

# Basics

## Learning Paradigms

The three settings we explore for in-context learning

### Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

Translate English to French: -- task description  
see otter --> loutre de mer -- example  
cheese --> -- prompt

### One-shot

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

Translate English to French: -- task description  
see otter --> loutre de mer -- example  
cheese --> -- prompt

### Few-shot

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

Translate English to French: -- task description  
see otter --> loutre de mer -- example  
peppercorn --> poivre noir -- example  
plush giraffe --> girafe peluche -- example  
cheese --> -- prompt

Traditional fine-tuning (not used for GPT-3)

### Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.

see otter --> loutre de mer -- example #1  
--  
peppercorn --> poivre noir -- example #2  
--  
plush giraffe --> girafe peluche -- example #N  
--  
cheese --> -- prompt

## Learning goals

- Understand the different learning paradigms
- Linguistic tasks vs. higher-level tasks

# CATEGORIZATION OF LEARNING

## **Disclaimer:**

- This categorization is rather coarse
- The list of paradigms is extendable
- Not everything is unambiguous, there might be overlap

## **Connection to tasks/data:**

- Given the task, some paradigms are more suitable
- Given the amount of data, a specific paradigm might be preferable
- Presence/Absence of labels makes certain paradigms (in)feasible

# CATEGORIZATION OF LEARNING

## **Distinction between:**

- Embedding texts
- Pre-training & fine-tuning a model
- Prompting
- Interaction & Generation
- Agents

# EMBEDDING

## Problem statement

- Words are discrete units composed of characters
- We can represent them (high-dimensional) one-hot vectors
- This makes it difficult/impossible to e.g. capture similarity between synonyms
- Documents can be represented as a vector of word occurrences (bag-of-words)

## Example (one-hot)

$$\vec{w}^{(\text{football})} = [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$\vec{w}^{(\text{basketball})} = [0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

# EMBEDDING

## Problems of one-hot embeddings

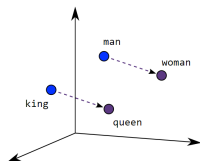
- high dimensionality
- not possible to measure similarity

## Example (dense embedding)

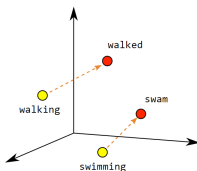
$$\vec{w}^{(\text{football})} = \begin{bmatrix} 0.359 \\ -0.174 \\ 0.701 \\ \vdots \\ 0.445 \\ -0.123 \\ 0.509 \end{bmatrix}$$

# EMBEDDING

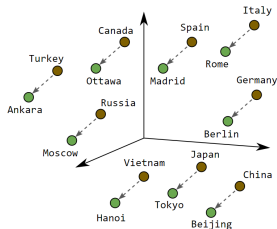
## Measuring similarity



Male-Female



Verb Tense

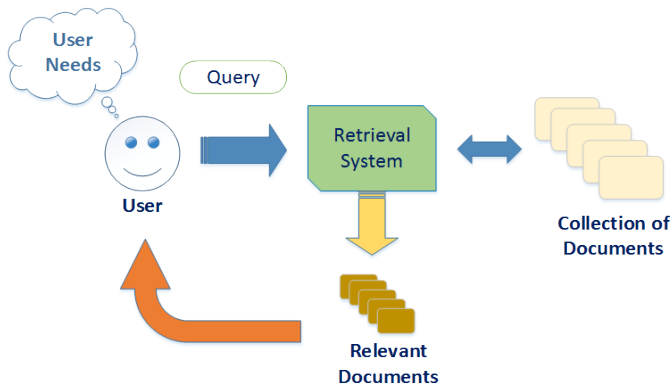


Country-Capital

► Source: Google

# EMBEDDING

## Document retrieval



► Source: Analytics Vidhya

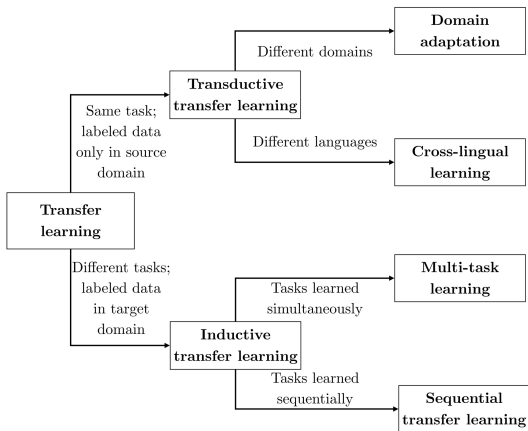
# PRE-TRAIN/FINE-TUNE

## Problem statement

- The larger the models, the more data is needed to train them
- (Labeled) Data is scarce and expensive!
- Many languages in the world are highly underrepresented in terms of existing resources  
Often: *Number of speakers  $\neq$  Amount of available written text*
- Unlabeled (English) text data is ubiquitous



# PRE-TRAIN/FINE-TUNE



► Taxonomy of transfer learning (Source: Ruder, 2019)

# PRE-TRAIN/FINE-TUNE

## Pre-training:

- Using unlabeled corpora in conjunction with self-supervised objectives is commonly referred to as *Pre-Training* the model
- Generation of samples for pre-training basically effortless, exploiting the inherent structure of the text
- Construction of different self-supervised objectives, which are assumed
  - to cover different phenomena better than the others
  - to work more efficiently for learning

# PRE-TRAIN/FINE-TUNE

## Fine-tuning:

- The second phase of transfer learning, i.e. adapting the pre-trained model to a labeled data set for a specific downstream task is referred to as *Fine-Tuning*
- Far less labeled data required compared to a scenario w/o pre-training

# PROMPTING

## Accessing pre-trained models:

- Fine-tune them
- Also possible: No fine-tuning, but ..
  - *Zero-Shot Transfer* w/o ANY labeled data
  - *Few-Shot Transfer* w/ FEW labeled data points
- In both of the latter cases, good pre-training becomes even more important

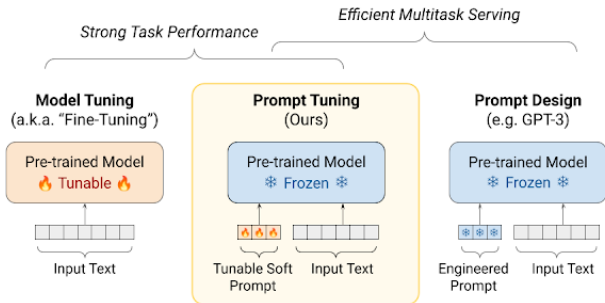
# PROMPTING

## Definition(s):

- *GPT-3 paper:*  
"Task Description" (accompanied by samples + labels)
- *Prompt:* Describing the task the model is expected to perform
- *Prompt Engineering:*  
Finding the best prompt(s) for one (or across multiple) task(s)
- *Prompt Tuning:*  
Add trainable weights ("soft prompt") to inputs and fine-tune

# PROMPTING

## Differences:



► Source: Google

# CHATting / GENERATION

## Interacting with the model

- Larger model sizes, reduced latency and improved training regimes enable conversations with the models
- Enables the user to ..
  - .. have multi-turn conversations, with the model "remembering" previous inputs
  - .. refine the prompt in case of unsatisfactory output
  - .. used increased context sizes for the prompts
- Still: Static, pre-trained model with "knowledge"

# CHATTING / GENERATION

## Interacting with the model

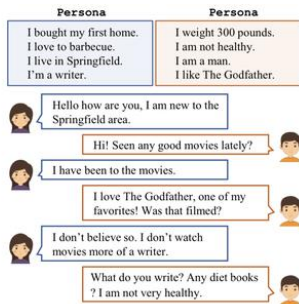


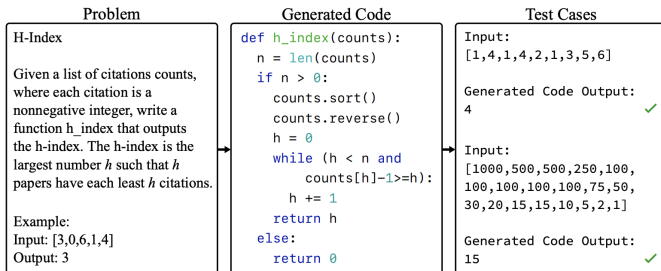
Figure 1: A clipped dialogue from PERSONA-CHAT.

► Source: Papers with code (example for Persona-Chat)



# CHATTING / GENERATION

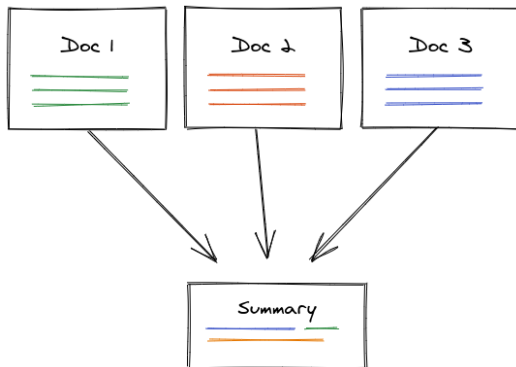
## Code generation



► Source: Papers with code

# CHATTING / GENERATION

## (Multi-)Document summarization



► Source: Aylien

# OUTLOOK

## Agents

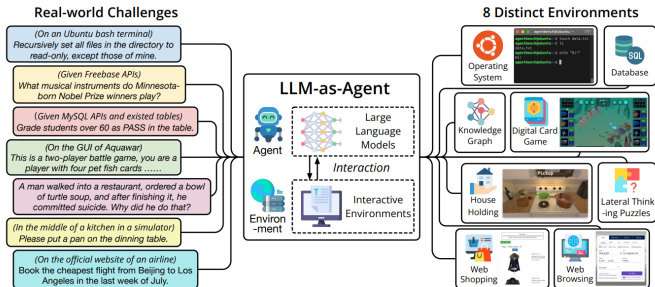


Figure 2: AgentBench is the first systematic benchmark to evaluate LLM-as-Agent on a wide array of real-world challenges and 8 distinct environments. In total, 25 LLMs are examined in its first edition.

► Source: Liu et al., 2023