Domain-Specific NLP

Contents

1. Domain-Specific Models
   a. *Don't Stop Pre-training*
   b. Specialized Models (BioBERT, SciBERT, Galactica)
2. Unsupervised Classification Models
   a. Out-of-the-box representations: limitations
   b. SimCSE, E5, GTE...
   c. Document Representation: DocBERT
3. Learning Long-Range Dependencies
   a. Long-range attention models
   b. State-space models: S4

Domain-Specific Models

Domain-Specific Models

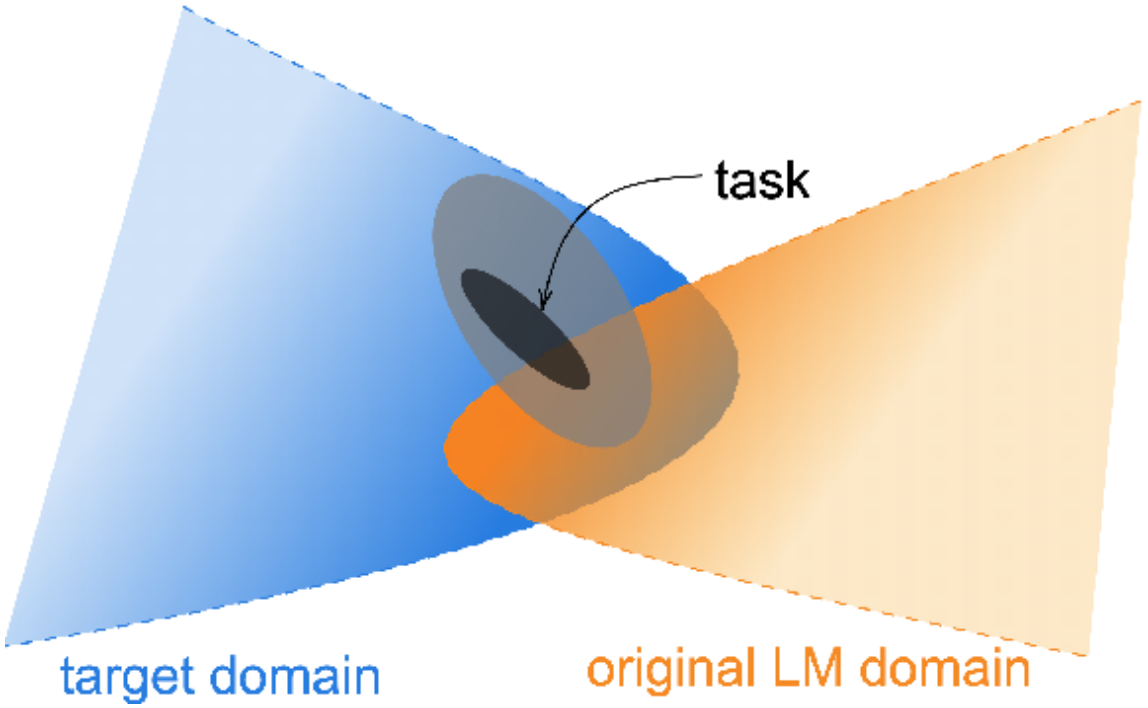
Pretrained (Large) Language Models are trained on content crawled over the internet, books, reports and news papers and are, hence **are open-domain**.

A **textual domain** is the **distribution over language characterizing a given topic or genre** [1].

- You are more likely to see the word "integer" in computer science than in news papers.
- An (L)LM will be more perplex to the word "integer" even though the input comes from a StackOverflow post.

Don’t Stop Pretraining

Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. [1]



Don’t Stop Pretraining

Domain	Pretraining Corpus	# Tokens	Size	$\mathcal{L}_{\text{RoB.}}$	$\mathcal{L}_{\text{DAPT}}$
BIOMED	2.68M full-text papers from S2ORC (Lo et al., 2020)	7.55B	47GB	1.32	0.99
CS	2.22M full-text papers from S2ORC (Lo et al., 2020)	8.10B	48GB	1.63	1.34
NEWS	11.90M articles from REALNEWS (Zellers et al., 2019)	6.66B	39GB	1.08	1.16
REVIEWS	24.75M AMAZON reviews (He and McAuley, 2016)	2.11B	11GB	2.10	1.93
ROBERTA (baseline)	see Appendix §A.1	N/A	160GB	‡1.19	-

Table 1: List of the domain-specific unlabeled datasets. In columns 5 and 6, we report ROBERTA’s masked LM loss on 50K randomly sampled held-out documents from each domain before $(\mathcal{L}_{ROB.})$ and after $(\mathcal{L}_{DAPT})$ $DAPT$ (lower implies a better fit on the sample). ‡ indicates that the masked LM loss is estimated on data sampled from sources similar to ROBERTA’s pretraining corpus.

Don’t Stop Pretraining

Figure 2: Vocabulary overlap (%) between domains. PT denotes a sample from sources similar to ROBERTA’s pretraining corpus. Vocabularies for each domain are created by considering the top 10K most frequent words (excluding stopwords) in documents sampled from each domain.

Don’t Stop Pretraining

Domain	Task	RoBERTa	Additional Pretraining Phases		
			DAPT	TAPT	DAPT + TAPT
BioMed	ChemProt	$81.9_{1.0}$	$84.2_{0.2}$	$82.6_{0.4}$	$\mathbf{84.4}_{0.4}$
	†RCT	$87.2_{0.1}$	$87.6_{0.1}$	$87.7_{0.1}$	$\mathbf{87.8}_{0.1}$
CS	ACL-ARC	$63.0_{5.8}$	$75.4_{2.5}$	$67.4_{1.8}$	$\mathbf{75.6}_{3.8}$
	SciERC	$77.3_{1.9}$	$80.8_{1.5}$	$79.3_{1.5}$	$\mathbf{81.3}_{1.8}$
News	HyperPartisan	$86.6_{0.9}$	$88.2_{5.9}$	$\mathbf{90.4}_{5.2}$	$90.0_{6.6}$
	†AGNews	$93.9_{0.2}$	$93.9_{0.2}$	$94.5_{0.1}$	$\mathbf{94.6}_{0.1}$
Reviews	†Helpfulness	$65.1_{3.4}$	$66.5_{1.4}$	$68.5_{1.9}$	$\mathbf{68.7}_{1.8}$
	†IMDB	$95.0_{0.2}$	$95.4_{0.1}$	$95.5_{0.1}$	$\mathbf{95.6}_{0.1}$

Table 5: Results on different phases of adaptive pretraining compared to the baseline RoBERTa (col. 1). Our approaches are $DAPT$ (col. 2, §3), $TAPT$ (col. 3, §4), and a combination of both (col. 4).

Don’t Stop Pretraining

"We show that **pretraining the model towards a specific task or small corpus can provide significant benefits**. Our findings suggest it may be valuable to complement work on ever-larger LMs with parallel efforts to **identify and use domain and task relevant corpora to specialize models**."

BioBERT

"[..] the word distributions of general and biomedical corpora are quite different, which can often be a problem for biomedical text mining models." [2]

BioBERT

: 1. List of text corpora used for BioBERT

Corpus	# of words (B)	Domain
English Wikipedia	2.5B	General
BooksCorpus	0.8B	General
PubMed Abstracts	4.5B	Biomedical
PMC Full-text articles	13.5B	Biomedical

Table 1. List of text corpora used for BioBERT

BioBERT

"We showed that **pre-training BERT on biomedical corpora is crucial in applying it to the biomedical domain**. Requiring minimal task-specific architectural modification, **BioBERT outperforms previous models on biomedical text mining tasks** such as NER, RE and QA."

SciBERT

"[...] while both BERT and ELMo have released pretrained models, they are still trained on general domain corpora such as news articles and Wikipedia." [3]

SciBERT

Field	Task	Dataset	SOTA	BERT-Base		SCIBERT	
				Frozen	Finetune	Frozen	Finetune
Bio	NER	BC5CDR (Li et al., 2016)	88.85[7]	85.08	86.72	88.73	**90.01**
		JNLPBA (Collier and Kim, 2004)	**78.58**	74.05	76.09	75.77	77.28
		NCBI-disease (Dogan et al., 2014)	**89.36**	84.06	86.88	86.39	88.57
	PICO	EBM-NLP (Nye et al., 2018)	66.30	61.44	71.53	68.30	**72.28**
	DEP	GENIA (Kim et al., 2003) - LAS	**91.92**	90.22	90.33	90.36	90.43
		GENIA (Kim et al., 2003) - UAS	**92.84**	91.84	91.89	92.00	91.99
	REL	ChemProt (Kringelum et al., 2016)	76.68	68.21	79.14	75.03	**83.64**
CS	NER	SciERC (Luan et al., 2018)	64.20	63.58	65.24	65.77	**67.57**
	REL	SciERC (Luan et al., 2018)	n/a	72.74	78.71	75.25	**79.97**
	CLS	ACL-ARC (Jurgens et al., 2018)	67.9	62.04	63.91	60.74	**70.98**
Multi	CLS	Paper Field	n/a	63.64	65.37	64.38	**65.71**
		SciCite (Cohan et al., 2019)	84.0	84.31	84.85	**85.42**	**85.49**
Average				73.58	77.16	76.01	79.27

Table 1: Test performances of all BERT variants on all tasks and datasets. [...]

SciBERT

Task	Dataset	BioBERT	SciBERT
NER	BC5CDR	88.85	90.01
	JNLPBA	77.59	77.28
	NCBI-disease	89.36	88.57
REL	ChemProt	76.68	83.64

Table 2: Comparing SciBERT with the reported BioBERT results on biomedical datasets.

SciBERT

NB: SciBERT was trained on curated textual data ; not trained on code or script for example --at leat not trained directly and purposefully on this kind of data

Galactica

"Computing has indeed revolutionized how research is conducted, but information overload remains an overwhelming problem [...]. In this paper, we argue for a better way through large language models. Unlike search engines, language models can potentially store, combine and reason about scientific knowledge." [4]

- Galactica was trained on a rather small highly curated dataset.
- All the data was standardized as markdown text.

Galactica

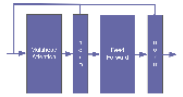
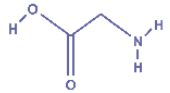
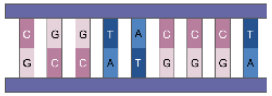
Modality	Entity	Sequence	
Text	Abell 370	`Abell 370 is a cluster...`	
LaTeX	Schwarzschild radius	`r_{s} = \frac{2GM}{c^2}`	$r_s = \frac{2GM}{c^2}$
Code	Transformer	`class Transformer(nn.Module)`	
SMILES	Glycine	`C(C(=O)O)N`	
AA Sequence	Collagen α-1(II) chain	`MIRLGAPQTL..`	
DNA Sequence	Human genome	`CGGTACCCTC..`	

Table 1: Tokenizing Nature. Galactica trains on text sequences that represent scientific phenomena.

Table 1: Tokenizing Nature. Galactica trains on text sequences that represent scientific phenomena.

Galactica

1. **Citations:** we wrap citations with special reference tokens [START_REF] and [END_REF].
2. **Step-by-Step Reasoning:** we wrap step-by-step reasoning with a working memory token , mimicking an internal working memory context.
3. **Mathematics:** for mathematical content, with or without LaTeX, we split ASCII operations into individual characters. Parentheses are treated like digits. The rest of the operations allow for unsplit repetitions. Operation characters are !"#$%&’*+,-./:;<=>?^_‘| and parentheses are ()[]{}.

4. **Numbers:** we split digits into individual tokens. For example 737612.62 -> 7,3,7,6,1,2,.,6,2.
5. **SMILES formula:** we wrap sequences with [START_SMILES] and [END_SMILES] and apply characterbased tokenization. Similarly we use [START_I_SMILES] and [END_I_SMILES] where isomeric SMILES is denoted. For example, C(C(=O)O)N → C,(,C,(,=,O,),O,),N.
6. **Amino acid sequences:** we wrap sequences with [START_AMINO] and [END_AMINO] and apply character-based tokenization, treating each amino acid character as a single token. For example, MIRLGAPQTL -> M,I,R,L,G,A,P,Q,T,L.

7. **DNA sequences:** we also apply a character-based tokenization, treating each nucleotide base as a token, where the start tokens are [START_DNA] and [END_DNA]. For example, CGGTACCCTC -> C, G, G, T, A, C, C, C, T, C.

Galactica

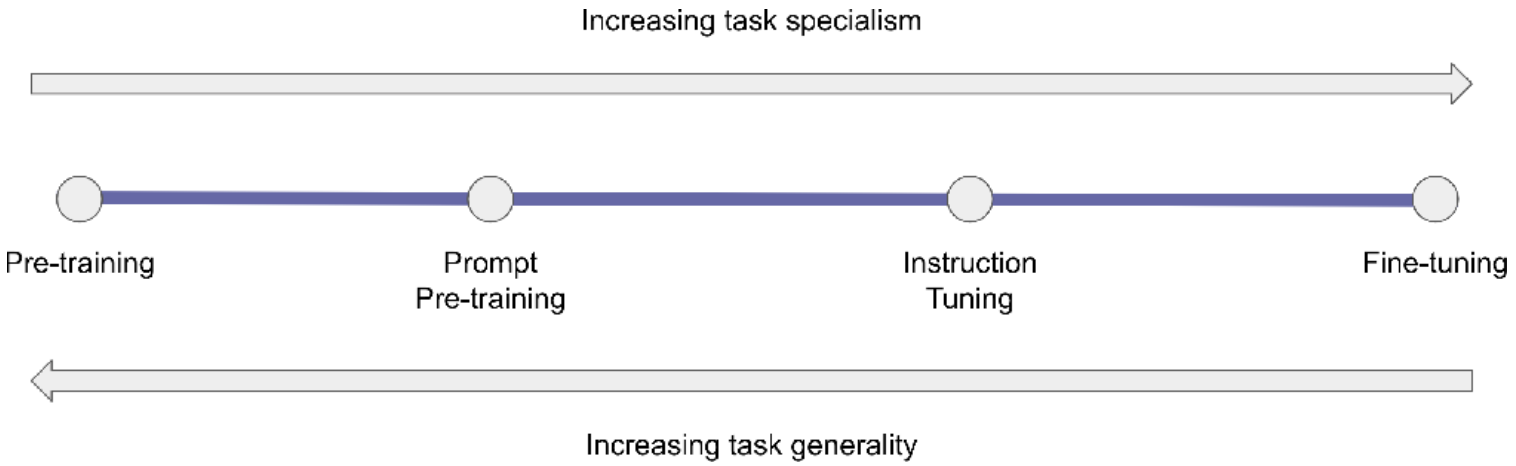


Figure 5: Prompt Pre-training. Pre-training weighs all tokens equally as part of the self-supervised loss. This leads to a weak relative signal for tasks of interest, meaning model scale has to be large to work. Instruction tuning boosts performance post hoc, and can generalize to unseen tasks of interest, but it risks performance in tasks that are distant from instruction set tasks. Prompt pre-training has a weaker task of interest bias than instruction tuning but less risk of

Galactica

- **GeLU Activation** - GeLU activations for all model sizes.
- **Context Window** - a 2048 length context window.
- **No Biases** - following PaLM, we do not use biases in any of the dense kernels or layer norms.
- **Learned Positional Embeddings** - learned positional embeddings for the model.
- **Vocabulary** - vocabulary of 50k tokens using BPE. The vocabulary was generated from a randomly selected 2% subset of the training data.

Galactica

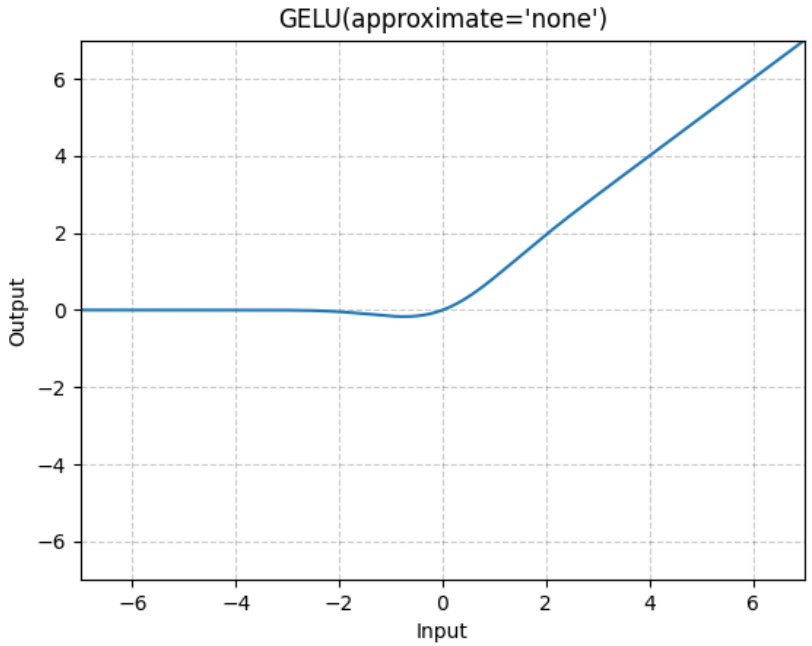
Gaussian Error Linear Units function (GeLu)

$$GELU(x) = x * \Phi(x)$$

Where $\Phi(x)$ is the Gaussian function.

$$GELU(x) \approx x * \frac{1}{2}(1 + Tanh(\frac{2}{\pi} * (x + 0.044715 * x^3)))$$

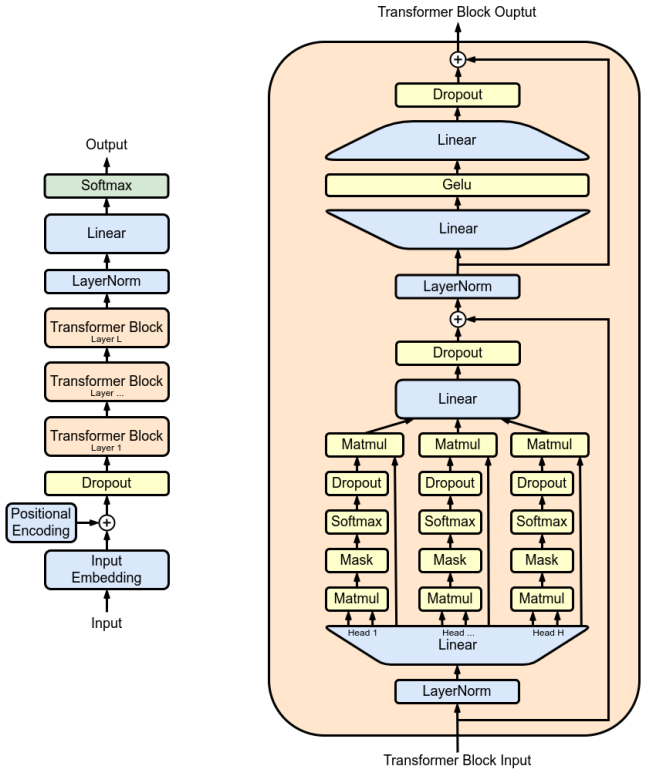
Galactica



- Allows small negative values when $x < 0$.
- Avoids the dying ReLU problem.

Galactica

Why no biases?



Unsupervised Classification Models

Out-of-the-box representations

pass

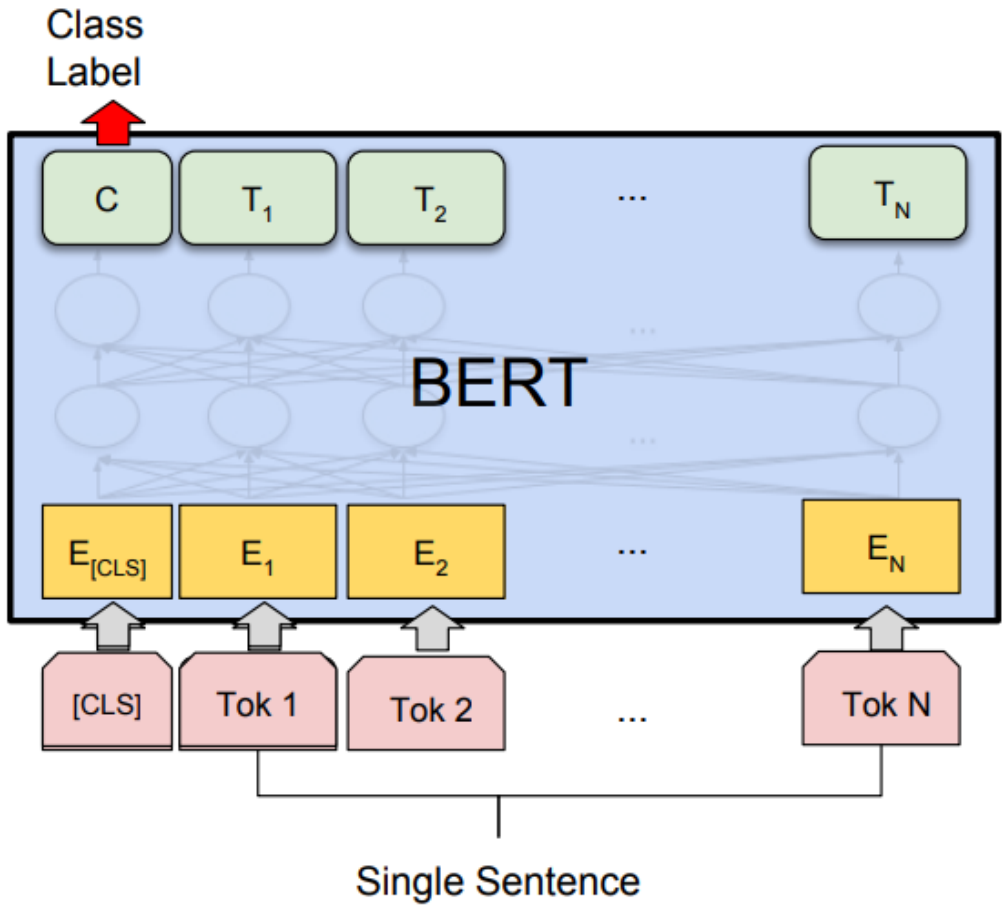
Sentiment Analysis

Sentiment Analysis

Sentiment analysis is a sentence classification task aiming at **automatically mapping data to their sentiment**.

It can be **binary** classification (e.g., positive or negative) or **multiclass** (e.g., enthusiasm, anger, etc)

Sentiment Analysis



Sentiment Analysis

The loss can be the likes of cross-entropy (CE), binary cross-entropy (BCE) or KL-Divergence (KL).

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{n'=1}^{N} y^{(n)}.log(f(\mathbf{x},\theta)^{(n)})$$

$$\mathcal{L}_{BCE} = -y^{(n)}.log(f(\mathbf{x},\theta)^{(n)}) + (1-y^{(n)}).(1-f(\mathbf{x},\theta)^{(n)})$$

$$\mathcal{L}_{KL} = -\frac{1}{N}\sum_{n'=1}^{N} y^{(n)}.log(\frac{y^{(n)}}{f(\mathbf{x},\theta)^{(n)}})$$

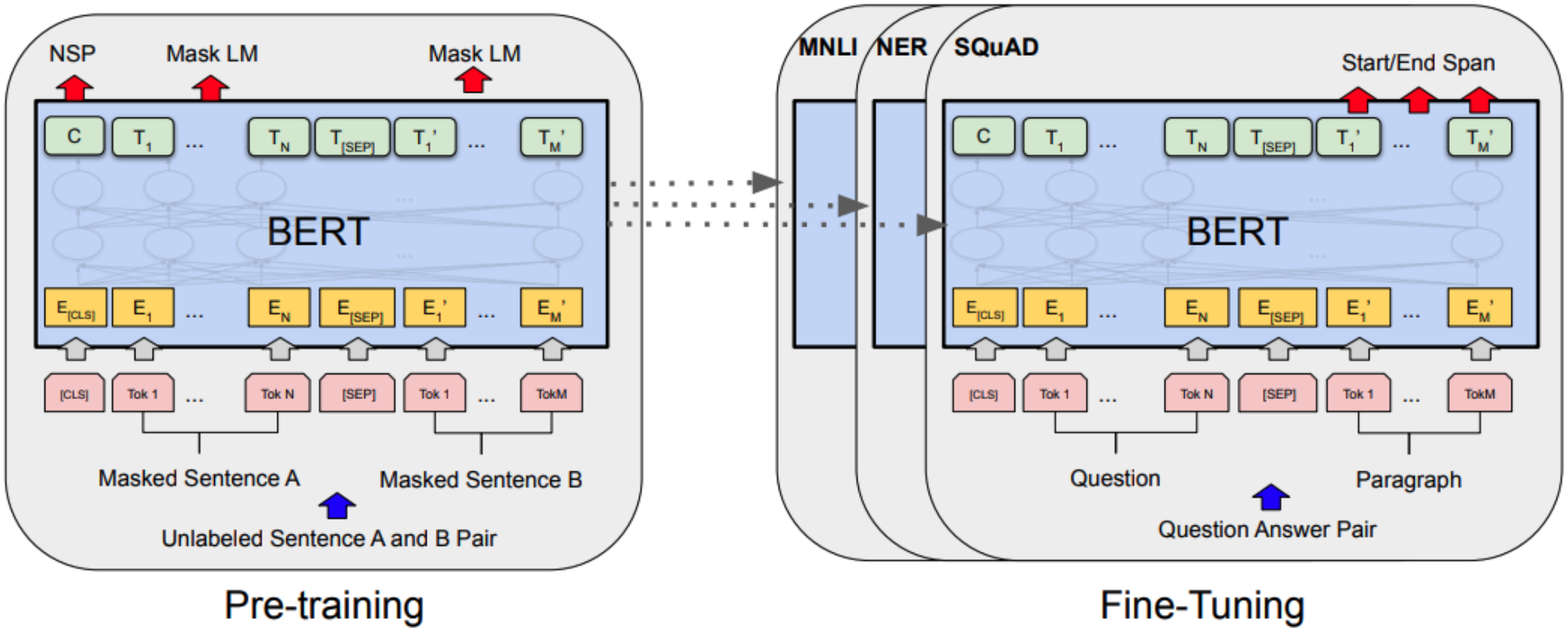
Question Answering (QA)

Question Answering (QA)

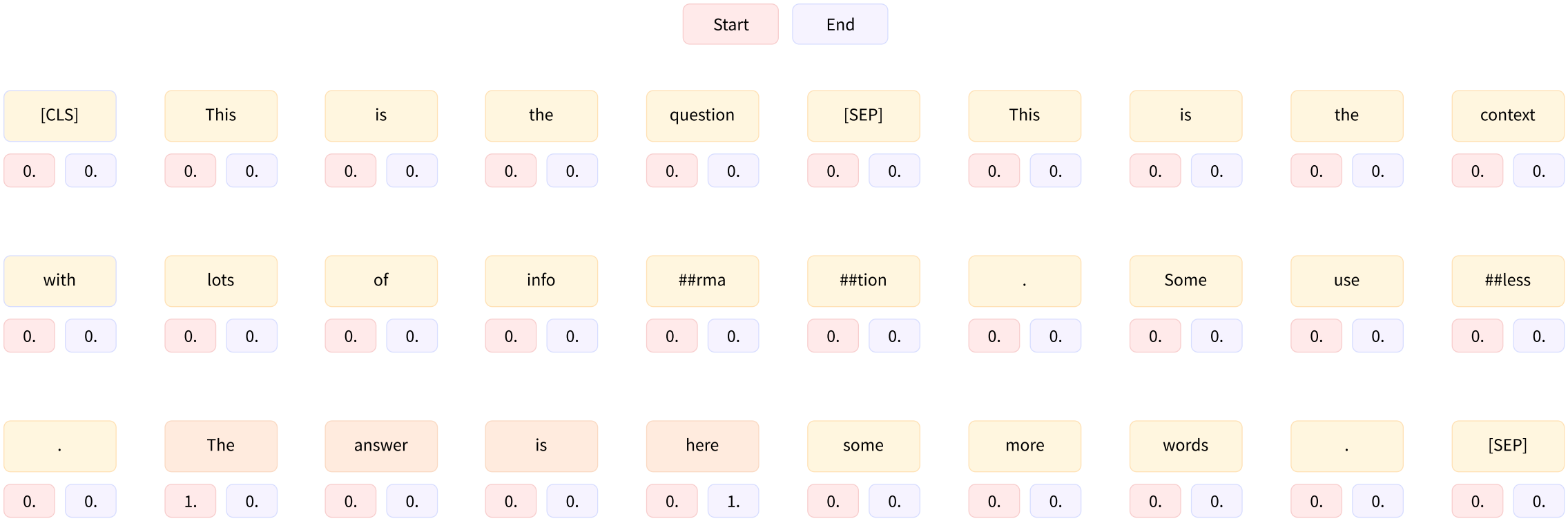
QA is the task of **retrieving a span of text from a context** that is best suited to answer a question.

This task is extractive -> **information retrieval**

Question Answering (QA)



Question Answering (QA)



Question Answering (QA)

The loss is the cross entropy over the output of the starting token and the ending one:

$$\mathcal{L}_{CE_{QA}} = \mathcal{L}_{CE_{start}} + \mathcal{L}_{CE_{end}}$$

Natural Language Inference (NLI)

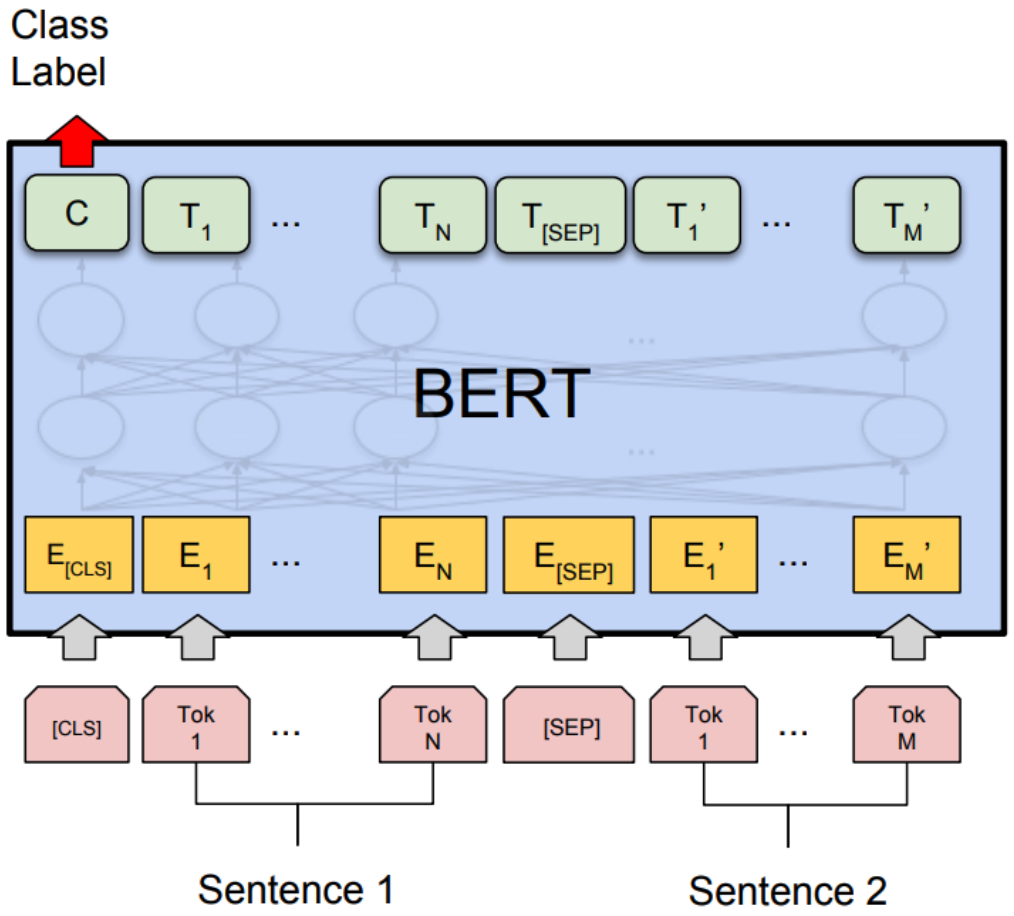
Natural Language Inference (NLI)

NLI is the task of **determining whether a "hypothesis" is true (entailment), false (contradiction), or undetermined (neutral)** given a "premise". [1]

Natural Language Inference (NLI)

Premise	Label	Hypothesis
A man inspects the uniform of a figure in some East Asian country.	contradiction	The man is sleeping.
An older and younger man smiling.	neutral	Two men are smiling and laughing at the cats playing on the floor.
A soccer game with multiple males playing.	entailment	Some men are playing a sport.

Natural Language Inference (NLI)



Natural Language Inference (NLI)

The loss is simply the cross entropy or the divergence over the output of the `CLS` token and the true label.

$$\mathcal{L}_{NLI} = \mathcal{L}_{CE_{CLS}}$$

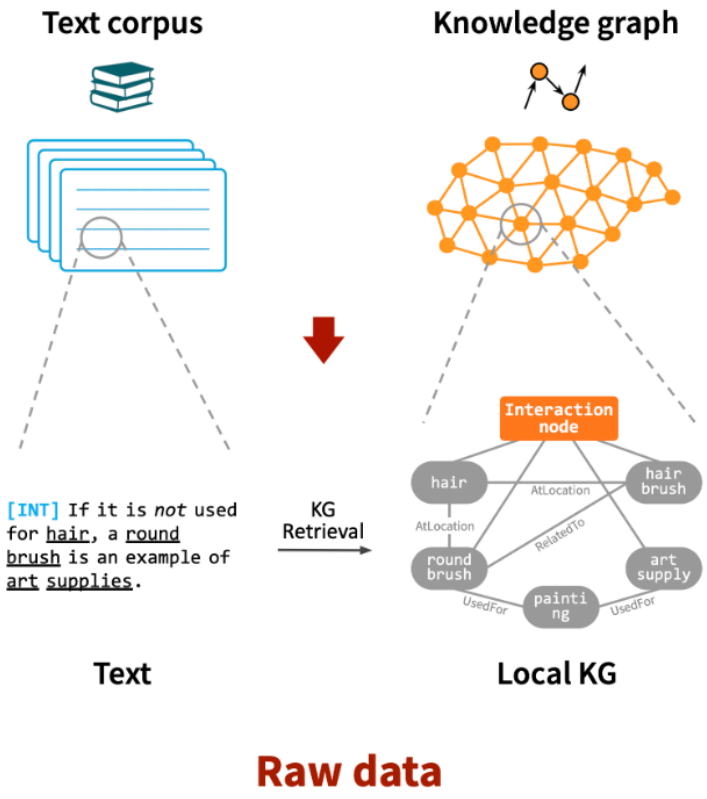
We are trying to compress the information about both sentence in one `CLS` token via attention and decide about their relationship.

Is it possible to help the model infering more information with les text data?

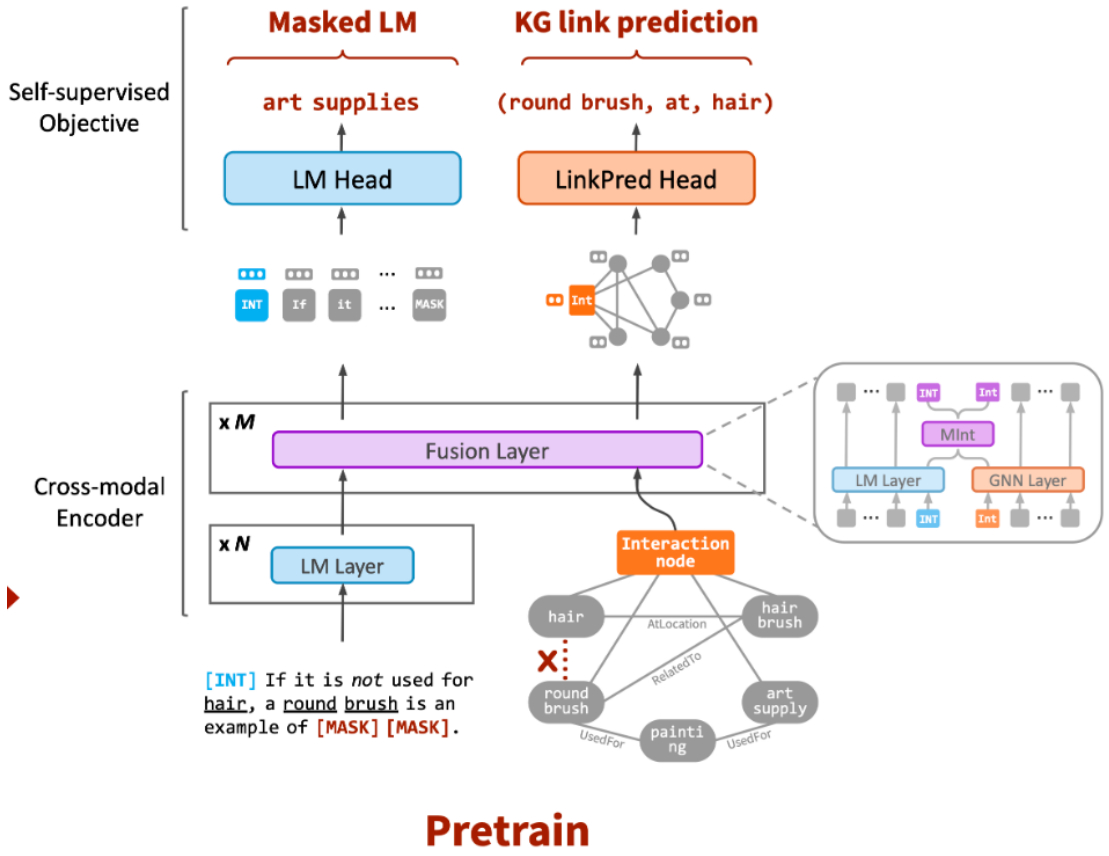
Going Further: LM as Knowledge Graphs

Yasunaga, M., Bosselut, A., Ren, H., Zhang, X., Manning, C. D., Liang, P. S., & Leskovec, J. (2022). Deep bidirectional language-knowledge graph pretraining. Advances in Neural Information Processing Systems, 35, 37309-37323.

Going Further: LM as Knowledge Graphs



Going Further: LM as Knowledge Graphs



Going Further: LM as Knowledge Graphs

This architecture ***involves a KG ready to use beforeheaad and pre-training from scratch***. How can we better **perform NLP task without having to retrain or fine-tune** a model?

Exploit LLMs capacities: Chain-of-thoughts & In context Learning

Exploit LLMs capacities

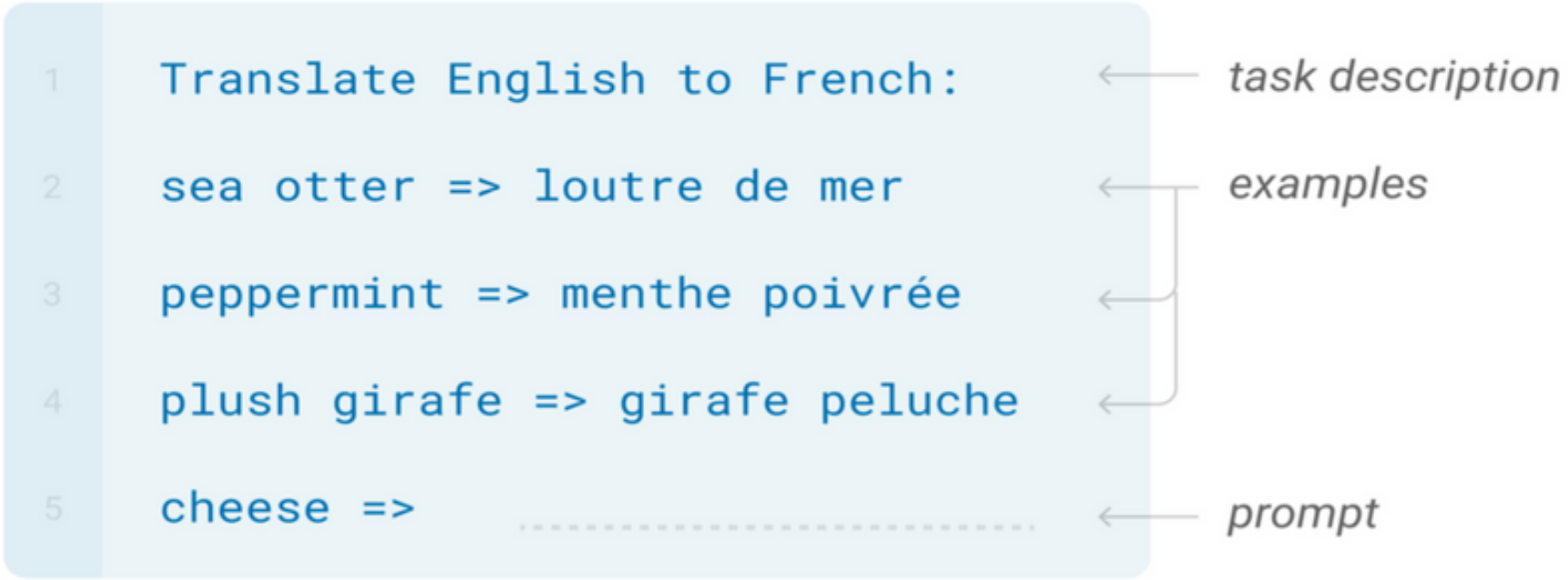
ICL enables LLMs to learn new tasks using natural language prompts without explicit retraining or fine-tuning.

The efficacy of ICL is closely tied to the model's scale, training data quality, and domain specificity.

Exploit LLMs capacities

Few-shot

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



Exploit LLMs capacities

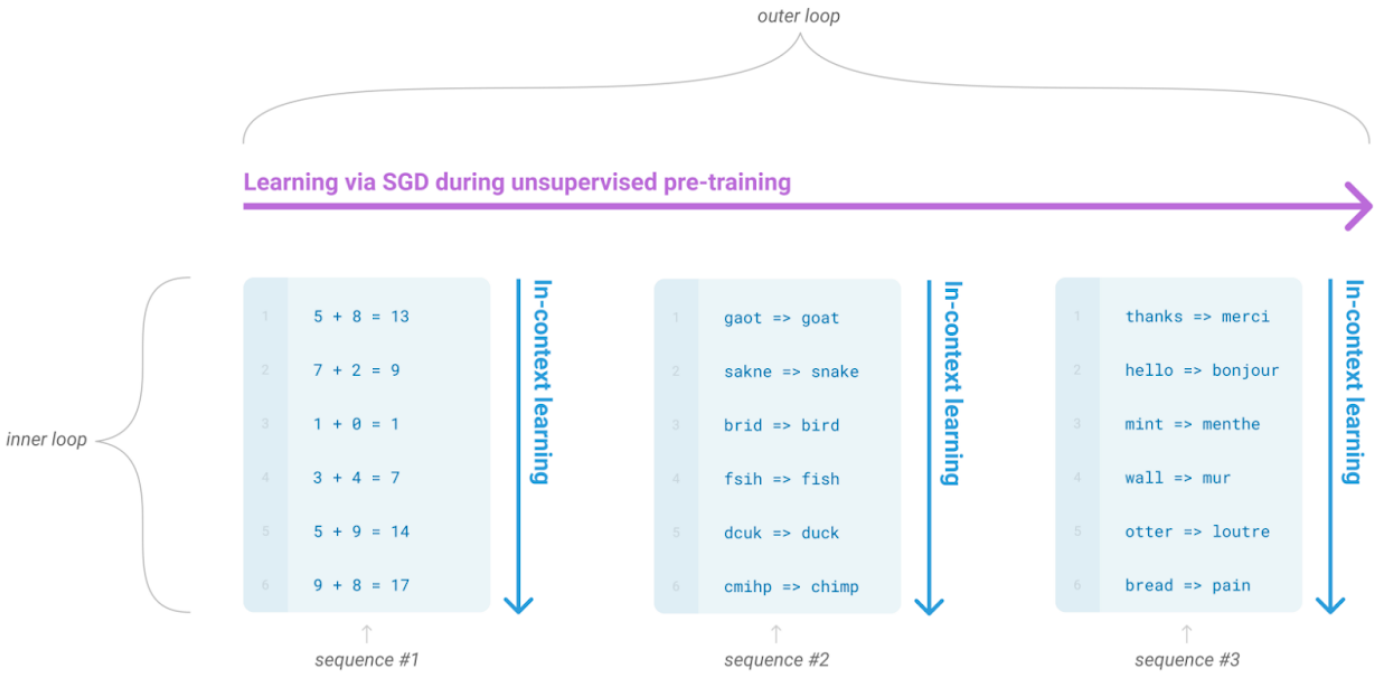


Figure 1.1: Language model meta-learning. During unsupervised pre-training, a language model develops a broad set of skills and pattern recognition abilities. It then uses these abilities at inference time to rapidly adapt to or recognize the desired task. We use the term "in-context learning" to describe the inner loop of this process, which occurs within the forward-pass upon each sequence. The sequences in this diagram are not intended to be representative of the data a model would see during pre-training, but are intended to show that there are sometimes repeated sub-tasks embedded within a single sequence.

Exploit LLMs capacities

$$p(\text{output}|\text{prompt}) = \int_{\text{concept}} p(\text{output}|\text{concept}, \text{prompt}) p(\text{concept}|\text{prompt}) d(\text{concept}). \qquad (1)$$

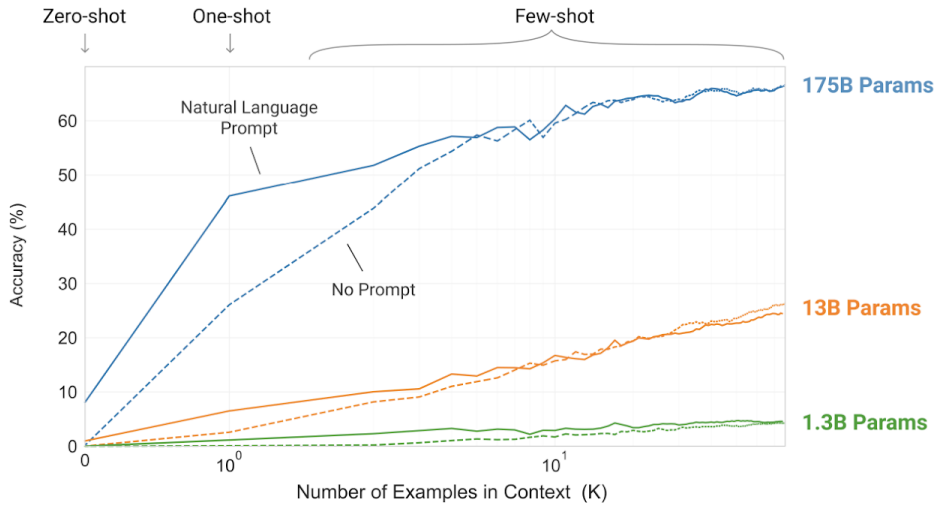
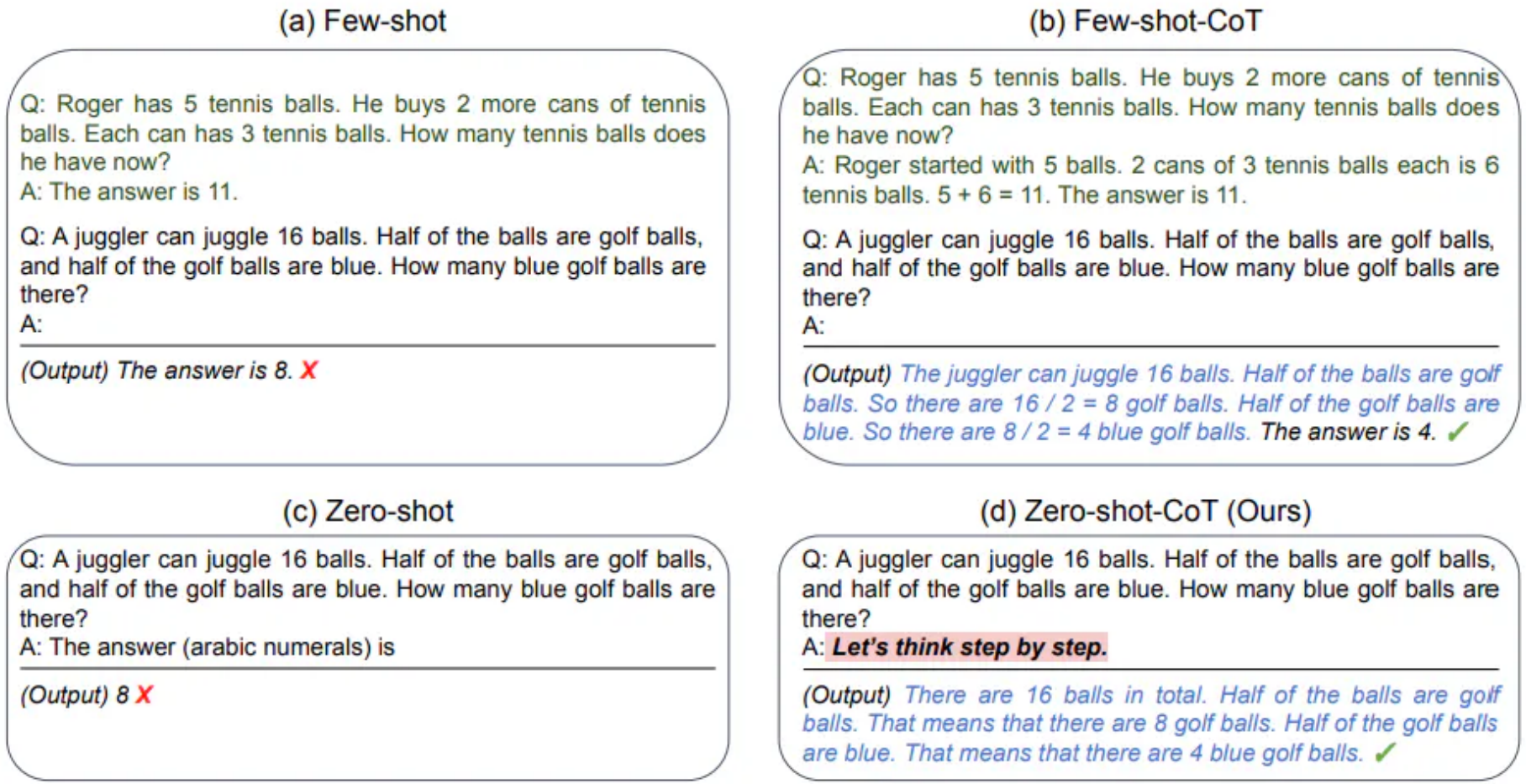


Figure 1.2: Larger models make increasingly efficient use of in-context information. We show in-context learning performance on a simple task requiring the model to remove random symbols from a word, both with and without a natural language task description (see Sec. 3.9.2). The steeper "in-context learning curves" for large models demonstrate improved ability to learn a task from contextual information. We see qualitatively similar behavior across a wide range of tasks.

Exploit LLMs capacities



Questions?

References

[1] Gururangan, Suchin, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. "Don't Stop Pretraining: Adapt Language Models to Domains and Tasks." In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, edited by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, 8342–60. Online: Association for Computational Linguistics, 2020.

[2] Lee, Jinhyuk, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. "BioBERT: A Pre-Trained Biomedical Language Representation Model for Biomedical Text Mining." Bioinformatics 36, no. 4 (February 15, 2020): 1234–40.

[3] Beltagy, Iz, Kyle Lo, and Arman Cohan. "SciBERT: A Pretrained Language Model for Scientific Text." In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), edited by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, 3615–20. Hong Kong, China: Association for Computational Linguistics, 2019.

[4] Taylor, Ross, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. “Galactica: A Large Language Model for Science.” arXiv, November 16, 2022.