



CMPT 413/713: Natural Language Processing

Conclusion

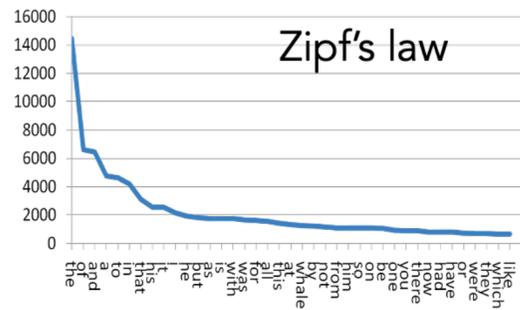
Spring 2025
2025-04-09

What is natural language?

- Way for humans to **communicate** with each other
 - Use it to express feelings, transmit information, and store **knowledge**
- **Discrete**, combinations of symbols
- **Learnable** by all humans

Key challenges of understanding language

- Ambiguity (at multiple levels)
- Scale
- Sparsity (Long tail)
- Variation
- Expressivity
- Unmodeled Variables (Implicit knowledge / Context)
- Unknown representation (representation)



I saw her duck



What has this course covered?

Representations

- Embeddings
- Compositional
- Structured

Methods

- Language Models
- Statistical Methods
- Neural Models
- Dynamic programming

LLMs

- Emergent capabilities
- Prompting
- Few-shot / In-context learning
- Efficient fine-tuning

Applications and tasks

- Machine translation, text generation, question answering, grounding

What has this course covered?

Representations

- Embeddings
- Compositional
- Structured

Methods

- Language Models
- Statistical Methods
- Neural Models
- Dynamic programming

LLMs

- Emergent capabilities
- Prompting
- Few-shot / In-context learning
- Efficient fine-tuning

Applications and tasks

- Machine translation, text generation, question answering, grounding

Key Questions

- How to **represent** language?
- How to **compose** meaning?
- How do we **parse** the “surface form” of language to a representation that a computer can process?
- How do we take an encoded representation and **generate** text from it?
- What are the key methods / algorithms for **learning** these representations?

Methods we covered

Rule-based methods

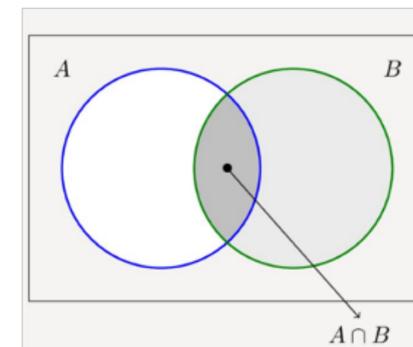
(O YOU O ME) [pattern]
→
(WHAT MAKES YOU THINK I 3 YOU) [transform]

You hate me

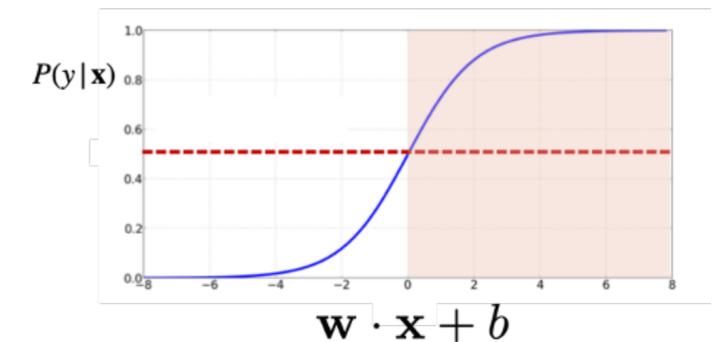
WHAT MAKES YOU THINK I HATE YOU

Use when you have no data
Good way to have something to bootstrap from

Statistical methods



Naive Bayes

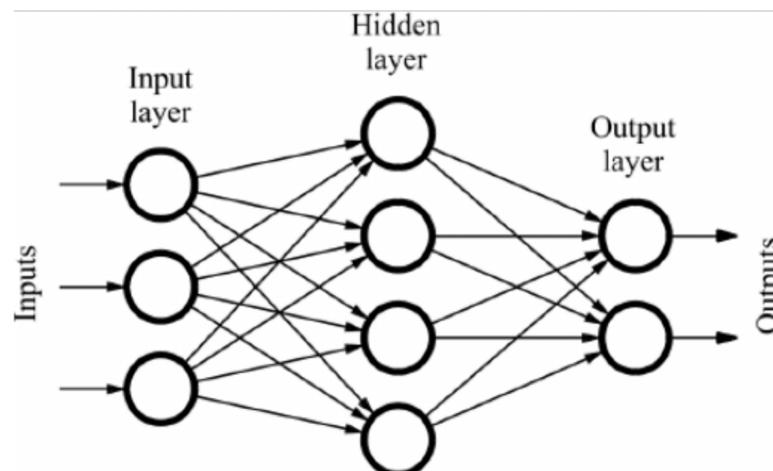


Logistic regression

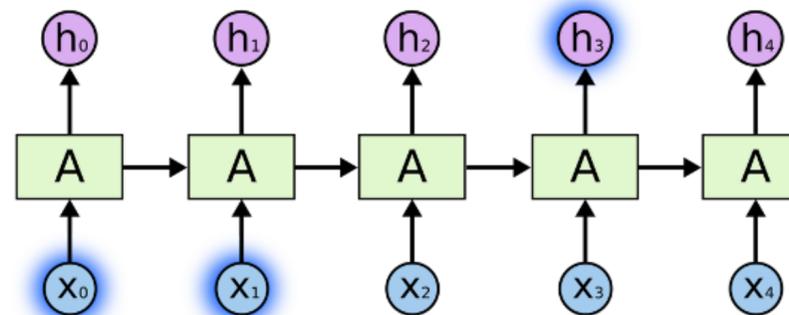
Simple baselines for text classification
Solid understanding helps understand NNs

Neural networks for NLP

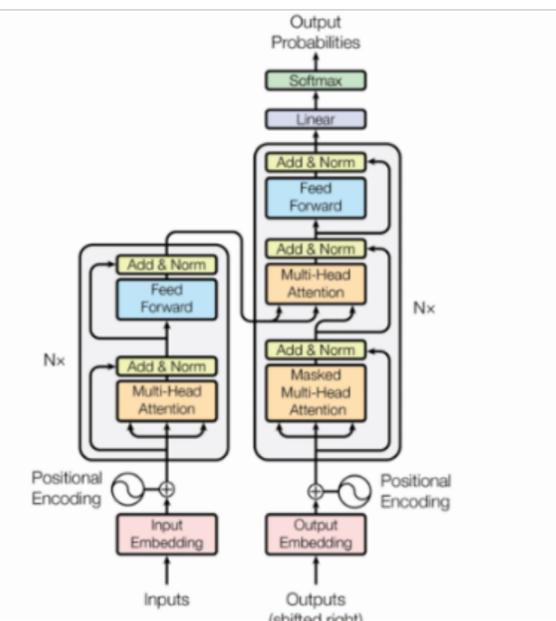
Feed-forward NNs



Recurrent networks (RNNs)

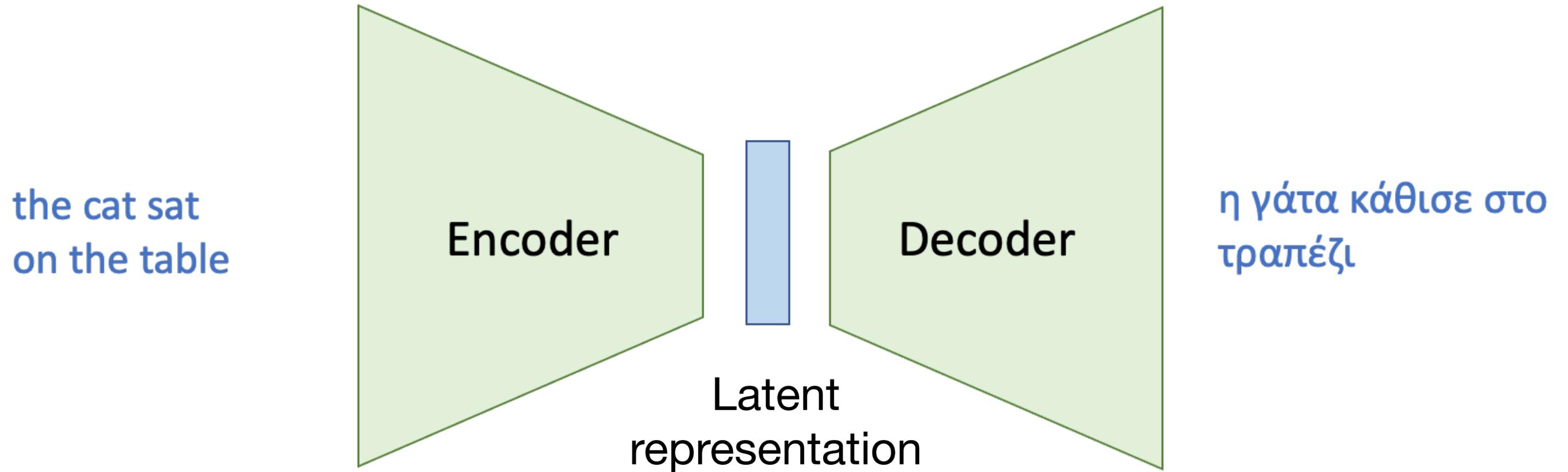


Transformers



Data hungry, compute intensive → Use pretrained models and then fine-tune on limited data

Encoder-Decoder Model



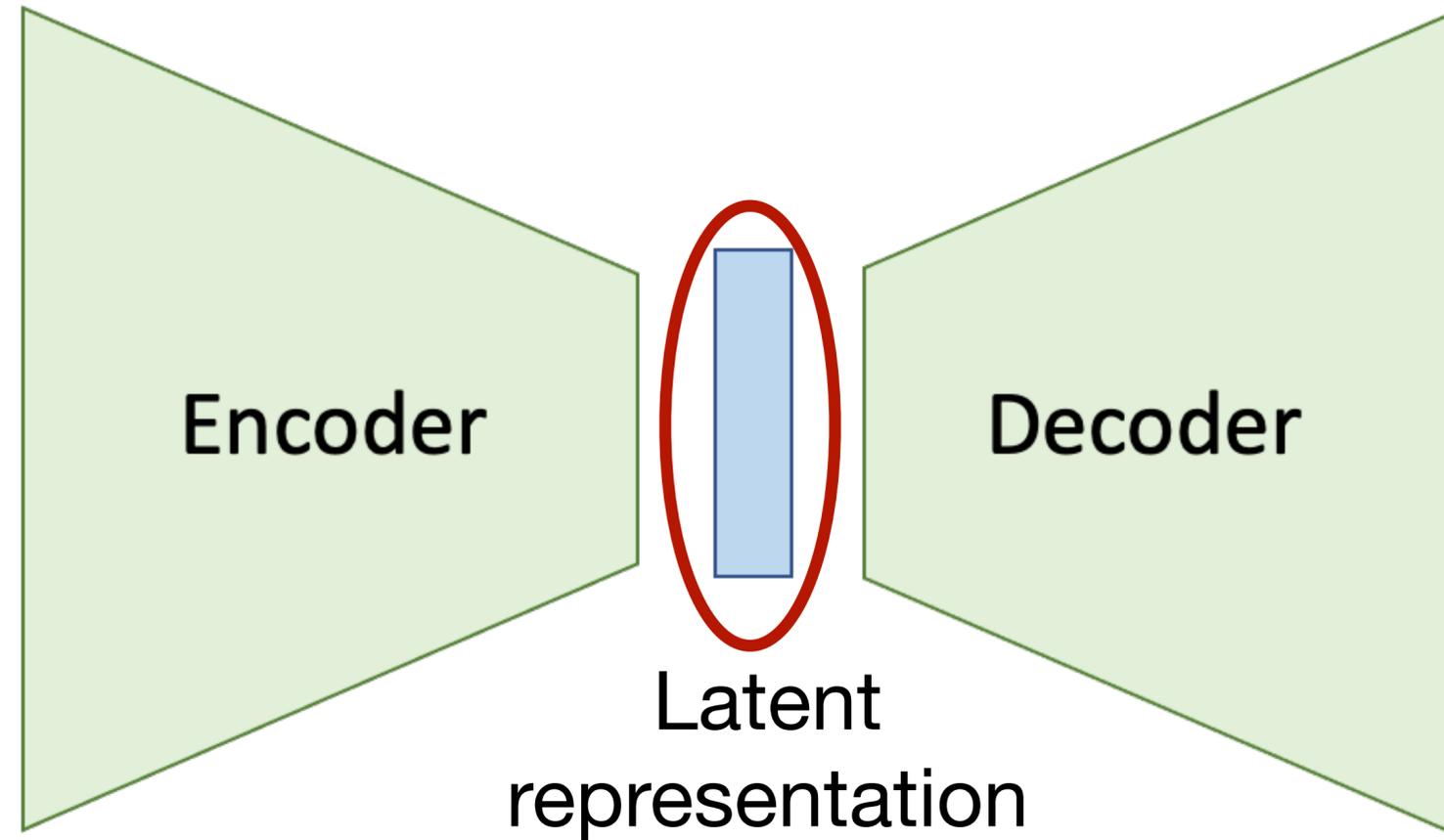
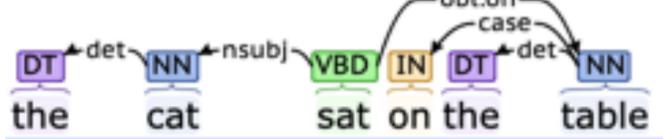
Understanding what is said
(encoding, parsing, feature extraction)

Deciding what to say
(decoding, generating)

How to represent language?

the cat sat
on the table

Dependency parse



η γάτα κάθισε στο
τραπέζι

Image

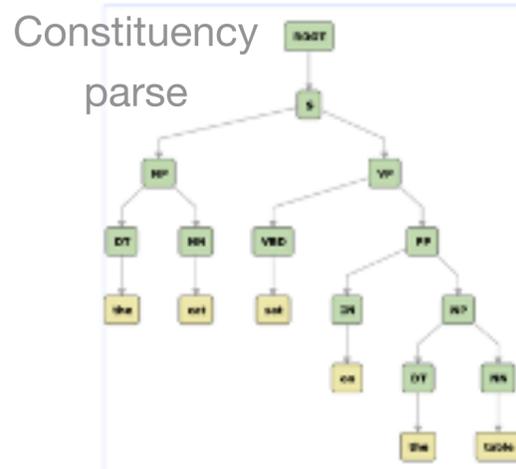


shutterstock.com - 629994974

What kind of representation should go here?
Structured? Unstructured? Grounded?

$$L = \begin{bmatrix} \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ |V| & & & & & \end{bmatrix}_n \text{ Tensor}$$

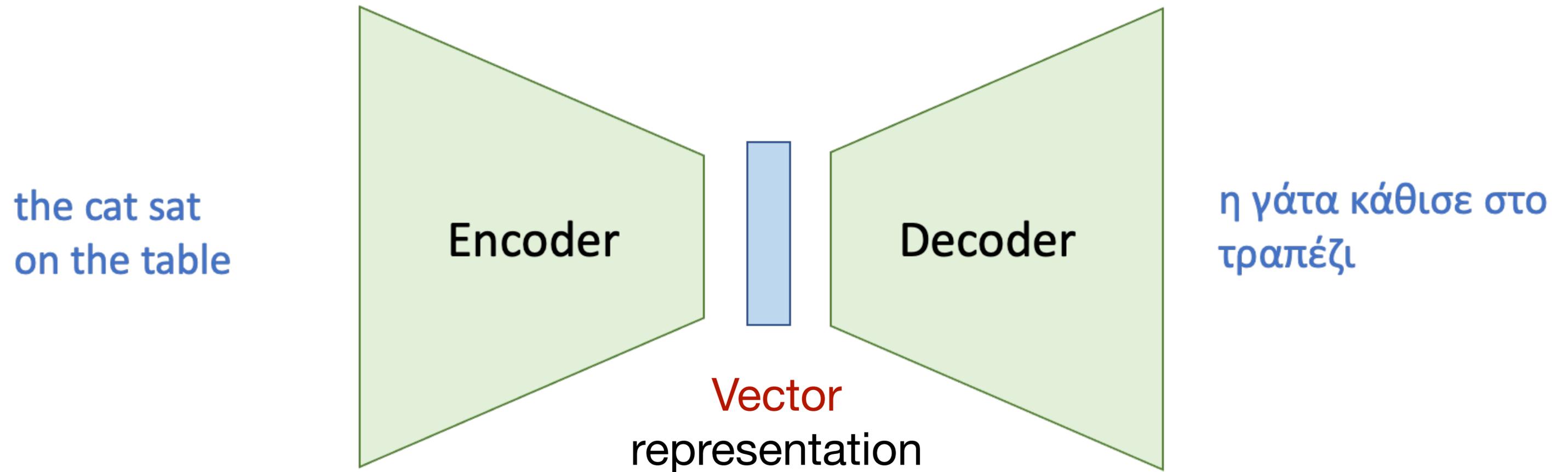
the cat mat ...



Lambda expression:

$((\lambda x . (\lambda y . x)) (\lambda z . z)) (\lambda a . a)$

Vector based representation



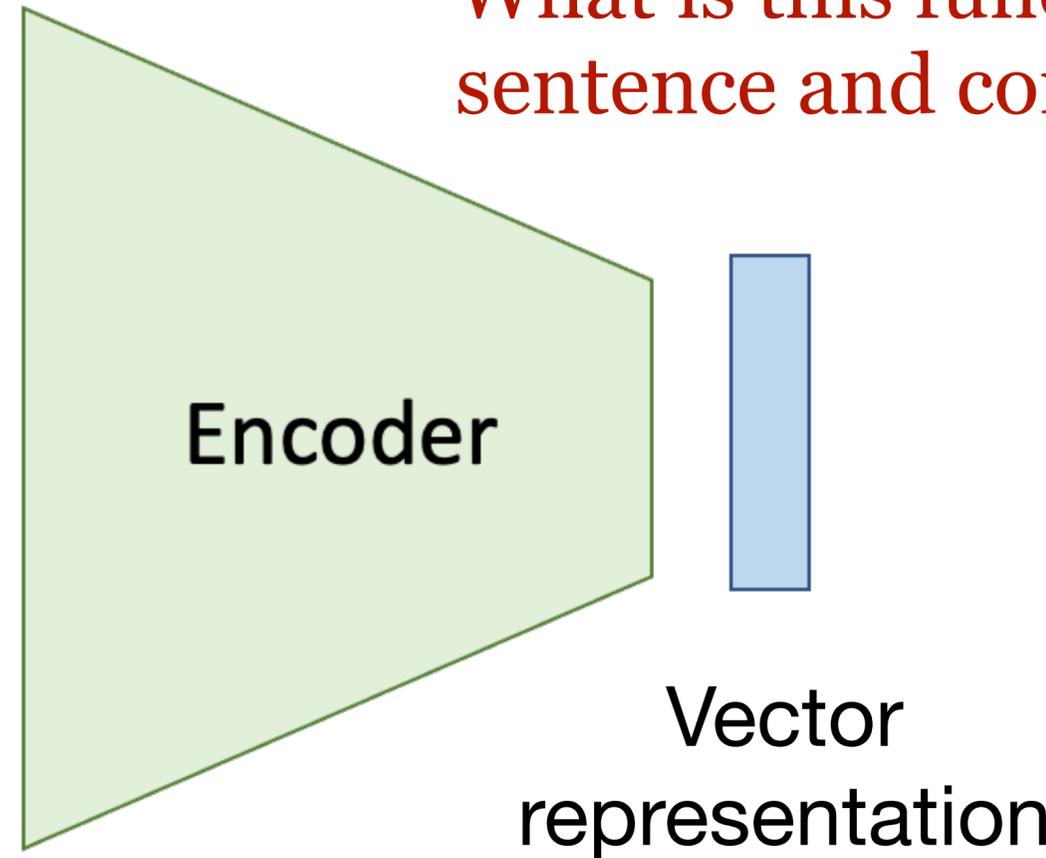
Common choice: flat unstructured vector of real numbers

- easy to work with
- well defined mathematical operations
- can be used to measure if two things are similar or not

How to **encode** language?

What is this function that takes the sentence and converts it into a vector?

the cat sat
on the table

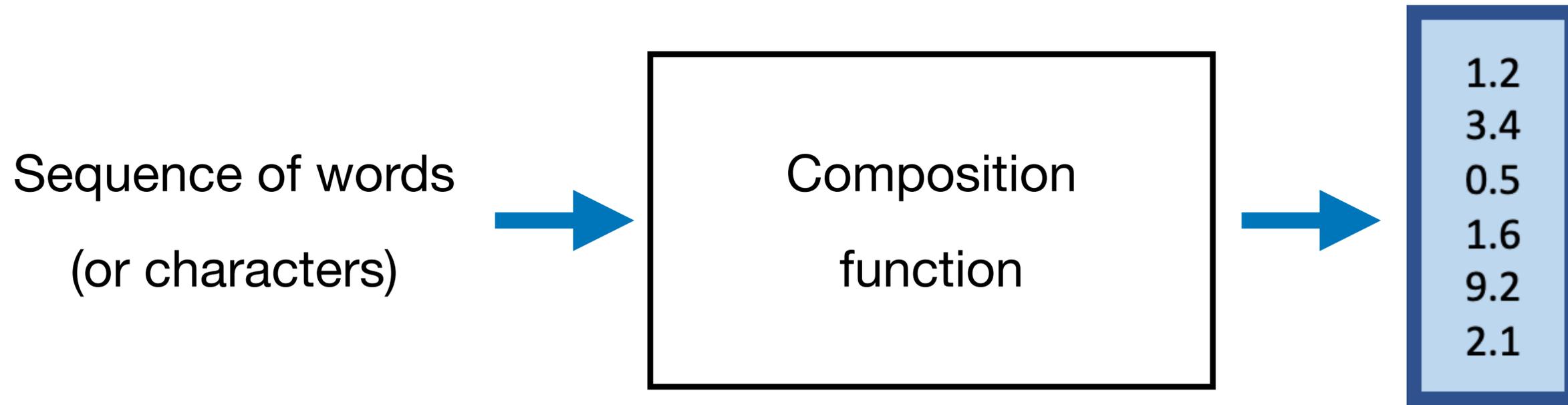


Properties we want

- Be able to represent a single word
- Be able to represent the meaning of a sentence as a composition of the meaning of each word

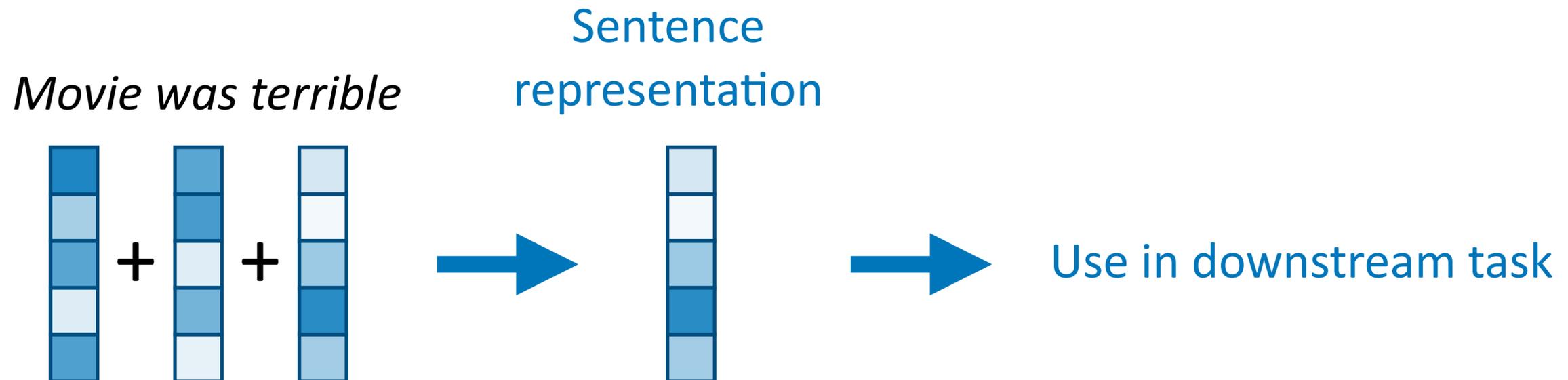
Understanding what is said
(encoding, parsing, feature extraction)

How to **compose** language?

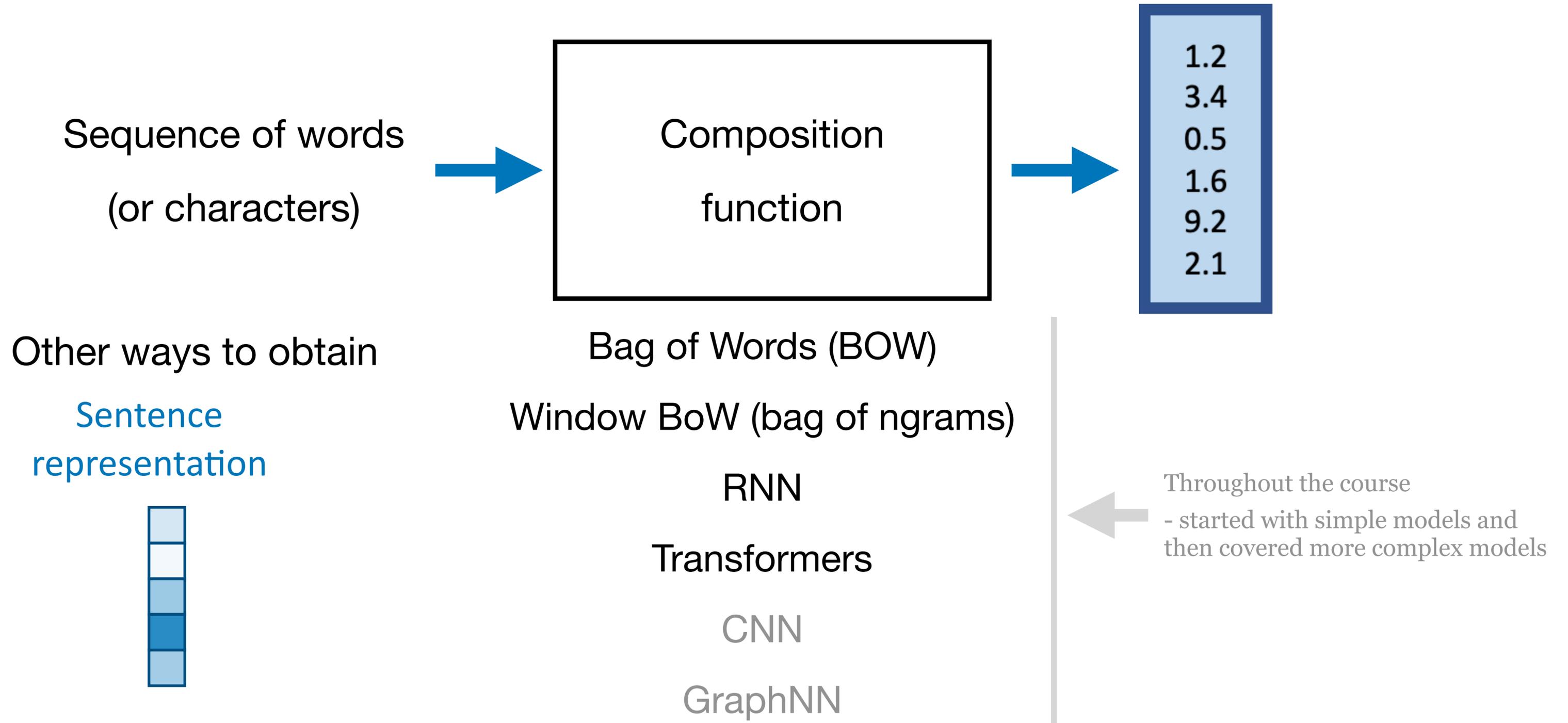


Simplest: take weighted average

Example:

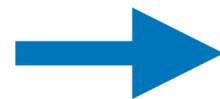


How to **compose** language?



How to use the learned **representation**?

What do we do with this vector representation after we get it?



Use in downstream task

Vector
representation

Do prediction: classify as spam/not spam

Use to generate text (also known as **decoding**)

Can also be used to generate other stuff (like images!)

How to **decode** language?

How do we generate text from this?

Auto-regressive text-generation

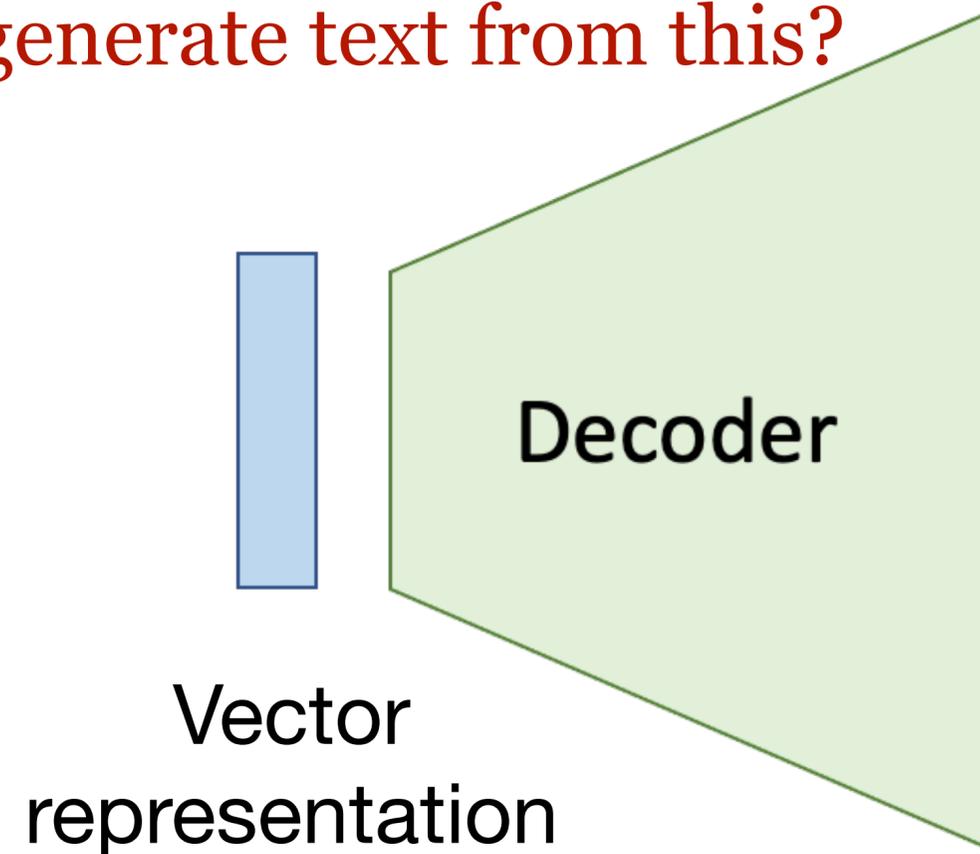
- Predict one word at time
- Big classification problem
- Can predict word from vocabulary or copy from somewhat else

Different training strategies

- Maximum likelihood estimation (cross-entropy loss for classification)

Different decoding strategies

- Greedy, Beam search, Sampling



Deciding what to say
(decoding, generating)

What has this course covered?

Representations

- Embeddings
- Compositional
- Structured

Methods

- Language Models
- Statistical Methods
- Neural Models
- Dynamic programming

LLMs

- Emergent capabilities
- Prompting
- Few-shot / In-context learning
- Efficient fine-tuning

Applications and tasks

- Machine translation, text generation, question answering, grounding

Rise of large pretrained models

FRONTIER AI MODELS + HIGHLIGHTS (MAR/2025)



poe.com



Microsoft phi
Google Gemma
IBM Granite
Mistral

+ hundreds more...



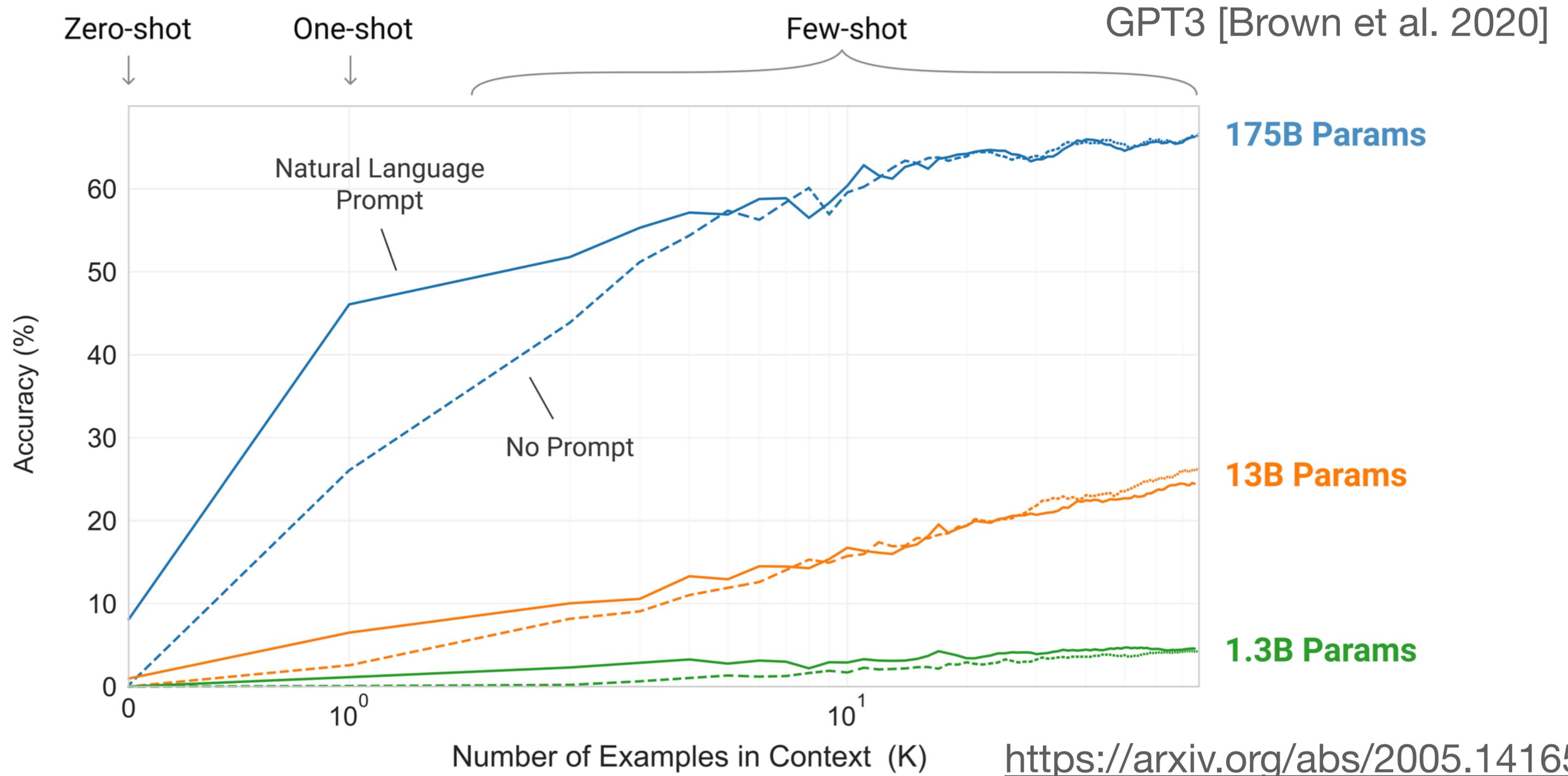
DeepSeek R
Cohere Command-R
AI21 Jamba
Alibaba Qwen/QwQ

+ hundreds more...

Some images by Flaticon.com. Selected highlights only. All 500+ models: <https://lifearchitect.ai/models-table/> Alan D. Thompson. 2021-2025.

<https://lifearchitect.ai/models>

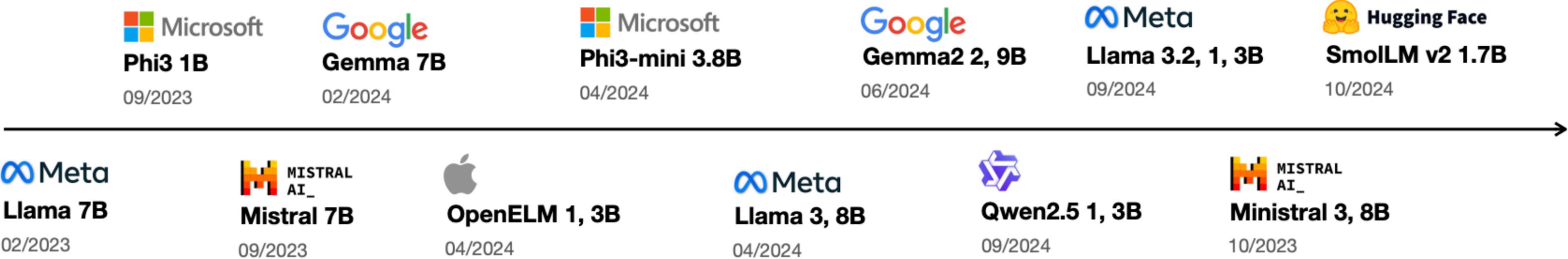
Emergence of capabilities in LLMs



SoTA methods: can use zero-shot/few-shot prompting, or fine-tune

Small language models

- SLM: language models <10B



Reduce model parameters and size through

- Pruning
- Distillation
- Quantization

Post-training

Pre-training

LM training on large, large amount of data

Fine-tuning

Supervised fine-tuning for instructions

Preference optimization

Align to human preferences

Model compression

Reduce size of model for efficient deployment

Task specific fine-tuning

Custom domains, improved performance on specialized tasks

Prompting

Generate responses

Use to build LLM agents

What has this course covered?

Representations

- Embeddings
- Compositional
- Structured

Methods

- Language Models
- Statistical Methods
- Neural Models
- Dynamic programming

LLMs

- Emergent capabilities
- Prompting
- Few-shot / In-context learning
- Efficient fine-tuning

Applications and tasks

Limited coverage of applications

- Machine translation, text generation, question answering, grounding

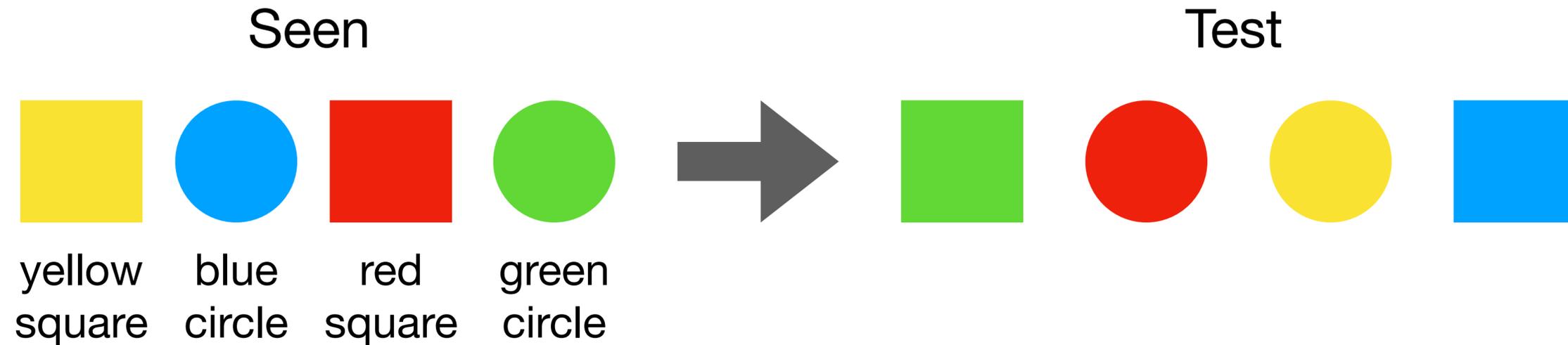
Topics in NLP research

- Dialogue and Interactive Systems
- Discourse and Pragmatics
- **Generation** and Summarization
- Information Extraction
- Question Answering
- Information Retrieval
- Language Resources and **Evaluation**
- **Language and Vision**
- Linguistic and Psycholinguistic Aspects of CL
- **Machine Learning for NLP**
- **Machine Translation**
- NLP for Web, Social Media and Social Sciences
- NLP-enabled Technology
- Phonology, Morphology and **Word Segmentation**
- Semantics
- Sentiment Analysis and Opinion Mining
- Spoken Language Processing
- **Tagging, Chunking, Syntax and Parsing**
- **Text Categorization** and Topic Models

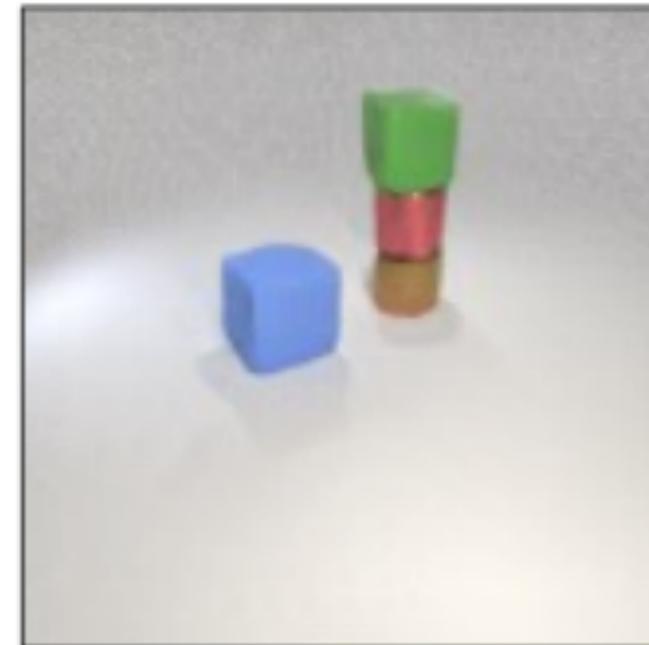
Remaining challenges and topics in NLP

- Understanding and interpreting these large data-driven models
- Compositional generalization
- Grounded natural language understanding
- Unsupervised and few-shot learning
- Language acquisition and emergence of language
- Building practical systems
- Improving evaluation
- Bias, trust, and ethics
- Efficient models
- Integrating domain-specific knowledge into NLP models

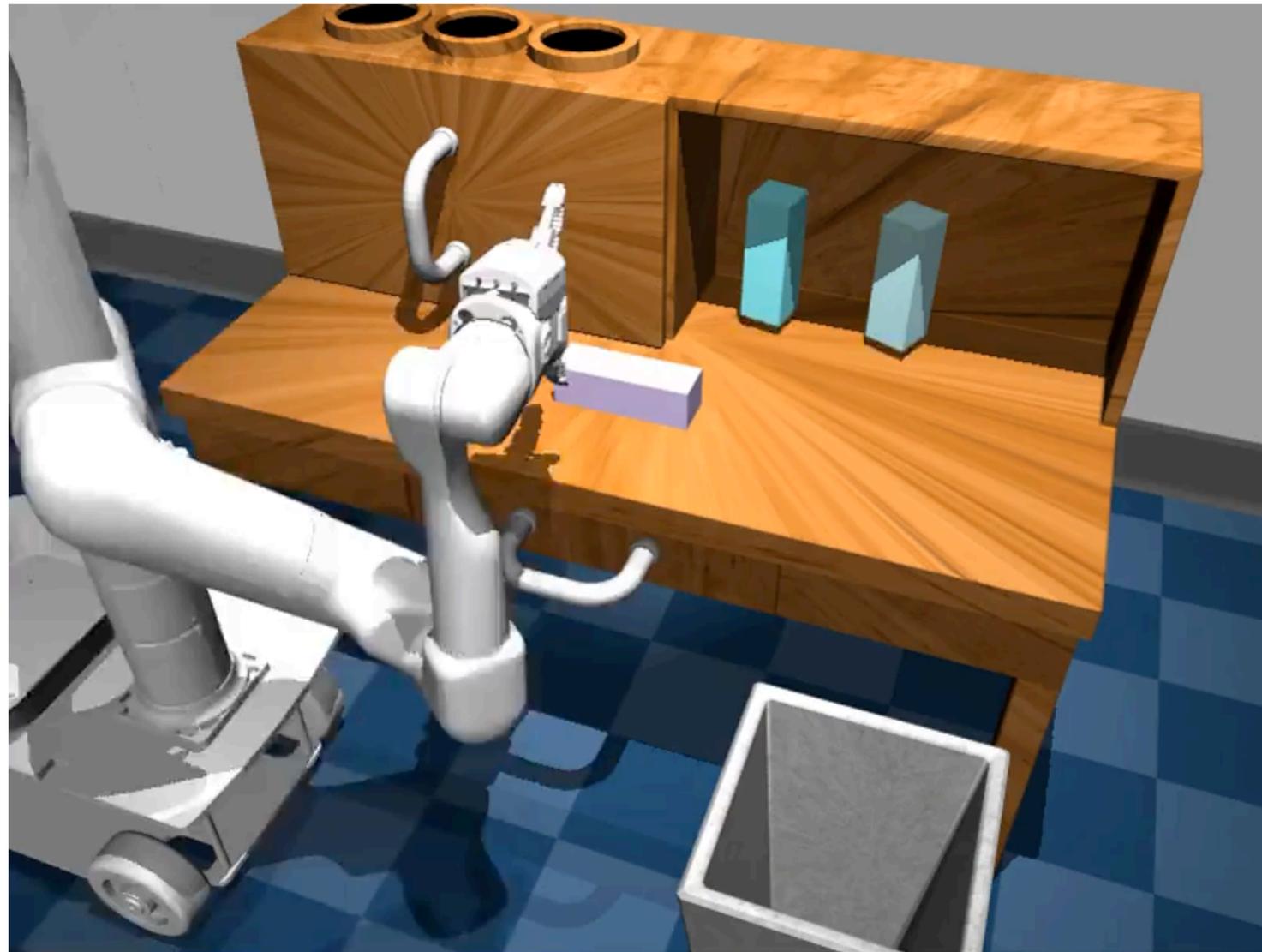
Compositional generalization



A small red metal cylinder *below*
a small green rubber cube
A small red metal cylinder *to the right of*
a large blue rubber cube
A small red metal cylinder *above*
a small brown metal cylinder



Grounded NLP / RoboNLP



now: **do not do anything**

next:

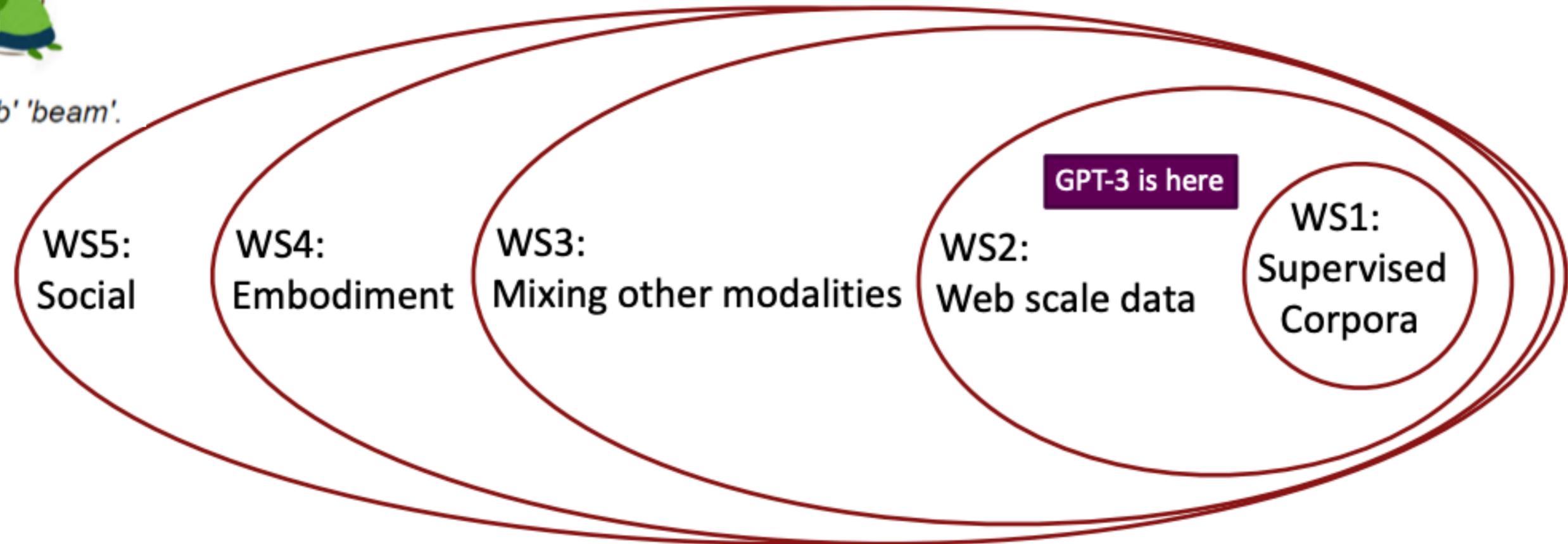
Language Conditioned Imitation Learning over Unstructured Data (<https://language-play.github.io/>)

Lynch and Sermanet, RSS 2021

Going beyond passive, supervised NLP



'block' 'pillar' 'slab' 'beam'.



*Experience Grounds Language (https://www.youtube.com/watch?v=cQasYLUC_00)
Bisk et al, EMNLP 2020*

More courses at SFU

- Advanced NLP: CMPT 839
 - Efficient training and fine-tuning of LLMs
 - Various techniques for text generation
 - Modeling long-sequences
 - Grounded natural language understanding
 - Other recent trends in NLP
- Special topics classes: CMPT 419/983
 - Human and Data-Centric AI with Nick Vincent
 - Trustworthy Deep Learning with Linyi Li
- Reinforcement learning (CMPT 729) with Jason Peng