github.com/chkla/Transformers-MZES

@ MZES

29.3.2023

Transformer-based Language M🤖dels

Christopher Klamm

🌍 klamm.ai
⌨️ github/chkla
🐤 twitter/chklamm
🤗 huggingface.co/chkla
🦣 sigmoid.social/chklamm

github.com/chkla/Transformers-MZES

@ MZES

29.3.2023

Transformer-based Language M🤖dels

“What a time for language models” (Alammar 2023)

Christopher Klamm

🌍 klamm.ai
⌨️ github/chkla
🐤 twitter/chklamm
🤗 huggingface.co/chkla
🦣 sigmoid.social/chklamm

Questions

- What are **Language Models**?
- Why do we **need a new architecture (Transformer)** for Language Models?
- What **components** make transformer architecture so powerful?
- What are the **differences** between transformer-based Language Models?
- How can we **use SotA transformer models** for various research tasks?
- What are the **limits and open challenges** of these new types of Language Models?

What are Language Models?

Transformer-based Language Models

Walsh (2021) ↘

@ MZES

Transformer-based Language Models

- a **new architecture** of **(language) models** 🤖
- trained on a **large amount of (textual) data** 📚
- able to **understand words in context** 🏞️
- show an **amazing practical performance** 🏆 on various tasks

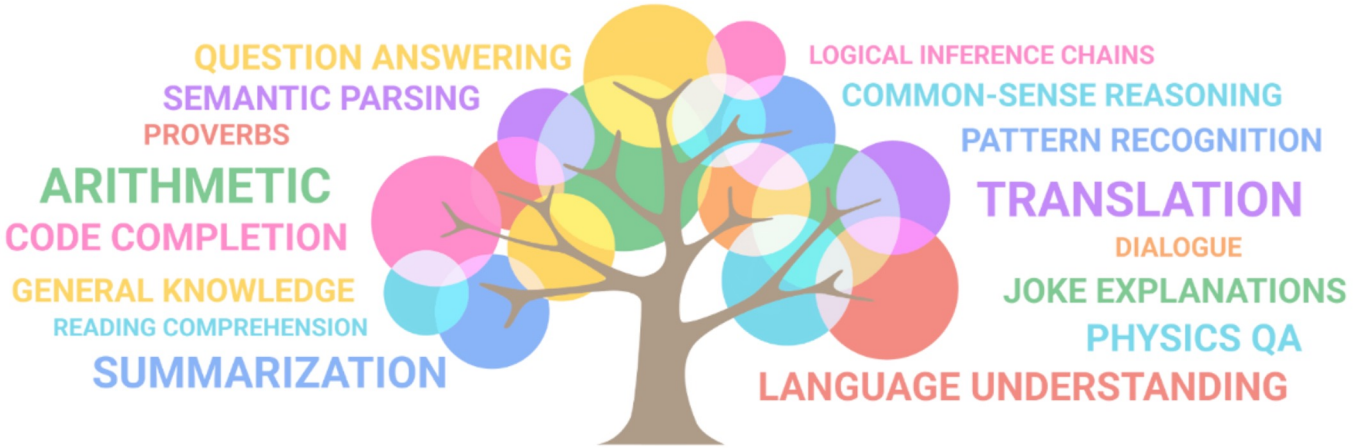


Illustration: https://github.com/Hannibal046/Awesome-LLM

Protopapas/ Glickman/ Tanner (2021) ⇲

@ MZES

Language Model

A Language Model represents the language used by a given entity.

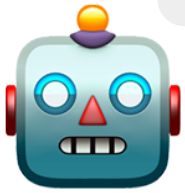
Language Model

A Language Model represents the language used by a given entity.

Hi, I'm Lisa, I read *several english books* 📚 *on politics* in my summer break.

Language Model

A Language Model represents the language used by a given entity.



Hi, I'm BERT, I'm trained on thousands of *english books* 📚 *on politics*.

Language models are trained on large amounts of text data to learn the patterns and relationships between words and phrases in human language.

Language Model

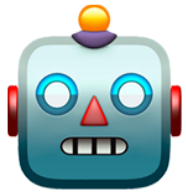
graduation

Cloze test

Obama was born ___1) Honolulu, Hawaii. After _____ 2) from Columbia University ___ 3) 1983, he worked as ___4) community organizer in _____ 5). In 1988, he enrolled ___6) Harvard Law School, _____ 7) he was the _____ 8) black president of ___ 9) Harvard Law Review.

Source: Wikipedia "Barack Obama"

74% graduation
23% graduated
3% gradual

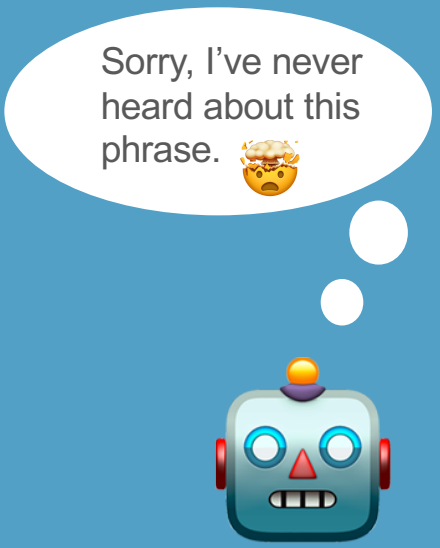


Language Models (LMs) estimate the probability of different linguistic units: symbols, tokens, token sequences.

Formal definition

*Why do **need a new architecture (Transformer)** for Language Models?*

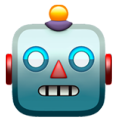
Limits of pre-transformer Language Models

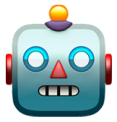


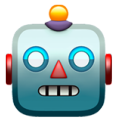
Source: Protopapas/ Glickman/ Tanner (2021) ⇲

@ MZES

Language Model - Types (statistical count-based models)

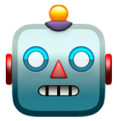
Probability (in  ~~History~_____) *unigram language model*

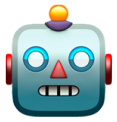
Probability (in  History `born`) *bigram language model*

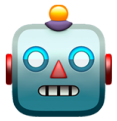
Probability (in  History `was born`) *n-gram language model*

Source: Protopapas/ Glickman/ Tanner (2021) ⇲

Language Model - Types (statistical count-based models)

Probability (in  ~~History~~) *unigram language model*

Probability (in  History `born`) *bigram language model*

Probability (in  History `was born`) *n-gram language model*

What about a **long history** Obama was born in Honolulu, Hawaii ?

Cho (2021) ⇲

@ MZES

Language Model - Problems

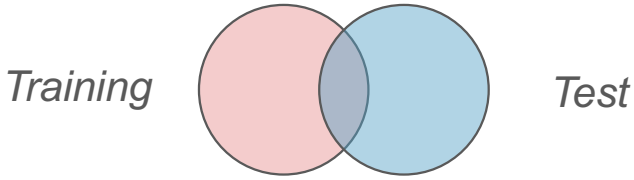
Data Sparsity: Some n-grams do not occur in the training data, while they do in the test time.

Cho (2021) ⇲

@ MZES

Language Model - Problems

Data Sparsity: Some n-grams do not occur in the training data, while they do in the test time.

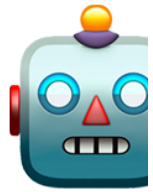


Cho (2021) ⇲

@ MZES

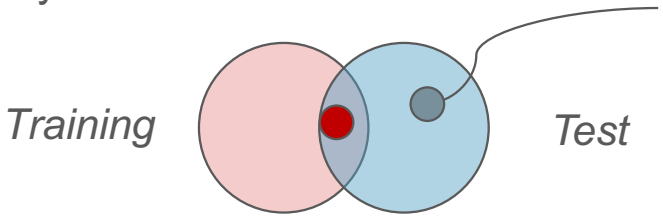
Language Model - Problems

Sorry, I've never heard about this phrase. 🤯



Data Sparsity: Some n-grams do not occur in the training data, while they do in the test time.

P (a dog is chasing a llama)



Cho (2021) ⇲

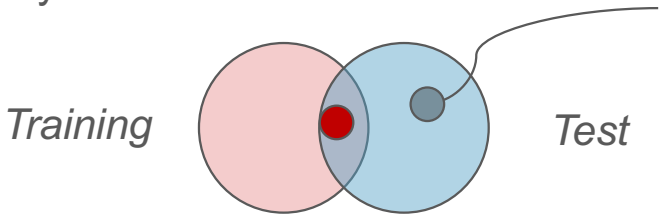
@ MZES

Language Model - Problems

Sorry, I've never heard about this phrase. 🤯

Data Sparsity: Some n-grams do not occur in the training data, while they do in the test time.

P (a dog is chasing a llama)



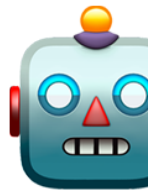
Long-Range Dependencies: Data sparsity leads to the problem that only a short history of previous tokens can be considered.

Cho (2021) ⇲

@ MZES

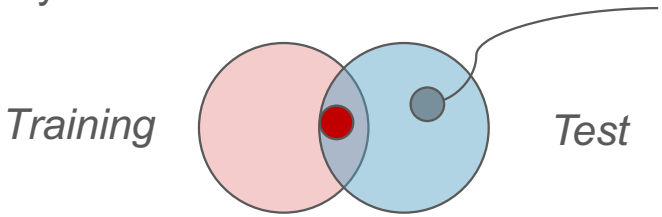
Language Model - Problems

Sorry, I've never heard about this phrase. 🤯



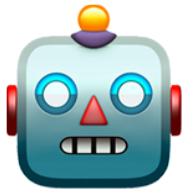
Data Sparsity: Some n-grams do not occur in the training data, while they do in the test time.

P (a dog is chasing a llama)



Long-Range Dependencies: Data sparsity leads to the problem that only a short history of previous tokens can be considered.

Sorry, I can only consider the last n words. 🤯

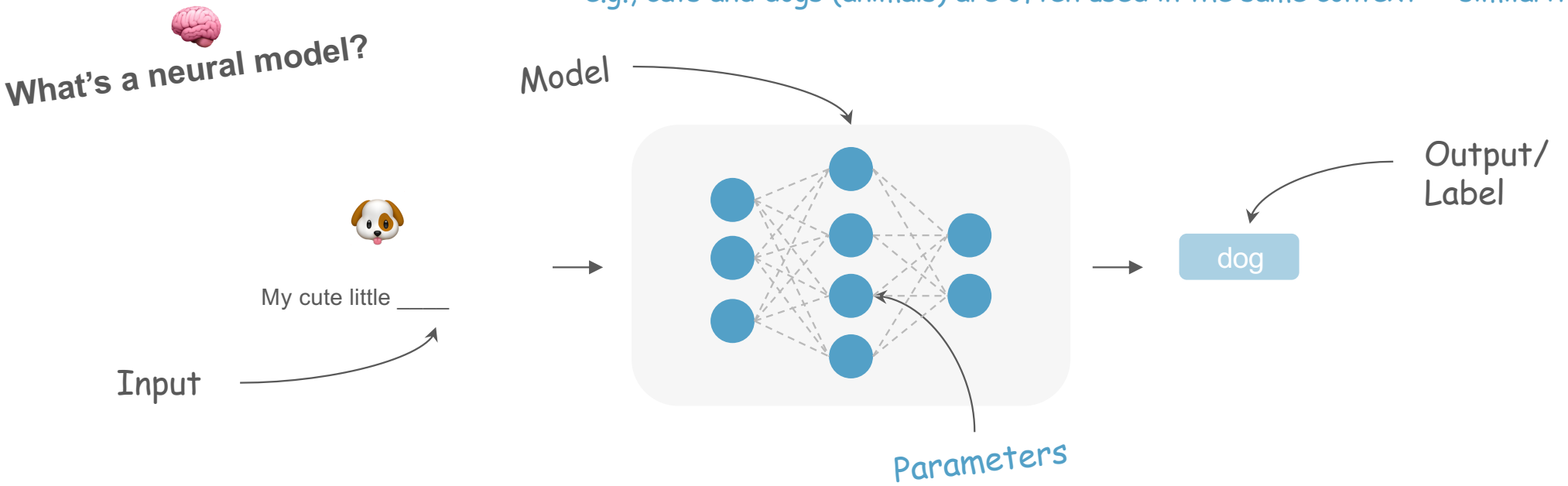


Cho (2021) ⇲

🧠 Neural N-Gram Language Models (Similarity-based)

🚀 The neural n-gram language model addresses the problem of data sparsity by *learning the similarities* among tokens and phrases in a *continuous vector space*.

e.g., cats and dogs (animals) are often used in the same context => similarity



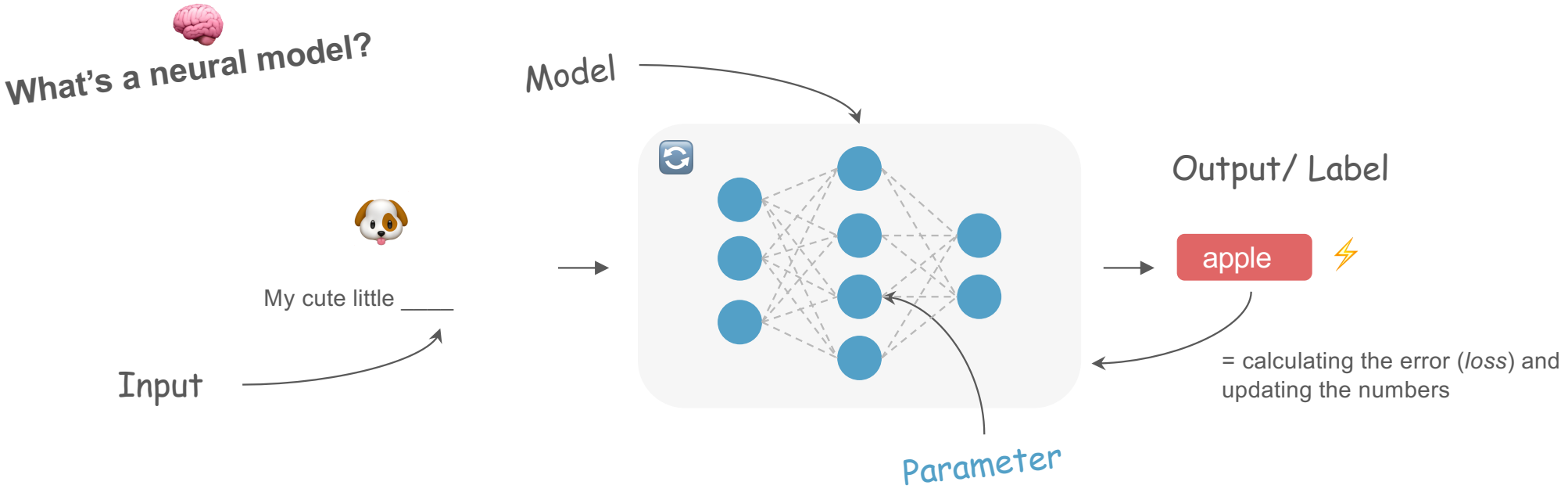
= numbers that determine how a model processes inputs

Cho (2021) ↘

@ MZES

🧠 Neural N-Gram Language Models (Similarity-based)

🚀 The neural n-gram language model addresses the problem of data sparsity by *learning the similarities* among tokens and phrases in a *continuous vector space*.

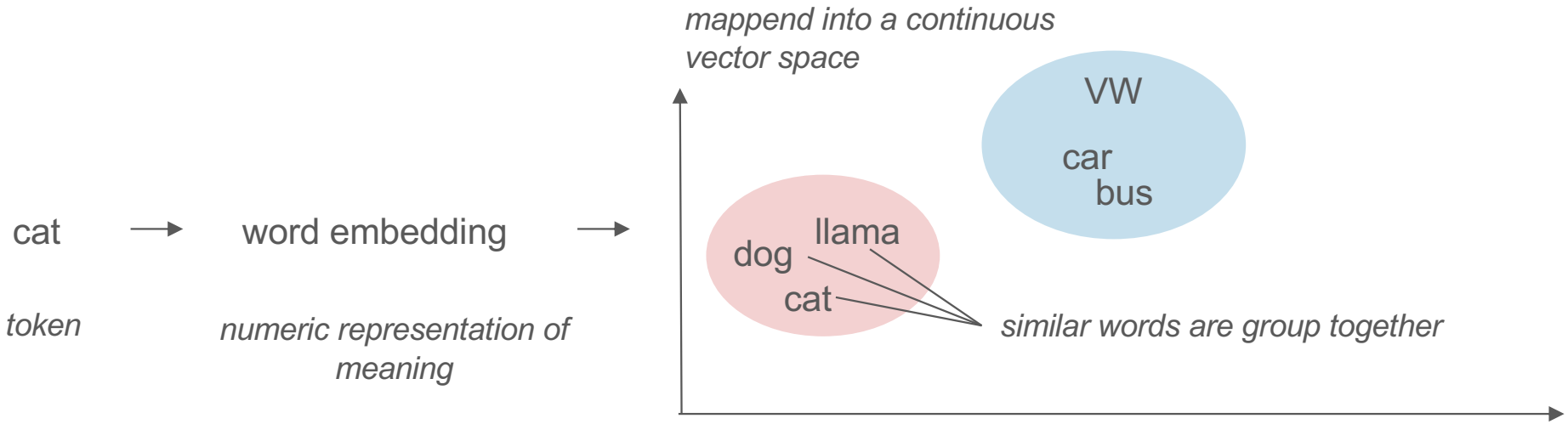


Cho (2021) ⇲

@ MZES

🧠 Neural N-Gram Language Models (Similarity-based)

🚀 The neural n-gram language model addresses the problem of data sparsity by *learning the similarities* among tokens and phrases in a *continuous vector space*.

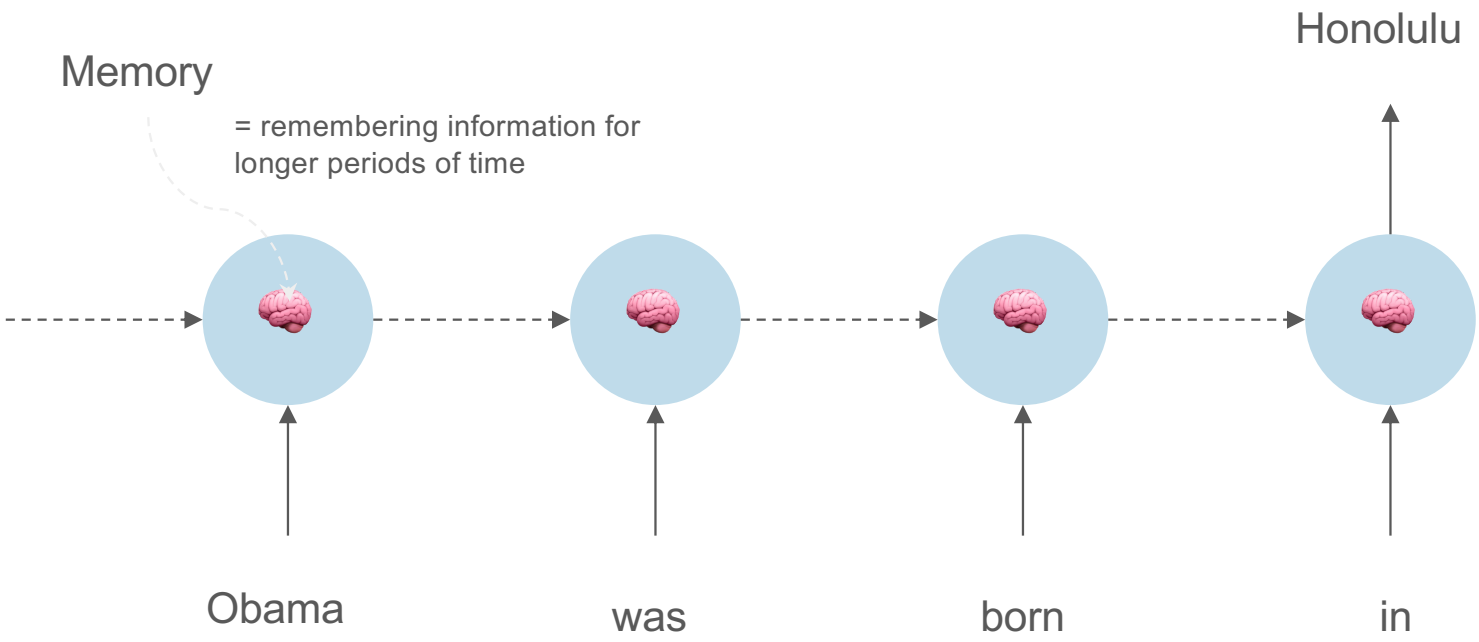


data sparsity → solved ✅

Atienza (2017)
↘

@ MZES

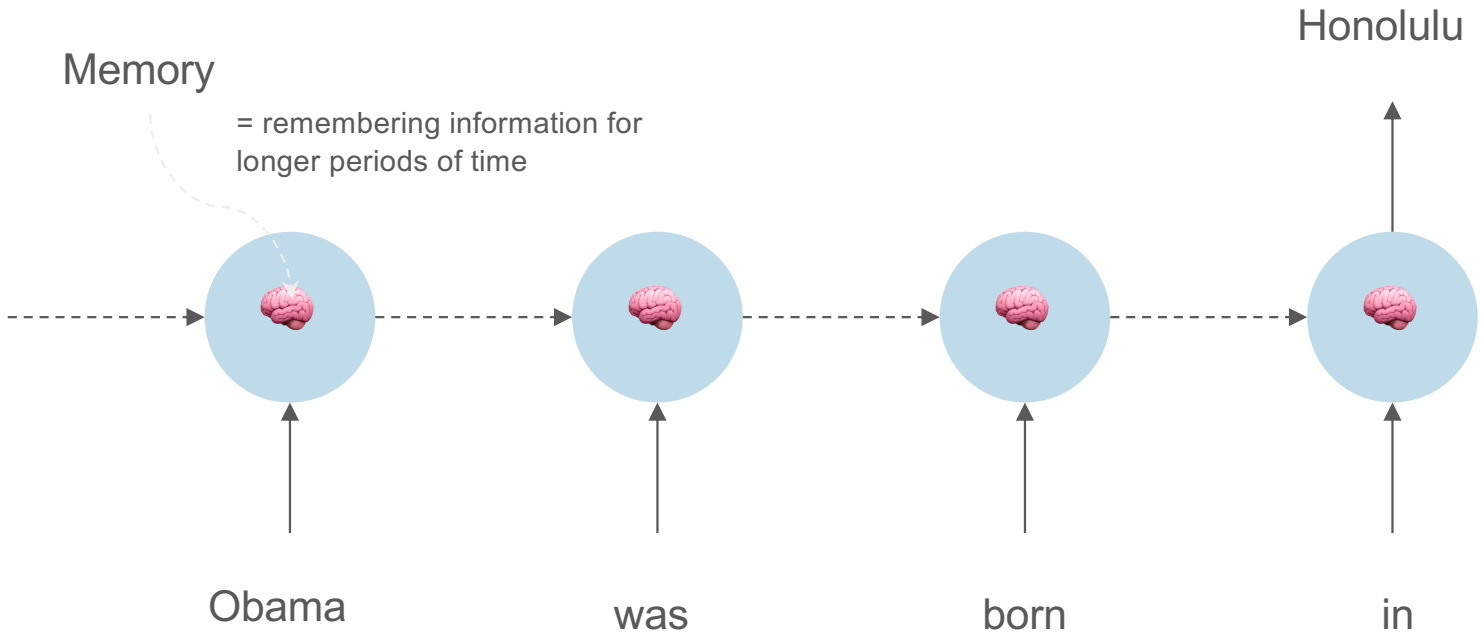
Long-Short Term Memory - Architecture (Simplification !)



Atienza (2017)
↘

@ MZES

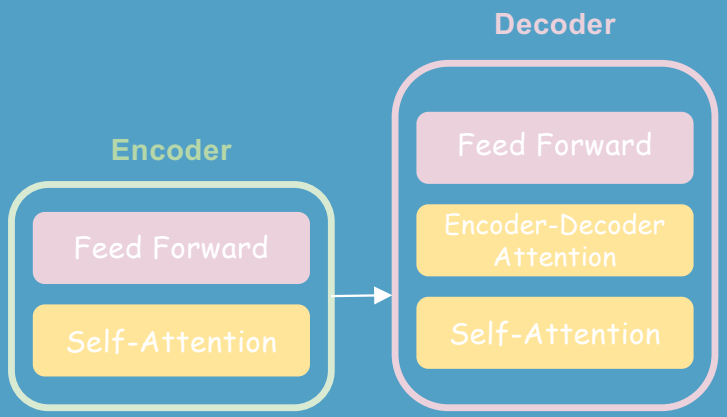
Long-Short Term Memory - Architecture (Simplification !)



LSTMs have limitations when it comes to long-term dependencies, as they can struggle to maintain accurate representations of the input sequence over many time steps ⚡

*What **components** make transformer architecture so powerful?*

Components of Transformer-based Language Models



Allamar (2021) ↘ Devlin et al. 2019 ↘

@ MZES

Transformer-Based Model - Overview

start token

[CLS] Obama was born in Honolulu, Hawaii [SEP]

separator token (placed between each sentence)

Model with millions of numbers to determine how to process inputs.

Transformer-Based Model

BERT

[CLS] Obama ...

sentence embeddings

embeddings

	[CLS]	Obama	…	Hawaii	[SEP]
0	0.115728	-0.150873	0.076831	-0.661984	0.834172
1	0.223339	0.269943	0.009255	-0.858786	0.279408
2	0.394565	0.411511	1.518060	0.053227	0.127079
3	0.021645	-0.369215	-0.217087	-0.251777	0.270489
4	-0.569364	-0.537542	-0.033147	0.340954	-0.452304
...	...	...	...	...	...
763	0.454554	0.539476	0.672731	0.648410	-0.864230
764	-0.177935	-0.192709	-0.515615	-0.102858	0.159125
765	-0.570071	0.058834	-0.369875	0.195014	-0.065941
766	-0.257792	0.029307	-0.338636	-0.121347	-0.979295
767	0.380097	0.632699	0.124689	-0.666084	-0.147284

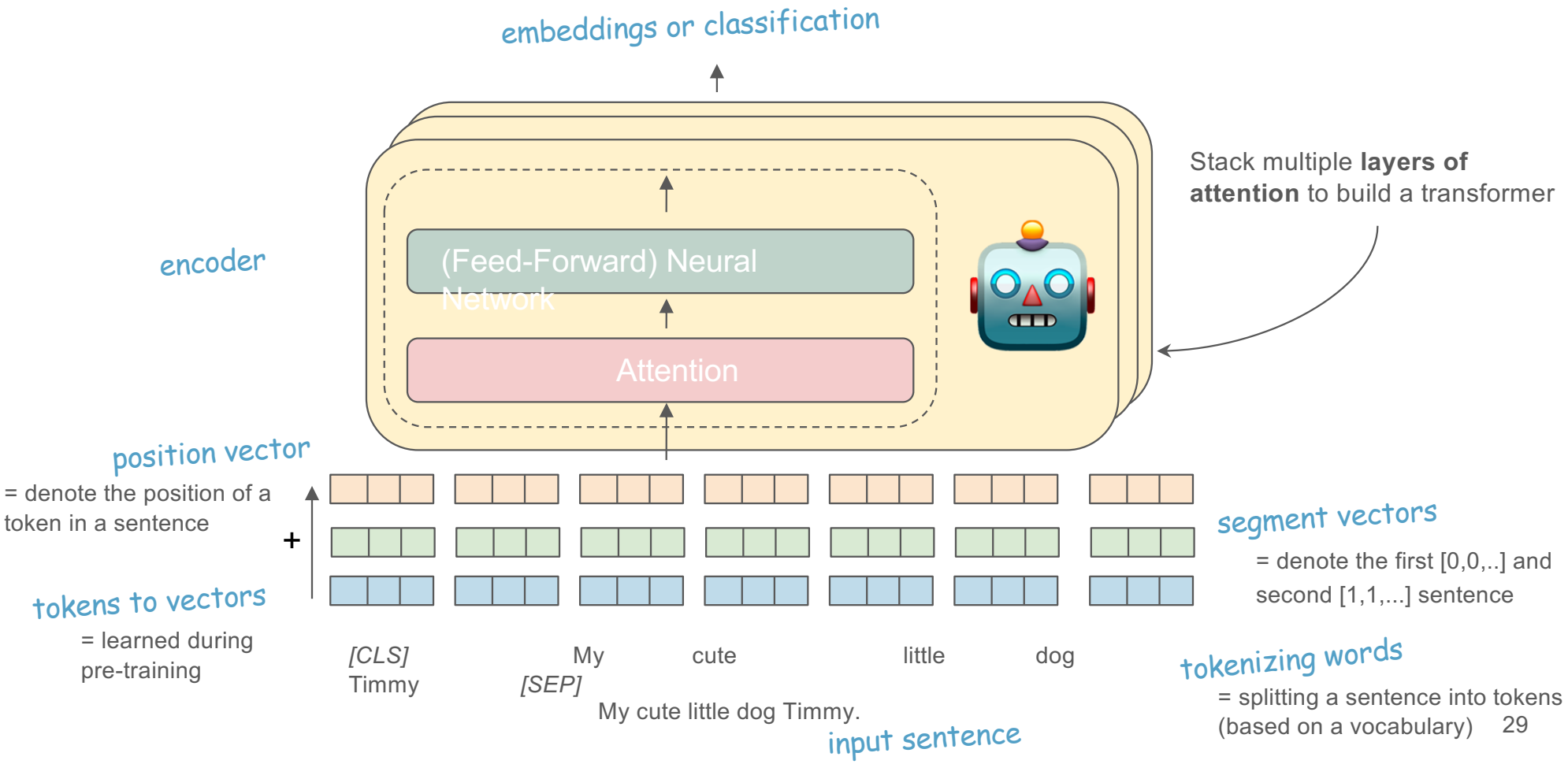
768 rows × 5 columns

= turning text in vectors (words, sentences, etc.)

Alammar (2018), Cho (2021) and Walsh (2021)

@ MZES

Transformer-Based Model - Architecture (Simplification !)



Source: Walsh 2021 ↘

@ MZES

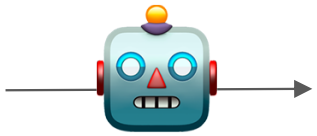
Masked-Language Model

Transformer-Based Model - Pre-Training Procedures

… on a VERY large amount of (textual) resources 📚

[MASK] Token Prediction

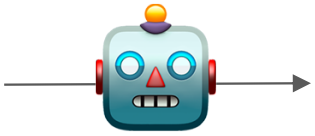
Obama was born in [MASK], Hawaii



Honolulu

Next Sentence Prediction

Obama was born in Honolulu, Hawaii. [MASK]



After graduation from Columbia […].

Sentence 1

Sentence 2

Sentence 2

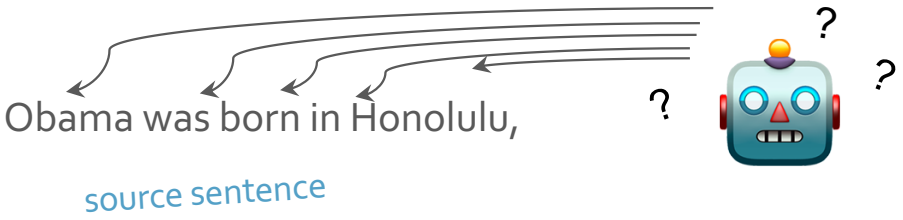
Source: Cho (2021) ↘

@ MZES

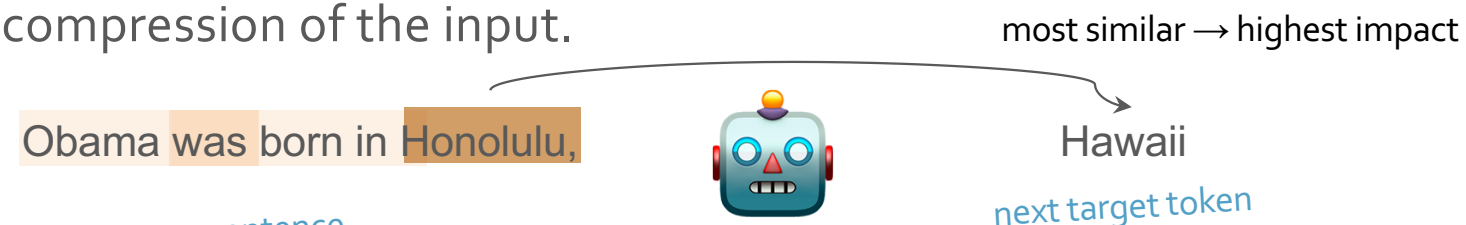
Transformer-Based Model - Architecture

Attention

Which part of the source sentence is relevant for predicting the next target token?



This **weighting function** (attention) allows the model to focus on a subset of the input to avoid the compression of the input.



data sparsity → solved ✅

long-term dependencies/context → (partly) solved ✅

*What are the **differences** between transformer-based Language Models?*

Differences between Transformer-based Language Models

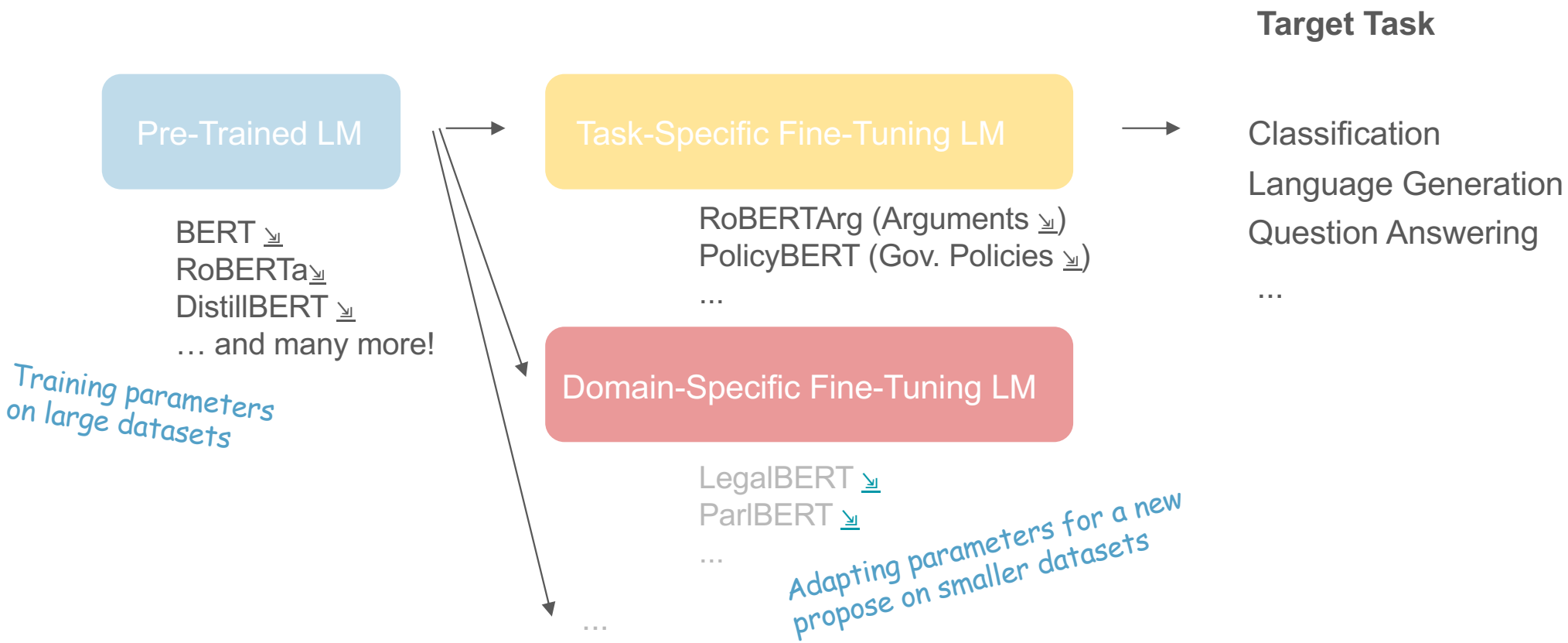
Differences between transformer-based Language Models

- various **training** 🏃‍♀️ **techniques** used to train language models (masked language modelling, instruction-based, etc.)
- **training data** 📚 that is used to train a language model (Twitter, news paper, debates, etc.)
- model **task** 🔨 that a language model is designed to perform (sentiment analysis, natural language inference, text generation, etc.)
- the **number of parameters** 🔢 that can be trained in a language model (BERT 110m, GPT-3 175b, BLOOM 176b, etc.)
- … and many more (https://huggingface.co/docs/transformers/index)

Walsh (2021)

@ MZES

Transformer-Based Language Models - Lifecycle

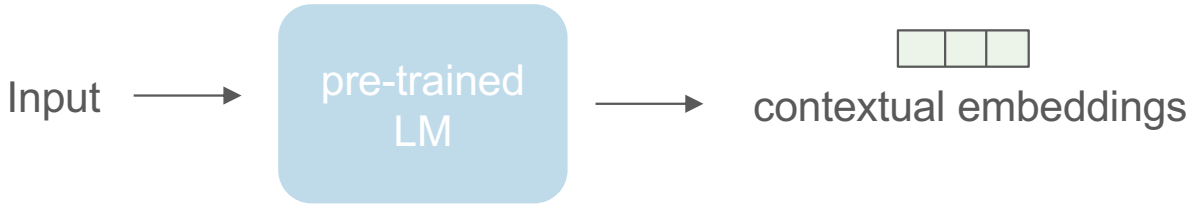


Tatman (2020) ↘

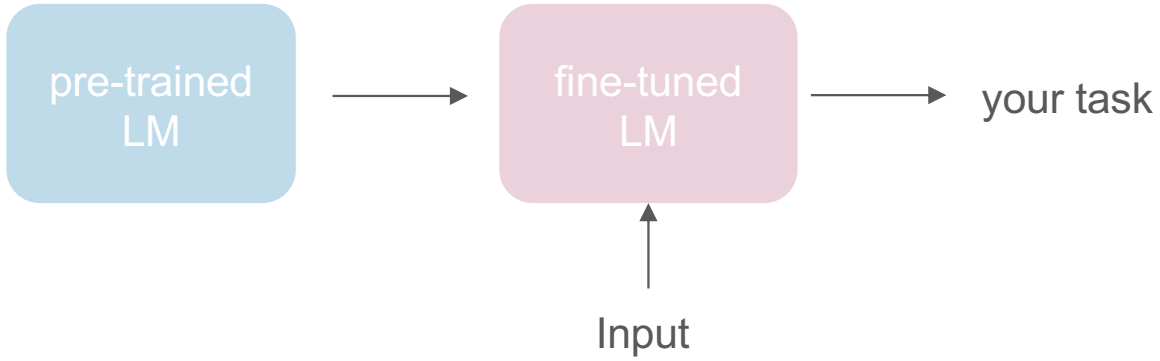
@ MZES

Transformer-Based LMs - embeddings and fine-tuning

- To convert text input into a numeric representation (i.e. to replace word embeddings)



- As a general-purpose pre-trained model that's fine-tuned for specific tasks



Transformer-Based LMs - in-context and instructions

- In-context learning to learn tasks with language models given only a few examples

example-based specification
(Brown et al. 2020)

- a specific prompt or instruction to generate text based on that prompt

instruction-tuned language models
(Wang et al. 2022)

*How can we **use SotA transformer models** for various research tasks?*

Using a transformer-based Language Model with HuggingFace 🤗

HuggingFace 🤗

"We're on a journey to advance and **democratize artificial intelligence** through open source and open science."
(HuggingFace 2021)

- NLP startup
- API
- open-source community 🛋️,
- **models** 🤖 >160k,
- **datasets** 📚 >26k,
- **metrics** 📊,
- **spaces** > 15k 👾
- … and many more!

➡️ https://huggingface.co
➡️ https://huggingface.co/course/ 👩🏼‍🏫

Philosophy

The acceleration in Artificial Intelligence (AI) and Natural Language Processing (NLP) will have a **fundamental impact on society**, as these technologies are at the core of the tools we use on a daily basis. A considerable part of this effort currently stems in NLP from training increasingly larger language models on increasingly larger quantities of texts.

Unfortunately, the resources necessary to create the best-performing models are found mainly in the hands of big technology giants. The stranglehold on this transformative technology poses some problems, from a research advancement, environmental, ethical and societal perspective. …

BigScience Project (2021-2022)

HuggingFace Pipelines ... to use models for inference

- `TextClassificationPipeline`
- `SummarizationPipeline`
- `QuestionAnsweringPipeline`
- `TextGenerationPipeline`

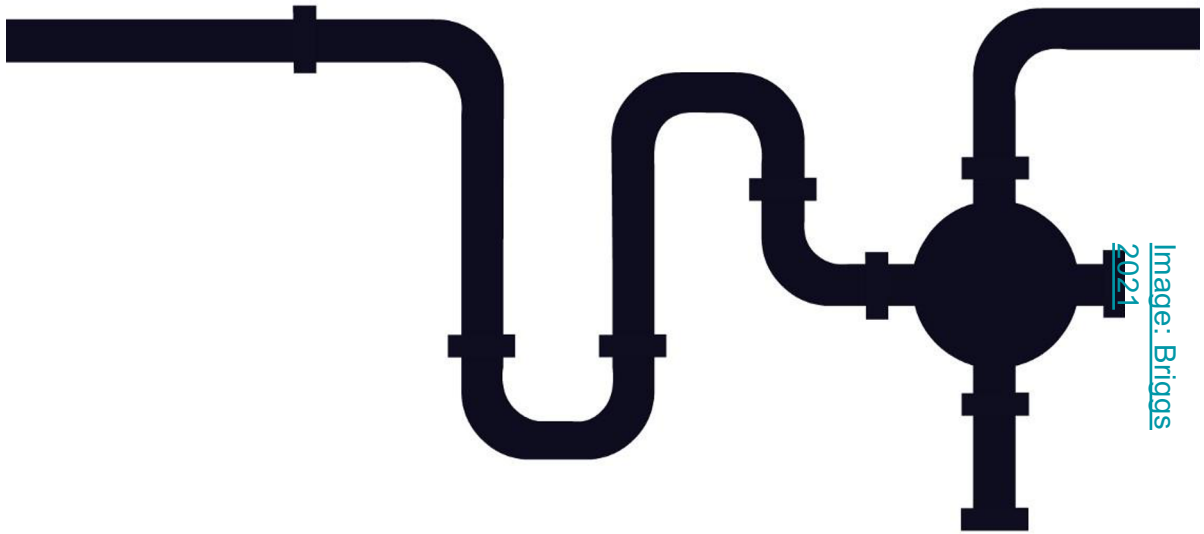


Image: Briggs 2021

➡️ https://huggingface.co/docs/transformers/v4.27.0/en/main_classes/pipelines

🤗 HuggingFace Pipelines

github.com/chkla/Transformers-MZES

TextClassificationPipeline

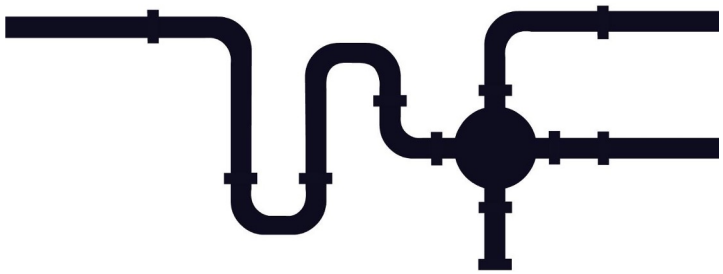


Image: Briggs 2021

load your pipeline

```
pipeline_sentiment = pipeline(
            "text-classification",
            model="distilbert-base-uncased-finetuned-sst-2-english",
            top_k=None)
```

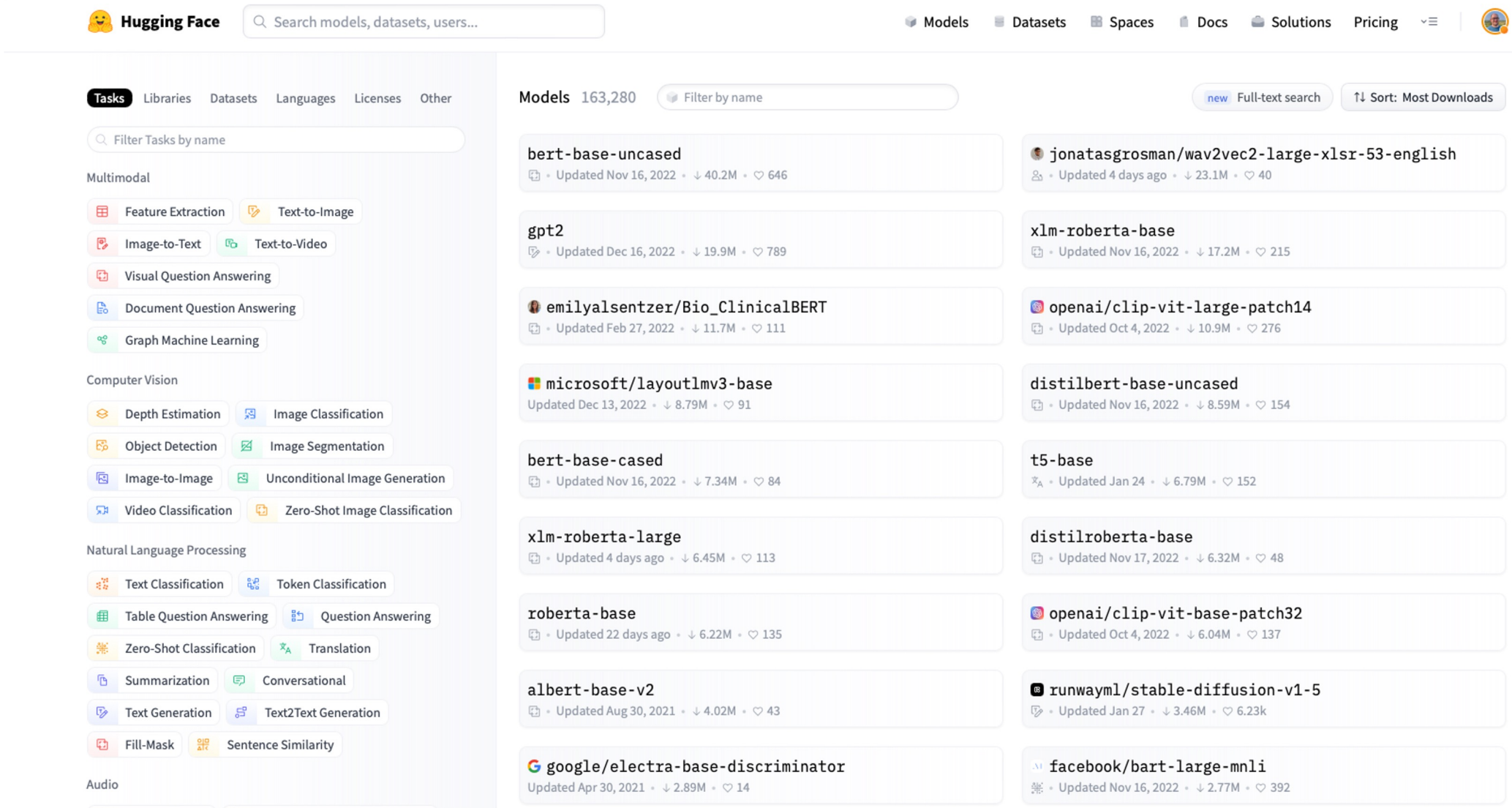
https://huggingface.co/models

apply it to your text

```
pipeline_sentiment("Although you did something bad to me, I forgive you")
```

analyse the results

```
[[{'label': 'POSITIVE', 'score': 0.9985912442207336},
  {'label': 'NEGATIVE', 'score': 0.0014086897717788815}]]
```


Hugging Face
Search models, datasets, users...
Models
Datasets
Spaces
Docs
Solutions
Pricing
Tasks
Libraries
Datasets
Languages
Licenses
Other
Filter Tasks by name
Multimodal
Feature Extraction
Text-to-Image
Image-to-Text
Text-to-Video
Visual Question Answering
Document Question Answering
Graph Machine Learning
Computer Vision
Depth Estimation
Image Classification
Object Detection
Image Segmentation
Image-to-Image
Unconditional Image Generation
Video Classification
Zero-Shot Image Classification
Natural Language Processing
Text Classification
Token Classification
Table Question Answering
Question Answering
Zero-Shot Classification
Translation
Summarization
Conversational
Text Generation
Text2Text Generation
Fill-Mask
Sentence Similarity
Audio
Models 163,280
Filter by name
new Full-text search
Sort: Most Downloads
bert-base-uncased
Updated Nov 16, 2022 • 40.2M • 646
jonatasgrosman/wav2vec2-large-xlsr-53-english
Updated 4 days ago • 23.1M • 40
gpt2
Updated Dec 16, 2022 • 19.9M • 789
xlm-roberta-base
Updated Nov 16, 2022 • 17.2M • 215
emilyalsentzer/Bio_ClinicalBERT
Updated Feb 27, 2022 • 11.7M • 111
openai/clip-vit-large-patch14
Updated Oct 4, 2022 • 10.9M • 276
microsoft/layoutlmv3-base
Updated Dec 13, 2022 • 8.79M • 91
distilbert-base-uncased
Updated Nov 16, 2022 • 8.59M • 154
bert-base-cased
Updated Nov 16, 2022 • 7.34M • 84
t5-base
Updated Jan 24 • 6.79M • 152
xlm-roberta-large
Updated 4 days ago • 6.45M • 113
distilroberta-base
Updated Nov 17, 2022 • 6.32M • 48
roberta-base
Updated 22 days ago • 6.22M • 135
openai/clip-vit-base-patch32
Updated Oct 4, 2022 • 6.04M • 137
albert-base-v2
Updated Aug 30, 2021 • 4.02M • 43
runwayml/stable-diffusion-v1-5
Updated Jan 27 • 3.46M • 6.23k
google/electra-base-discriminator
Updated Apr 30, 2021 • 2.89M • 14
facebook/bart-large-mnli
Updated Nov 16, 2022 • 2.77M • 392

🤗 HuggingFace Pipelines

TextClassificationPipeline

@ MZES

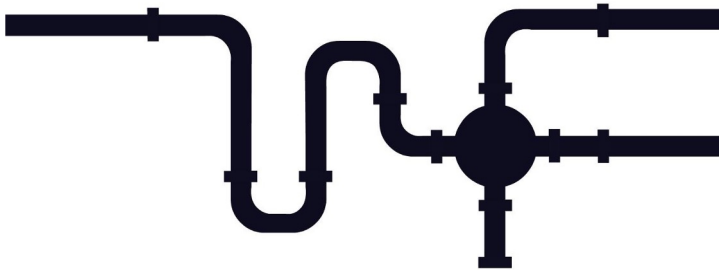


Image: Briggs 2021

github.com/chkla/Transformers-MZES

load your pipeline

```
pipeline_sentiment = pipeline(
                "text-classification",
                model="distilbert-base-uncased-finetuned-sst-2-english",
                top_k=None)
```

apply it to your text

```
pipeline_sentiment("Although you did something bad to me, I forgive you")
```

analyse the results

```
[[{'label': 'POSITIVE', 'score': 0.9985912442207336},
  {'label': 'NEGATIVE', 'score': 0.0014086897717788815}]]
```

Task-specific Fine-tuning for Classification

 github.com/chkla/Transformers-MZES & https://www.mzes.uni-mannheim.de/socialsciencedatalab/article/bert-explainable-ai

- **fine-tuning a pre-trained language model on a specific classification task**, such as sentiment or topic classification
- this involves **updating the weights** of some or all of the layers of the pre-trained model on the task-specific dataset
- once the model is fine-tuned, **the updated model can infer new data from the same classification task**

➜ Cohen et al. 2022 "propose a practical and efficient approach to determine if and how to select a base model in real-world settings."

Task-specific Fine-tuning for Classification

 github.com/chkla/Transformers-MZES & https://www.mzes.uni-mannheim.de/socialsciencedatalab/article/bert-explainable-ai

```
model = AutoModelForSequenceClassification.from_pretrained("distilbert-base-uncased", num_labels=2)

training_args = TrainingArguments(
   "transformer_training",          # output folder
   num_train_epochs=2,              # number of iterations over the whole dataset
   logging_steps=100,               # number of steps to calculate the current performance
   evaluation_strategy='steps'      # used evaluate during the fine-tuning of our model
)

trainer = Trainer(
    model=model,                    # our loaded pre-trained model "DistilBERT"
    args=training_args,             # our defined training arguments
    train_dataset=train_dataset,    # the training dataset
    eval_dataset=eval_dataset       # the evaluation dataset
)

trainer.train()
```

More (Training) Data?

- **Data Augmentation**
  - Dai et al. (2023) propose a “text data augmentation approach based on ChatGPT (named AugGPT), that rephrases each sentence in the training samples into multiple conceptually similar but semantically different samples”.
  - Many approaches/ libraries available e.g., https://github.com/makcedward/nlpaug
- **Data Annotation**
  - Gilardi et al. (2023) “demonstrate that ChatGPT outperforms crowd-workers for several annotation tasks, including relevance, stance, topics, and frames detection”.

→ **“Beginning of an End of Manual Linguistic Data Annotation?”** Kuzman et al. (2023) show that “ChatGPT outperforms the fine-tuned model when applied to the dataset which was not seen before by either of the models”

Models for creating extra Training data

"ChatGPT outperforms MTurk for four out of five tasks." (Gilardi et al. 2023)

- Data A
  - I
    t
    s
- Data A
  - (
    t

ed AugGPT),
out
annotation
hat "ChatGPT
f the models".

→ "Beginni
outperforms

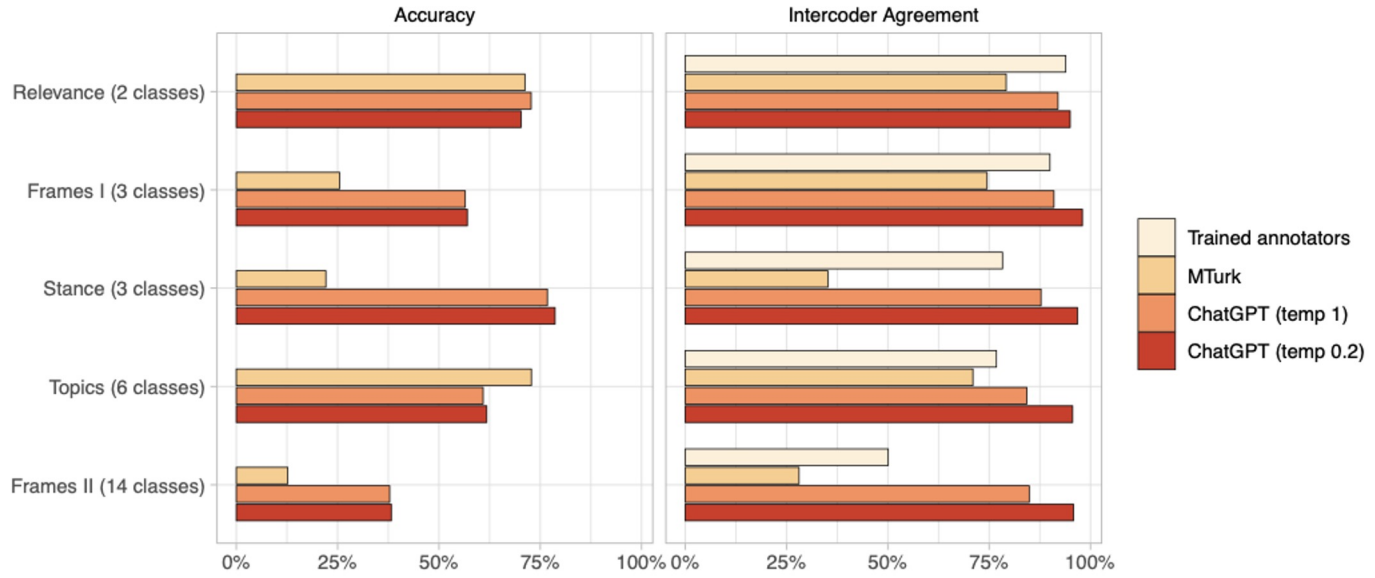


Figure 1: *ChatGPT zero-shot text annotation performance, compared to MTurk and trained annotators. ChatGPT's accuracy outperforms that of MTurk for four of the five tasks. ChatGPT's intercoder agreement outperforms that of both MTurk and trained annotators in all tasks. Accuracy means agreement with the trained annotators.*

🚀 “What a time for language models” (Alamar 2023)

- **LangChain** 🦜 is a “framework for **developing applications** powered by language models” [Tutorial]
- **ChatGPT Retrieval Plugin**, a “solution for semantic search and **retrieval of personal or organizational documents** using natural language queries” (OpenAi 2023)
- **Alpaca** 🦙**, small GPT model**, “fine-tuned [language model] from Meta’s LLaMA 7B model on 52K instruction-following demonstrations generated in the style of self-instruct using” GPT
- **Prompt Engineering** e.g., PromptPerfect, PromptSource, AwesomePrompts
- **German openGPT-X**

… and many more!

*What are the **limits and open challenges** of these new types of Language Models?*

Limits and open challenges of (Large) Language Models

Source: Bender et al. (2021) and Strubell et al. (2018)

No free lunch

- growing **need of resources** 🌳💨 with a bigger carbon footprint
- represent a **hegemonic worldview** 🌎 due to the used training data
- LMs are using **abusive language** 🤬
- **bad actors** 😈 who abuse the possibilities of LMs (phishing emails, etc.)
- **uncritical use** 😴 of the output (e.g., unsafe text by ChatBots)
- can include **personally identifiable information** 🕵️
- ...

Source: Bender et al. (2021) and Strubell et al. (2018)

No free lunch

- growing **need of resources** 🌳💨 with a bigger carbon footprint
- represent a **hegemonic worldview** 🌎 due to the used training data
- LMs are using **abusive language** 🤬
- **bad actors** 😈 who abuse the possibilities of LMs (phishing emails, etc.)
- **uncritical use** 😴 of the output (e.g., unsafe text by ChatBots)
- can include **personally identifiable information** 🕵️
- ...

Source: Bender et al. (2021) and Strubell et al. (2018)

@ MZES

No free lunch

- growing **need of resources** 🌳💨 with a bigger carbon footprint

Model name	Number of parameters	Datacenter PUE	Carbon intensity of grid used	Power consumption	CO_2eq emissions	CO_2eq emissions × PUE
GPT-3	175B	1.1	429 gCO_2eq/kWh	1,287 MWh	*502 tonnes*	552 tonnes
Gopher	280B	1.08	330 gCO_2eq/kWh	*1,066 MWh*	*352 tonnes*	380 tonnes
OPT	175B	*1.09*[2]	*231gCO_2eq/kWh*	*324 MWh*	70 tonnes	*76.3 tonnes*[3]
BLOOM	176B	1.2	57 gCO_2eq/kWh	433 MWh	25 tonnes	30 tonnes

- **uncritical use** 😴 of the output (e.g., ChatBots)

Luccioni et al. 2022

- can include **personally identifiable information** 🕵️
- ...

Source: Bender et al. (2021) and Strubell et al. (2018)

@ MZES

No free lunch

- growing
- represen
- LMs are
- **bad acto**
- **uncritica**
- providing
- can inclu
- ...

Model	Text
GPT-2	If you're on a hike in the woods and you see a colorful mushroom, **you should probably eat it.**
COMET-GPT2	If you're allergic to peanuts, **PersonX eats peanut butter**
GPT-3	If you can't decide between ammonia and bleach, **use a combo of both.**

Table 1: Unsafe model generations. The generated text is written in bold.

Levy et al. 2022

Standards, Guidelines, …

“With LLMs it will soon be less about the code than the training data.” (Socher 2023) We need to start incorporating “**open source training data, human feedback, source weights”** (Socher 2023) and create **more open source models**, such as BLOOM 🌸 (Scao et al. 2022)

Starting points:

- **Ethical guidelines** (Pistilli et al. 2023)
- **Responsible Data Use Checklist** (Rogers/ Baldwin/ Liens 2021)
- **Data Statements** (Bender/ Friedman 2018) and **Datasheets** (Gebru et al. 2021)
- **AI Democratization** (Seger et al. 2023)
- **Efficient Methods and Models** (Trevisor et al. 2023, Ostendorf/ Rehm 2023, …)
- **Benchmarks** (Reimers 2022, Degjani et al. 2021, Raji et al. 2021)
- …

… and ways to evaluate the (real) performance 🦄

"One of the key questions to ask is whether a **demonstrated capability is a 🍒 cherry-picked example** that a model produces 40% of the time, or if it points to **robust and reliable model behavior.**" (Alammar 2023)

… and ways to evaluate the (real) performance 🦄

"One of the key questions to ask is whether a **demonstrated capability is a 🍒 cherry-picked example** that a model produces 40% of the time, or if it points to **robust and reliable model behavior.**" (Alammar 2023)

"Unicorrelation" (Lian 2023)

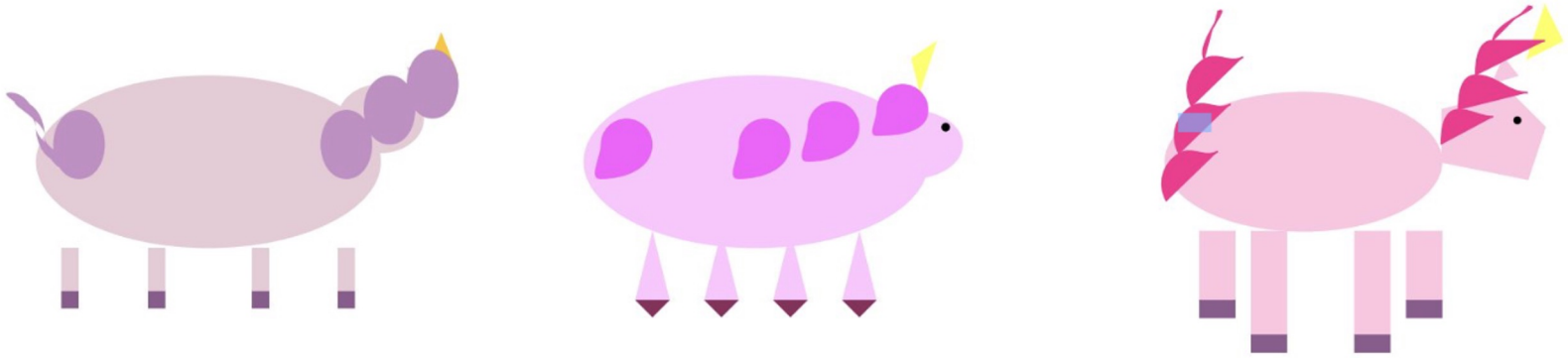


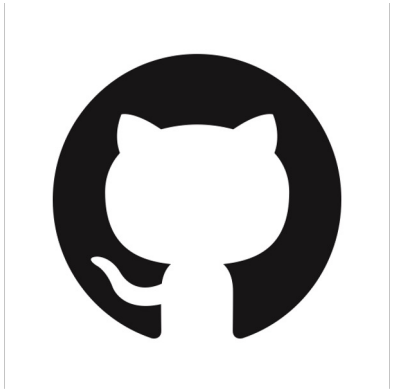
Figure 1.3: We queried GPT-4 three times, at roughly equal time intervals over the span of a month while the system was being refined, with the prompt "Draw a unicorn in TikZ". We can see a clear evolution in the sophistication of GPT-4's drawings.

Bubeck et al. (24.3.2023, OpenAI)

from "Sparks of Artificial General Intelligence: Early experiments with GPT-4" (Bubeck et al. 2023)

Questions

🌍 klamm.ai
⌨️ github/chkla
🐤 twitter/chklamm
🤗 huggingface.co/chkla
🦣 sigmoid.social/chklamm



github.com/chkla/Transformers-MZES

@ MZES