Bias in AI:
Week #1: Introduction to Modern NLP

Instructor:

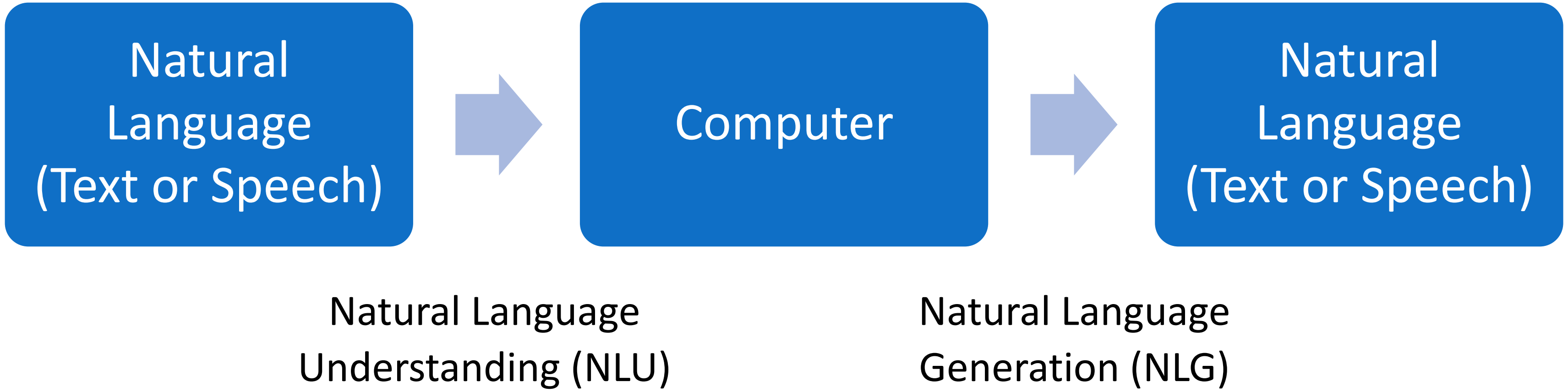
Sayyed Nezhadi

Summer 2022

Definition

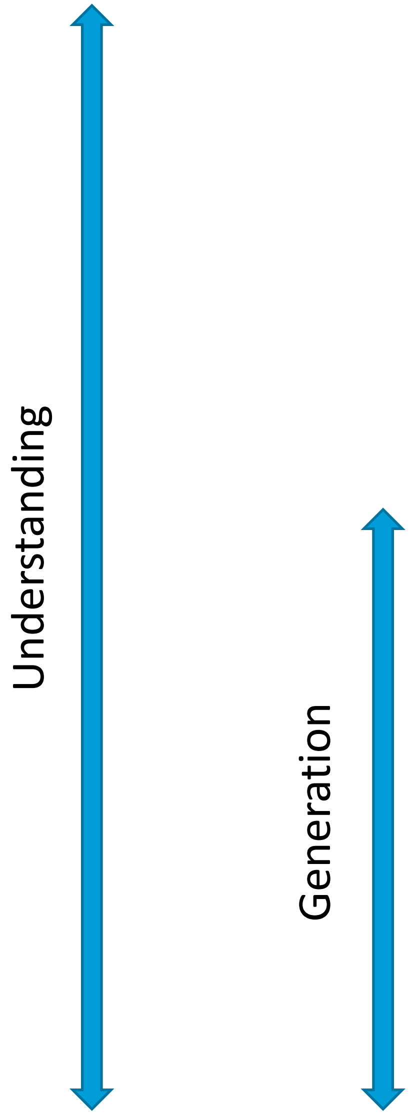
- **Natural Language Processing (NLP)**, is a branch of artificial intelligence that deals with the interaction between computers and humans using the natural language.

- **The main goals of NLP:** To understand and generate natural language.


Natural Language (Text or Speech)
Computer
Natural Language (Text or Speech)
Natural Language Understanding (NLU)
Natural Language Generation (NLG)

Sample NLP Applications

- Text Classification (e.g., Spam Filtering)
- Sentiment Analysis / Market Intelligence
- Information Retrieval / Document Parsing
- Search Engines (e.g., Google, Bing)
- Content Recommendation
- Machine Translation (e.g. Google Translate)
- Chatbots (e.g. IBM Watson-powered chatbot)
- Personal Assistants (Siri, Google Home, Alexa)
- Text to Speech (Voice Generation)
- Speech to Text (Voice Transcription)
- Text Summarization
- Question and Answering



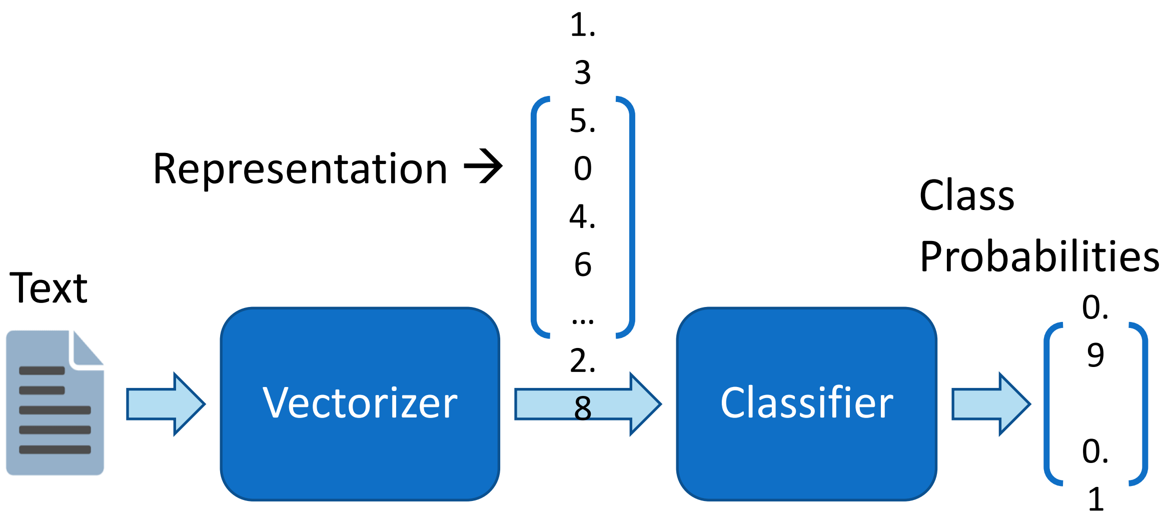
NLP is Hard (Ambiguity in Language)

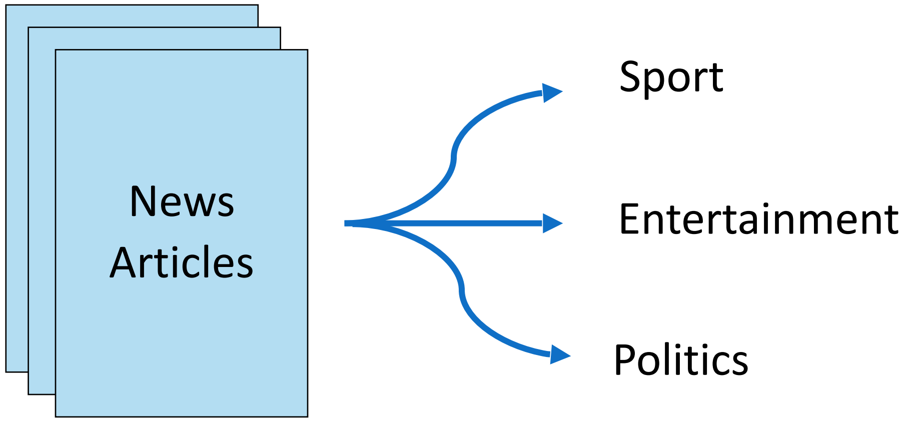
- Lexical Ambiguity:
  - I saw a bat.
  - The fisherman went to the bank.
- Syntactic Ambiguity:
  - The chicken is ready to eat.
  - I ate a salad with dressing from Italy.
  - I threw a brick through the window. It broke.
- Semantic Ambiguity:
  - John and Mary are married." (To each other? or separately?)
  - Can you lift this bag? (Can? Or Will?)
- Misspellings
- Acronyms, slangs, …

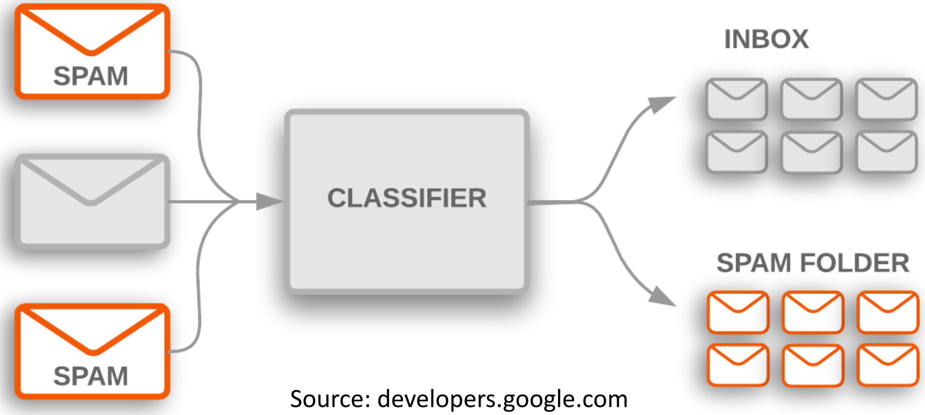
Text Classification

To use a classifier, we need to represent the text by a vector (Vectorization):

- How to represent a word?
- How to represent a text? (variable number of words)







Source: developers.google.com

Word Representation - Discrete

In traditional NLP, words are represented as discrete entities (one-hot vectors):

$$\text{Word} = [x_1, x_2, \ \ldots, x_D], x_i \epsilon [0, 1]$$

D= number of words in vocabulary (e.g., 30,000)

Problems:

- Very large and sparse vectors (not suitable for deep networks)
- Similarities (meaning and word family) are not captured:

House = [0, 0, …, 0, 1, …]

Home = [0, 0, …, 1, 0, …]

* The above vectors are orthogonal

Word Representation – by Context

The idea:

- Represent any word with a dense vector (a.k.a., word embedding), so that the similar words have similar vectors.
- They are also called, “word embedding”

How?

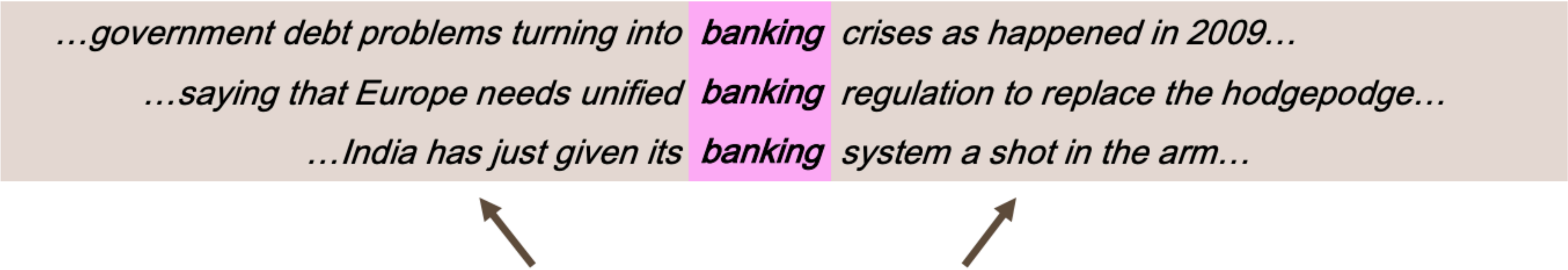
- Use different context of a word to generate a representation (a very successful idea in modern NLP)
- In each text, a word context is the set of words that appear close to the word.

$$\text{house} = \begin{pmatrix} 0.23 \\ 0.57 \\ -0.45 \\ 0.77 \\ 0.17 \\ -0.99 \\ 0.35 \\ 0.01 \end{pmatrix}$$

Word Representation – by Context

You shall know a word by the company it keeps

(Firth, J. R. 1957:11)



…government debt problems turning into **banking** crises as happened in 2009…

…saying that Europe needs unified **banking** regulation to replace the hodgepodge…

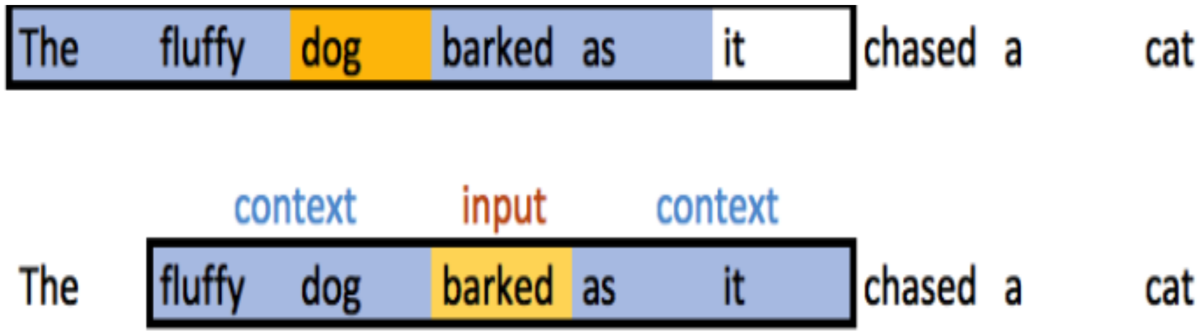
…India has just given its **banking** system a shot in the arm…

These context words will represent ***banking***

Source: cs224n slides Stanford

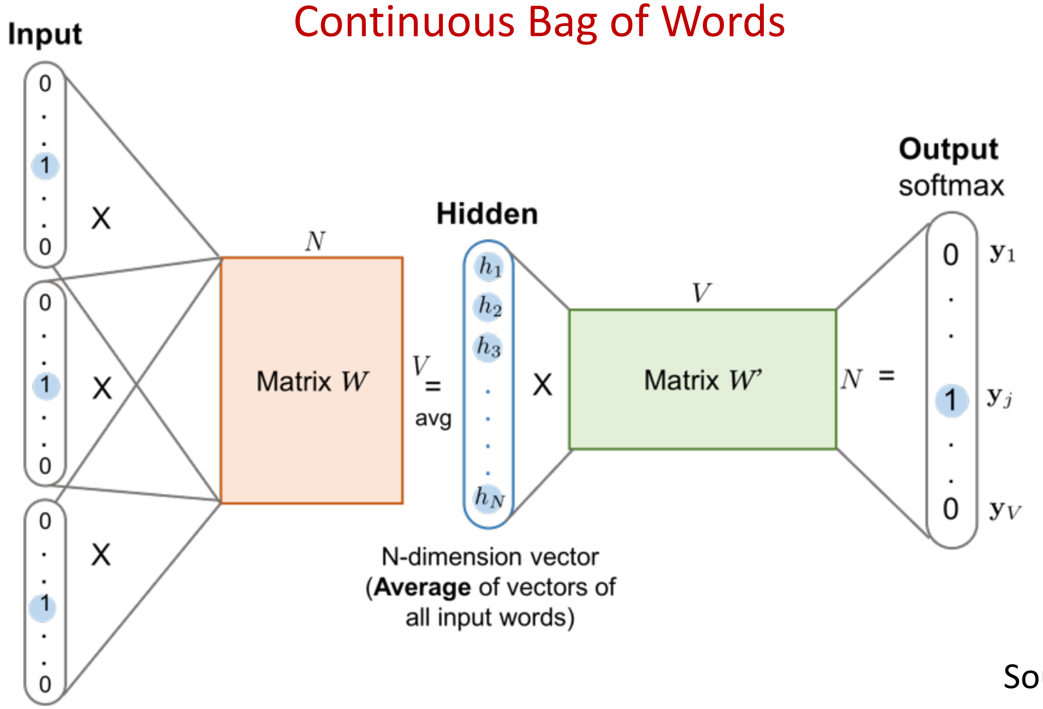
Example: word2vec (Mikolov et al. 2013)

- Using a sliding window to choose a word and its context

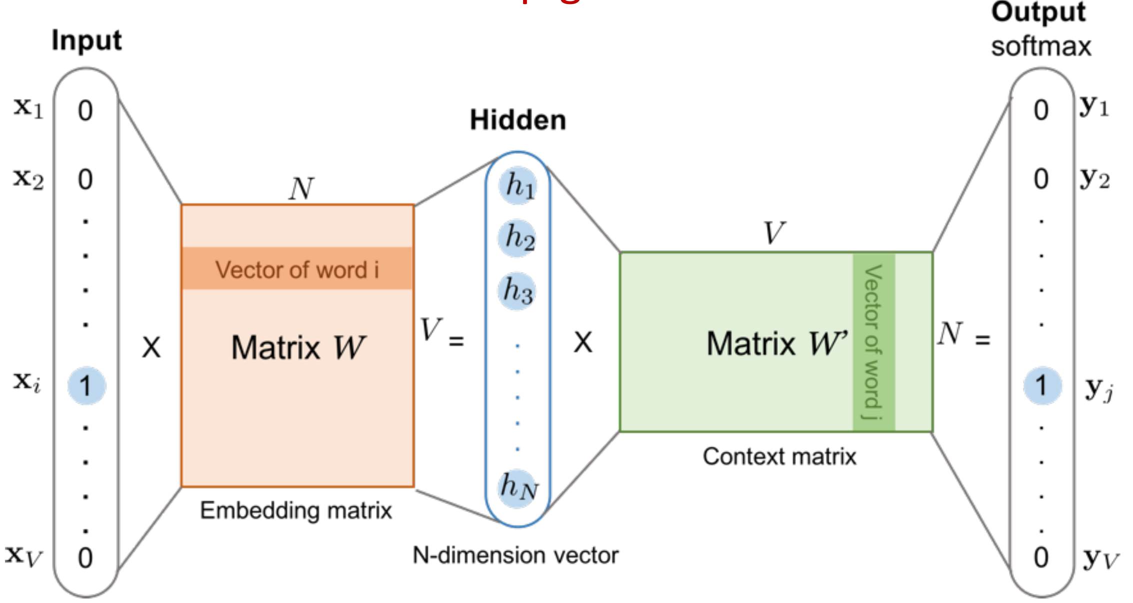

The fluffy dog barked as it chased a cat
context
input
context
The fluffy dog barked as it chased a cat

- Two variants:

Continuous Bag of Words



Skip-gram



Source: Lil'Log

Other Methods

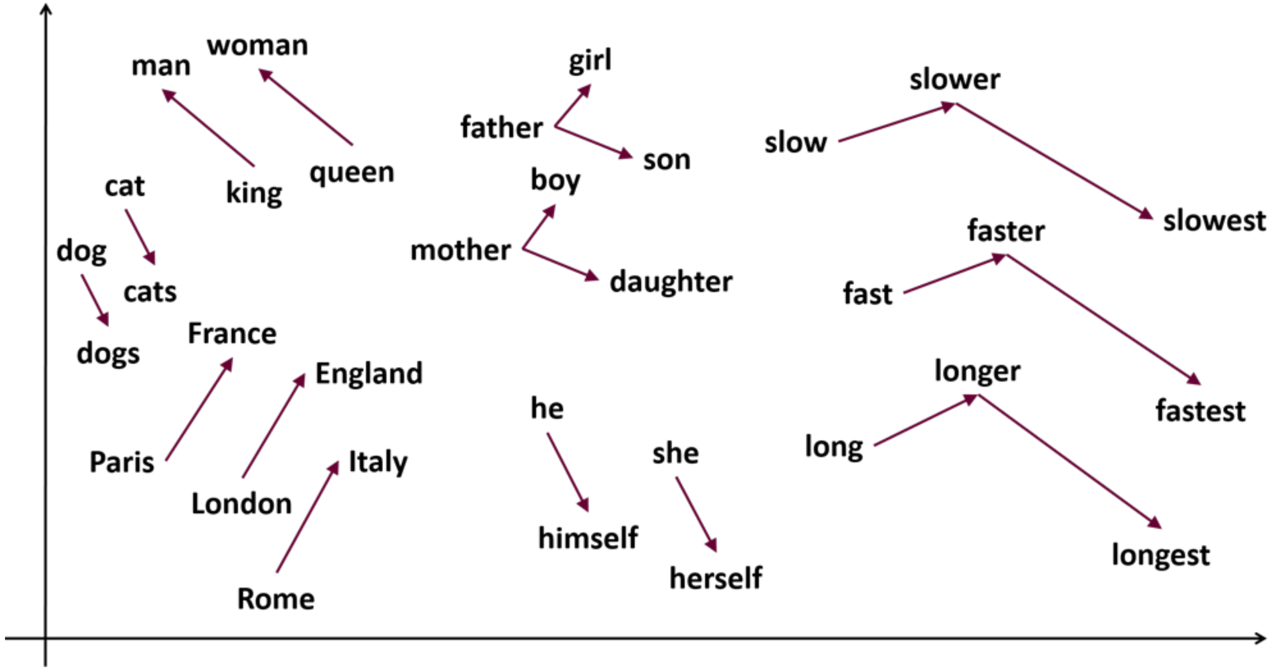
- GloVe: Global Vectors for Word Representation (Pennington et al. 2014)
- fastText: Enriching Word Vectors with Subword Information (Facebook Research 2016)

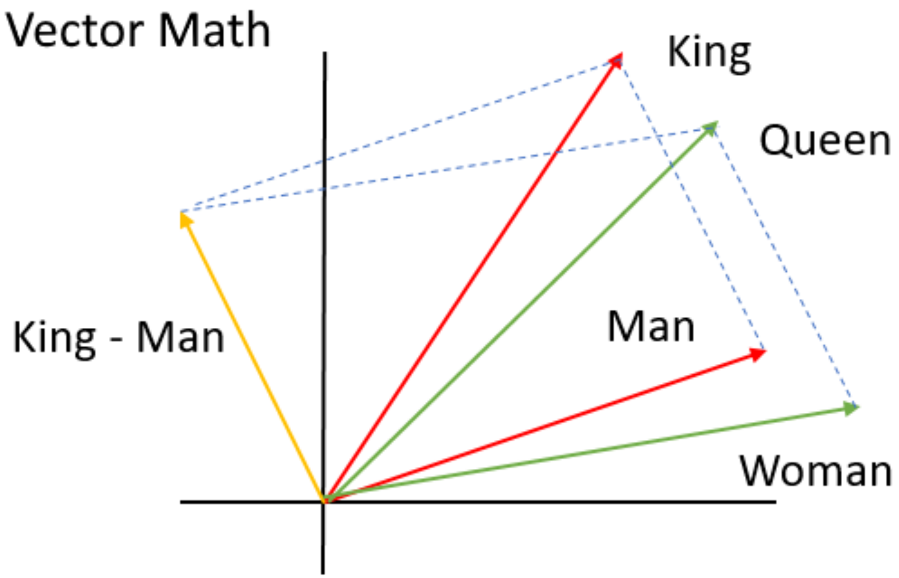
Word Embedding Properties

- Capturing similarities
- Dense vector (e.g., 300 dimensions)

- Surprising property: meaningful arithmetic operations

$$Closest\Big(V\big(king\big) - V(man) + V(woman)\Big) = V(queen)$$





Bias in Word Embedding

$$\overrightarrow{man} - \overrightarrow{woman} \approx \overrightarrow{computer\ programmer} - \overrightarrow{homemaker}$$

$$\overrightarrow{man} - \overrightarrow{woman} \approx \overrightarrow{doctor} - \overrightarrow{nurse}$$

Source: Bolukbasi et al. 2016

How to Represent a Text

- A couple of ideas:
  - Average the word embedding vectors (remember arithmetic operations are meaningful).
  - Concatenate the word embedding vectors (Challenge: the representation vector will be very long. Also, the number of words is not the same in each text)
- Other methods like "Doc2Vec" (Mikolov et al. 2014)
- But, what about the order of the words? The following sentences are different:

"The dog bit the baby" ≠ "The baby bit the dog"

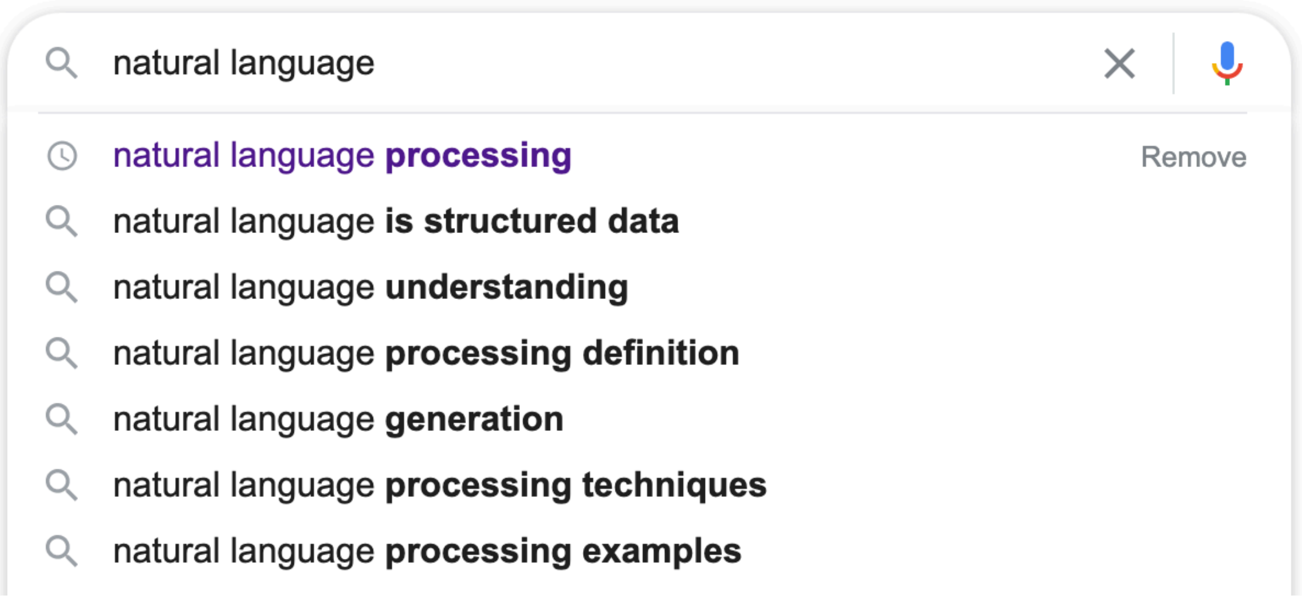
** We need a sequential model **

Language Model

- Consider a “Word Prediction” or “Auto Suggestion” use case.
- Predicting the next word is called Language Modeling.

- A probability distribution over sequences of words, is called Language Model.

$$P\left(w_{t+1} \middle| w_t,\ w_{t-1},\ \ldots,\ w_1\right)$$



natural language

natural language **processing** Remove

natural language **is structured data**

natural language **understanding**

natural language **processing definition**

natural language **generation**

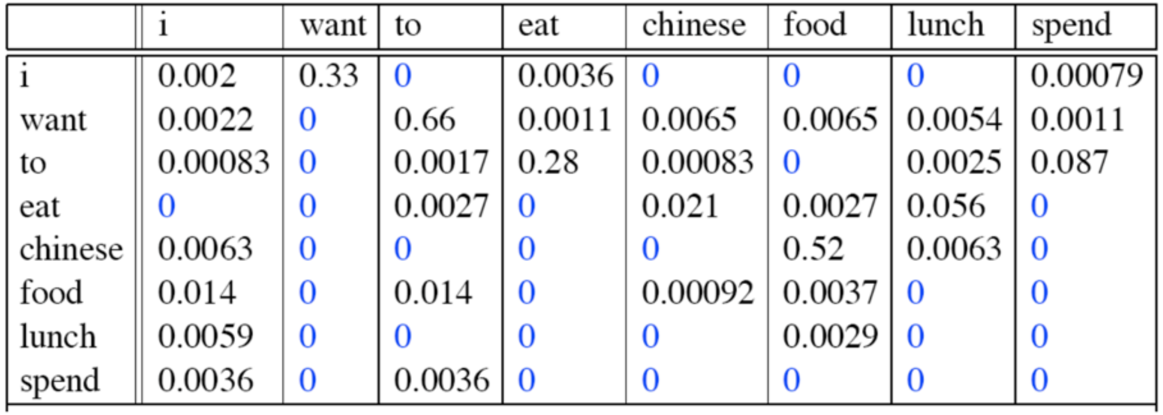
natural language **processing techniques**

natural language **processing examples**

Example Language Models

- N-gram Language Model (pre-deep learning era): build and store a probability table for occurrence of words based on previous (n-1) words
  - Sparsity problem
  - Storage problem
  - Fixed window (small)

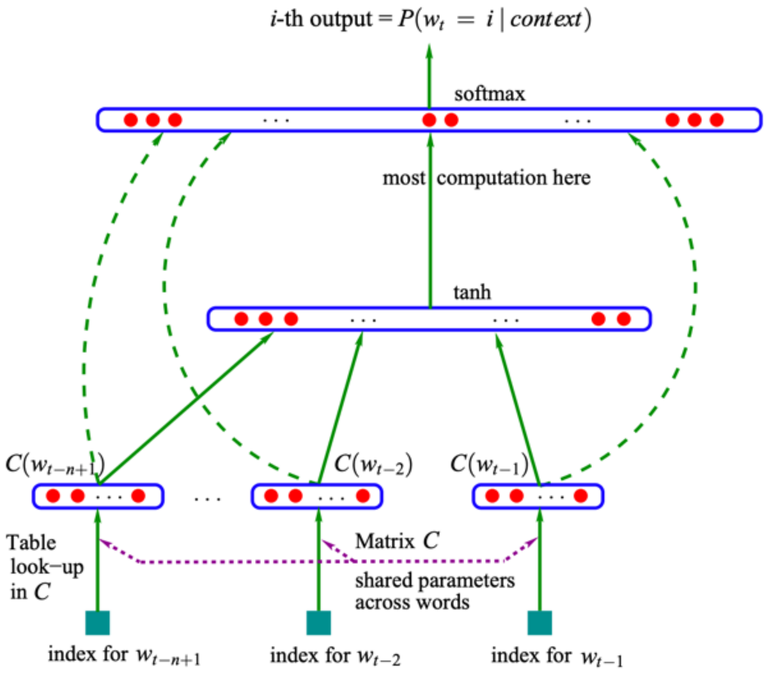
Bigram example (Dan Jurafsky)



	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

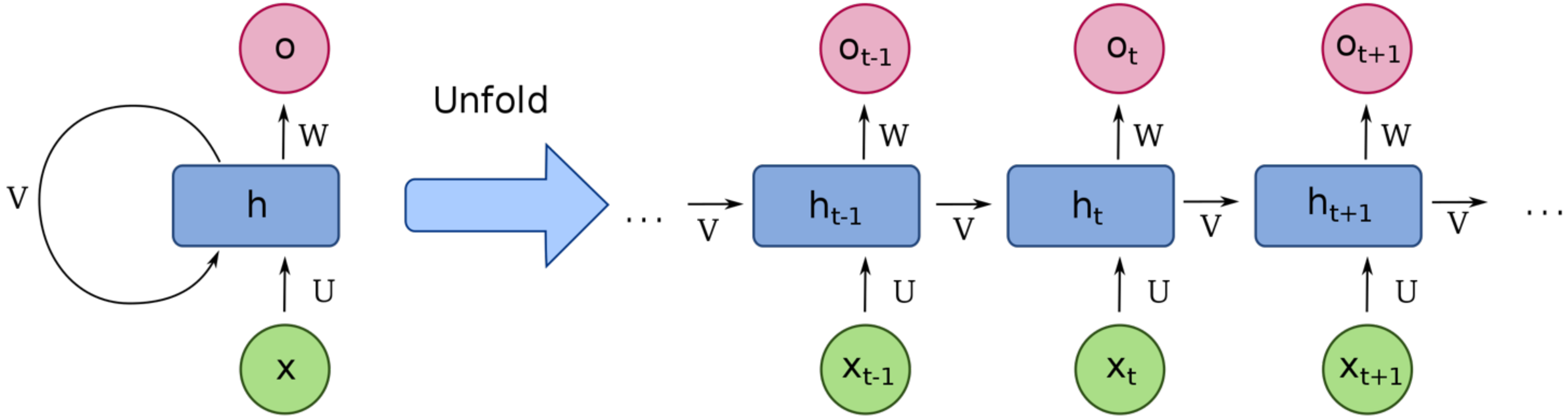
- Fixed-window Neural Model (Bengio, et al. 2000-2003):
  - Fixed window problem

** Need a model for any length **



Recurrent Neural Networks (RNN)

- A class of neural networks that are suited to process time-series and other sequential data.
- It stores the information from the previous experiences to be used for future prediction.
- Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs) are two common RNN architectures.



By fdeloche - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=60109157

RNN Language Model

- The input sequence can be of any size
- In theory, it can hold info from a long time in the past
- Model size doesn't depend on the input size
- The weights are shared (symmetry)

output distribution

$$\hat{\boldsymbol{y}}^{(t)} = \text{softmax}\left(\boldsymbol{U}\boldsymbol{h}^{(t)} + \boldsymbol{b}_2\right) \in \mathbb{R}^{|V|}$$

hidden states

$$\boldsymbol{h}^{(t)} = \sigma\left(\boldsymbol{W}_h\boldsymbol{h}^{(t-1)} + \boldsymbol{W}_e\boldsymbol{e}^{(t)} + \boldsymbol{b}_1\right)$$

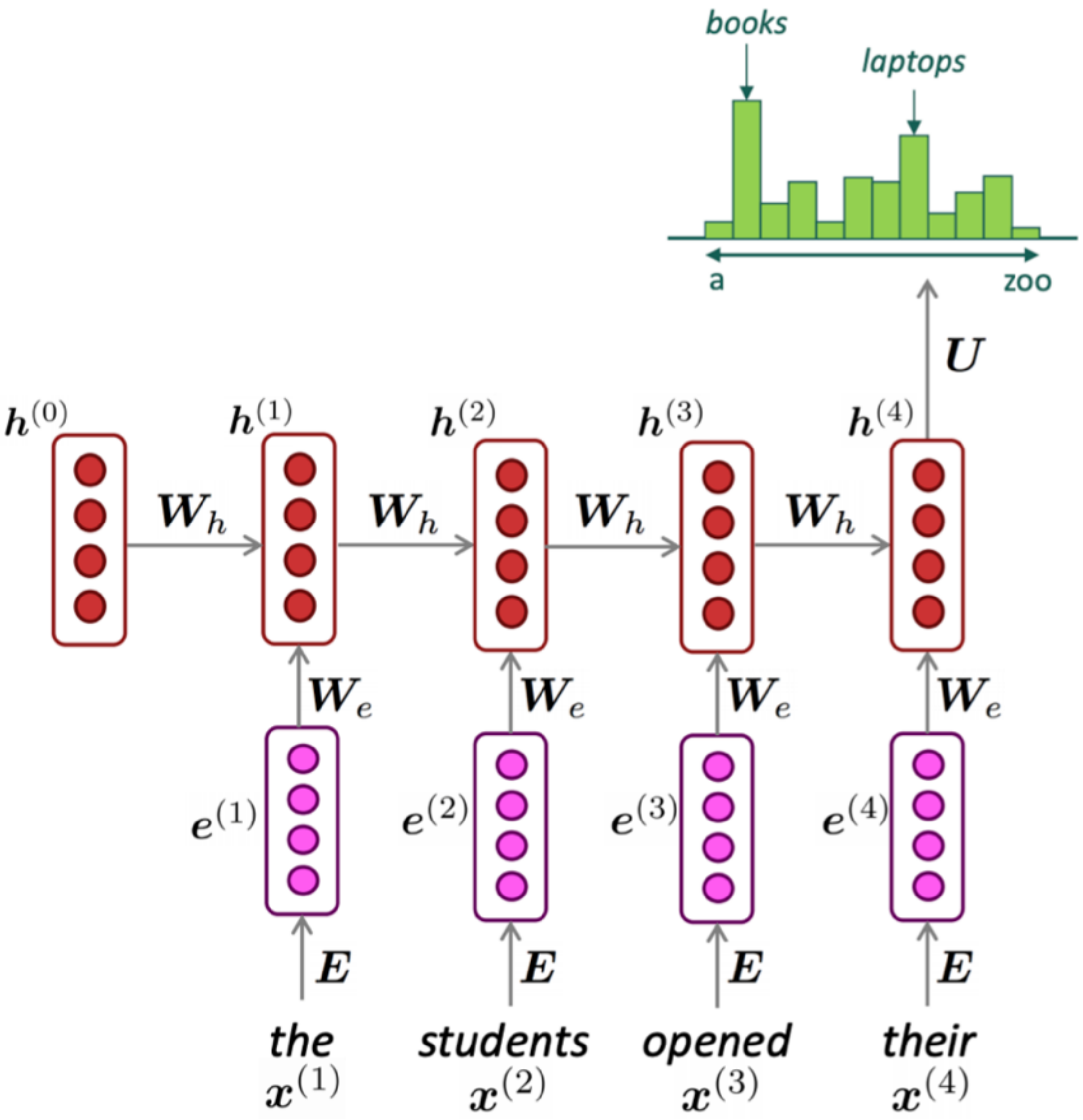
$\boldsymbol{h}^{(0)}$ is the initial hidden state

word embeddings

$$\boldsymbol{e}^{(t)} = \boldsymbol{E}\boldsymbol{x}^{(t)}$$

words / one-hot vectors

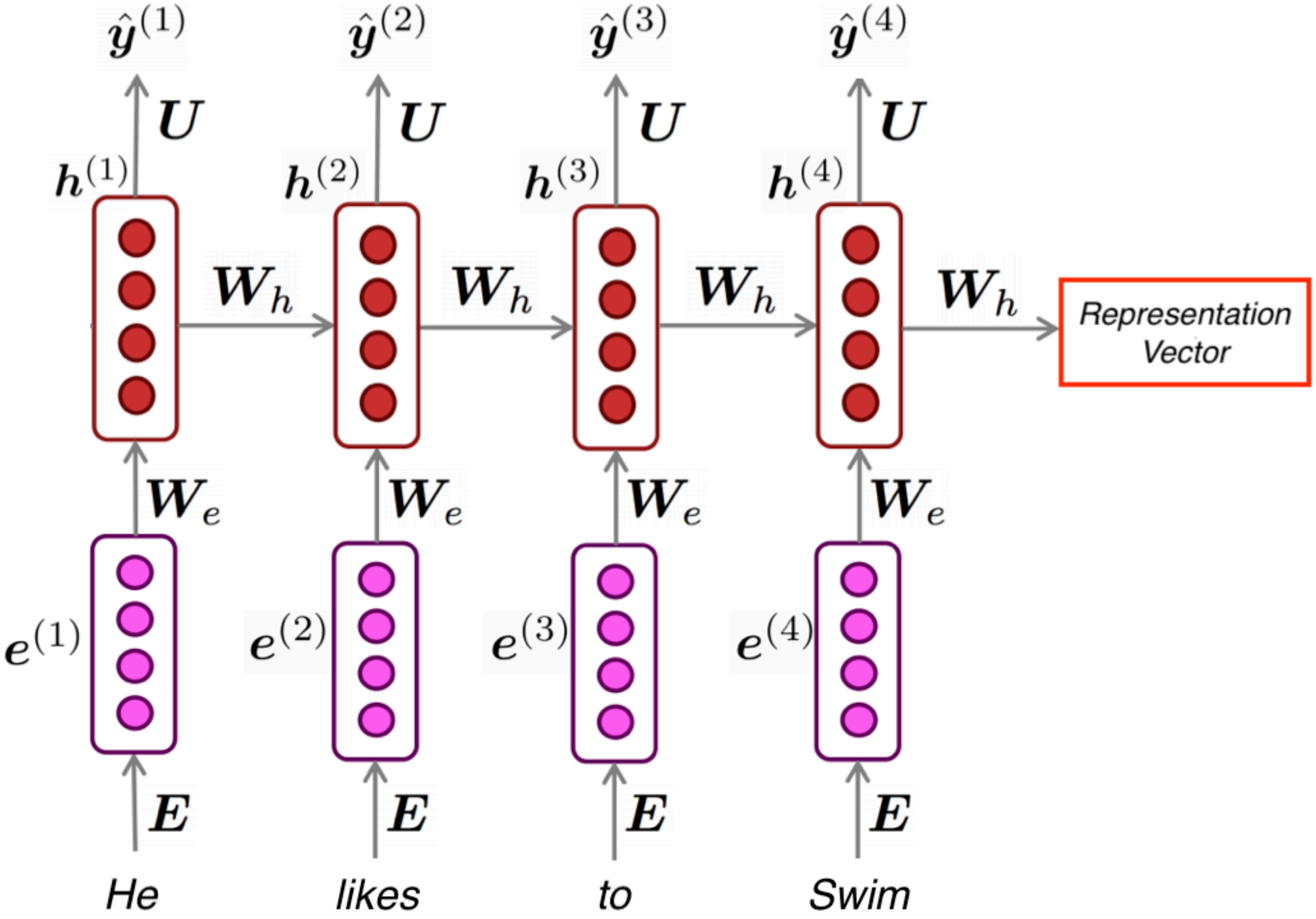
$$\boldsymbol{x}^{(t)} \in \mathbb{R}^{|V|}$$



Source: cs224n slides Stanford

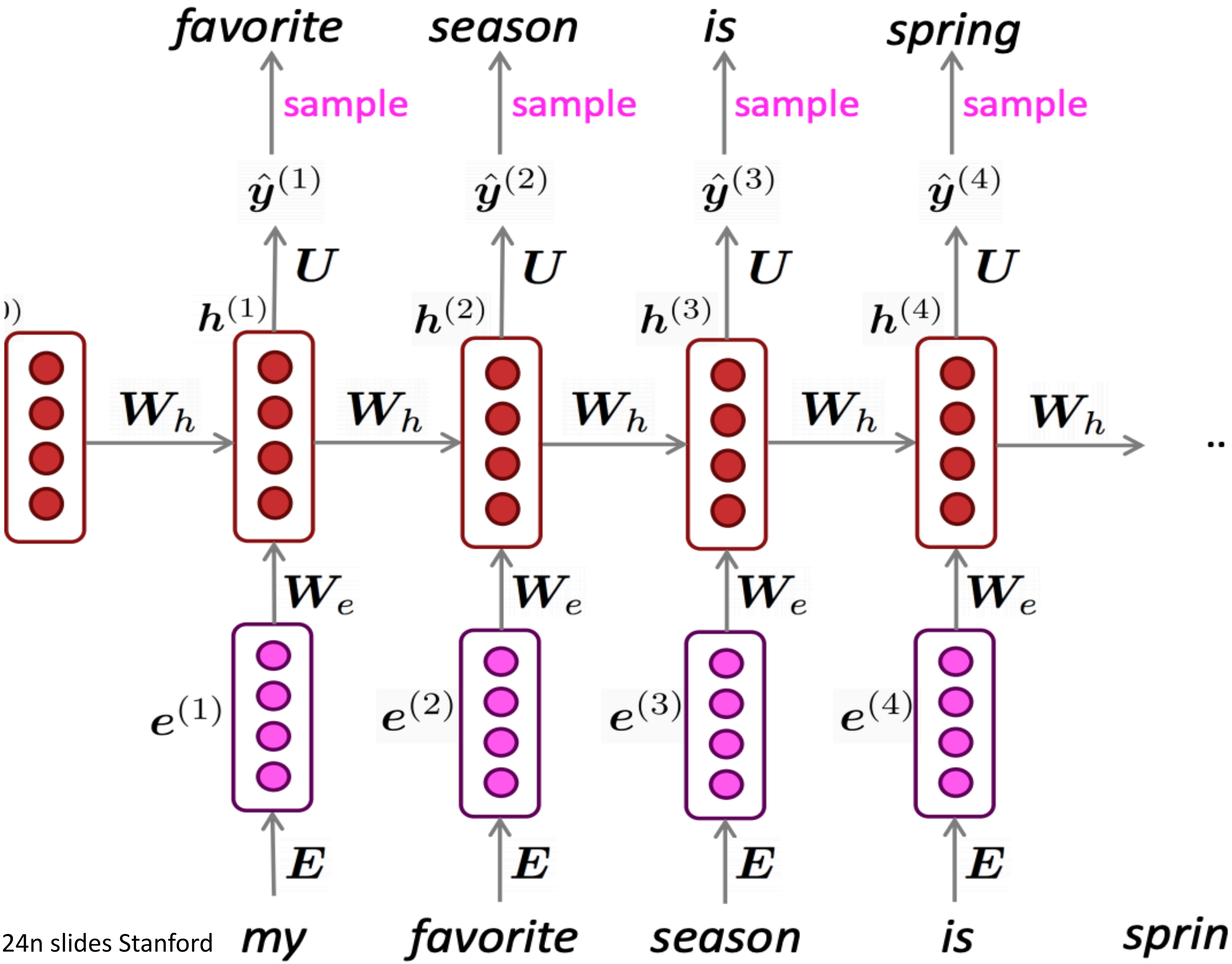
Text Representation using RNN

- Run the entire text through a RNN sequentially (word by word)
- Ignore the outputs
- Use the last state vector as the text representation



Text Generation using RNN

- Starting with an initial state (seed)
- The output of each step will be the input of the next step
- Continue until the output is ”End of Text”.



Source: cs224n slides Stanford

Text Generation Example

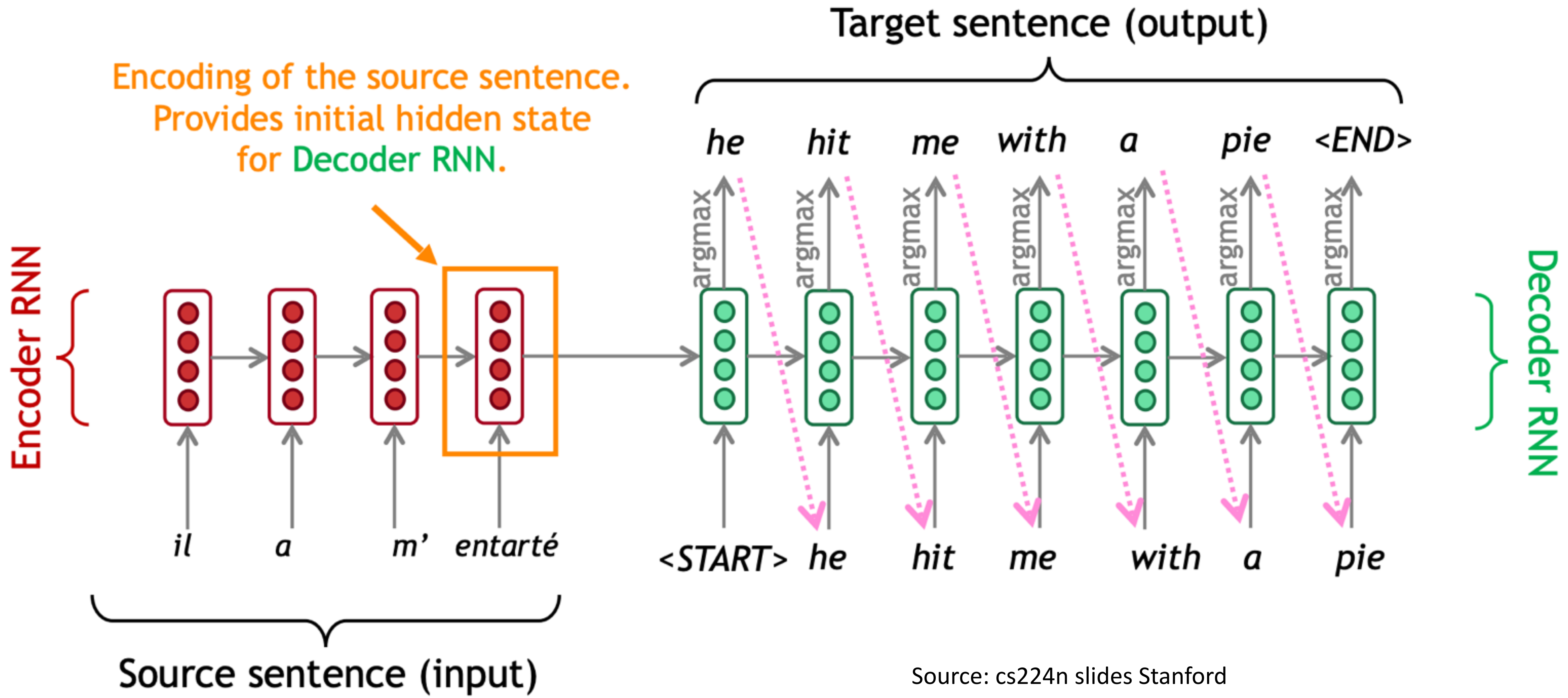
OBAMA-RNN. (source: https://medium.com/@samim/obama-rnn-machine-generated-political-speeches-c8abd18a2ea0)

SEED: Jobs

“The United States will step up to the cost of a new challenges of the American people that will share the fact that we created the problem. They were attacked and so that they have to say that all the task of the final days of war that I will not be able to get this done. The promise of the men and women who were still going to take out the fact that the American people have fought to make sure that they have to be able to protect our part. It was a chance to stand together to completely look for the commitment to borrow from the American people. And the fact is the men and women in uniform and the millions of our country with the law system that we should be a strong stretcks of the forces that we can afford to increase our spirit of the American people and the leadership of our country who are on the Internet of American lives.”

Neural Machine Translation (NMT)

- Machine translation using sequence-to-sequence (a.k.a. seq2seq) architecture using <u>two RNNs</u> (encoder and decoder)



Source: cs224n slides Stanford

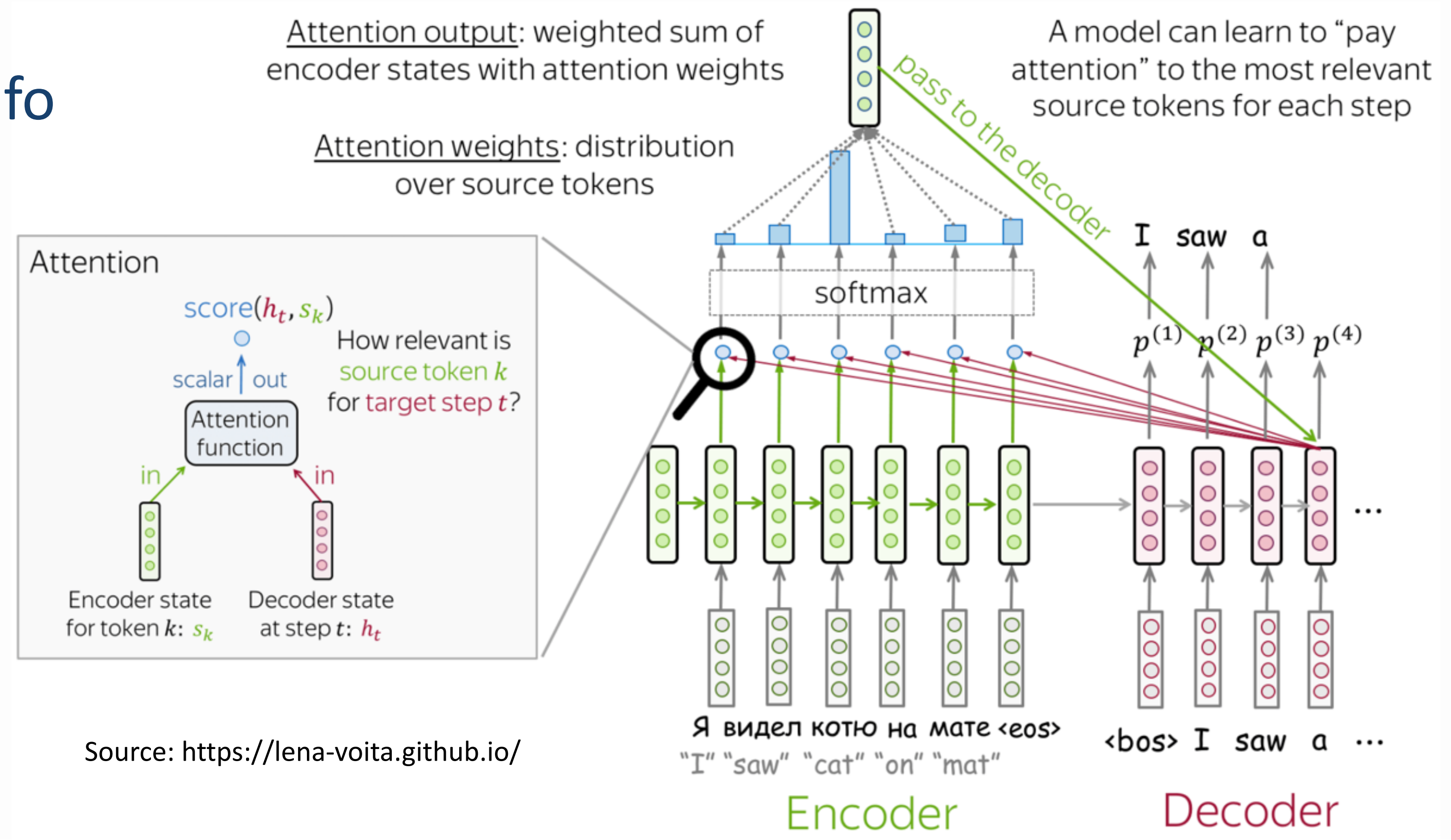
Seq2seq beyond NMT

- NMT is one of the biggest success stories of Modern NLP
- Google Translate switched to NMT in 2016
- There have been many improvements to the original seq2seq NMT system

- The same model (encoder-decoder) is used in many other applications:
  - Dialog system or Chatbot (previous utterance -> next utterance)
  - Image captioning (image representation -> caption)
  - Text summarization (long text -> short text)
  - Code generation (natural language -> programming code)
  - Sentence parsing (input text → output parse as sequence)
  - And many others

Seq2seq with Attention

- Problem with regular seq2seq model:
  - The encoder needs to capture all information about the source sentence.
  - Hard for the encoder
  - Decoder needs relevant info for each step
- Attention:
  - Significantly improves NMT performance.
  - Provides some interpretability
  - is a general and common Deep Learning technique



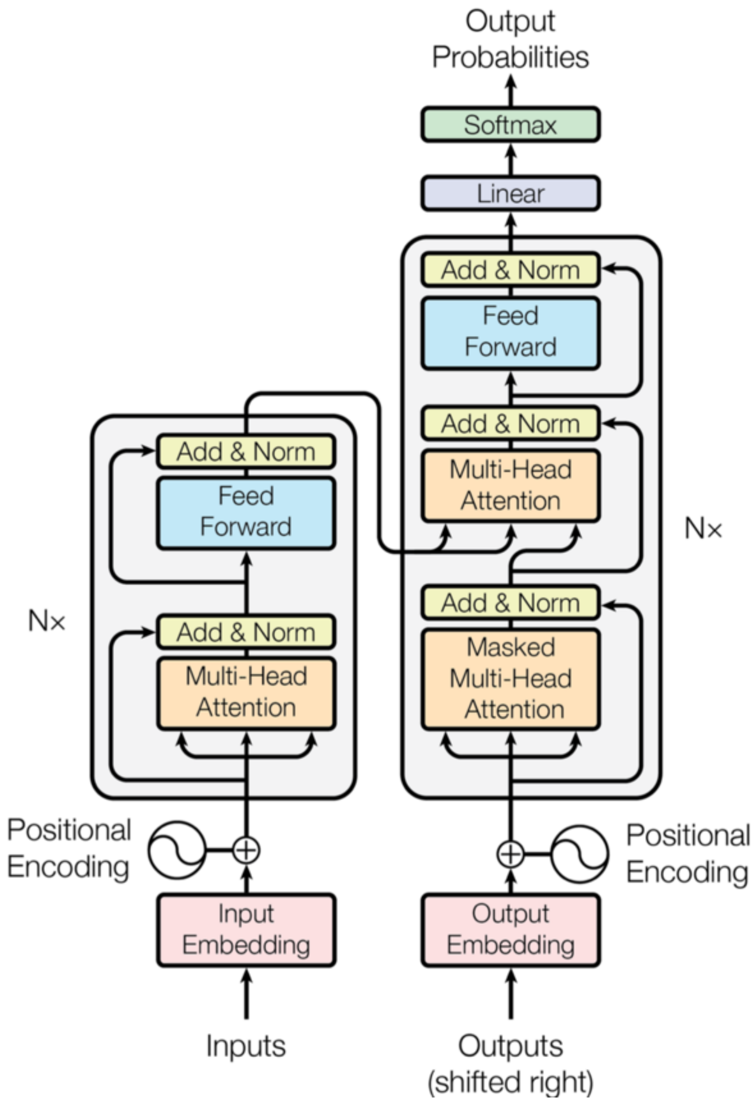
Attention output: weighted sum of encoder states with attention weights
A model can learn to "pay attention" to the most relevant source tokens for each step
pass to the decoder
Attention weights: distribution over source tokens
softmax
Attention
score(h_t, s_k)
scalar
out
How relevant is source token k for target step t?
Attention function
in
in
Encoder state for token k: s_k
Decoder state at step t: h_t
I saw a
$p^{(1)}$ $p^{(2)}$ $p^{(3)}$ $p^{(4)}$
...
Я видел котю на мате <eos>
"I" "saw" "cat" "on" "mat"
<bos> I saw a ...
Encoder
Decoder

Source: https://lena-voita.github.io/

Transformers (Vaswani et al. 2017)

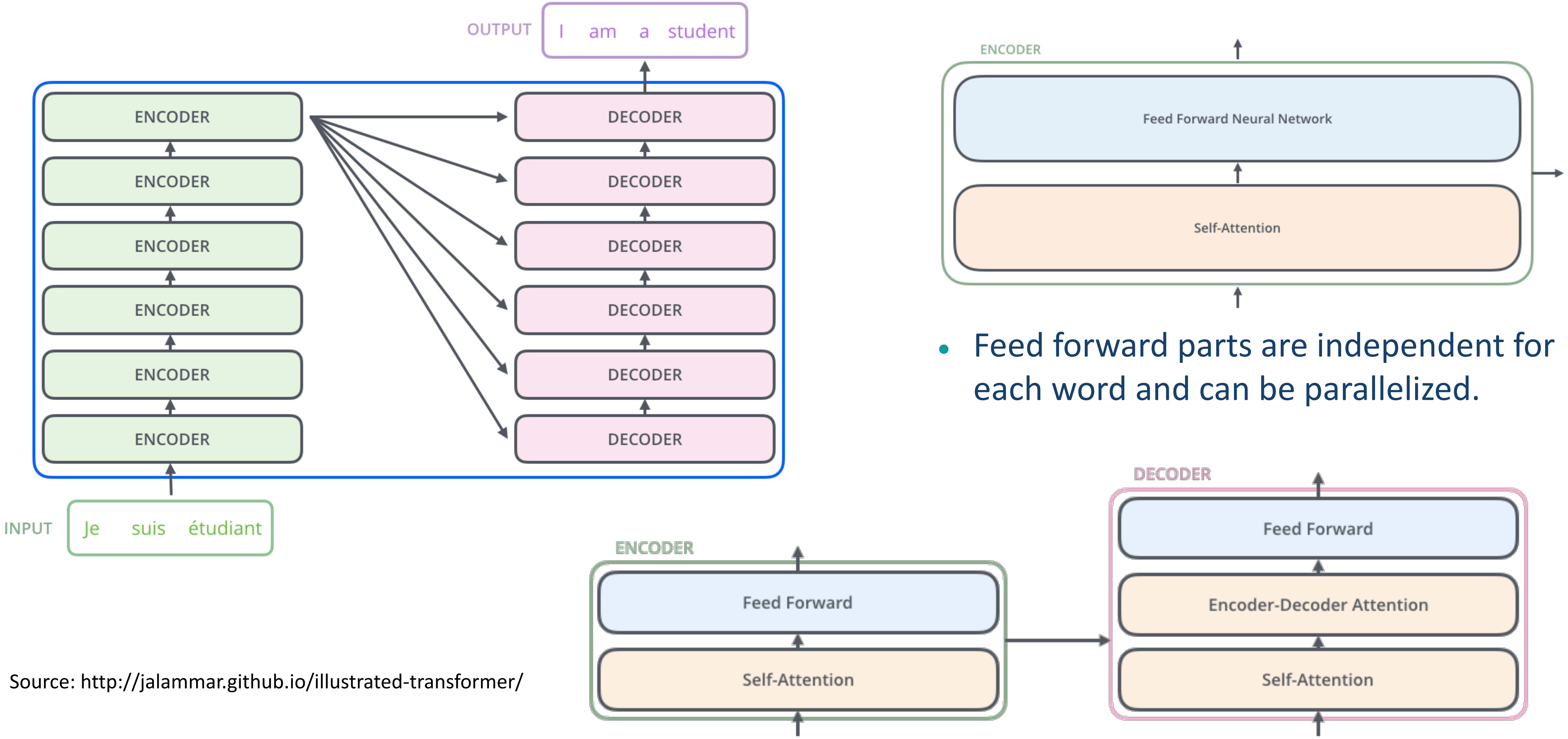
- We successfully used RNNs for both encoder and decoder in seq2seq models. But RNNs are slow to train because they are inherently sequential. We need parallelization.
- Also, RNNs (even LSTMs) become inefficient when there is a long gap between related information.
- In addition, RNNs generally need attention mechanism to deal with long range dependencies.
- Perhaps, “All we need is attention” not RNNs.

- **Transformer**: a non-recurrent encoder-decoder model fully based on attention



Source: cs224n slides Stanford

Transformer Architecture



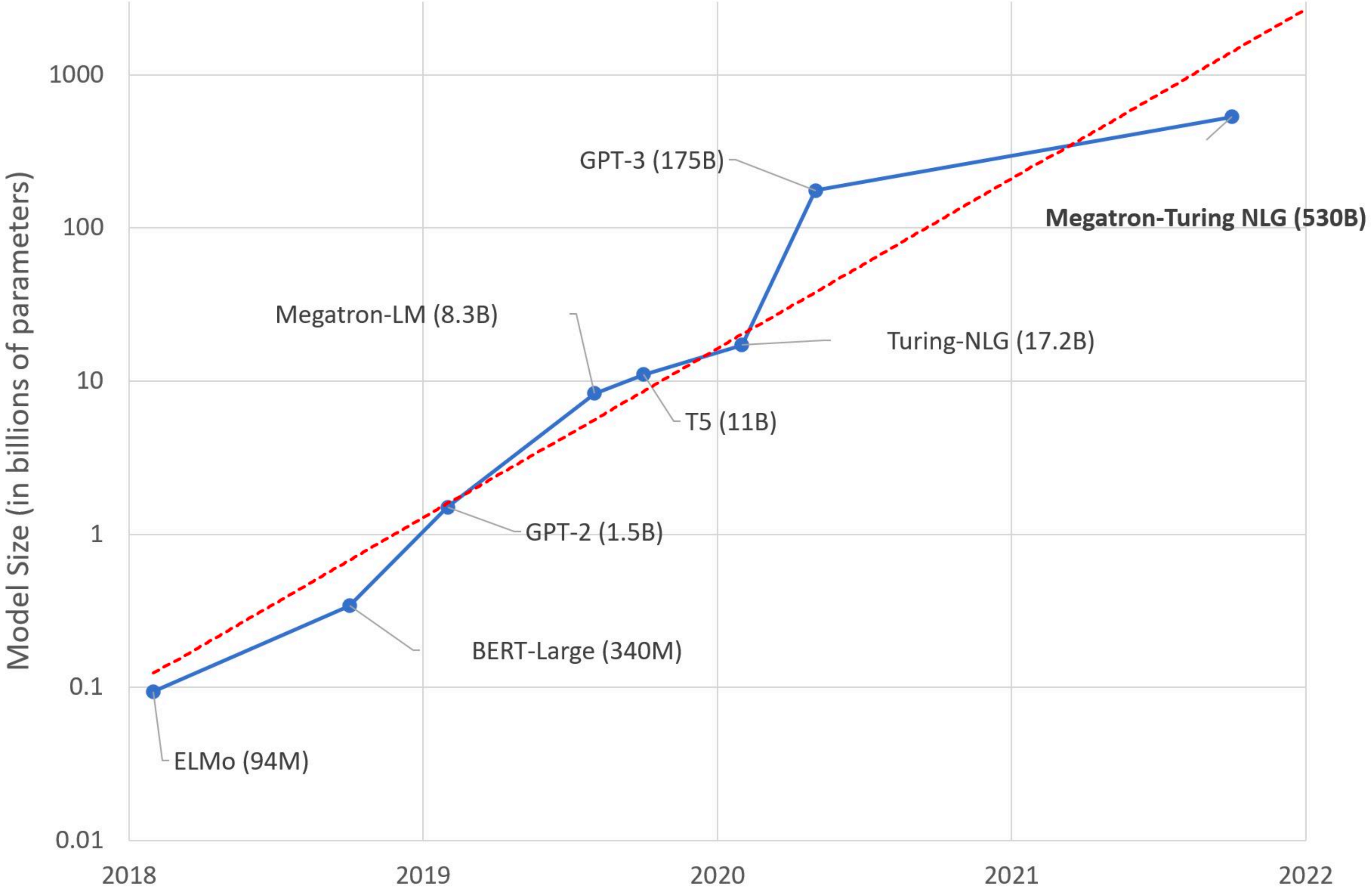
- Feed forward parts are independent for each word and can be parallelized.

Source: http://jalammar.github.io/illustrated-transformer/

BERT: Pre-training of Deep Bidirectional Transformers (Google AI 2018)

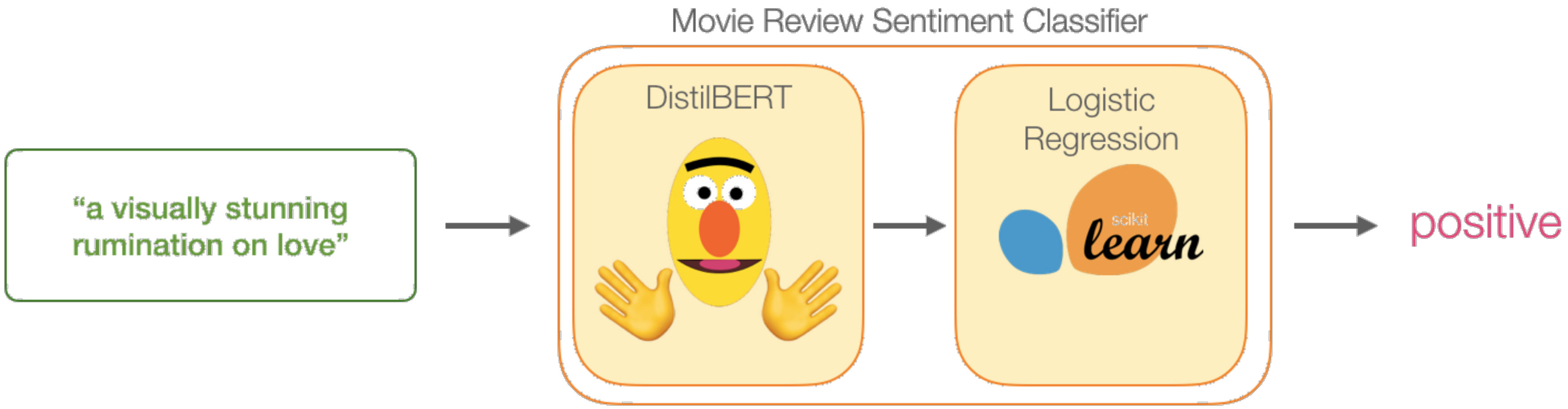
- A novel Transformer-approach, pre-trained on large corpora and open-sourced that can be fine-tuned for any specific NLP task to get high accuracy.
- One of the biggest challenges in NLP is the lack of enough training data. Pre-trained language representation models like BERT are the key to get good results on small task-specific datasets.
- BERT was trained on Wikipedia and Book Corpus, a dataset containing +10,000 books of different genres.
- BERT in a contextual model (word representation is based on the other words in the sentence) and bidirectional. Example: “bank account” vs. “bank of the river”.
- The best part about BERT is that it can be download and used for free.

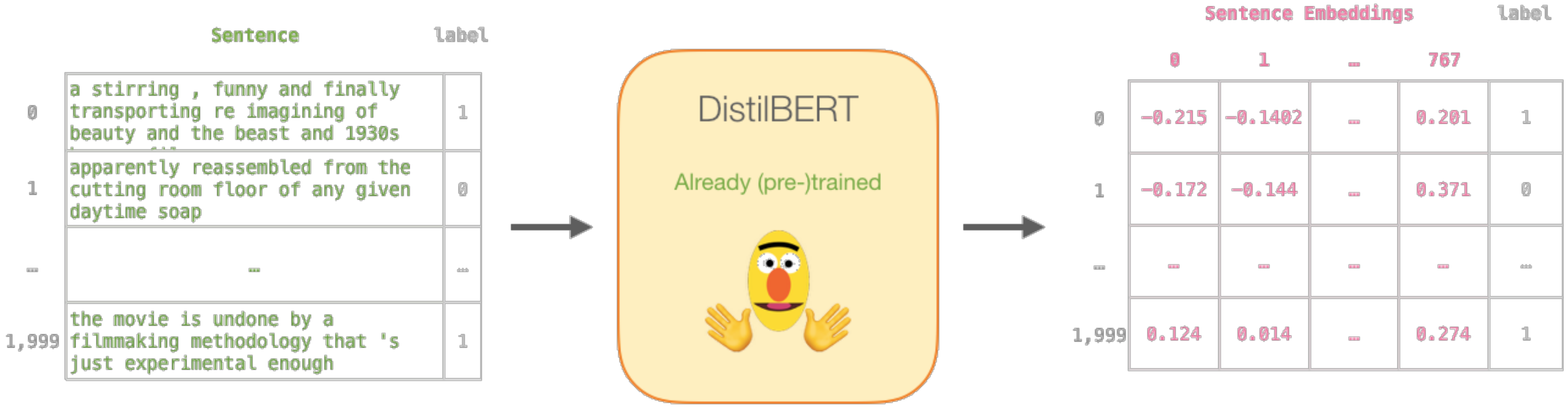
Pre-trained Language Models



Source: Microsoft Research Blog

Text Classification with BERT


Movie Review Sentiment Classifier
"a visually stunning rumination on love"
DistilBERT
Logistic Regression
scikit learn
positive

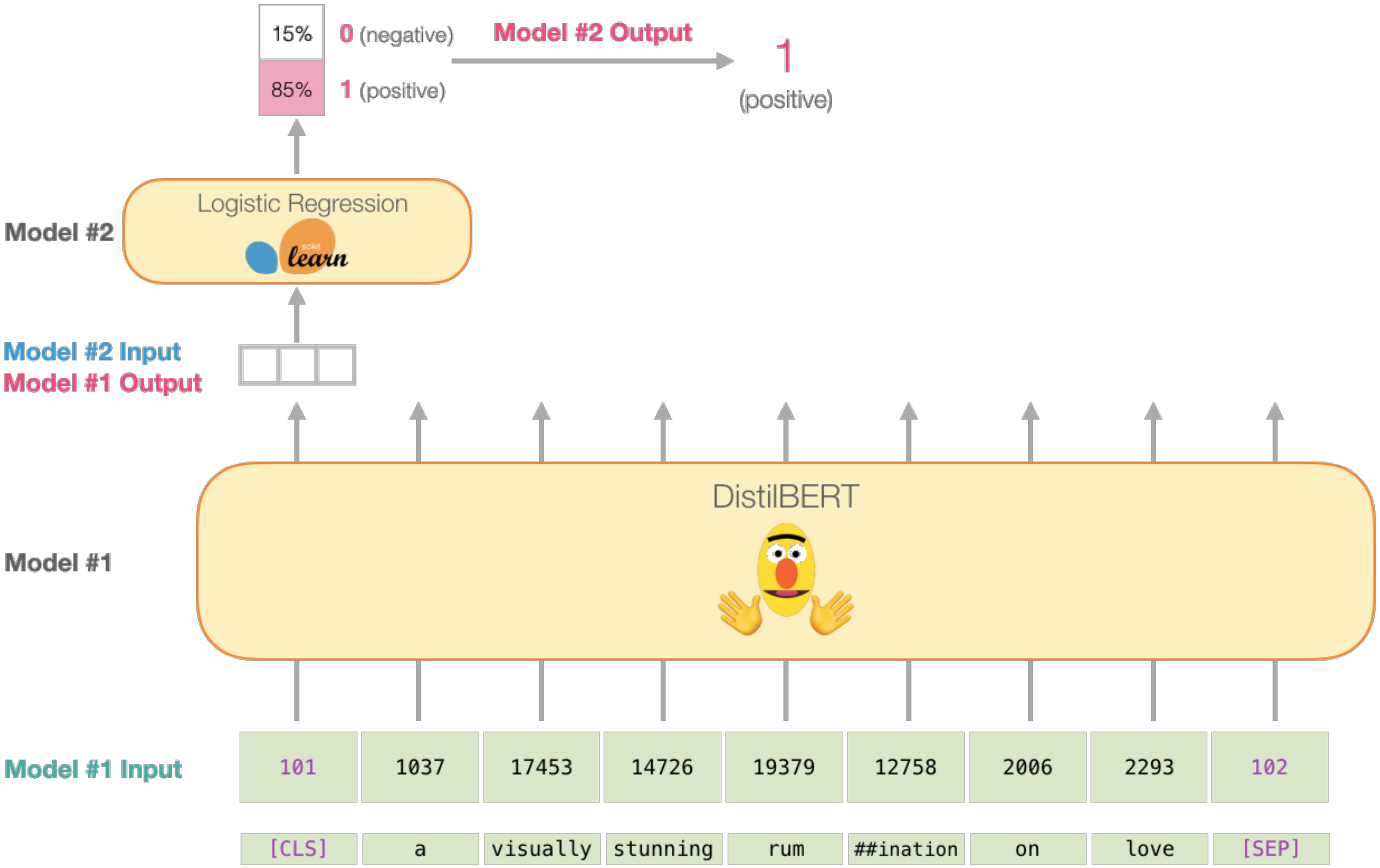


	Sentence	label
0	a stirring , funny and finally transporting re imagining of beauty and the beast and 1930s	1
1	apparently reassembled from the cutting room floor of any given daytime soap	0
…	…	…
1,999	the movie is undone by a filmmaking methodology that 's just experimental enough	1

	Sentence Embeddings				label
	0	1	…	767	
0	-0.215	-0.1402	…	0.201	1
1	-0.172	-0.144	…	0.371	0
…	…	…	…	…	…
1,999	0.124	0.014	…	0.274	1

Source: http://jalammar.github.io/a-visual-guide-to-using-bert-for-the-first-time/

Text Classification with BERT



Source: http://jalammar.github.io/a-visual-guide-to-using-bert-for-the-first-time/

GPT: Generative Pre-trained Transformer (OpenAI 2018)

Please see the assignment.

VECTOR INSTITUTE