NLP: Vorstellung Assignment

KI Labor - Wintersemester 2022

Stefan Käser, Jochen Gietzen, Maximilian Blanck,
Adrian Westermaier, Tim Bossenmaier, Pascal Fecht

Karlsruhe, 18. November 2022

Zeitplan

Datum	Thema	Inhalt	Präsenz
30. Sept..	**Allg.**	**Organisation, Teamfindung, Vorstellung CV**	**Ja**
7. Okt.	Ausfall (DMA Techday)		
14. Okt.	CV	Q&A Sessions	Nein
21. Okt.	CV	Sprintwechsel, Vorstellung Assignment	Ja
28. Okt.	CV	Q&A Sessions	Nein
4. Nov.	**CV / NLP**	**Abgabe CV, Vorstellung NLP**	**Ja**
11. Nov.	NLP	Q&A Sessions	Nein
18. Nov.	NLP	Sprintwechsel, Vorstellung Assignment	Ja
25. Nov.	NLP	Q&A Sessions	Nein
2. Dez.	Ausfall (Winter Plenum)		
9. Dez.	**NLP / RL**	**Abgabe NLP, Vorstellung RL**	**Ja**
16. Dez.	RL	Q&A Sessions	Nein
23. Dez.	RL	Sprintwechsel, Vorstellung Assignment	Ja / Nein
13. Jan.	RL	Q&A Sessions	Nein
20. Jan.	**RL**	**Abgabe RL, Abschluss KI Labor**	**Ja**

inovex

Agenda

- **Besprechung Übungsaufgaben**
  - Word Embeddings Alice im Wunderland (Aufgabe 1)
  - Sentiment Analyse für Twitter Posts (Aufgabe 2)
- **Vorstellung Assignment**
  - Fine-Tuning *bzw.* Prompting mit Transformern

inovex

Übungsaufgaben

inovex

Theorie

Scenario for this lecture

- *Task*: Detecting sentiment on poem verses
- Dataset from *Investigating Societal Biases in a Poetry Composition Systems*; Emily Sheng, David Uthus; 2011.02686
- Around 1100 crowd-sourced samples

Verse	*Sentiment*
that has a charmingly bourbon air.	Positive
ah, what a pang of aching sharp surprise	Negative
down in the west upon the ocean floor	No Impact (Neutral)

inovex

Examples in this lecture:

- **Implement** sentiment analysis for english poem verses
  - **Fine-Tune** a pre-trained model
  - Use **prompting** for a pre-trained model
- Compare different approaches and results
- Demo in Google Colab

Pre-Trained Language Models

Pre-Trained Language Models (PLM)

- Language Model (LM): Given a context, predict the next word:
  - the weather was *[MASK]* ⇒ [MASK] = (0.5 hot, 0.3 cold, …)
- Semi-supervised learning task (without labels)
  - Trained on very large datasets.
- **Transformer**-architecture scales up to trillions of parameters
- Large LM encode general knowledge features

inovex

Why “Pre-Trained”?

Pre-Training

Language modeling

Source task

Fine-Tuning or Prompting

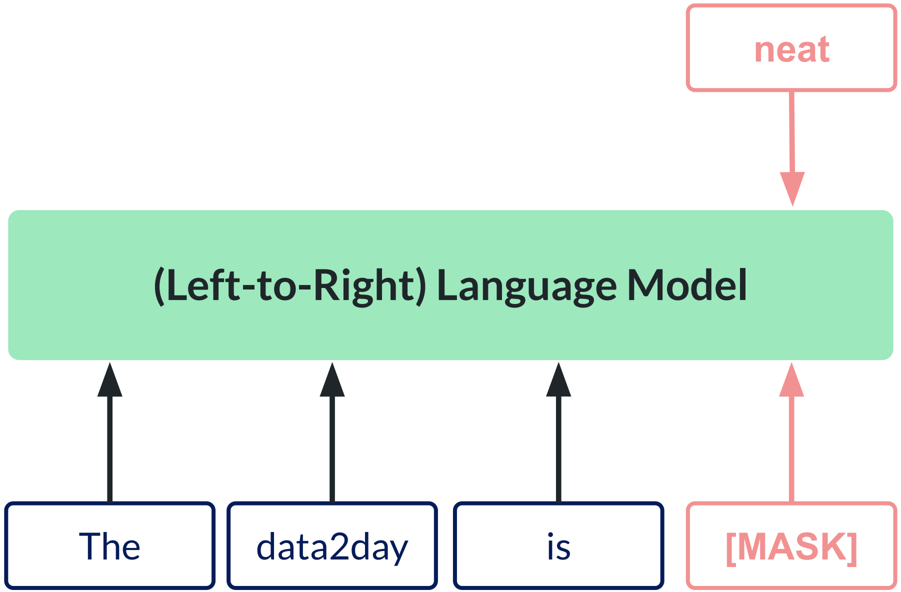
Sentence Classification

Token Classification (NER)

...

Target task(s)

Left-to-Right language models



Predicts the next token given a sequence of tokens.

Models

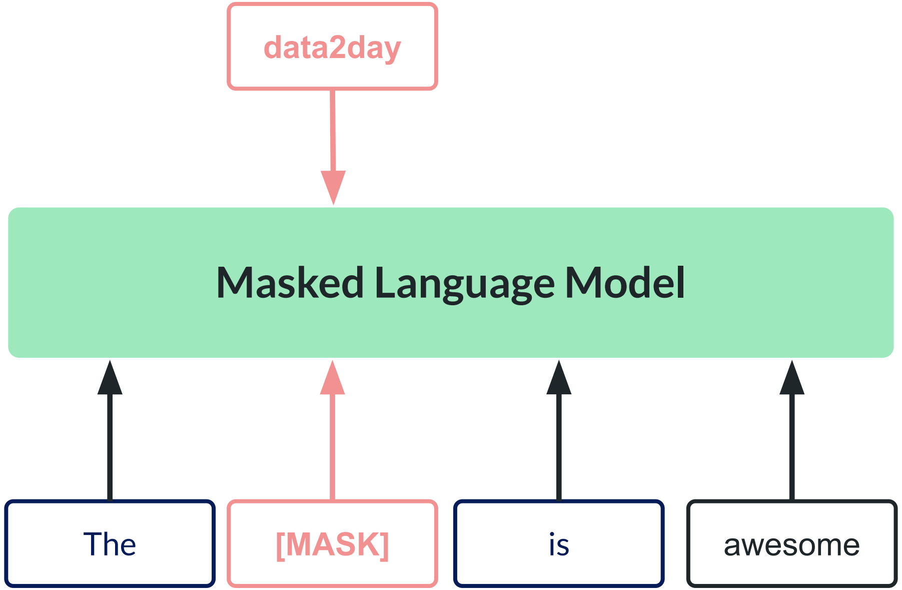
- GPT-2, GPT-3

(Main) application

- Text generation

inovex

Masked Language Model (MLM)



Predicts a masked token in a sequence of tokens (cloze task).

Models

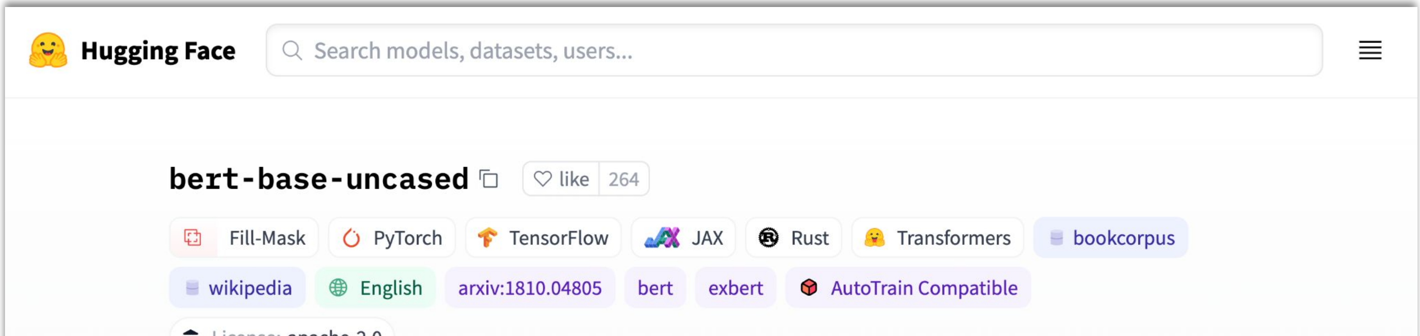
- BERT

(Main) application

- Classification

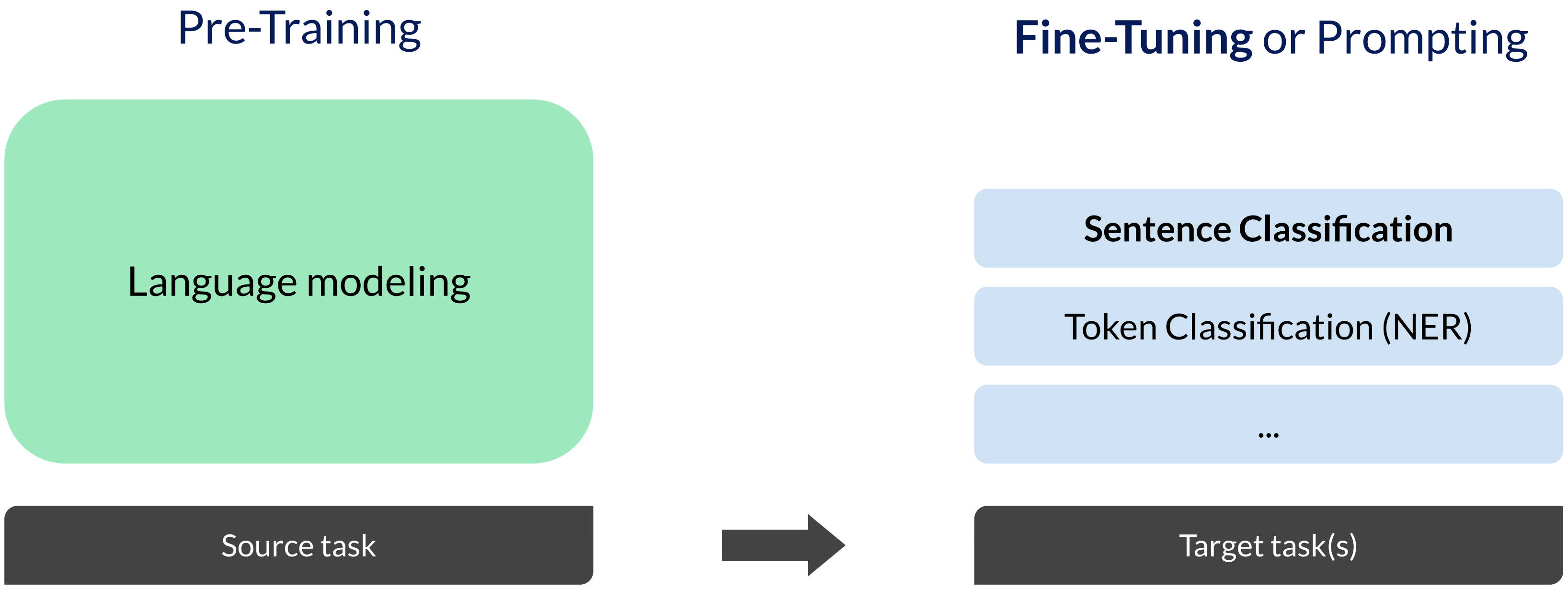
Our Scenario: Model and dataset

- Selecting the type of model
  - **Masked language model** for sentiment analysis
  - load [bert-base-uncased] from [huggingface transformers]
- Prepare the dataset
  - Load the poem sentiment dataset with [datasets]
  - Tokenization

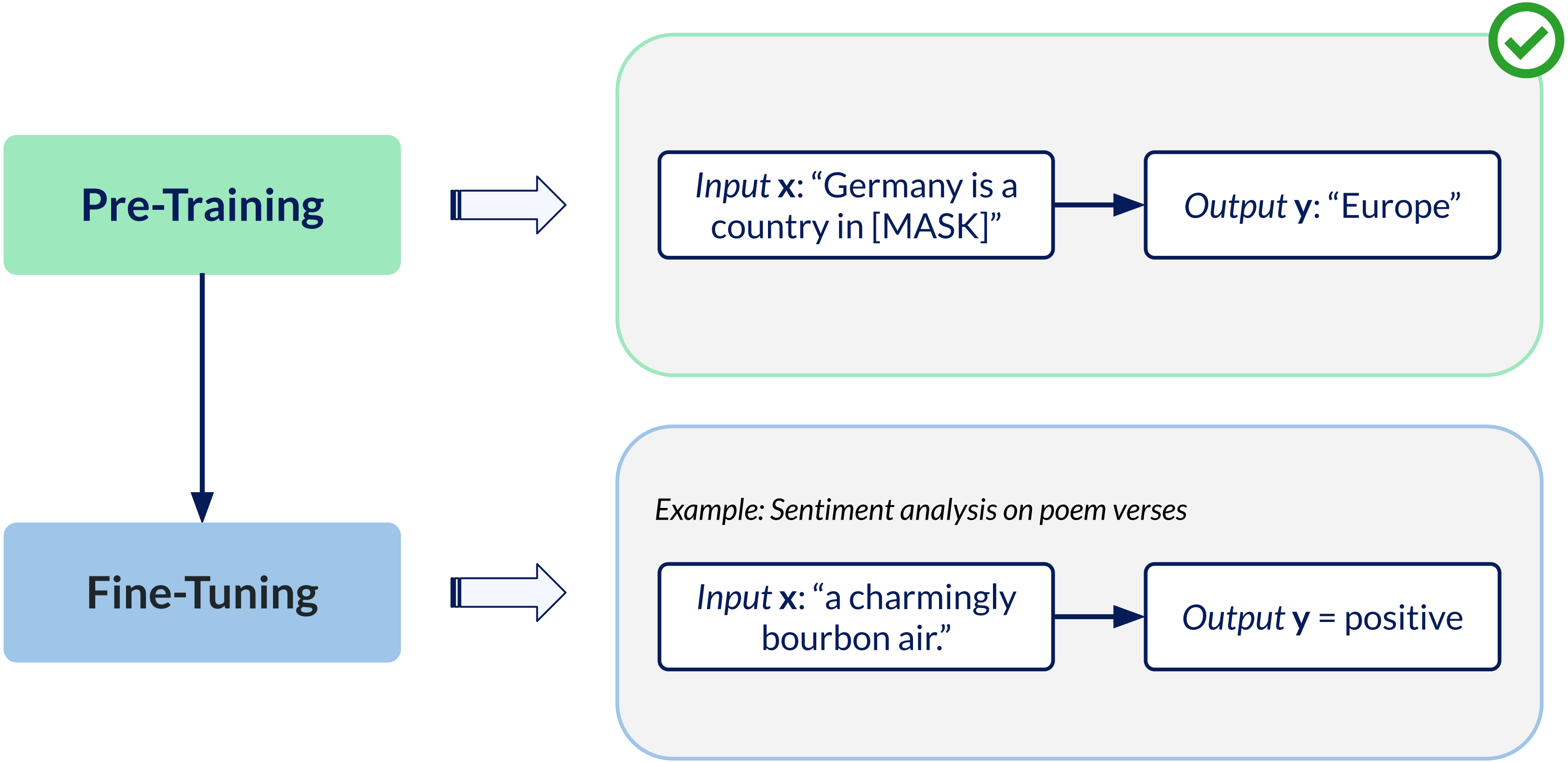


Fine-Tuning

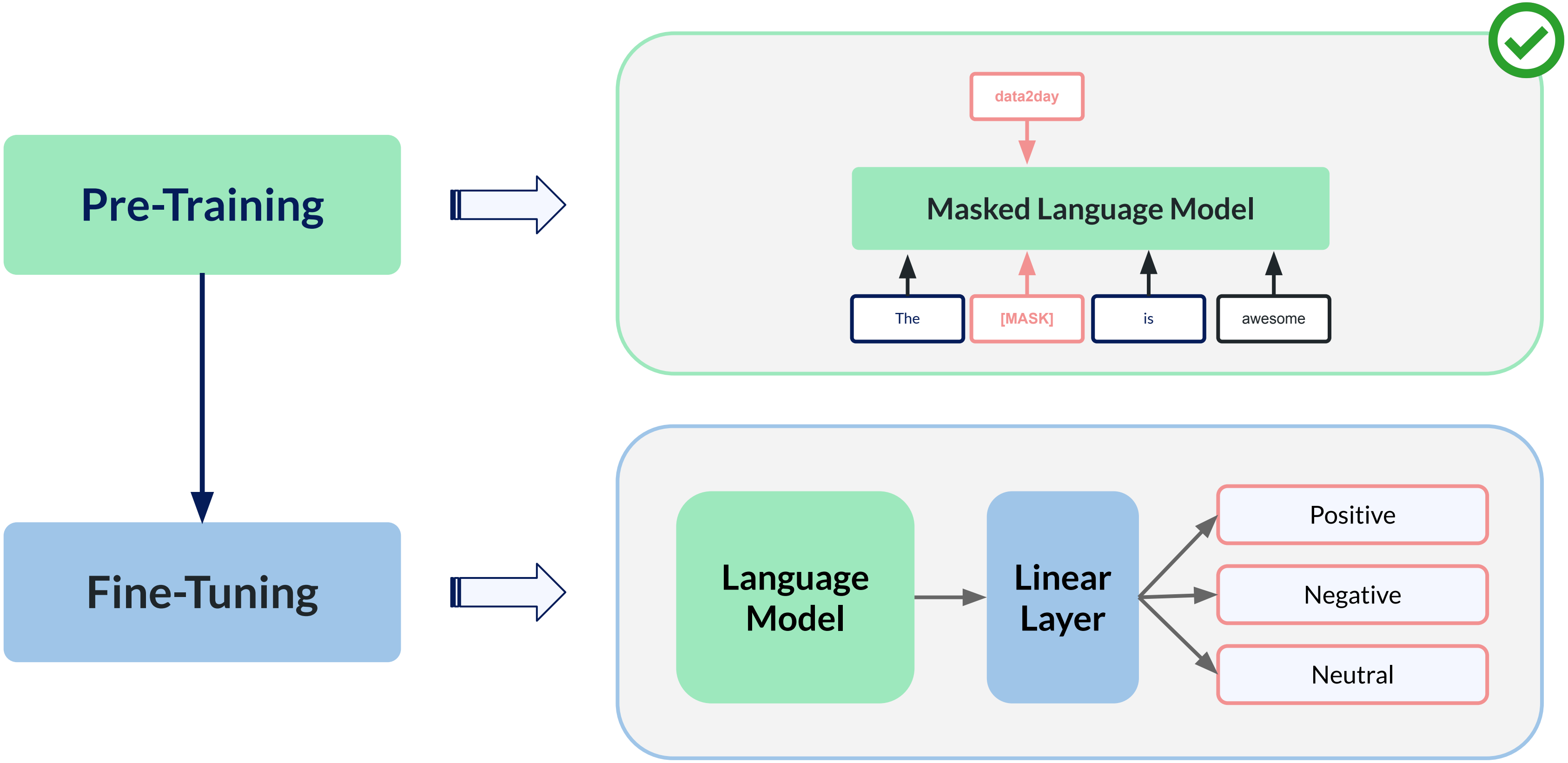
Pre-Training and Fine-Tuning



Fine-Tuning: Sentiment Analysis



Pre-Training and Fine-Tuning



Our Scenario: Fine-Tuning the model

Pre-Training

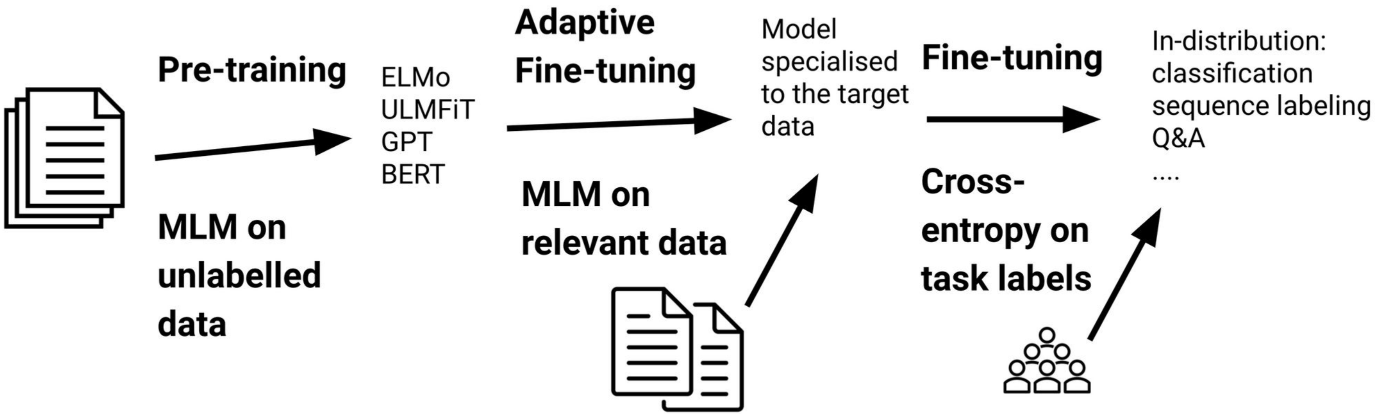
Download `bert-base-uncased` model from huggingface model hub.

Fine-Tuning

Fine-tune `bert-base-uncased` for text classification and the sentiment task.

Adaptive Fine-Tuning

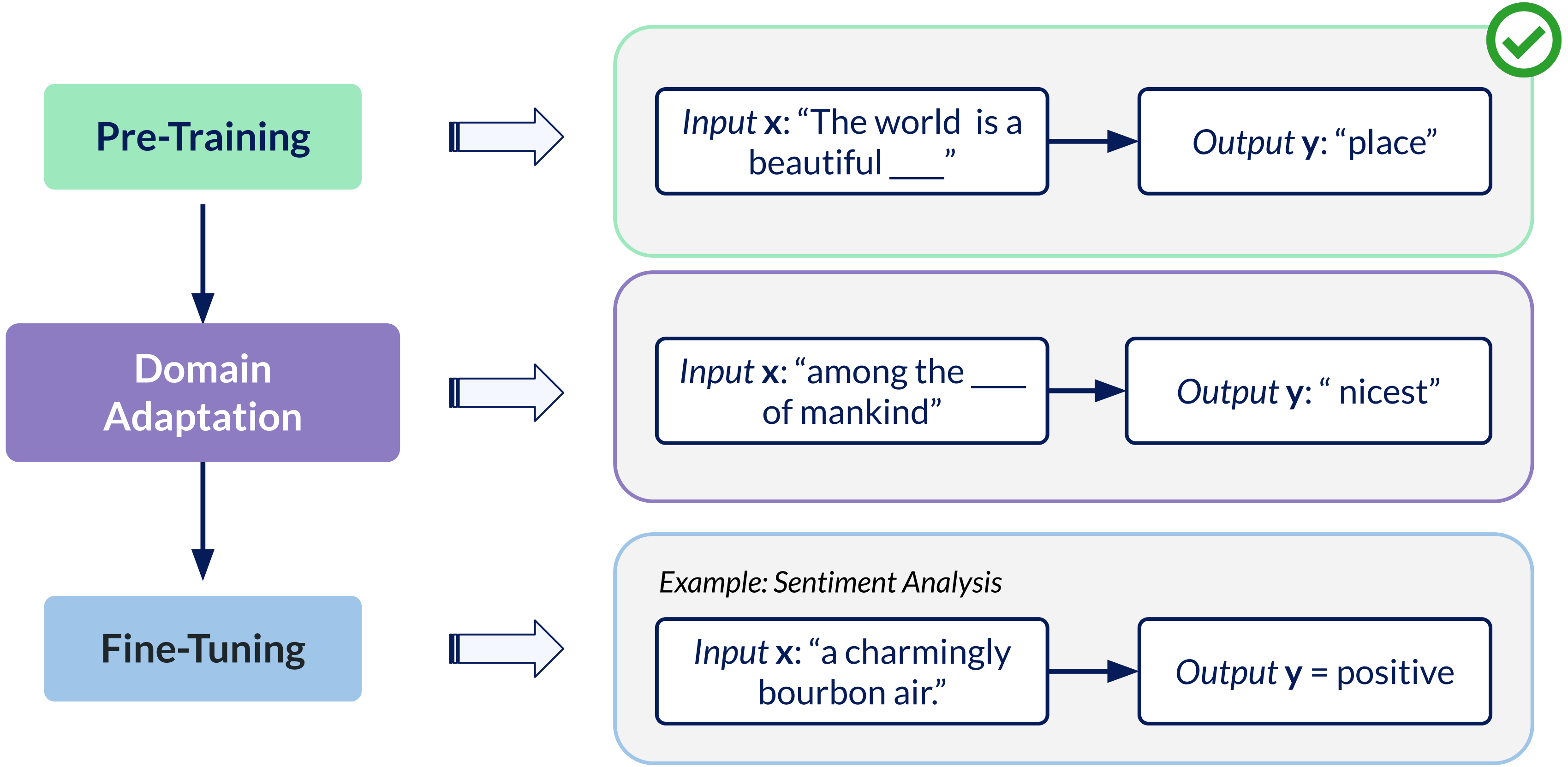
- Additional step to continue the pre-training on **different data**
- Can be **domain-**, **task-** or **language-specific**.
- Improves performance while losing the generalization capabilities.



https://ruder.io/recent-advances-lm-fine-tuning/

inovex

Adaptive Fine-Tuning: Domain Adaptation



inovex

Our scenario: Domain Adaptation

Pre-Training

Download `bert-base-uncased` from huggingface model hub.

Domain Adaptation

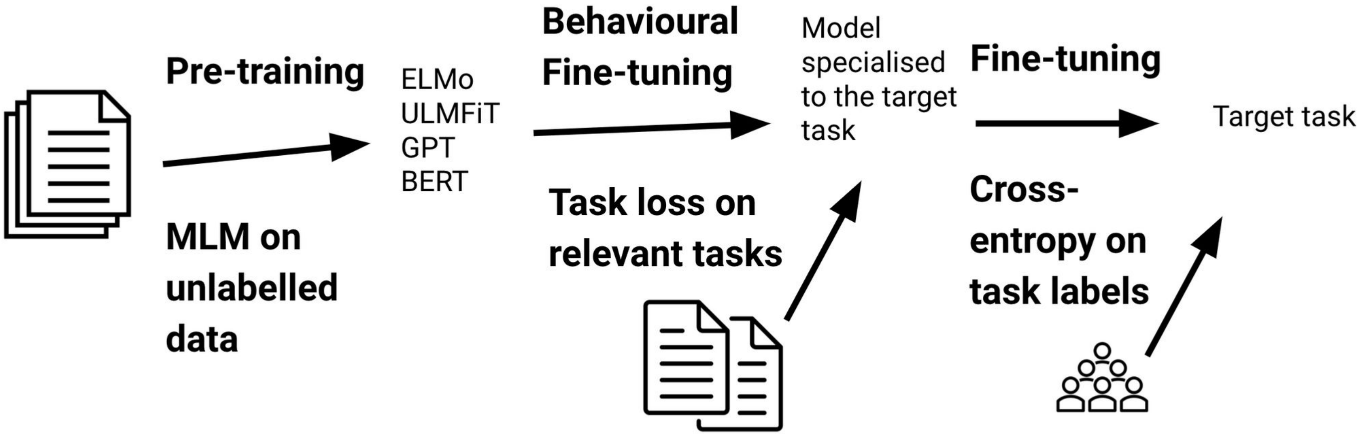
Continue Pre-training (!) with public available poem dataset (not the poem sentiment used for fine-tuning!)

Fine-Tuning

Fine-tune `adapted-lm` on sentiment task with sentiment poem dataset (as before).

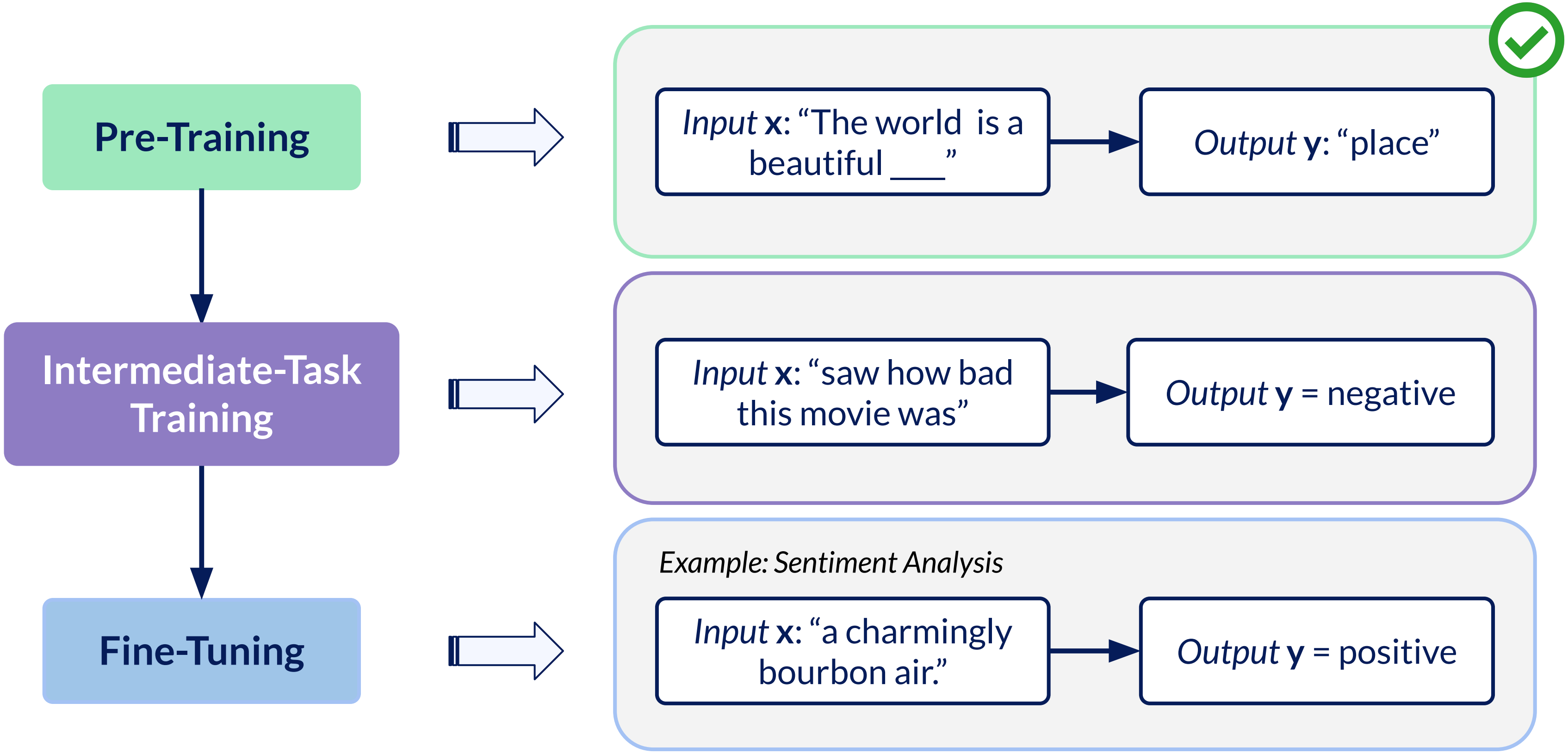
Behavioural Fine-Tuning

- Additional step to teach the model **task-specific behaviour**
- **Intermediate-task training**: Supervised learning of a related task (for instance for named entities)



https://ruder.io/recent-advances-lm-fine-tuning/

Intermediate-task training



inovex

Our scenario: Intermediate-task training

Pre-Training

Intermediate-Task Training

Fine-Tuning

Use model `bert-base-uncased-sst2`

- provided by the community (!)
- trained on SST2 and GLUE dataset

Fine-tune `bert-base-uncased-sst2` on sentiment task with poem dataset.

inovex

Summary: Fine-Tuning

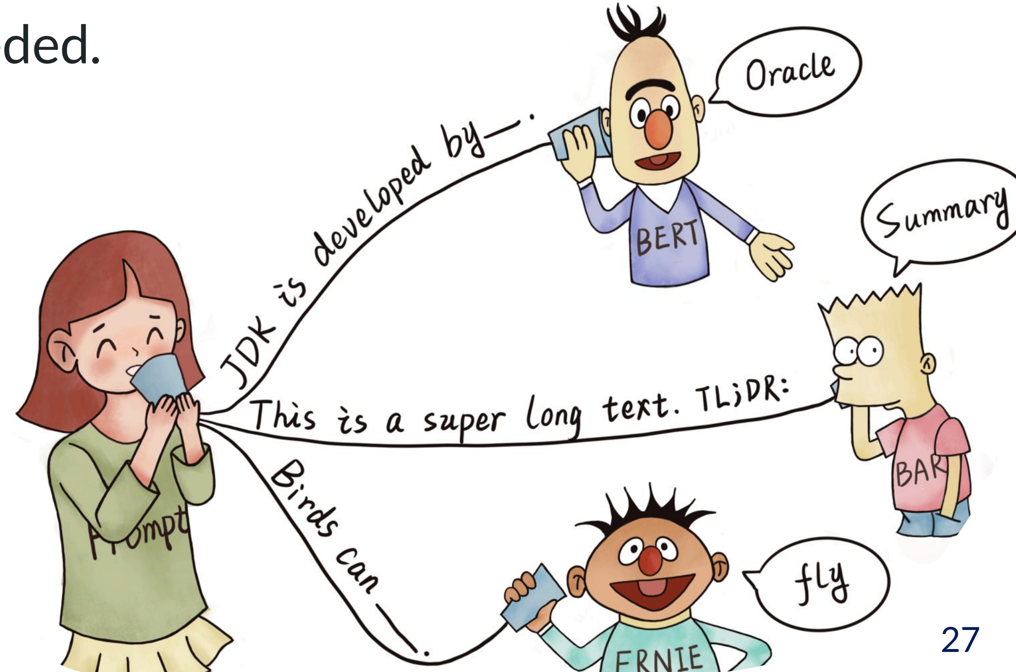
- Fine-Tuning has become fast and straight-forward
- Domain adaptation can help to transfer to a different domain / language
- Intermediate task-training can teach task-specific knowledge .

Prompting

Prompting Overview

Define a **prompt** to formulate the original task as language modeling problem.

- No architectural changes needed.
- Popularized by GPT-3
- Mainly possible with large models



https://arxiv.org/pdf/2107.13586.pdf

Zero- / Few-Shot-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
cheese => ........................ ← prompt
```

Few-shot

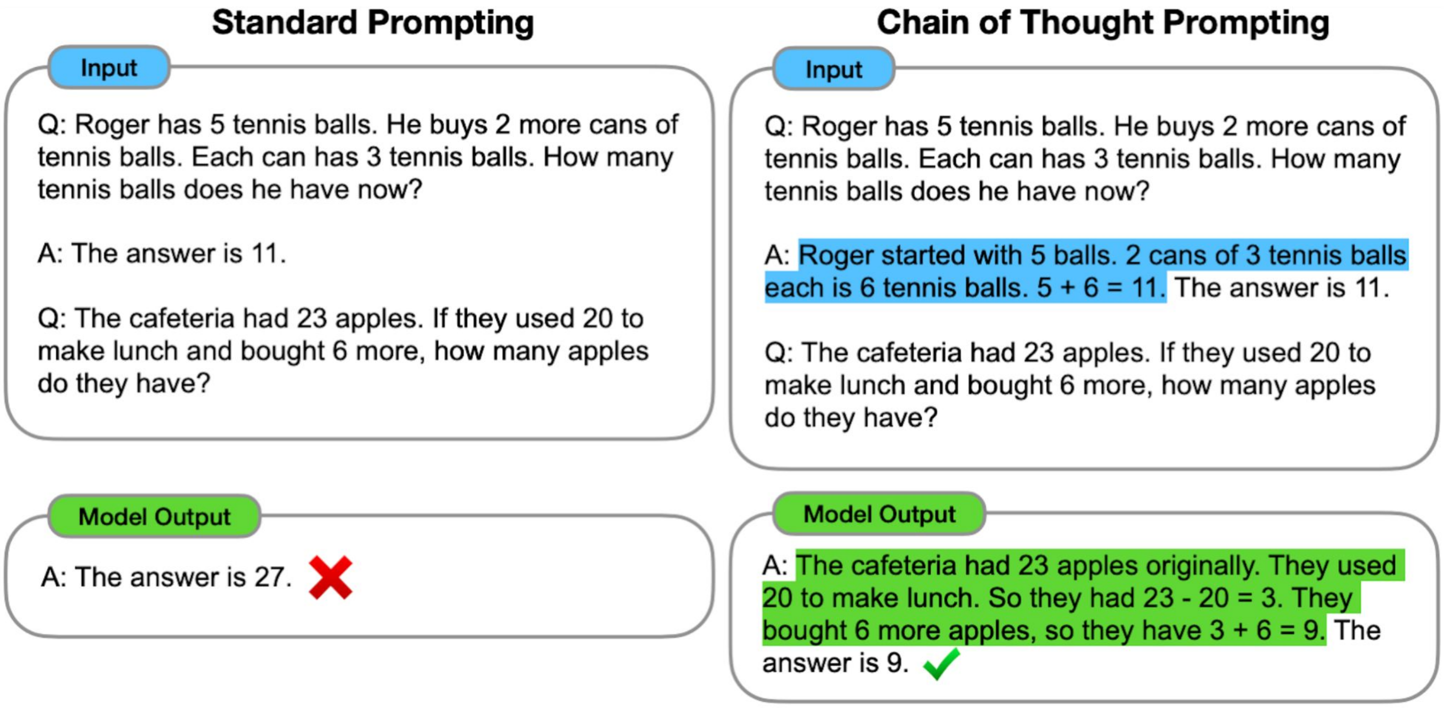
In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
Translate English to French:        ← task description
sea otter => loutre de mer          ← examples
peppermint => menthe poivrée        ←
plush girafe => girafe peluche      ←
cheese => ........................   ← prompt
```

Beispiele: https://beta.openai.com/examples

Language Models are Few Shot Learners, https://arxiv.org/pdf/2005.14165.pdf

Chain-of-Thought Prompting



https://ai.googleblog.com/2022/05/language-maodels-perform-reasoning-via.html

Aspects of Prompting

Prompt Design

Answer Engineering (Verbalizer)

Input **x**: "a charmingly bourbon air."

↓

"**x**. It was such a **[MASK]** feeling." — *Template*

↓

"a charmingly bourbon air. It was such a **[MASK]** feeling." — *Prompting using Pre-Trained LM*

↓

[MASK] = fantastic — *Prediction (most likely word)*

↓

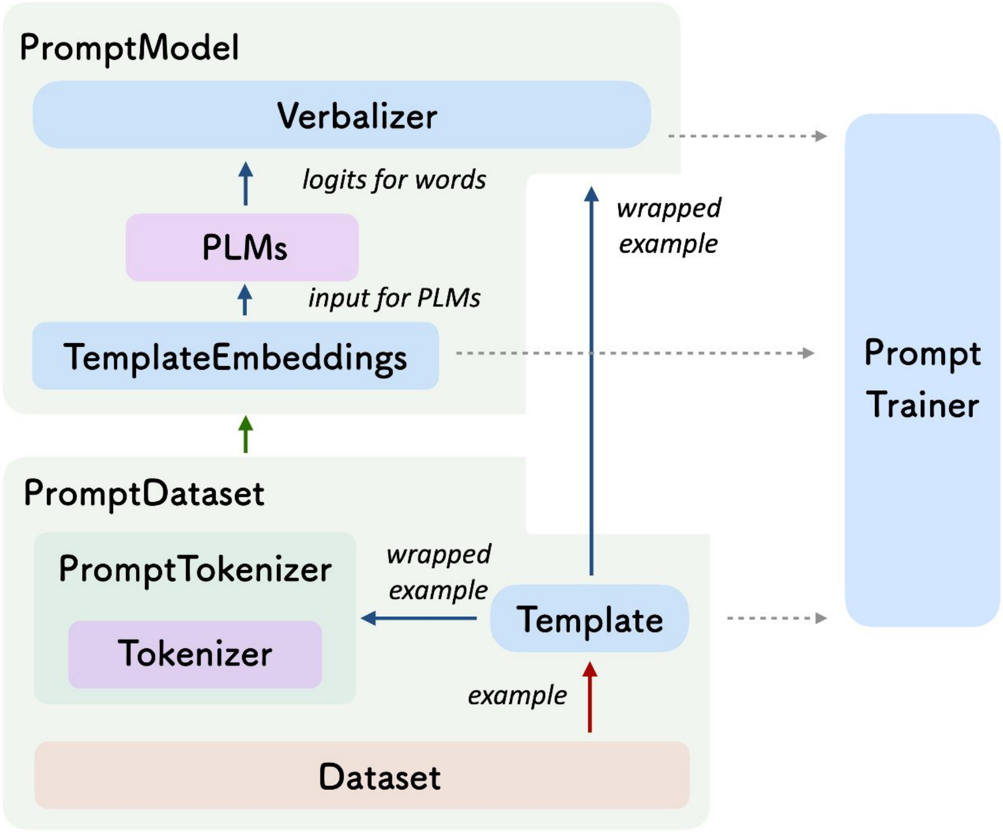
Answer Mapping: **[MASK]** = fantastic ⇒ **y** = positive — *Answer Mapping*

http://phontron.com/class/anlp2021/assets/slides/anlp-10-prompting.pdf

inovex

OpenPrompt

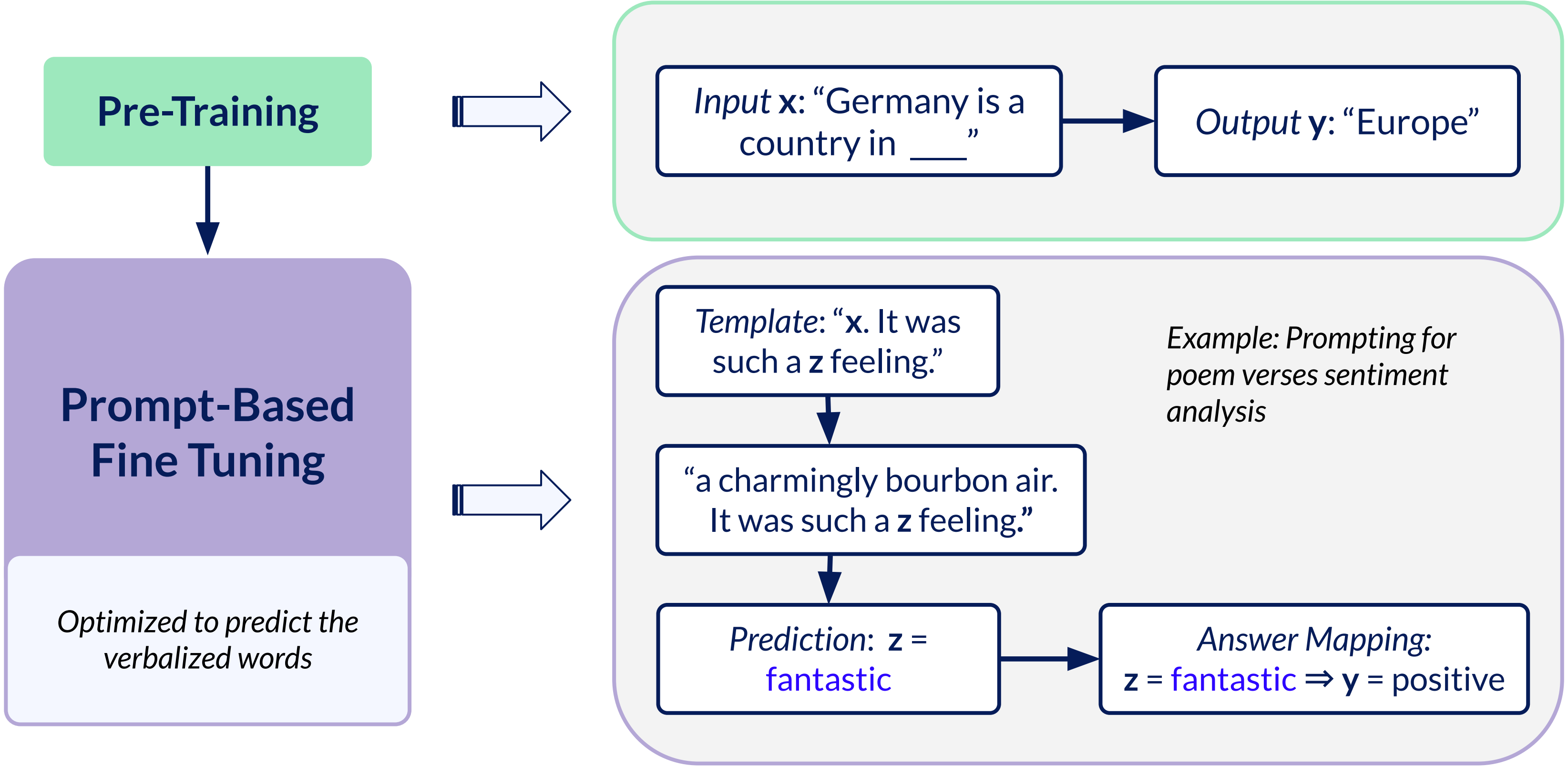
- OS Framework for Prompt Learning
- Simplifies usage and generation of prompts
- Integrates huggingface transformers



https://github.com/thunlp/OpenPrompt

inovex

Prompt-Based Fine-Tuning

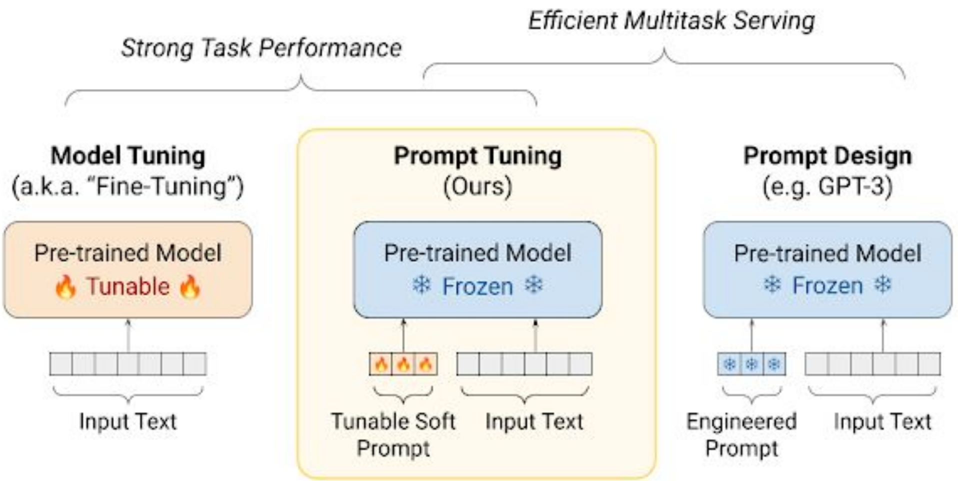


inovex

http://phontron.com/class/anlp2021/assets/slides/anlp-10-prompting.pdf

There is much more in prompting...

- Automatic Prompt Search
- [Soft Prompts] instead of discrete prompts



https://ai.googleblog.com/2022/02/guiding-frozen-language-models-with.html

inovex

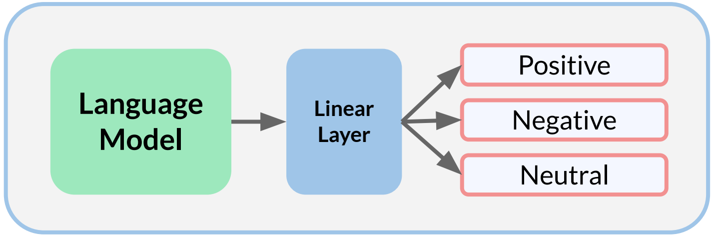
Assignment

Assignment

Open-Ended Assignment Fine-Tuning and/or Promting

Anforderungen

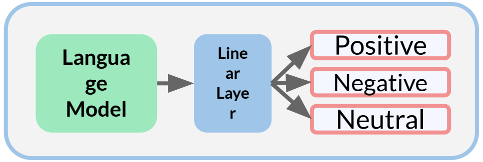
- Wähle passende Transformer-Modelle (https://huggingface.co/) und einen Datensatz zu einem NLP-Task (Translation, Text Generation, Classification, QA, …)
- Datensatz und Task erklären können
- Wähle eine der folgenden Möglichkeiten auf den nächsten Slides



Fine-Tuning

Promting

Assignment



Fine-Tuning

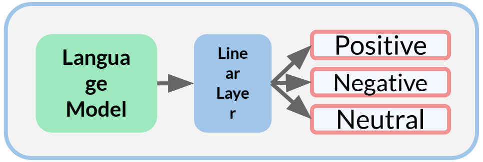
&

Promting

Möglichkeit 1

- Fine-Tuning **und** Prompting
- Beide initialen Ansätze ausprobieren
- Beispiel: Fine-Tuning auf Dataset und Zero-Shot Prompting → Evaluation auf Test daten
- Fokus liegt auf Vergleich der beiden Ansätze
    - Welcher Ansatz funktioniert besser?
    - Wie verändern sich die Ergebnisse bei mehr Daten?
    - Vergleich der Ansätze auf großem vs. kleinem Sprachmodell?
    - …

Assignment



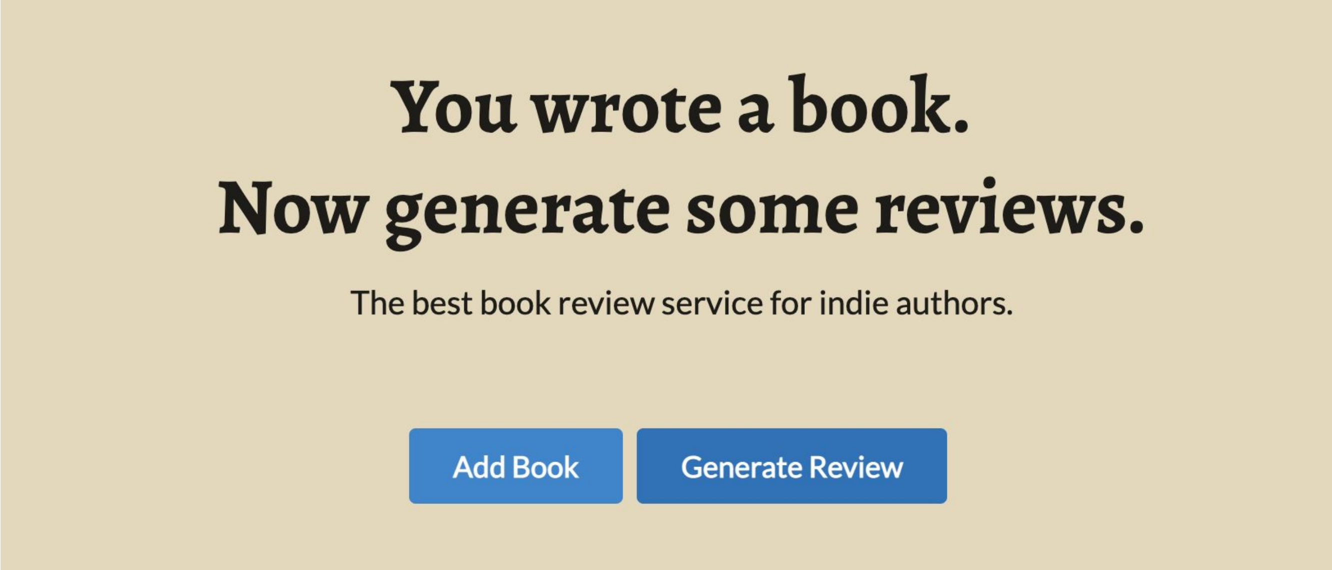
Fine-Tuning

Prompting

Möglichkeit 2

- Fine-Tuning **oder** Prompting
  - Neben den initialen Ansätzen
  - Fine-Tuning: Ansätze wie Domain-Adaption und Intermediate-Task-Training
  - Prompting: Ansätze wie Automatic Prompt-Search, Prompt-Based Fine-Tuning, Parameter-Efficient Fine-Tuning
  - Fokus liegt auf Vergleich der verschiedenen Ansätze
    - Bspw. Wie verändern sich die Ergebnisse, wenn Domain Adaptation bzw. Prompt-Based Fine-Tuning verwendet wird?
    - Welcher Ansatz liefert die besten Ergebnisse?

Beispiel: Generierung von Book Reviews



https://www.bookreview.io/

Assignment

- Quellen zur Inspiration
  - https://beta.openai.com/examples
  - https://www.buildgpt3.com/
  - https://paperswithcode.com/methods/area/natural-language-processing
  - https://huggingface.co/models
- Weitere Links
  - Recent advances in Fine-Tuning: https://ruder.io/recent-advances-lm-fine-tuning
  - Tutorial auf ACL22 zum Thema Prompting: https://github.com/allenai/acl2022-zerofewshot-tutorial/blob/main/acl2022-zerofewshot-tutorial.pdf

inovex

Vielen Dank

inovex GmbH
Ludwig-Erhard-Allee 6
76131 Karlsruhe

inovex