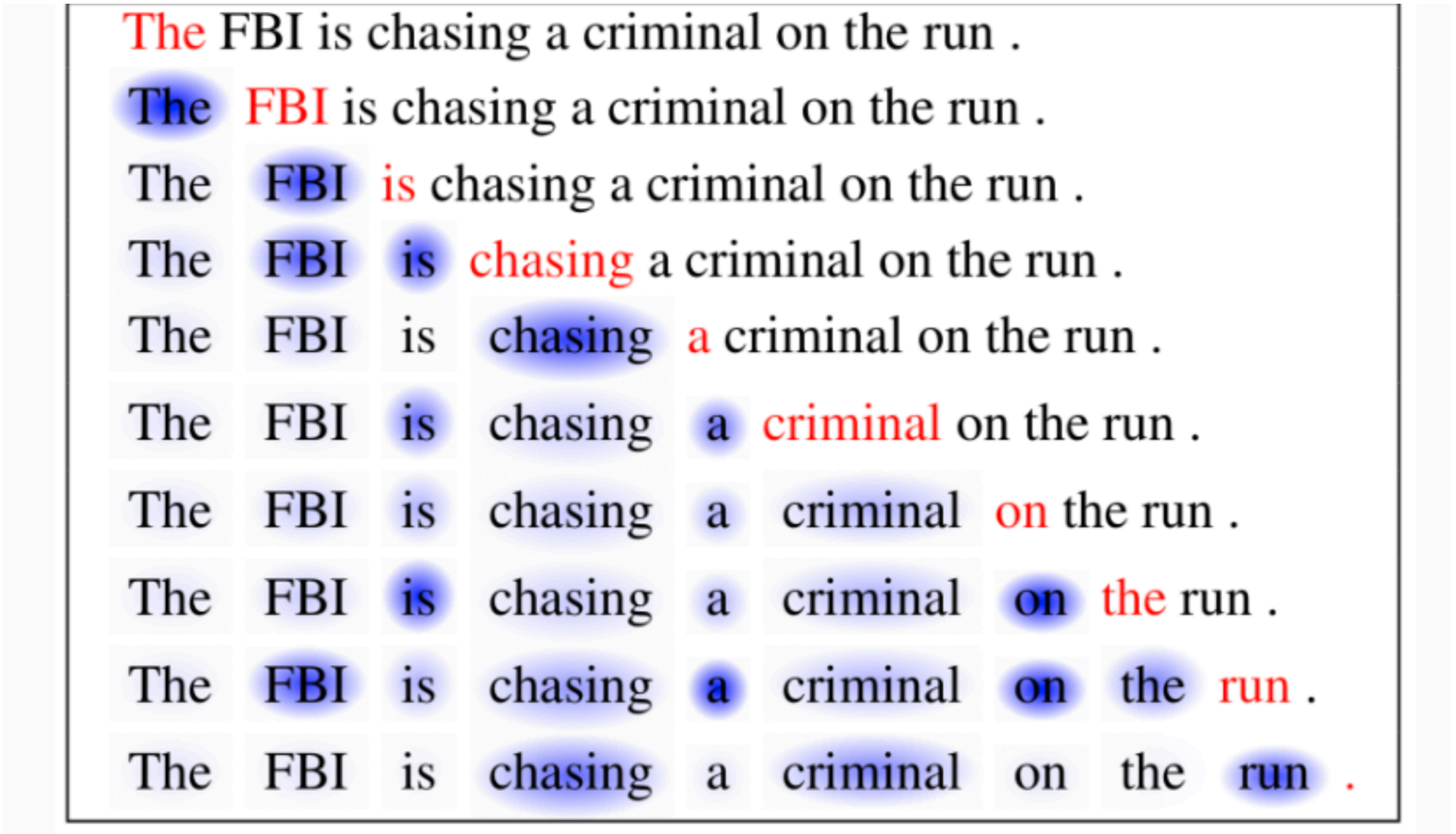
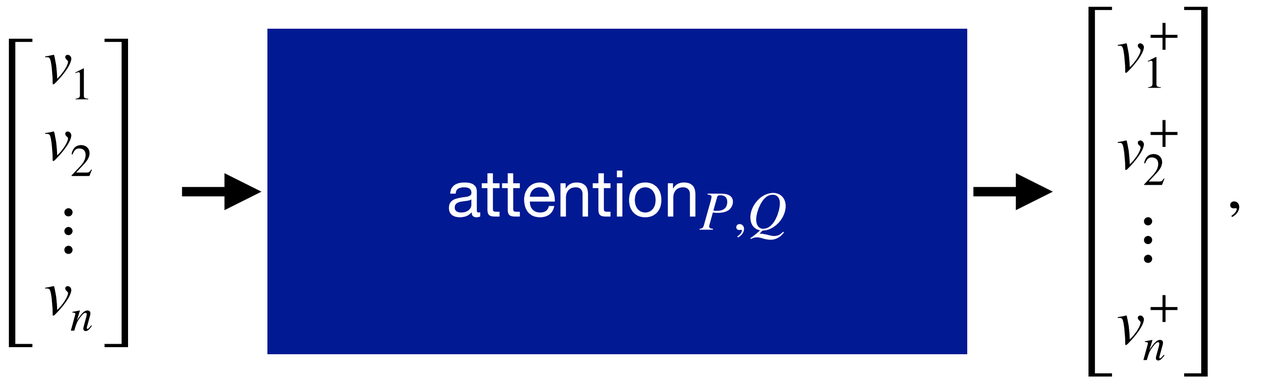


Image credit: https://www.analyticsvidhya.com/blog/2019/11/comprehensive-guide-attention-mechanism-deep-learning/

Attention layer

- The FBI is chasing a criminal on the run

$$\underbrace{v_1}\ \underbrace{v_2} \qquad\qquad \underbrace{v_i} \in \mathbb{R}^d$$



$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \rightarrow \begin{bmatrix} v_1^+ \\ v_2^+ \\ \vdots \\ v_n^+ \end{bmatrix}, \qquad \{P, Q\} \in \mathbb{R}^{d \times d}$$

$$v_i^+ = v_i + P \sum_{j=1}^{n} \alpha_{ij} v_j, \quad \alpha_{ij} = \frac{e^{v_j^\top Q v_i}}{\sum_k e^{v_k^\top Q v_i}} \qquad \text{[softmax]}$$

UNIVERSITY of VIRGINIA

Example

The FBI is chasing a criminal on the run .

$$v^{+}_{chasing} = v_{chasing} + P\left(\alpha_1 v_{the} + \alpha_2 v_{FBI} + \alpha_3 v_{is}\right)$$

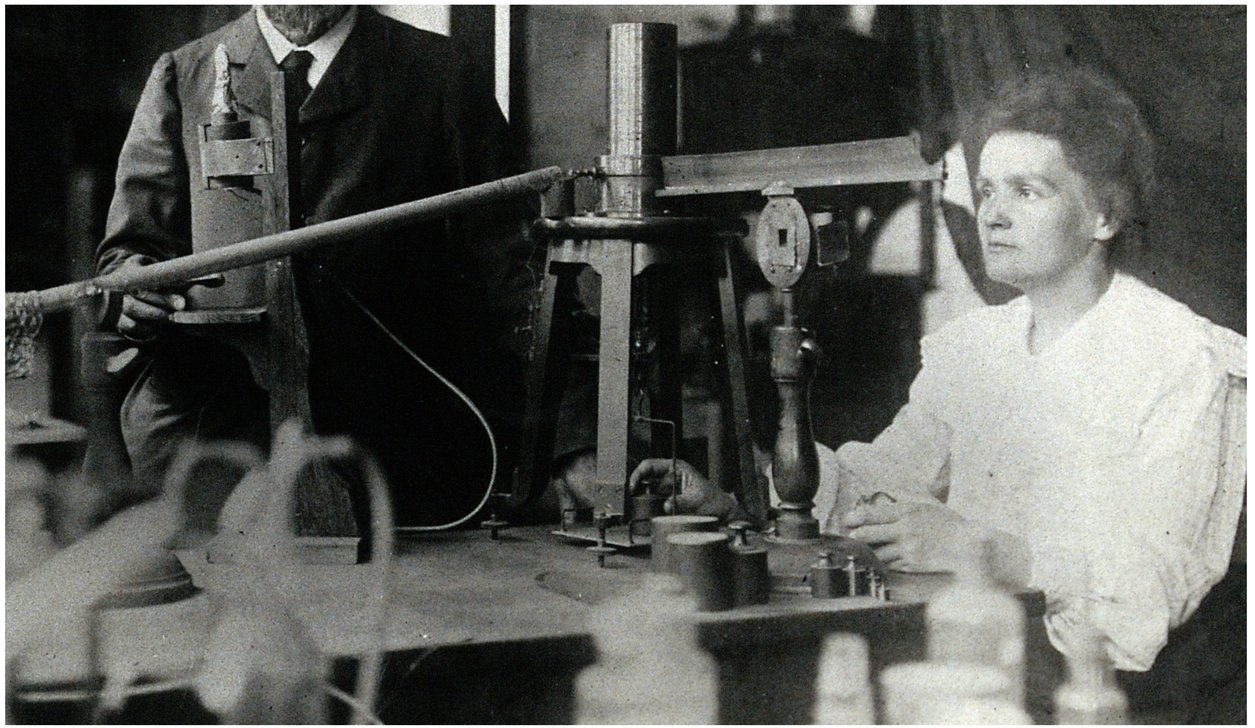
$$\alpha_1 = \frac{e^{v_{the}^{\top} Q v_{chasing}}}{e^{v_{the}^{\top} Q v_{chasing}} + e^{v_{FBI}^{\top} Q v_{chasing}} + e^{v_{is}^{\top} Q v_{chasing}}}$$

[softmax]

Image credit: https://www.analyticsvidhya.com/blog/2019/11/comprehensive-guide-attention-mechanism-deep-learning/

Outline

- Introduction to LLMs and self-supervised learning
- Attention component in LLMs
- A group activity

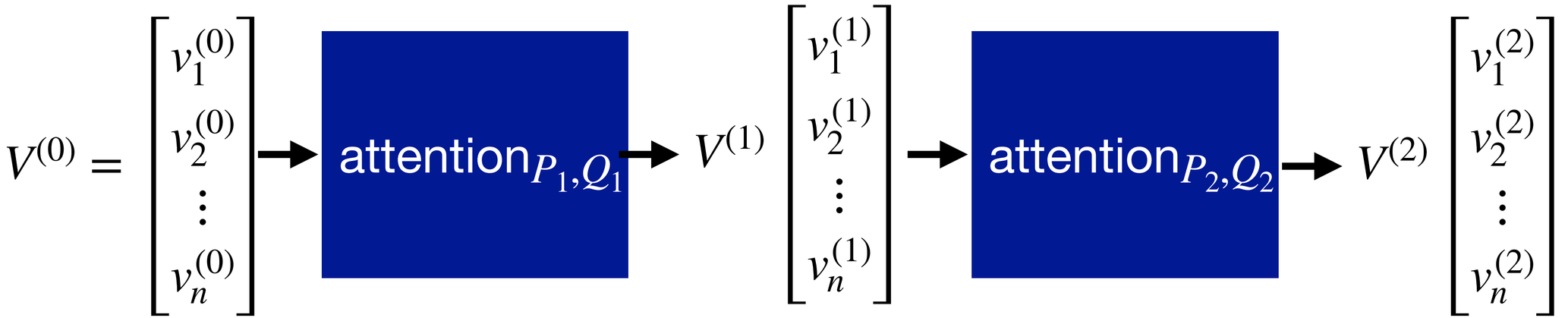


wikipedia: Marie Curie

Experiments

Get ready for hands-on group activity

Goal: word embeddings after attention layers

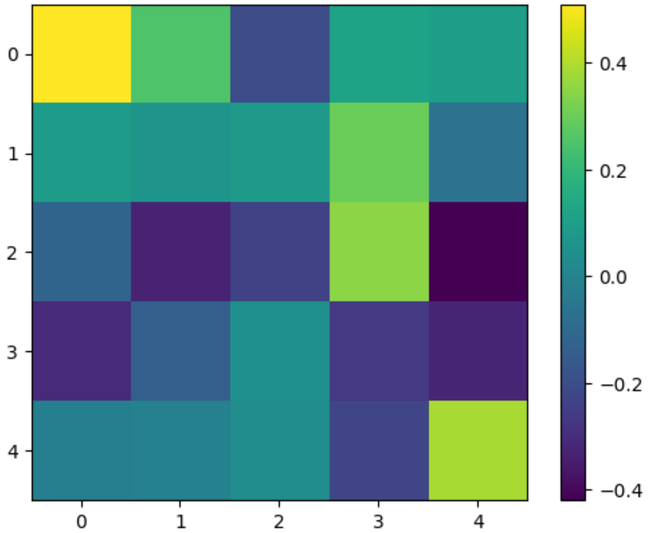


- Let $P_i = -0.01 I_{n\times n}$ (identity matrix), $Q_i = GG^\top, G \in R^{d\times d}, G_{ij} \sim N(0,1/d)$
- Task: Given $V_{ij}^{(0)} \sim N(0,1/2\sqrt{d})$, plot $V^{(2)}$.

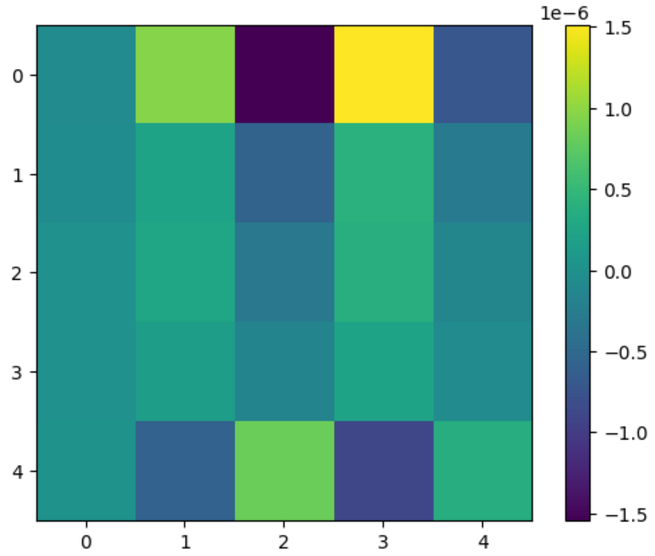
Colab

- https://shorturl.at/2clWn
- https://colab.research.google.com/drive/1w_O9xFkfvsgrPmQXBAZqVhHOLexriEqZ?usp=sharing
- Details are provided in the colab

An observation



$$V^{(0)} = \begin{bmatrix} v_1^{(0)} \\ v_2^{(0)} \\ \vdots \\ v_n^{(0)} \end{bmatrix}$$



$$V^{(2)} = \begin{bmatrix} v_1^{(2)} \\ v_2^{(2)} \\ \vdots \\ v_n^{(2)} \end{bmatrix}$$

Reference: Geshkovski, Borjan, et al. "The emergence of clusters in self-attention dynamics." *Advances in Neural Information Processing Systems* 36 (2024).

The next lecture

- Why word embeddings become low-rank? (Explaining the observation)
- Why attention layers are linked to gradient descent?

Thank you very much!

?

Next word predication and supervised learning

- This text is used to teach a model
- Questions:
  - What are response variables y (outputs)?
    - All words or characters. Would you prefer characters or words?
  - What are input features x?
    - $\underbrace{\text{This}}\ \text{text is used to}\ \underbrace{\text{teach}}\ \text{a model}$

$$x_1 \in \mathbb{R}^d \qquad x_i \in \mathbb{R}^d \rightarrow \text{word embeddings}$$

The power of self-supervised learning

Sentiment Analysis

Prompt. “I do not like this product” is positive?

Output. No

Translation

Prompt. Translate “this is a book” to Dutch

Output: Dit is een boek

The skill of "in-context learning”

- **In-context learning:** Language models can learn from in-context data

Example: swapping characters

Prompt: ac ca ; qm mq ; mn ?

GPT3: nm

- Reference: Language Models are Few-Shot Learners, Tom B. Brown et al.