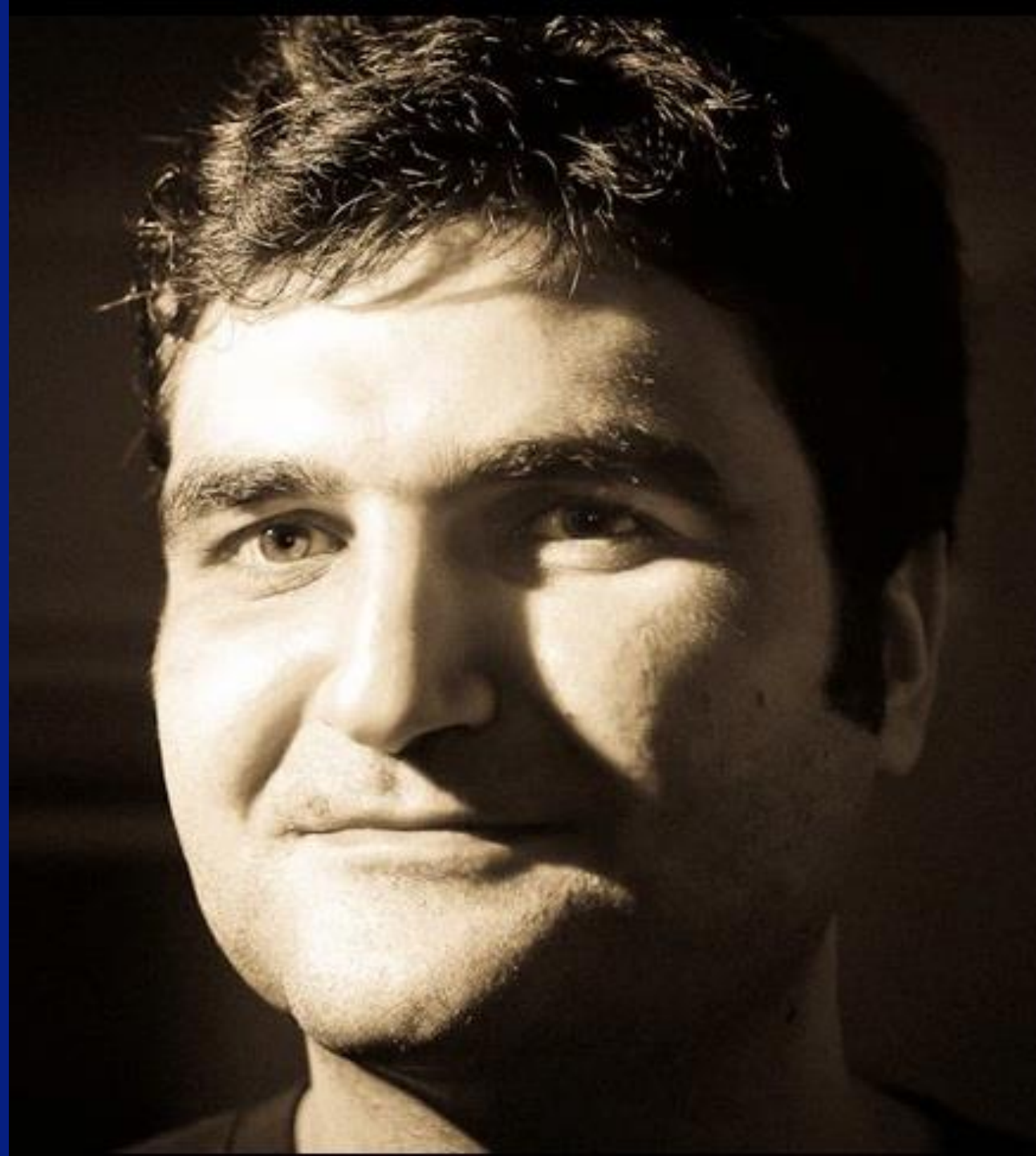
About the instructor

Alfio Gliozzo is a Principal Research Staff Member and Manager at IBM T.J. Watson Research. He has over 25 years of experience in the field of Artificial Intelligence. He is the Chief Science Catalyst for the area of Data and AI Platforms in IBM research. His research focuses on advancing the state of the art of genAI platforms with applications on enterprise data. He published approximately 150 scientific publications, including books, papers and patents. He was part of the Deep QA team that developed IBM Watson, the system that defeated the Jeopardy! grand masters. He is the main author and contributor of Agentics.



Agentics Tutorial

Lecture 1: Large Language Models

Guest lectures at Columbia University Class on Agentic AI, Fintech, and the Data Economy. - Prof. Agostino Capponi

Alfio Massimiliano Gliozzo

IBM research

gliozzo@us.ibm.com

Outline

- from Word Embeddings …
- … to Large Language Models
- Enhancing LLMs with human supervision
- Structured Decoding
- Hands on: your first llm call

Outline

- from Word Embeddings …
- … to Large Language Models
- Enhancing LLMs with human supervision
- Structured Decoding
- Hands on: your first llm call

Structuralism in Linguistics (early 1900's)

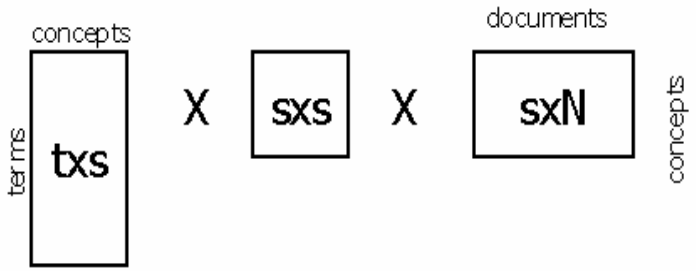
- **structuralism** is the theory that elements of **human culture** must be understood in terms of their **relationship to a larger system** or **structure**
- In **lexical semantics** structuralism states that meaning of words emerges from their relationship with other words (De Saussure,1916)
- **A.k.a. distributional hypothesis**
  - Words with similar meanings tend to occur in similar context

Latent Semantic Analysis Deerwester et al. (1990): Acquiring Domain Models by SVD

M =

	d_1	d_2	d_3	d_4	d_5	d_6
shuttle	1	0	1	0	0	0
astronaut	0	1	0	0	0	0
moon	1	1	0	0	0	0
car	1	0	0	1	1	0
truck	0	0	0	1	0	1

$$\mathbf{M} = \mathbf{K}_{t\times s}\mathbf{S}_{s\times s}\mathbf{D}^{T}_{s\times N}$$



K =

	dim_1	dim_2	dim_3	dim_4	dim_5
shuttle	-0.44	-0.30	0.57	0.58	0.25
astronaut	-0.13	-0.33	-0.59	0.00	0.73
moon	-0.48	-0.51	-0.37	0.00	-0.61
car	-0.70	0.35	0.15	-0.58	0.16
truck	-.0.26	0.65	-0.41	0.58	-0.09

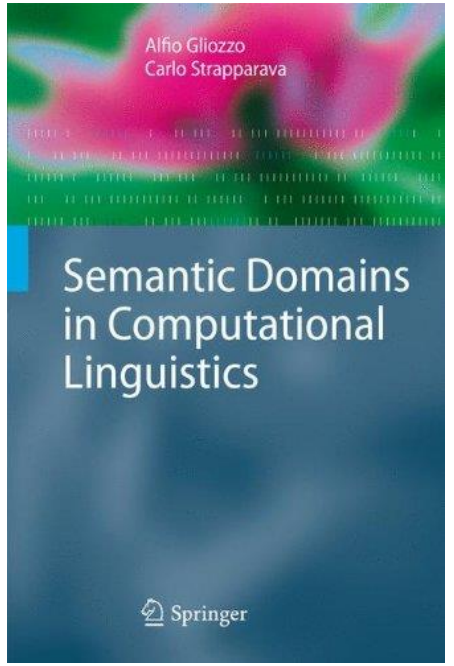
S =

2.16	0.00	0.00	0.00	0.00
0.00	1.59	0.00	0.00	0.00
0.00	0.00	1.28	0.00	0.00
0.00	0.00	0.00	1.00	0.00
0.00	0.00	0.00	0.00	0.39

Term Vectors are mapped into a lower dimensional Space

Alternative Word Embedding: LDA, Word2Vect

Gliozzo and Strapparava, 2005

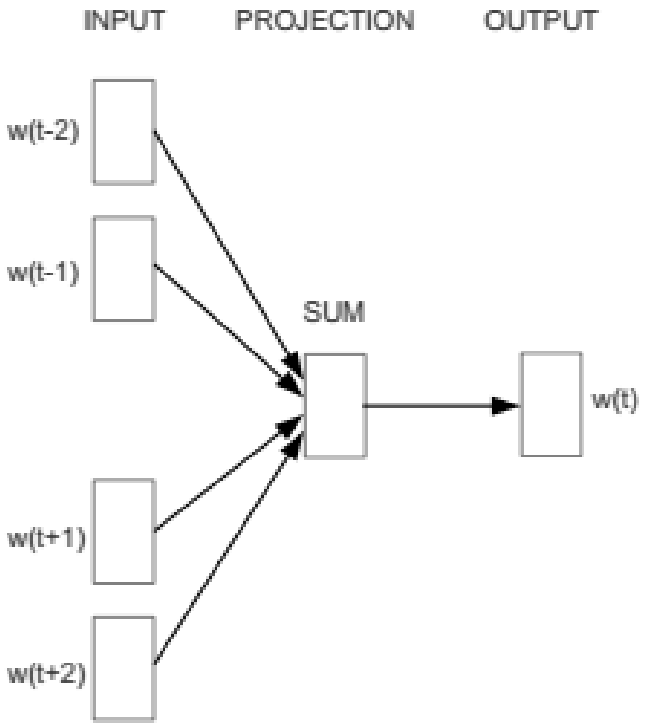


Applications

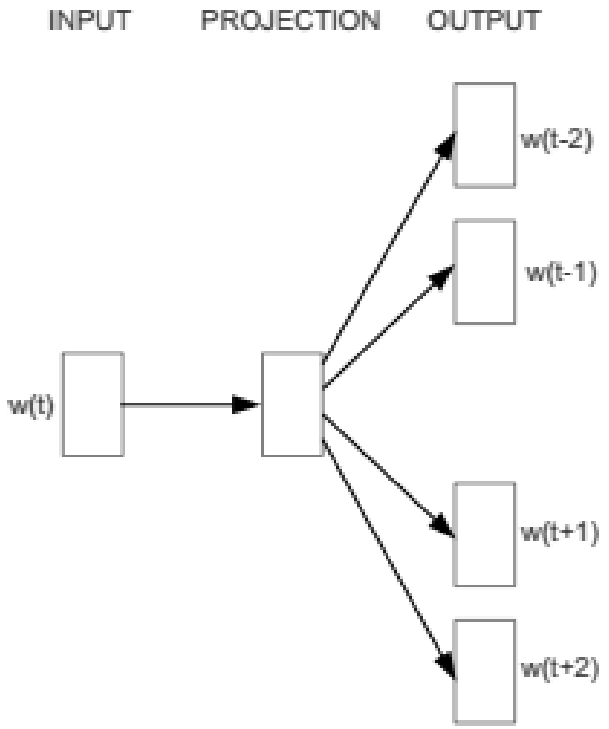
- Text Categorization
- Word Sense Disambiguation
- Information Retrieval
- …

Word2Vect (Mikolov et al., 2013)

Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." , ICLR



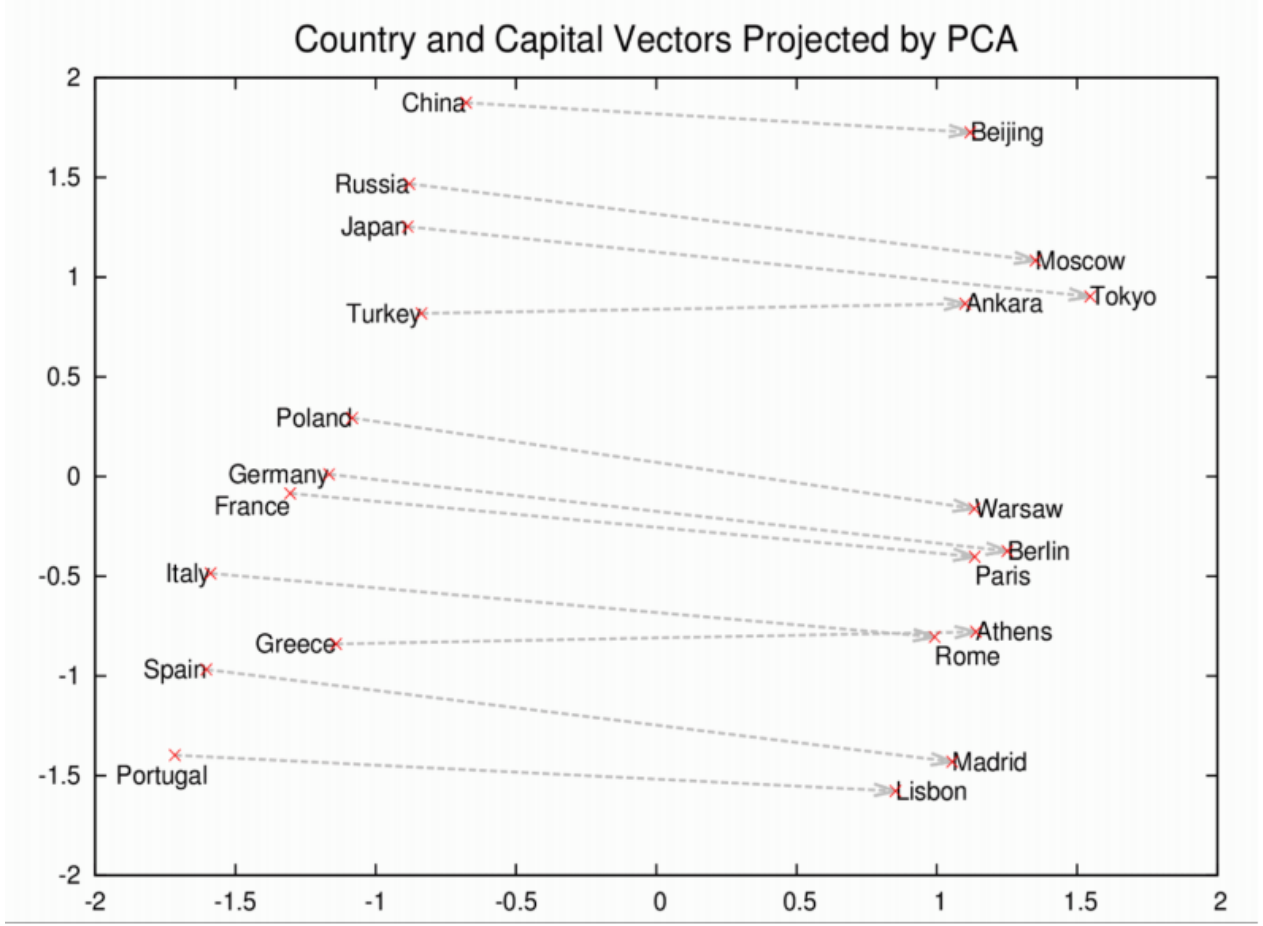
CBOW



Skip-gram

Word Analogies with W2V

Rome – Italy + China = Beijing



Applications:

- Knowledge Induction
- Relation Extraction
- Clustering
- …

Outline

- from Word Embeddings …
- … to Large Language Models
- Enhancing LLMs with human supervision
- Structured Decoding
- Hands on: your first llm call

The Transformer Architecture

- Language Modeling (LM): Predict future token given history.

$$p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i | x_1, \ldots, x_{i-1}) \qquad \mathcal{L} = -\sum_{i=1}^{n} \log p(x_i | x_1,$$

- E.g. P(account|open, a, bank)
- Minimizing the log likelihood corresponds to maximizing the probability of correctly guessing words.
- Training is about fining the optimal set of parameters that minimize loss over a large corpus of text
- GPT 2 has been trained on 40GB of text data from over 8 million documents, 1.5 billion parameters

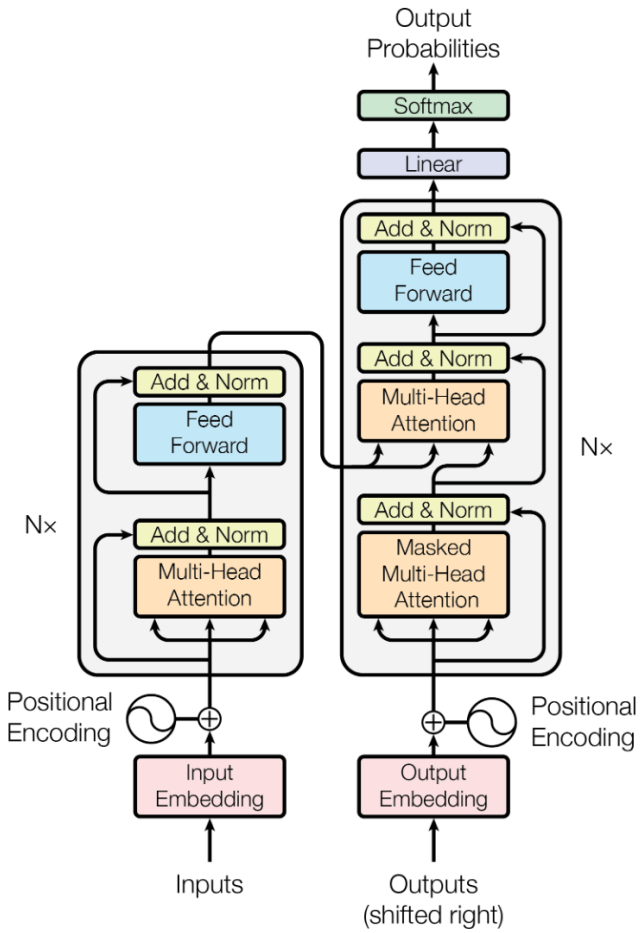


Figure 1: The Transformer - model architecture.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, **Attention Is All You Need,** NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems December 2017

What is BERT (Bidirectional Encoder Representations from Transformers)?

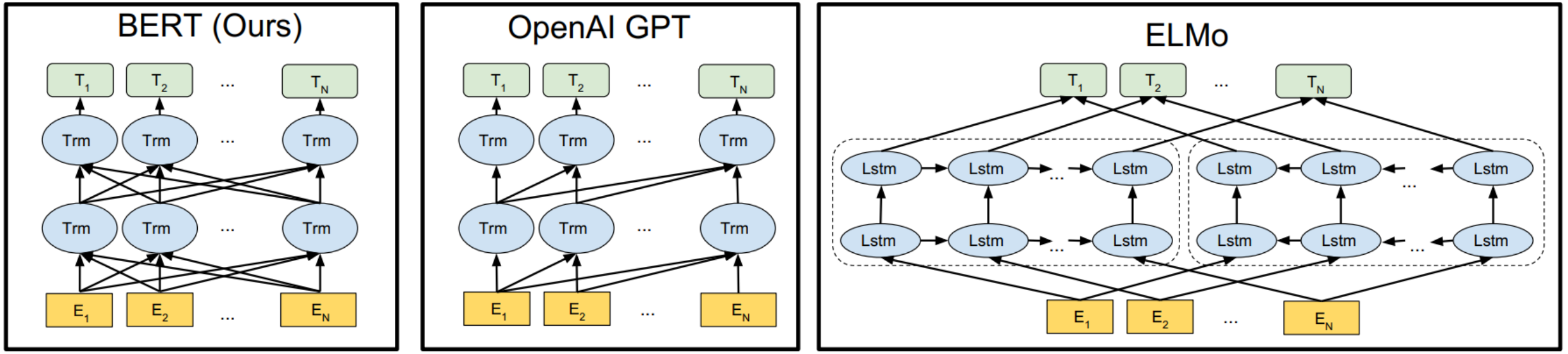


Figure 1: Differences in pre-training model architectures. BERT uses a bidirectional Transformer. OpenAI GPT uses a left-to-right Transformer. ELMo uses the concatenation of independently trained left-to-right and right-to-left LSTM to generate features for downstream tasks. Among three, only BERT representations are jointly conditioned on both left and right context in all layers.

Figure in (Devlin *et al.*, 2018)

GPT-3: Scaling Up enables In Context Learning

Unsupervised Learning, No human supervision

The task is predicting the next token

Loss decreases with parameter size

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

Few-shot

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

```
1  Translate English to French:          ← task descrip
2  sea otter => loutre de mer            ← examples
3  peppermint => menthe poivrée          ←
4  plush girafe => girafe peluche        ←
5  cheese =>                             ← prompt
```

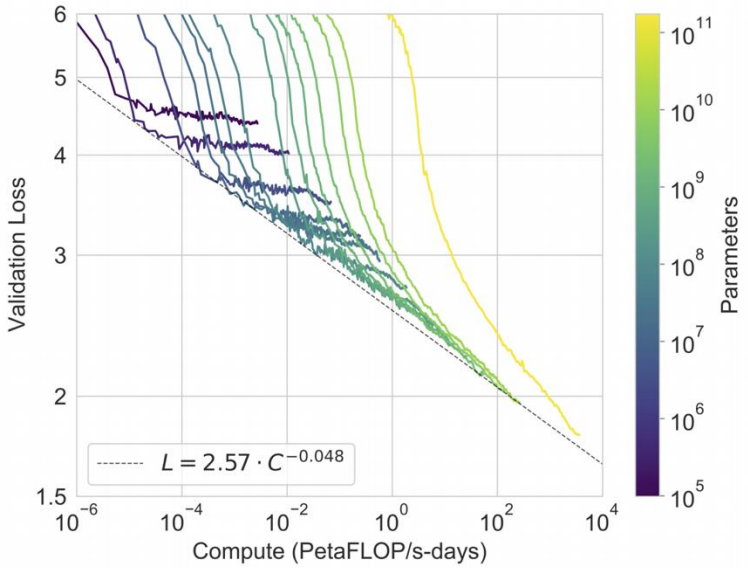
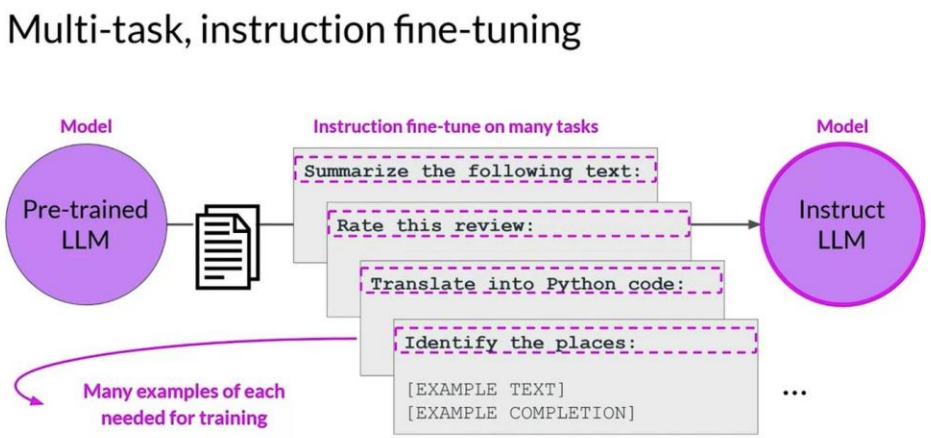


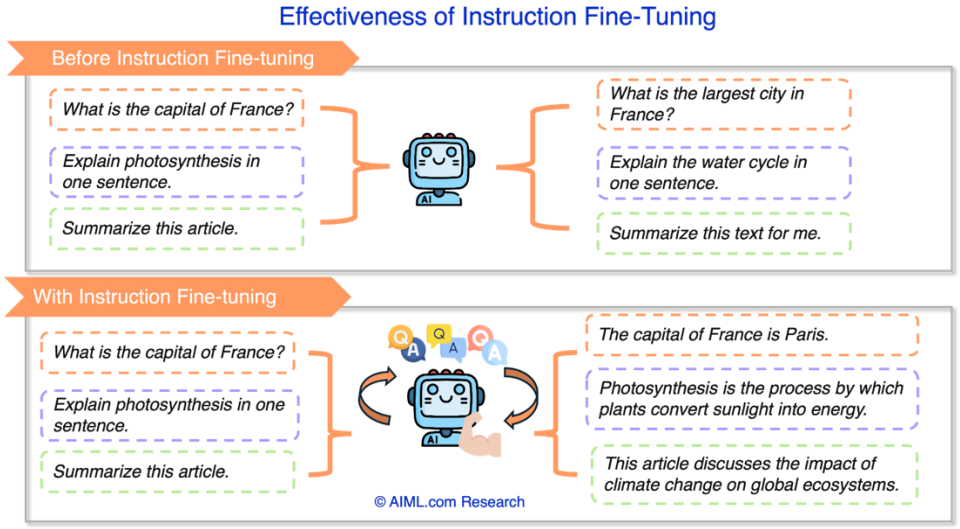
Figure 3.1: Smooth scaling of performance with compute. Performance (measured in terms of cross-entropy validation loss) follows a power-law trend with the amount of compute used for training. The power-law behavior observed in [KMH+20] continues for an additional two orders of magnitude with only small deviations from the predicted curve. For this figure, we exclude embedding parameters from compute and parameter counts.

Outline

- from Word Embeddings …
- … to Large Language Models
- Enhancing LLMs with human supervision
- Structured Decoding
- Hands on: your first llm call

Human Supervision





- **instruction-tuning:** A fine-tuning method where LLMs are trained on instruction–response pairs.
- **Reinforcement Learning from Human Feedback (RLHF)** : Human based feedback on ranking different LLM outputs with rationale

Outline

- from Word Embeddings …
- … to Large Language Models
- Enhancing LLMs with human supervision
- Structured Decoding
- Hands on: your first llm call

Structured Decoding

*Structured decoding is a mechanism that enforces LLMs to generate outputs that follow a predefined structure — such as a **Pydantic schema, JSON schema, or grammar***.

```
class get_weather(BaseModel):
        location: str
        date: Time
```

e.g. “What’s the weather in Rome tomorrow?”

```
{
  "function": "get_weather",
  "arguments": {"location": "Rome", "date": "2025-09-03"}
}
```

What is pydantic?

- A Python library for data validation and parsing using type annotations.
- Based on Python 3.6+ type hints.
- **Why use it?**
  - Guarantees type safety and validation at runtime.
  - Popular in data models, APIs, configuration management, agentic frameworks
- **Key Features**
  - Automatic type coercion
  - Rich error messages
  - Easy JSON/serialization support
  - Documentation

Example Pydantic Class Declaration

```python
class AnswerEvaluation(BaseModel):
    question_grade:int = Field(..., description="""A judgment on wether the question
contains all the necessary clues to actually get to a specific and anabiguous answer,
in a scale 0 to 10""", ge=0,lt=10)
    answer_grade:int = Field(..., description="""A judgment on wether the system
generated answer is correct for the question based on the retrieved supported evidence,
in a scale 0 to 10 """, ge=0,lt=10)
    system_correctly_provided_no_answers:bool = Field(..., description="""True if the
system generated an empty, None or a statement sayng that there is no evidence to
provide an answer, and there is actually no evidence from the retrieved passages
supporting that.""")
    explanation:str = Field(..., description="""Explanation of why the grades above
have been assigned""")
    alternate_question:Optional[str] = Field(None, description="""A reformulation
of the input question that could have better chances to generate a correct answer
given the observed passage""")
```

Types

Required

Constraints

LLM/Human readable Descriptions

Optional

Outline

- from Word Embeddings …
- … to Large Language Models
- Enhancing LLMs with human supervision
- Structured Decoding
- Hands on: your first llm call

Today’s LLM Ecosystem

- **Fine-tuned for dialogue** with guardrails and structured reasoning
- Integrate with **tools**, search, and APIs
- **Commercial**: GPT-4 (OpenAI), Claude (Anthropic), Gemini (Google)
- **OpenSource**: LLaMA 3 (Meta), Mistral, Granite

ChatGPT

Gemini

Claude

LLaMA by Meta

watsonx

Ollama

Granite

Hands on: Use LLMs in Agentics

- Get your API key
  - google https://aistudio.google.com/apikey (free with limited quota)
- Optional: set up google drive https://drive.google.com/drive/my-drive
  - Copy this file in your MyDrive folder https://github.com/IBM/Agentics/blob/main/.env_sample and remane it to .env
  - Edit the file to add your API key (you can use the texteditor app in colab)
  - Optionally Copy this folder in https://github.com/IBM/Agentics/blob/main/tutorials/data your MyDrive
- Follow this tutorial notebook
  - https://colab.research.google.com/github/IBM/Agentics/blob/main/tutorials/llms.ipynb

Homework: Familiarize with Agentics

- Install VS code:
  - https://code.visualstudio.com/download
- Set up the agentic repo
  - Fork your local version of agentics
  - Follow installation instructions here https://ibm.github.io/Agentics/getting_started/
- Learn about Agentics
  - Documentaton https://ibm.github.io/Agentics/
  - Paper: https://arxiv.org/pdf/2508.15610
- Play with agentics tutorials
  - https://colab.research.google.com/github/IBM/Agentics/blob/main/tutorials/agentics_basics.ipynb
  - https://colab.research.google.com/github/IBM/Agentics/blob/main/tutorials/transduction.ipynb
  - https://colab.research.google.com/github/IBM/Agentics/blob/main/tutorials/amap_reduce.ipynb

References

De Saussure, F. (1916). *Cours de linguistique générale*. Lausanne & Paris: Payot.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), 391–407.

Gliozzo, A., & Strapparava, C. (2009). *Semantic Domains in Computational Linguistics*. Springer Berlin Heidelberg. ISBN 978-3-540-68158-8 (eBook); 978-3-540-68156-4 (print).

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space*. arXiv preprint arXiv:1301.3781.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, **Attention Is All You Need,** NIPS'17

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., & Amodei, D. (2017). *Deep Reinforcement Learning from Human Preferences.* In *Advances in Neural Information Processing Systems* (NeurIPS), 30.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv preprint arXiv:1810.04805.

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving Language Understanding by Generative Pre-Training*. OpenAI.

GPT-2 (2019): Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language Models are Unsupervised Multitask Learners*. OpenAI.

GPT-3 (2020): Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., … & Amodei, D. (2020). *Language Models are Few-Shot Learners*. In *Advances in Neural Information Processing Systems* (NeurIPS).

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., … & Lowe, R. (2022). *Training language models to follow instructions with human feedback.* arXiv preprint arXiv:2203.02155.