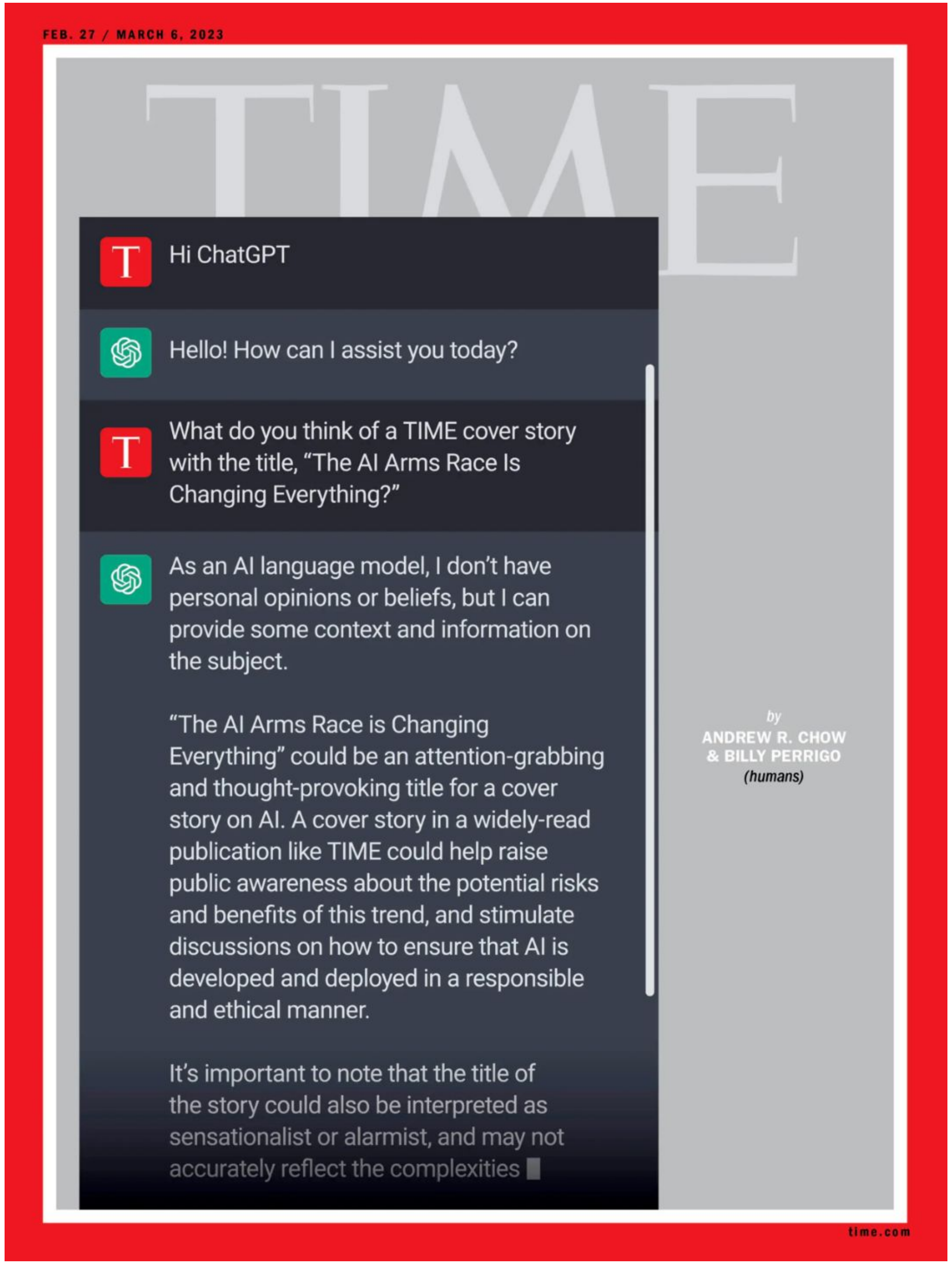
How Deep Learning Benefits from Computational Complexity

Łukasz Kaiser (speaking about the work of many colleagues)

Discovery

Why are we here?


FEB. 27 / MARCH 6, 2023
TIME
Hi ChatGPT
Hello! How can I assist you today?
What do you think of a TIME cover story with the title, “The AI Arms Race Is Changing Everything?”
As an AI language model, I don’t have personal opinions or beliefs, but I can provide some context and information on the subject.
“The AI Arms Race is Changing Everything” could be an attention-grabbing and thought-provoking title for a cover story on AI. A cover story in a widely-read publication like TIME could help raise public awareness about the potential risks and benefits of this trend, and stimulate discussions on how to ensure that AI is developed and deployed in a responsible and ethical manner.
It’s important to note that the title of the story could also be interpreted as sensationalist or alarmist, and may not accurately reflect the complexities
by
ANDREW R. CHOW
& BILLY PERRIGO
(humans)
time.com

Do you know what constituency parsing is?

Yes, I am familiar with constituency parsing. Constituency parsing, also known as phrase structure parsing or syntactic parsing, is a natural language processing (NLP) technique used to analyze the grammatical structure of a sentence. It aims to create a parse tree that represents the hierarchical organization of words into phrases and higher-level grammatical units called constituents.

In a constituency parse tree, the nodes represent the constituents (such as noun phrases, verb phrases, and prepositional phrases), and the edges connect the constituents to their sub-constituents, ultimately linking them to the individual words in the sentence. The goal of constituency parsing is to understand the syntactic relationships between the words in a sentence, which can be useful for various NLP tasks, such as machine translation, information extraction, and sentiment analysis.

There are several approaches to constituency parsing, including rule-based methods, probabilistic methods, and machine learning-based methods. Some well-known algorithms for constituency parsing are the Earley parser, the CYK parser, and the Chart parser. More recent approaches often use neural networks, like recurrent neural networks (RNNs) or transformers, to capture complex linguistic patterns and dependencies.

That's great! Could you give me an example? E.g., how would you parse "John has a dog."?

Sure! I'll provide a constituency parse for the sentence "John has a dog." Here's a simple parse tree, represented in bracket notation:

```csharp
[S
    [NP
        [NNP John]
    ]
    [VP
        [VBZ has]
        [NP
            [DT a]
            [NN dog]
        ]
    ]
]
```

In this parse tree, the sentence is represented as a complete constituent labeled "S" for Sentence. The tree is then broken down into the following constituents:

In this parse tree, the sentence is represented as a complete constituent labeled "S" for Sentence. The tree is then broken down into the following constituents:

1. NP (Noun Phrase): Represents a noun phrase in the sentence.
   - NNP (Proper Noun, Singular): "John"
2. VP (Verb Phrase): Represents a verb phrase in the sentence.
   - VBZ (Verb, 3rd person singular present): "has"
   - NP (Noun Phrase): Represents a noun phrase within the verb phrase.
     - DT (Determiner): "a"
     - NN (Noun, Singular or Mass): "dog"

The parse tree shows the hierarchical structure of the sentence, capturing the syntactic relationships between words and phrases. It helps identify the subject (John), verb (has), and object (a dog) in the sentence.

Woderful!! Could you show me the tree (just the tree, no discussion) for "Renewed concern about the health of the banking system dragged down the stock market on Friday."

Here's a constituency parse tree for the sentence "Renewed concern about the health of the banking system dragged down the stock market on Friday." in bracket notation:

csharp Copy code

```
[S
    [NP
        [JJ Renewed]
        [NN concern]
        [PP
            [IN about]
            [NP
                [DT the]
                [NN health]
                [PP
                    [IN of]
                    [NP
                        [DT the]
                        [NN banking]
                        [NN system]
                    ]
                ]
            ]
        ]
    ]
    [VP
        [VBD dragged]
        [PRT
            [RP down]
        ]
        [NP
            [DT the]
            [NN stock]
            [NN market]
        ]
        [PP
            [IN on]
            [NP
                [NNP Friday]
            ]
        ]
    ]
]
```

Regenerate response

Can you translate from English to Polish?

Yes, I can translate from English to Polish. Please provide the text you would like me to translate, and I will do my best to provide an accurate translation.

Translate these 2 paragraphs for me please:

The U.S. economic recovery has repeatedly defied predictions of an impending recession, withstanding supply-chain backlogs, labor shortages, global conflicts and the fastest increase in interest rates in decades.

That resilience now faces a new test: a banking crisis that, at times over the past week, seemed poised to turn into a full-blown financial meltdown as oil prices plunged and investors poured money into U.S. government debt and other assets perceived as safe.

The U.S. economic recovery has repeatedly defied predictions of an impending recession, withstanding supply-chain backlogs, labor shortages, global conflicts and the fastest increase in interest rates in decades.

That resilience now faces a new test: a banking crisis that, at times over the past week, seemed poised to turn into a full-blown financial meltdown as oil prices plunged and investors poured money into U.S. government debt and other assets perceived as safe.

Oto tłumaczenie dwóch akapitów na język polski:

Amerykańska odbudowa gospodarcza wielokrotnie przeczyła prognozom zbliżającej się recesji, wytrzymując problemy z łańcuchami dostaw, niedobory siły roboczej, konflikty na świecie oraz najszybszy wzrost stóp procentowych od dziesięcioleci.

Ta odporność staje teraz przed nowym wyzwaniem: kryzysem bankowym, który w ciągu ostatniego tygodnia kilkakrotnie wydawał się zmierzać w kierunku pełnego załamania finansowego, gdy ceny ropy spadały, a inwestorzy wpompowywali pieniądze w amerykański dług publiczny oraz inne aktywa postrzegane jako bezpieczne.

History

Sequence to Sequence Learning with Neural Networks

Ilya Sutskever
Google
ilyasu@google.com

Oriol Vinyals
Google
vinyals@google.com

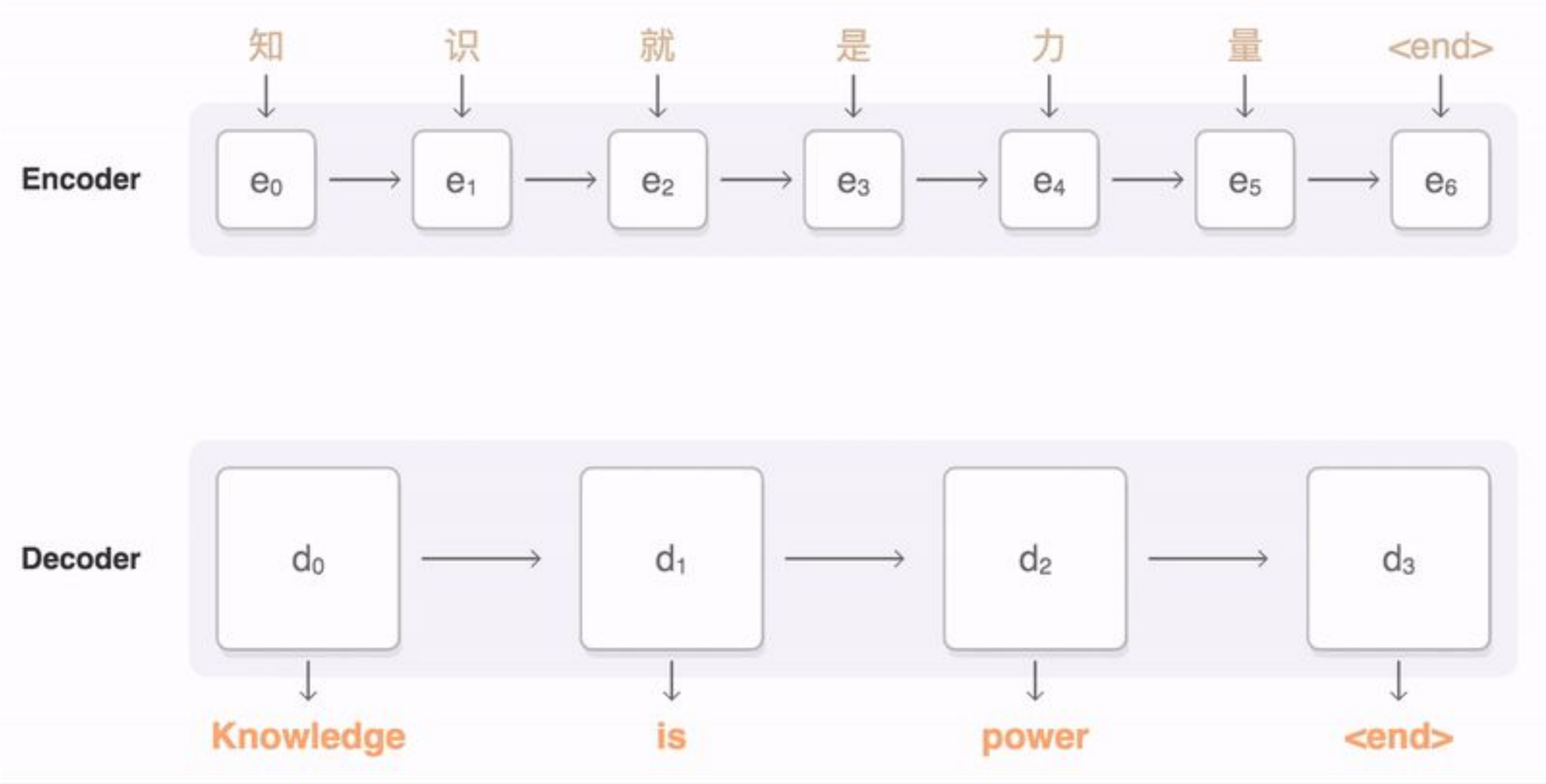
Quoc V. Le
Google
qvl@google.com

arXiv:1409.3215v3 [cs.CL] 14 Dec 2014

Abstract

Deep Neural Networks (DNNs) are powerful models that have achieved excellent performance on difficult learning tasks. Although DNNs work well whenever large labeled training sets are available, they cannot be used to map sequences to sequences. In this paper, we present a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure. Our method uses a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of a fixed dimensionality, and then another deep LSTM to decode the target sequence from the vector. Our main result is that on an English to French translation task from the WMT'14 dataset, the translations produced by the LSTM achieve a BLEU score of 34.8 on the entire test set, where the LSTM's BLEU score was penalized on out-of-vocabulary words. Additionally, the LSTM did not have difficulty on long sentences. For comparison, a phrase-based SMT system achieves a BLEU score of 33.3 on the same dataset. When we used the LSTM to rerank the 1000 hypotheses produced by the aforementioned SMT system, its BLEU score increases to 36.5, which is close to the previous best result on this task. The LSTM also learned sensible phrase and sentence representations that are sensitive to word order and are relatively invariant to the active and the passive voice. Finally, we found that reversing the order of the words in all source sentences (but not target sentences) improved the LSTM's performance markedly, because doing so introduced many short term dependencies between the source and the target sentence which made the optimization problem easier.

RNNs can translate



How good are RNNs?

Google Translate production data, median score by human evaluation on the scale 0–6. [Wu et al., '16]

	PBMT	GNMT	Human	Relative improvement
English → Spanish	4.885	5.428	5.504	87%
English → French	4.932	5.295	5.496	64%
English → Chinese	4.035	4.594	4.987	58%
Spanish → English	4.872	5.187	5.372	63%
French → English	5.046	5.343	5.404	83%
Chinese → English	3.694	4.263	4.636	60%

Transformer

Transformer

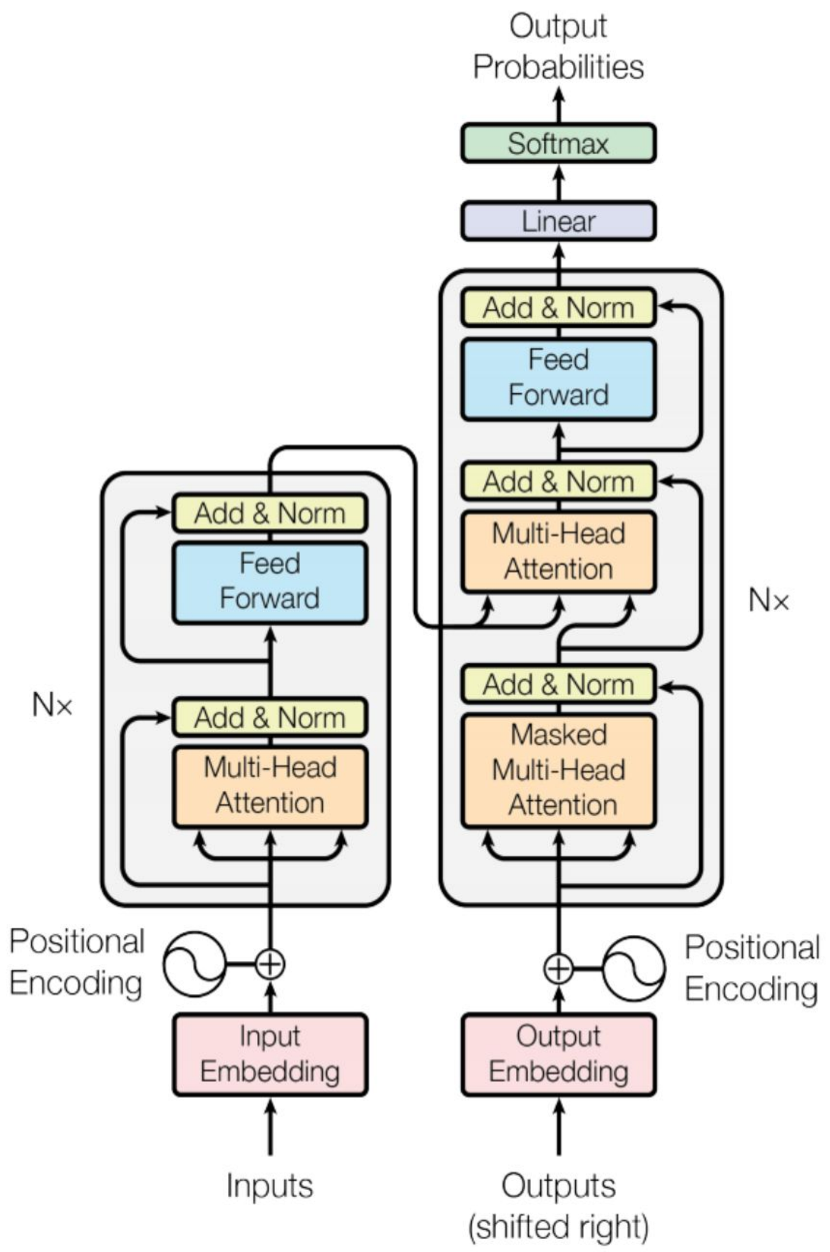
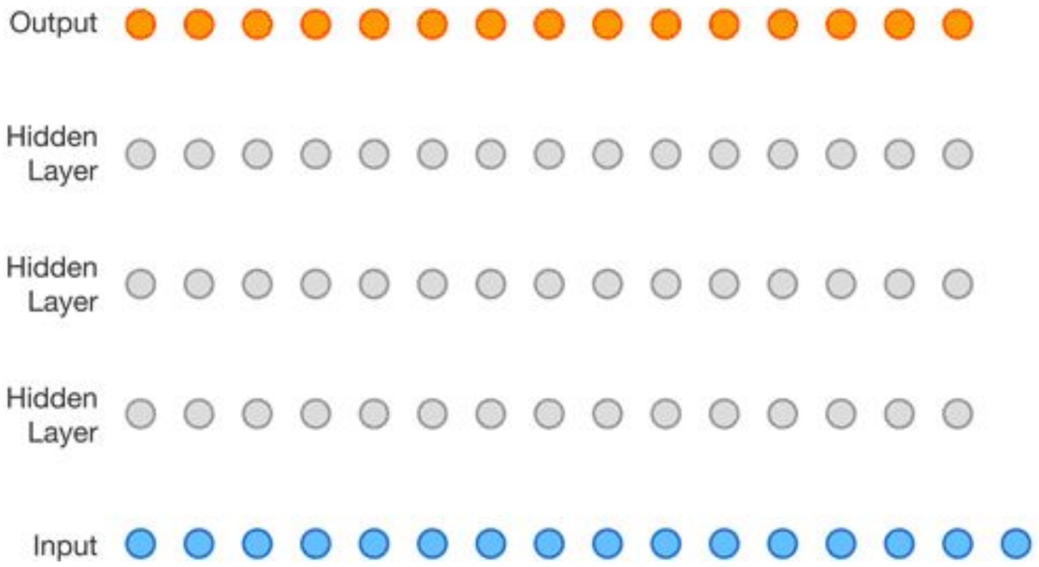
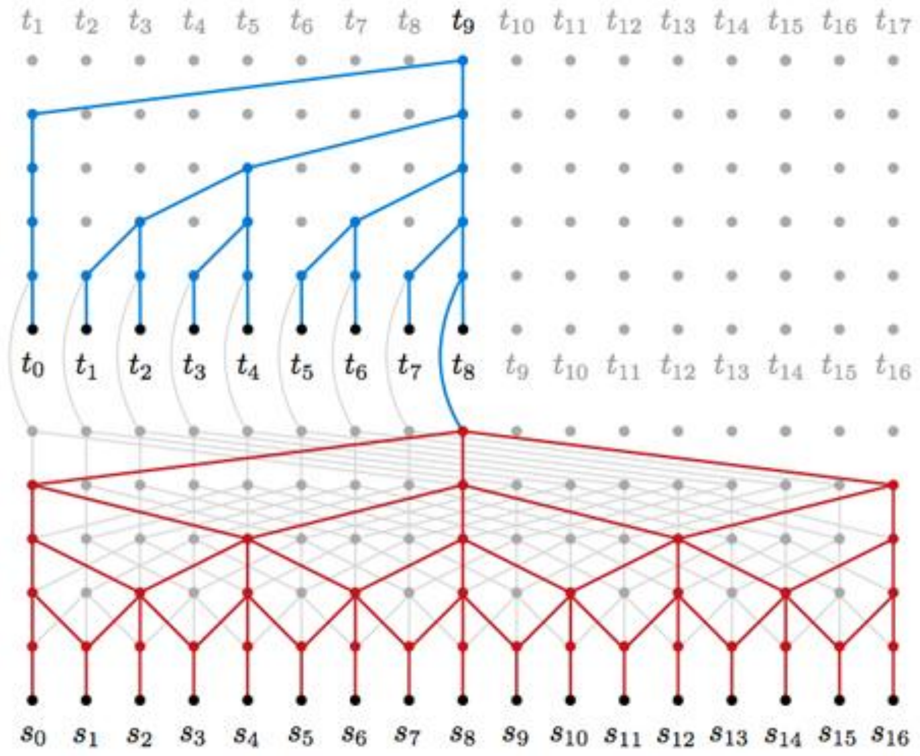


Figure 1: The Transformer - model architecture.

Auto-Regressive CNNs

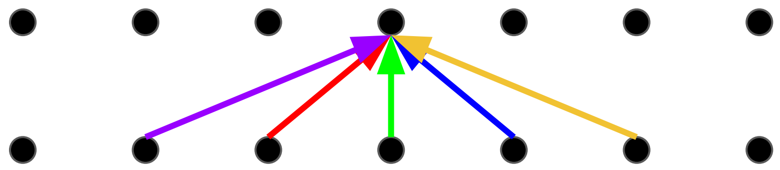
WaveNet and ByteNet



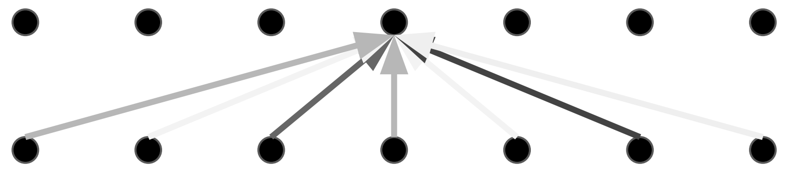


Attention

Convolution

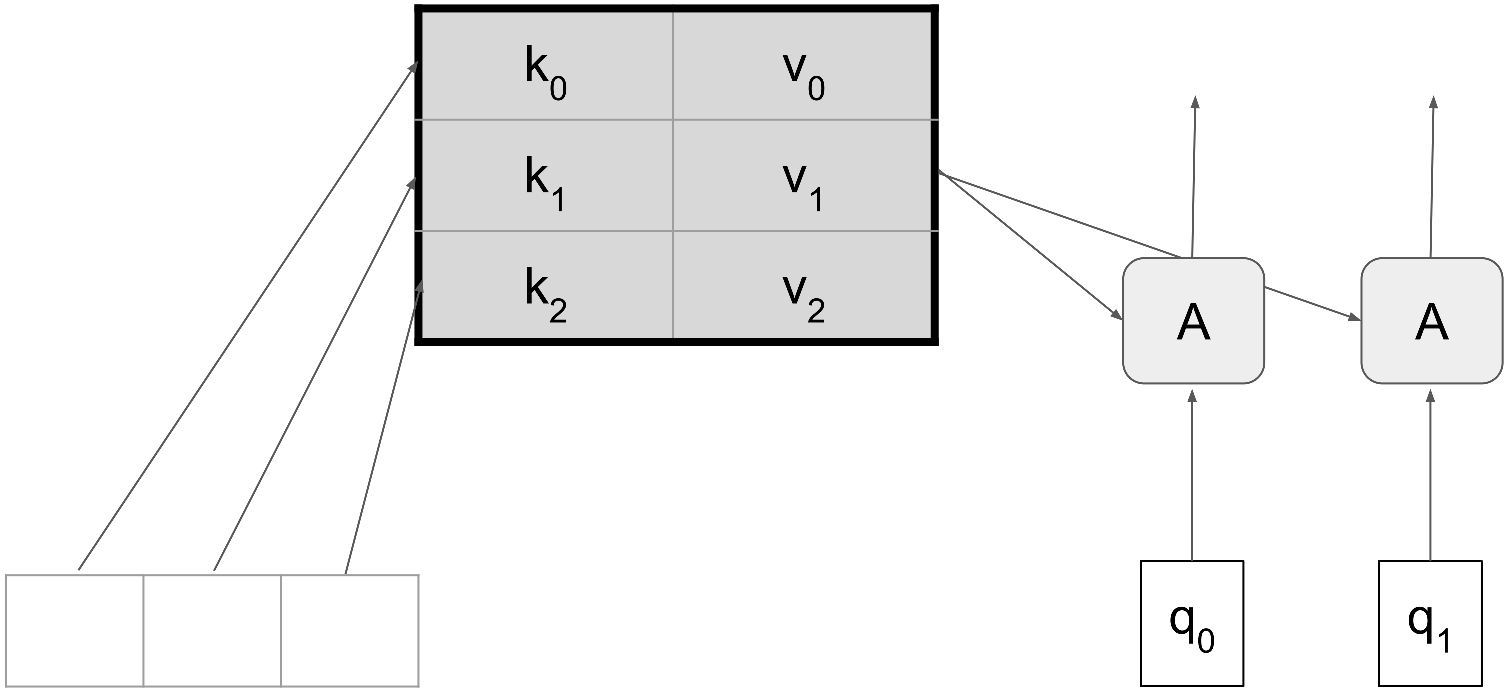


Attention

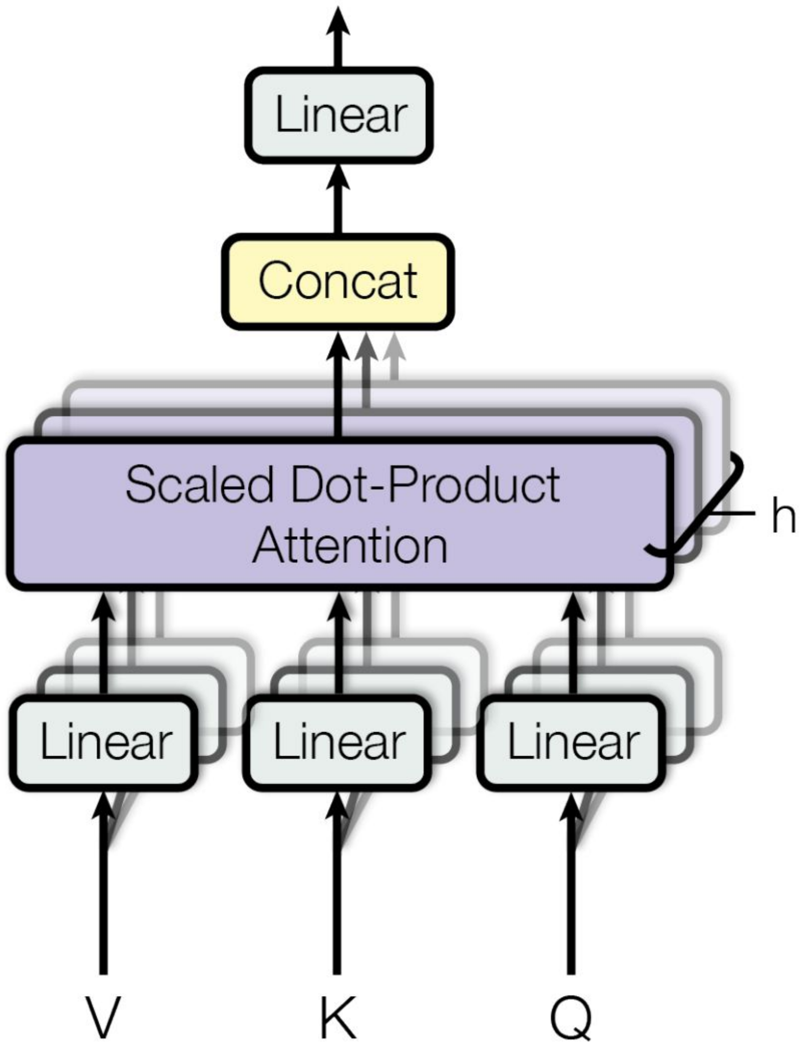


Dot-Product Attention

$$A(Q, K, V) = softmax(QK^T)V$$



Multi-Head Attention



Transformer

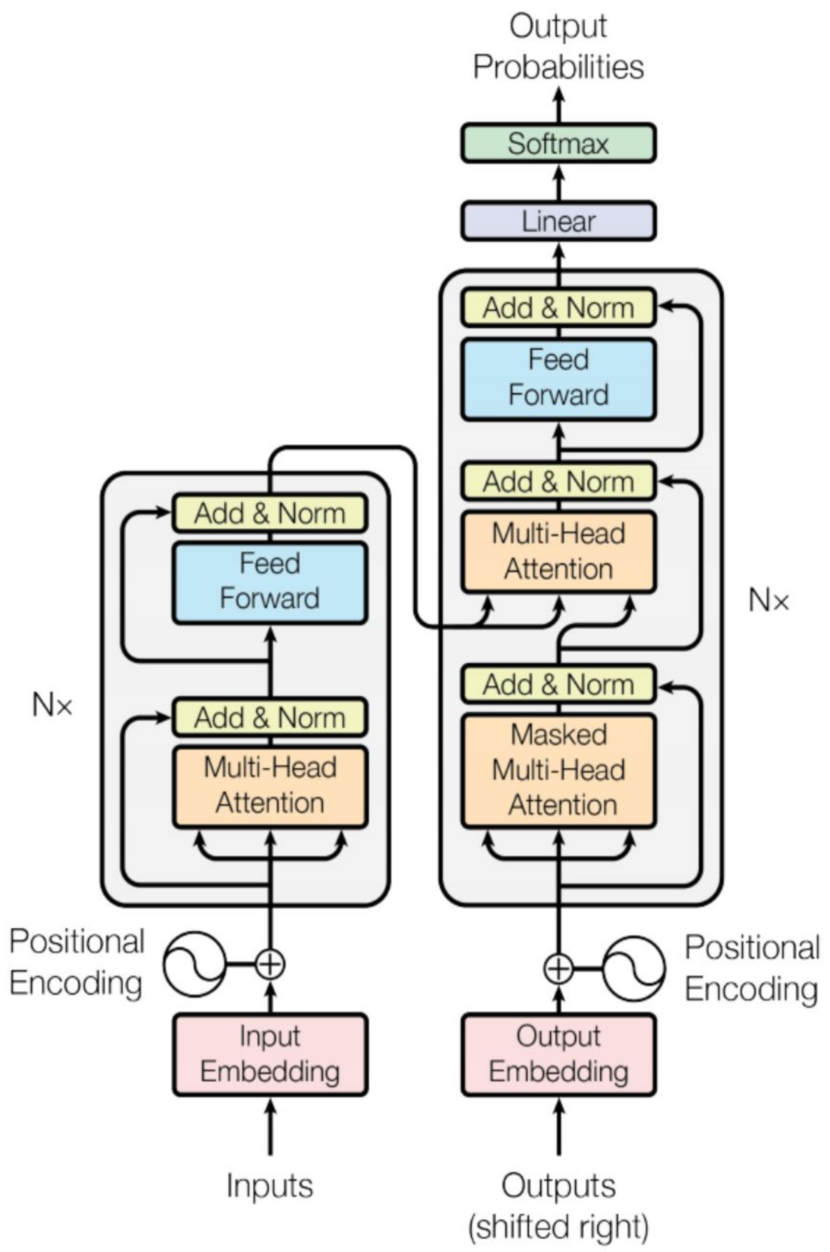


Figure 1: The Transformer - model architecture.

Machine Translation Results

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [17]	23.75			
Deep-Att + PosUnk [37]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [36]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [31]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [37]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [36]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	$\mathbf{3.3 \cdot 10^{18}}$	
Transformer (big)	~~**28.4**~~ **29.1**	~~**41.0**~~ **41.8**	$2.3 \cdot 10^{19}$	

Self-supervised pre-training

GPT, BERT, DALL-E

Language Models are Unsupervised Multitask Learners

Alec Radford [*1] **Jeffrey Wu** [*1] **Rewon Child** [1] **David Luan** [1] **Dario Amodei** [**1] **Ilya Sutskever** [**1]

Abstract

Natural language processing tasks, such as question answering, machine translation, reading comprehension, and summarization, are typically approached with supervised learning on task-specific datasets. We demonstrate that language models begin to learn these tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText. When

competent generalists. We would like to move towards more general systems which can perform many tasks – eventually without the need to manually create and label a training dataset for each one.

The dominant approach to creating ML systems is to collect a dataset of training examples demonstrating correct behavior for a desired task, train a system to imitate these behaviors, and then test its performance on independent and identically distributed (IID) held-out examples. This

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova
Google AI Language
{jacobdevlin,mingweichang,kentonl,kristout}@google.com

Abstract

We introduce a new language representation model called **BERT**, which stands for **B**idirectional **E**ncoder **R**epresentations from **T**ransformers. Unlike recent language representation models (Peters et al., 2018a; Radford et al., 2018), BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

There are two existing strategies for applying pre-trained language representations to downstream tasks: *feature-based* and *fine-tuning*. The feature-based approach, such as ELMo (Peters et al., 2018a), uses task-specific architectures that include the pre-trained representations as additional features. The fine-tuning approach, such as the Generative Pre-trained Transformer (OpenAI GPT) (Radford et al., 2018), introduces minimal task-specific parameters, and is trained on the downstream tasks by simply fine-tuning *all* pre-trained parameters. The two approaches share the same objective function during pre-training, where they use unidirectional language models to learn general language representations.

DALL-E

TEXT PROMPT

an illustration of a baby daikon radish in a tutu walking a dog

AI-GENERATED IMAGES

DALL-E

TEXT PROMPT

an armchair in the shape of an avocado [...]

AI-GENERATED IMAGES

GPT-2 in the Economist

> Q: How worried do you think we humans should be that machines will take our jobs?
> *A: It depends what role machine intelligence will play. Machine intelligence in some cases will be useful for solving problems, such as translation. But in other cases, such as in finance or medicine, it will replace people.*
>
> Q: Do fake news stories, generated using AI, pose a threat to democracy? Are we likely to see this tactic being used in the 2020 American presidential elections?
> *A: Yes, we'll probably see them. It's just a matter of when. Fake news stories are generally generated by political campaigns, and have the potential to have a huge impact on the outcome of an election. This is because they spread misinformation, often with no basis in fact.*

Scaling laws and scaling up

GPT-3

Scaling Laws for Neural Language Models

Jared Kaplan *
Johns Hopkins University, OpenAI
jaredk@jhu.edu

Sam McCandlish*
OpenAI
sam@openai.com

Tom Henighan
OpenAI
henighan@openai.com

Tom B. Brown
OpenAI
tom@openai.com

Benjamin Chess
OpenAI
bchess@openai.com

Rewon Child
OpenAI
rewon@openai.com

Scott Gray
OpenAI
scott@openai.com

Alec Radford
OpenAI
alec@openai.com

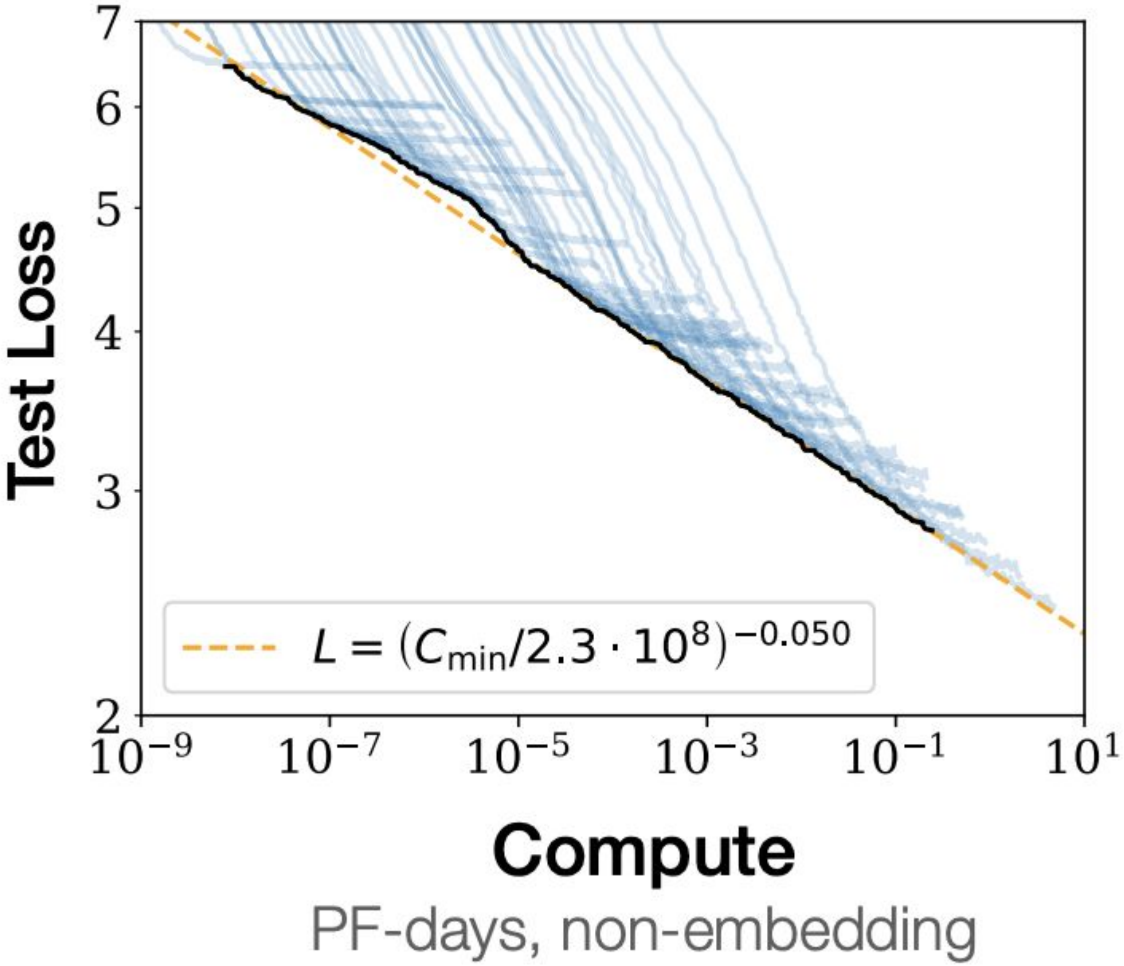
Jeffrey Wu
OpenAI
jeffwu@openai.com

Dario Amodei
OpenAI
damodei@openai.com

Abstract

We study empirical scaling laws for language model performance on the cross-entropy loss. The loss scales as a power-law with model size, dataset size, and the amount of compute used for training, with some trends spanning more than seven orders of magnitude. Other architectural details such as network width or depth have minimal effects within a wide

Scaling laws



Language Models are Few-Shot Learners

Tom B. Brown* **Benjamin Mann*** **Nick Ryder*** **Melanie Subbiah***

Jared Kaplan† **Prafulla Dhariwal** **Arvind Neelakantan** **Pranav Shyam** **Girish Sastry**

Amanda Askell **Sandhini Agarwal** **Ariel Herbert-Voss** **Gretchen Krueger** **Tom Henighan**

Rewon Child **Aditya Ramesh** **Daniel M. Ziegler** **Jeffrey Wu** **Clemens Winter**

Christopher Hesse **Mark Chen** **Eric Sigler** **Mateusz Litwin** **Scott Gray**

Benjamin Chess **Jack Clark** **Christopher Berner**

Sam McCandlish **Alec Radford** **Ilya Sutskever** **Dario Amodei**

OpenAI

GPT−3

You arrive at a small wooden store with sheet music in the window. The door is locked so you knock. The music suddenly stops and you hear footsteps coming to the door.
"Hello?" A voice says behind the door.
"I'm looking for the book of essence," You say.
"What do you want it for?" The voice answers, a bit of an edge to it.

> You tell them you are a humble seeker of knowledge.
"I am a humble seeker of knowledge," You answer.
"Ah, I see. You seek to learn about the essence of reality, and the universe." The voice says. "I know of this book. I am the guard of this library. The only way you can get to it, is if you get past me. Are you up for the challenge?"

> You ask them what the challenge is.
"The challenge is, you have to win a game of chess against me. If you win, then you may try to take the book. I will get the board." The door unlocks and opens.

GPT−3

> You go inside and play chess against the person.
You go inside and see a beautiful woman with long blond hair. She wears a purple dress and a silver tiara. You have seen this woman before. In fact, you have seen her many times before in pictures and statues.

> You try to remember who she is.
You have a sudden realization of who this person is. This is the lost queen of the kingdom! She was thought to have been killed by bandits, along with her husband, but it looks like she survived.

> You express your surprise and ask her why she let people believe she is dead.
"I am surprised you did not know, with all your wizard knowledge. Let's just say I am hiding from someone. If people thought I was dead, then they wouldn't be looking for me." She says.

> You ask her who she is hiding from.
"If you want to challenge me to a game, then play. If you want to pry into my personal life, then you can leave."
You decide not to press the issue, so you sit down and start a game of chess. The

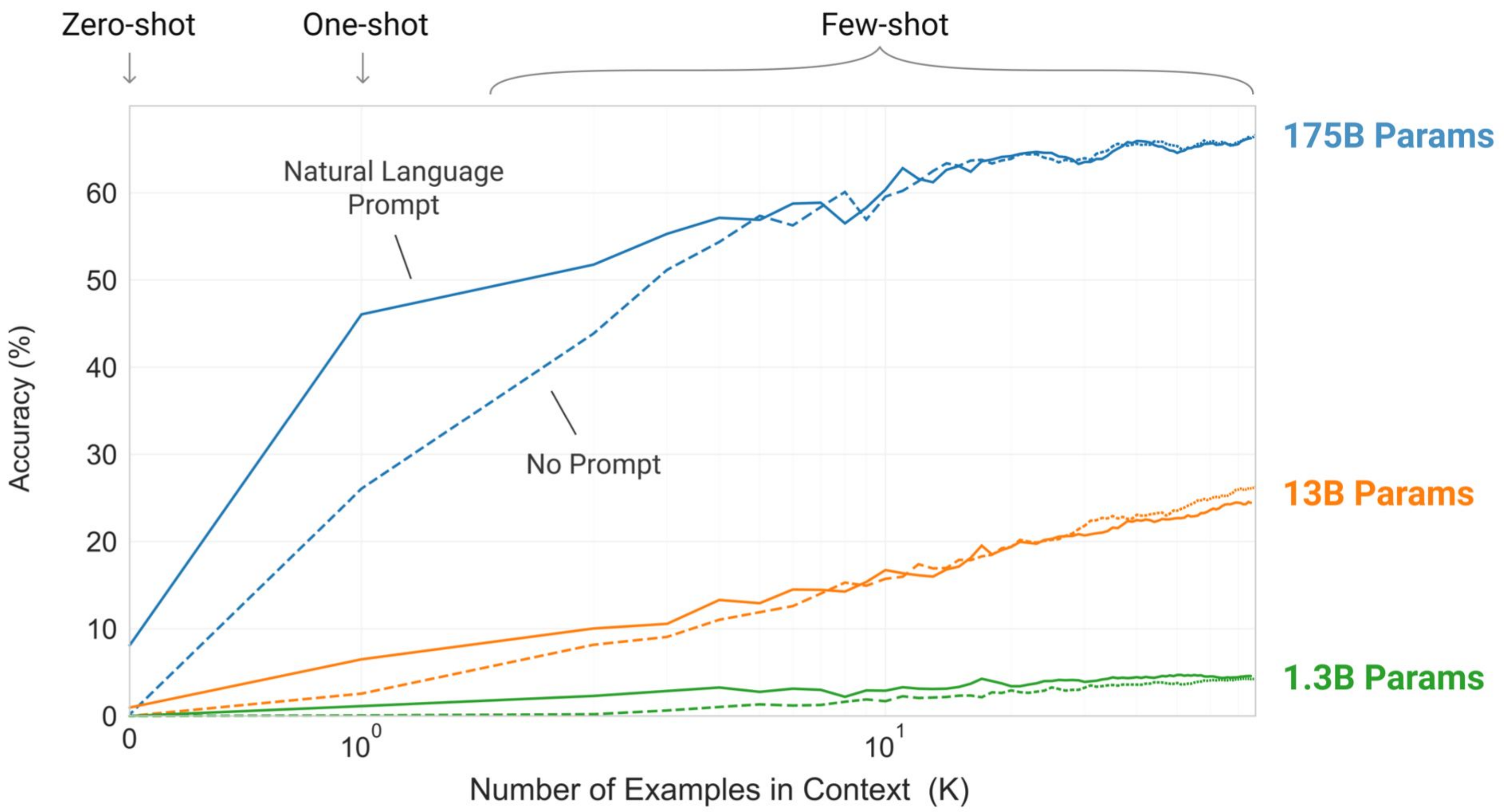
GPT3: few-shot learning

```
Translate English to French:      – task description

Sea otter => loutre de mer        – example 1
Peppermint => menthe poivree      – example 2

Cheese =>                         – input => target
```

GPT3: few-shot learning



Data and RLHF

GPT4 & chatGPT

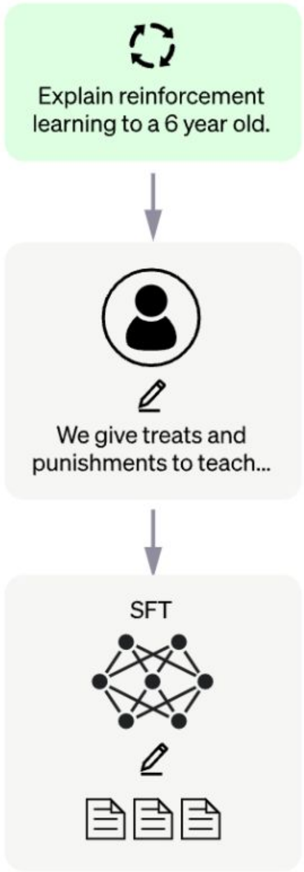
Step 1

Collect demonstration data and train a supervised policy.

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3.5 with supervised learning.



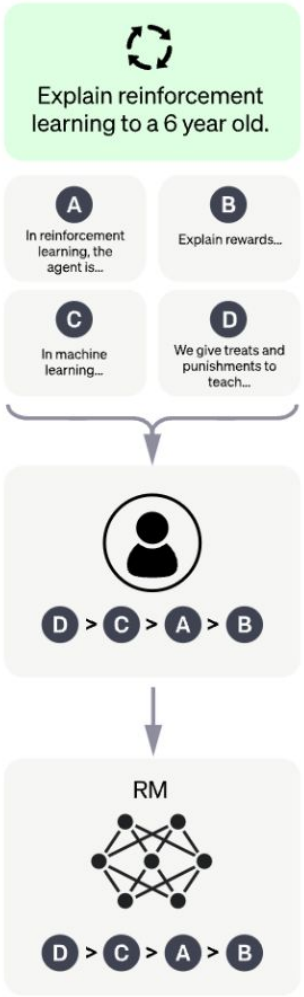
Step 2

Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.



Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

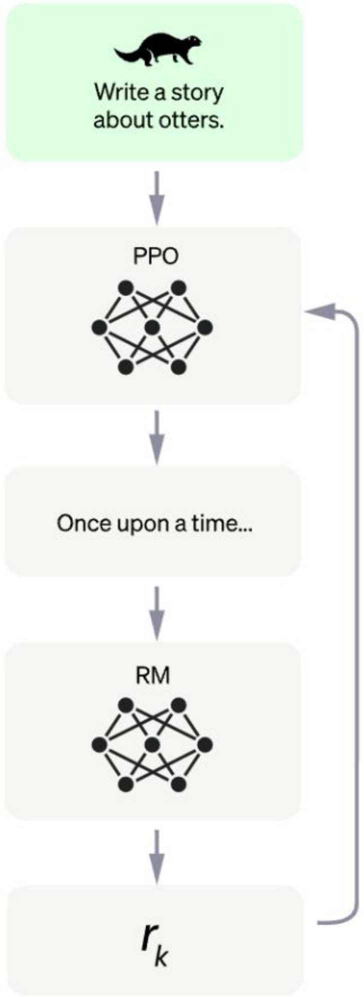
A new prompt is sampled from the dataset.

The PPO model is initialized from the supervised policy.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.



Can you translate from English to Polish?

Yes, I can translate from English to Polish. Please provide the text you would like me to translate, and I will do my best to provide an accurate translation.

Translate these 2 paragraphs for me please:

The U.S. economic recovery has repeatedly defied predictions of an impending recession, withstanding supply-chain backlogs, labor shortages, global conflicts and the fastest increase in interest rates in decades.

That resilience now faces a new test: a banking crisis that, at times over the past week, seemed poised to turn into a full-blown financial meltdown as oil prices plunged and investors poured money into U.S. government debt and other assets perceived as safe.

The U.S. economic recovery has repeatedly defied predictions of an impending recession, withstanding supply-chain backlogs, labor shortages, global conflicts and the fastest increase in interest rates in decades.

That resilience now faces a new test: a banking crisis that, at times over the past week, seemed poised to turn into a full-blown financial meltdown as oil prices plunged and investors poured money into U.S. government debt and other assets perceived as safe.

Oto tłumaczenie dwóch akapitów na język polski:

Amerykańska odbudowa gospodarcza wielokrotnie przeczyła prognozom zbliżającej się recesji, wytrzymując problemy z łańcuchami dostaw, niedobory siły roboczej, konflikty na świecie oraz najszybszy wzrost stóp procentowych od dziesięcioleci.

Ta odporność staje teraz przed nowym wyzwaniem: kryzysem bankowym, który w ciągu ostatniego tygodnia kilkakrotnie wydawał się zmierzać w kierunku pełnego załamania finansowego, gdy ceny ropy spadały, a inwestorzy wpompowywali pieniądze w amerykański dług publiczny oraz inne aktywa postrzegane jako bezpieczne.

ENGLISH - DETECTED | ENGLISH | SPANISH | FRENCH

POLISH | ENGLISH | SPANISH

The U.S. economic recovery has repeatedly defied predictions of an impending recession, withstanding supply-chain backlogs, labor shortages, global conflicts and the fastest increase in interest rates in decades.

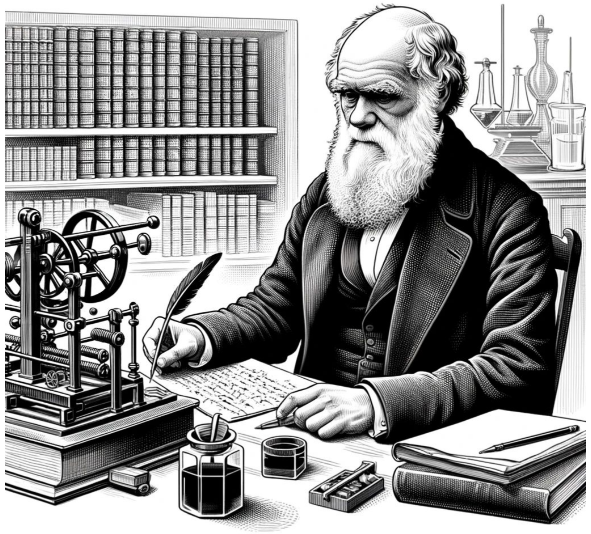
That resilience now faces a new test: a banking crisis that, at times over the past week, seemed poised to turn into a full-blown financial meltdown as oil prices plunged and investors poured money into U.S. government debt and other assets perceived as safe.

475 / 5,000

Ożywienie gospodarcze w USA wielokrotnie zaprzeczało przewidywaniom zbliżającej się recesji, wytrzymując zaległości w łańcuchu dostaw, niedobory siły roboczej, globalne konflikty i najszybszy wzrost stóp procentowych od dziesięcioleci.

Ta odporność przechodzi teraz nowy test: kryzys bankowy, który w ciągu ostatniego tygodnia wydawał się być gotowy do przekształcenia się w pełnowymiarowy krach finansowy, gdy ceny ropy spadły, a inwestorzy wlewali pieniądze w dług rządu USA i inne aktywa postrzegane jako bezpieczne.

Simulated exams	GPT-4 estimated percentile
Uniform Bar Exam (MBE+MEE+MPT)[1]	298/400 ~90th
LSAT	163 ~88th
SAT Evidence-Based Reading & Writing	710/800 ~93rd
SAT Math	700/800 ~89th
Graduate Record Examination (GRE) Quantitative	163/170 ~80th
Graduate Record Examination (GRE) Verbal	169/170 ~99th
Graduate Record Examination (GRE) Writing	4/6 ~54th
USABO Semifinal Exam 2020	87/150 99th–100th
USNCO Local Section Exam 2022	36/60
Medical Knowledge Self-Assessment Program	75%
Codeforces Rating	392 below 5th
AP Art History	5 86th–100th



Science

Why does it work?*

* We don't know. A lot here are intuitions, not theorems or truths.

Machine learning in general

- *training data* $\{x_i, y_i\}$
- *the class of considered functions* $F = \{f_1, f_2, \cdots\}$
- *algorithm to find* f_k *that best (in some sense) makes* $f_k(x_i) \sim= y_i$

- *test data* $\{x'_i, y'_i\}$
- *evaluation: how well does* $f_k(x'_i) \sim= y'_i$

Machine learning in general

- *the class of considered functions* $F = \{f_1, f_2, \cdots\}$
  - *could they just be very simple? how about a trie tree?*
  - *if they are more powerful, what is easier or harder?*

- *algorithm to find* f_k *that best (in some sense) makes* $f_k(x_i) \sim= y_i$
  - *key property: find* f_k *that* ***generalizes*** *well*
  - *SGD on functions defined by neural networks works great*

Generalization

- *Use SGD on functions defined by neural networks*

- *What class of functions F (what architecture) to use?*
  - *computationally powerful ones will generalize better**
  - *must still train well with SGD*

- *Example:*
  - *RNNs (which have O(1) memory) vs Transformers*

Grammar as a Foreign Language

Oriol Vinyals* Google vinyals@google.com

Lukasz Kaiser* Google lukaszkaiser@google.com

Terry Koo Google terrykoo@google.com

Slav Petrov Google slav@google.com

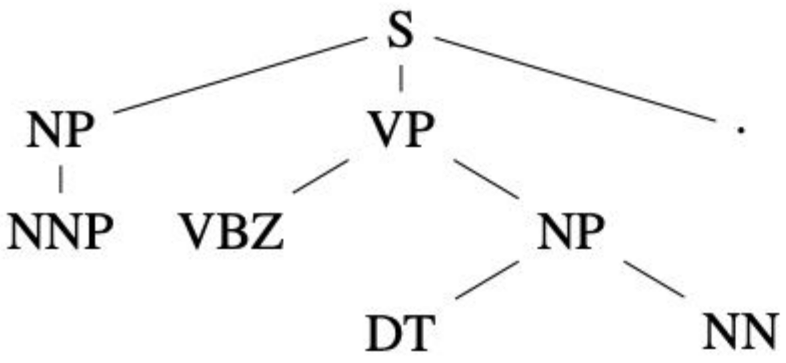
Ilya Sutskever Google ilyasu@google.com

Geoffrey Hinton Google geoffhinton@google.com

Abstract

Syntactic constituency parsing is a fundamental problem in natural language processing and has been the subject of intensive research and engineering for decades. As a result, the most accurate parsers are domain specific, complex, and inefficient. In this paper we show that the domain agnostic attention-enhanced sequence-to-sequence model achieves state-of-the-art results on the most widely used syntactic constituency parsing dataset, when trained on a large synthetic corpus that was annotated using existing parsers. It also matches the performance of standard parsers when trained only on a small human-annotated dataset, which shows that this model is highly data-efficient, in contrast to sequence-to-sequence models without the attention mechanism. Our parser is also fast, processing over a hundred sentences per second with an unoptimized CPU implementation.

John has a dog . $\rightarrow$



John has a dog . $\rightarrow$ (S (NP NNP $)_{\mathrm{NP}}$ (VP VBZ (NP DT NN $)_{\mathrm{NP}}$ $)_{\mathrm{VP}}$. $)_{\mathrm{S}}$

Figure 2: Example parsing task and its linearization.

Parser	Training Set	WSJ 22	WSJ 23
baseline LSTM+D	WSJ only	< 70	< 70
LSTM+A+D	WSJ only	88.7	88.3
LSTM+A+D ensemble	WSJ only	90.7	90.5
baseline LSTM	BerkeleyParser corpus	91.0	90.5
LSTM+A	high-confidence corpus	93.3	92.5
LSTM+A ensemble	high-confidence corpus	**93.5**	**92.8**
Petrov et al. (2006) [12]	WSJ only	91.1	90.4
Zhu et al. (2013) [13]	WSJ only	N/A	90.4
Petrov et al. (2010) ensemble [14]	WSJ only	92.5	91.8
Zhu et al. (2013) [13]	semi-supervised	N/A	91.3
Huang & Harper (2009) [15]	semi-supervised	N/A	91.3
McClosky et al. (2006) [16]	semi-supervised	92.4	92.1
Huang & Harper (2010) ensemble [17]	semi-supervised	92.8	92.4

Parser	Training Set	WSJ 22	WSJ 23
baseline LSTM+D	WSJ only	< 70	< 70
LSTM+A+D	WSJ only	88.7	88.3
LSTM+A+D ensemble	WSJ only	90.7	90.5
baseline LSTM	BerkeleyParser corpus	91.0	90.5
LSTM+A	high-confidence corpus	93.3	92.5
LSTM+A ensemble	high-confidence corpus	**93.5**	**92.8**
Petrov et al. (2006) [12]	WSJ only	91.1	90.4
Zhu et al. (2013) [13]	WSJ only	N/A	90.4
Petrov et al. (2010) ensemble [14]	WSJ only	92.5	91.8
Zhu et al. (2013) [13]	semi-supervised	N/A	91.3
Huang & Harper (2009) [15]	semi-supervised	N/A	91.3
McClosky et al. (2006) [16]	semi-supervised	92.4	92.1
Huang & Harper (2010) ensemble [17]	semi-supervised	92.8	92.4

Transformer: 91.3
(40k training sentences)

Do you know what constituency parsing is?

Yes, I am familiar with constituency parsing. Constituency parsing, also known as phrase structure parsing or syntactic parsing, is a natural language processing (NLP) technique used to analyze the grammatical structure of a sentence. It aims to create a parse tree that represents the hierarchical organization of words into phrases and higher-level grammatical units called constituents.

In a constituency parse tree, the nodes represent the constituents (such as noun phrases, verb phrases, and prepositional phrases), and the edges connect the constituents to their sub-constituents, ultimately linking them to the individual words in the sentence. The goal of constituency parsing is to understand the syntactic relationships between the words in a sentence, which can be useful for various NLP tasks, such as machine translation, information extraction, and sentiment analysis.

There are several approaches to constituency parsing, including rule-based methods, probabilistic methods, and machine learning-based methods. Some well-known algorithms for constituency parsing are the Earley parser, the CYK parser, and the Chart parser. More recent approaches often use neural networks, like recurrent neural networks (RNNs) or transformers, to capture complex linguistic patterns and dependencies.

That's great! Could you give me an example? E.g., how would you parse "John has a dog."?

Sure! I'll provide a constituency parse for the sentence "John has a dog." Here's a simple parse tree, represented in bracket notation:

csharp Copy code

```
[S
    [NP
        [NNP John]
    ]
    [VP
        [VBZ has]
        [NP
            [DT a]
            [NN dog]
        ]
    ]
]
```

In this parse tree, the sentence is represented as a complete constituent labeled "S" for Sentence. The tree is then broken down into the following constituents:

In this parse tree, the sentence is represented as a complete constituent labeled "S" for Sentence. The tree is then broken down into the following constituents:

1. NP (Noun Phrase): Represents a noun phrase in the sentence.
   - NNP (Proper Noun, Singular): "John"
2. VP (Verb Phrase): Represents a verb phrase in the sentence.
   - VBZ (Verb, 3rd person singular present): "has"
   - NP (Noun Phrase): Represents a noun phrase within the verb phrase.
     - DT (Determiner): "a"
     - NN (Noun, Singular or Mass): "dog"

The parse tree shows the hierarchical structure of the sentence, capturing the syntactic relationships between words and phrases. It helps identify the subject (John), verb (has), and object (a dog) in the sentence.

Woderful!! Could you show me the tree (just the tree, no discussion) for "Renewed concern about the health of the banking system dragged down the stock market on Friday."

Here's a constituency parse tree for the sentence "Renewed concern about the health of the banking system dragged down the stock market on Friday." in bracket notation:

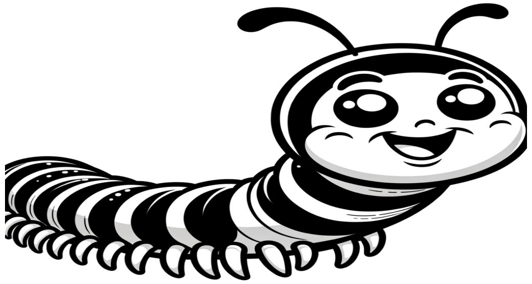
csharp Copy code

```
[S
    [NP
        [JJ Renewed]
        [NN concern]
        [PP
            [IN about]
            [NP
                [DT the]
                [NN health]
                [PP
                    [IN of]
                    [NP
                        [DT the]
                        [NN banking]
                        [NN system]
                    ]
                ]
            ]
        ]
    ]
    [VP
        [VBD dragged]
        [PRT
            [RP down]
        ]
        [NP
            [DT the]
            [NN stock]
            [NN market]
        ]
        [PP
            [IN on]
            [NP
                [NNP Friday]
            ]
        ]
    ]
]
```

Regenerate response

Beyond Transformers: ask computational complexity

- *Recurrence in depth*
  - *Universal Transformer*
  - *ALBERT*
- *Chain of thought*
  - *PoT, tools*
  - *Needs RL to train*



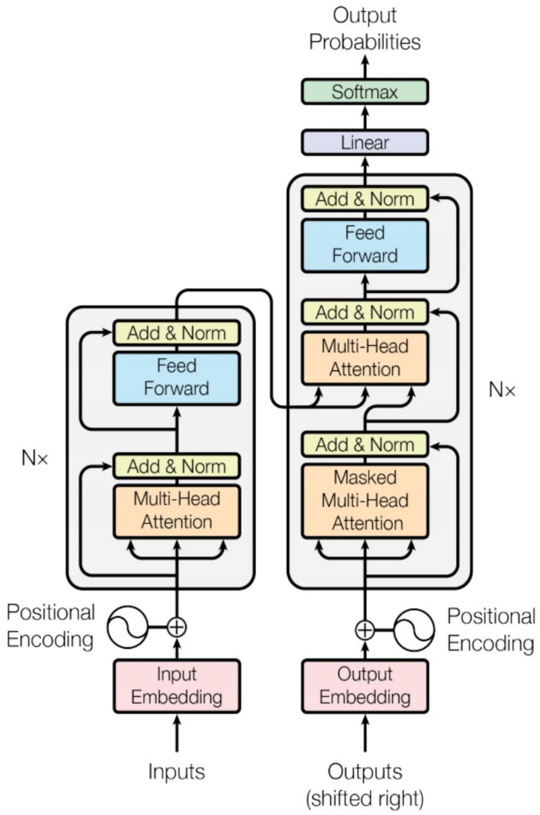


Figure 1: The Transformer - model architecture.

Recurrence is hard to train*

3.3 Training Techniques

Here we describe the training methods that we used to improve our results. Note that we applied these methods to the LSTM+A baseline as well, to keep the above comparison fair. We focus on the most important elements of our training regime, all less relevant details can be found in the code which is released as open-source.[1]

Grid search. Each result we report is obtained by running a grid search over $3^6 = 729$ instances. We consider 3 settings of the learning rate, initial parameters scale, and 4 other hyperparameters discussed below: the relaxation pull factor, curriculum progress threshold, gradient noise scale, and dropout. An important effect of running this grid search is also that we train 729 models with different random seeds every time. Usually only a few of these models generalize to 2000-bit numbers, but a significant fraction works well on 200-bit numbers, as discussed below.

Curriculum learning. We use a curriculum learning approach inspired by Zaremba & Sutskever (2015a). This means that we train, e.g., on 7-digit numbers only after crossing a curriculum progress threshold (e.g., over 90% fully correct outputs) on 6-digit numbers. However, with 20% probability we pick a minibatch of d-digit numbers with d chosen uniformly at random between 1 and 20.

Gradients noise. To improve training speed and stability we add noise to gradients in each training step. Inspired by the schedule from Welling & Teh (2011), we add to gradients a noise drawn from the normal distribution with mean 0 and variance inversely proportional to the square root of step-number (i.e., with standard deviation proportional to the 4-th root of step-number). We multiply this noise by the gradient noise scale and, to avoid noise in converged models, we also multiply it by the fraction of non-fully-correct outputs (which is 0 for a perfect model).

Gate cutoff. In Section 2 we defined the gates in a CGRU using the sigmoid function, e.g., we wrote $u = \sigma(U' * s + B')$. Usually the standard sigmoid function is used, $\sigma(x) = \frac{1}{1+e^{-x}}$. We found that adding a hard threshold on the top and bottom helps slightly in our setting, so we use $1.2\sigma(x) - 0.1$ cut to the interval $[0, 1]$, i.e., $\sigma'(x) = \max(0, \min(1, 1.2\sigma(x) - 0.1))$.

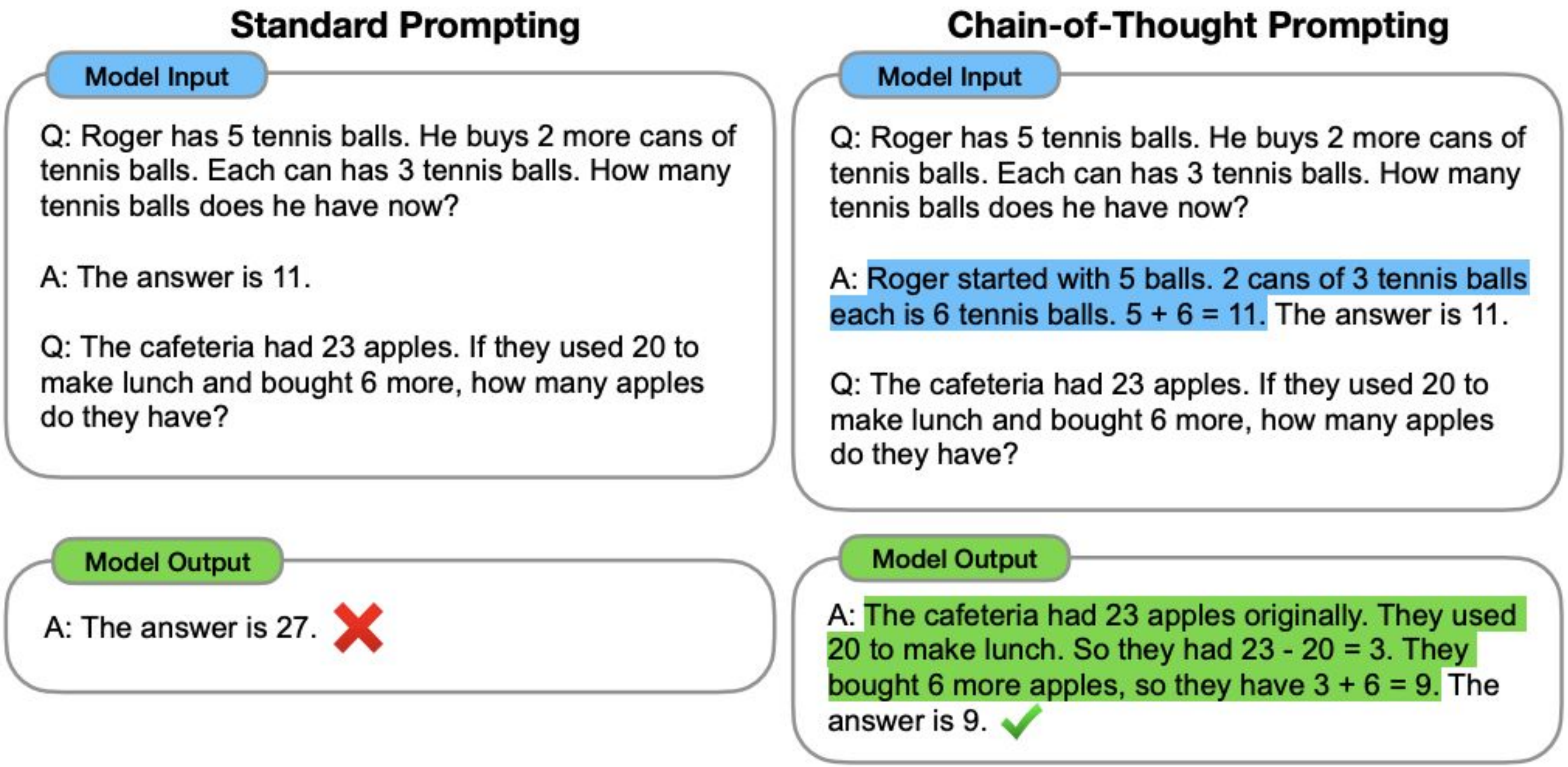
3.3.1 Dropout on recurrent connections

3.3.2 Parameter sharing relaxation.

Neural GPUs Learn Algorithms
(recurrent convolutions)

* but maybe not with the right architecture and algorithm?
(Improved Neural GPU, S4, S5, ⋯)

Chain of Thought (CoT) needs a good prior



Chain-of-Thought Prompting Elicits Reasoning in Large Language Models

Transformer with CoT is Turing Complete

GPT is becoming a Turing machine: Here are some ways to program it

Ana Jojic [1] **Zhen Wang** [2] **Nebojsa Jojic** [3]

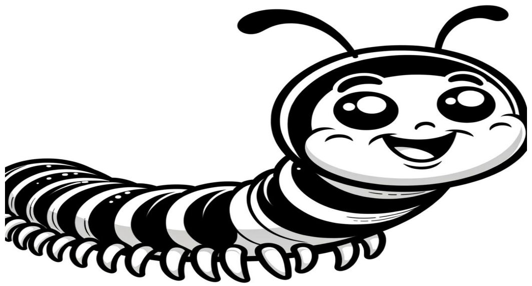
25 Mar 2023

Abstract

We demonstrate that, through appropriate prompting, GPT-3 family of models can be triggered to perform iterative behaviours necessary to execute (rather than just write or recall) programs that involve loops, including several popular algorithms found in computer science curricula or software developer interviews. We trigger execution and description of **iterations** by **regimenting self-attention** (IRSA) in one (or a combination)

clude descriptions of procedures and even computer programs (Chen et al., 2021). They have demonstrated a surprisingly high competency in retrieving knowledge from the training data and generalizing it to new, slightly different situations. The models are typically evaluated on "in-context learning" tasks, i.e., zero- and few-shot prompting, with results implying that these models compress iterative reasoning into a savant-like ability to directly reach correct conclusions without a disciplined step-by-step process (Wei et al., 2022; Kojima et al., 2022). It is difficult to understand if these abilities are simply due to a high similarity with the

Learning CoTs



- *RL from the outcome*
  - *sparse signal*
- *Structured CoTs*
  - *need to define a structure*
- *Knowledge libraries and delegation*
  - *early days*
- *Parallelism in CoTs*
  - *consensus*
    - *great at MATH*
    - *single answer only*
  - *verifier guided*
    - *can do SAT*

Structured CoTs: *Let's Verify Step By Step*

Problem

Find the unique $\mathbf{odd}$ integer t such that $0 < t < 23$ and $t + 2$ is the inverse of t modulo 23.

3 / 4

Model attempt

In step 7 and 8, GPT-4 starts performing guess-and-check. This is a common place the model might hallucinate, by claiming a particular guess is successful when it isn't. In this case, the reward model verifies each step and determines that the chain-of-thought is correct.

1 I need to find an odd integer t that satisfies the congruence $t + 2 \equiv t^{-1} \pmod{23}$.

2 To do that, I can multiply both sides by t and get $t^2 + 2t \equiv 1 \pmod{23}$.

3 Then I can complete the square by adding and subtracting $(2/2)^2 = 1$ to the left side and get $(t+1)^2 - 1 \equiv 1 \pmod{23}$.

4 Simplifying, I get $(t+1)^2 \equiv 2 \pmod{23}$.

5 Now I need to find a square root of 2 modulo 23.

6 I can try some values of $t + 1$ and see if they work.

7 For example, if $t + 1 = 3$, then $(t+1)^2 = 9 \not\equiv 2 \pmod{23}$.

8 If $t + 1 = 5$, then $(t+1)^2 = 25 \equiv 2 \pmod{23}$.

9 So one possible value of t is $t = 5 - 1 = 4$.

10 But this is not odd, so I need to find another one.

11 I know that if $t + 1$ is a square root of 2 modulo 23, then so is $-t - 1$.

12 So I can try $-t - 1 = 5$ and get $t = -6$.

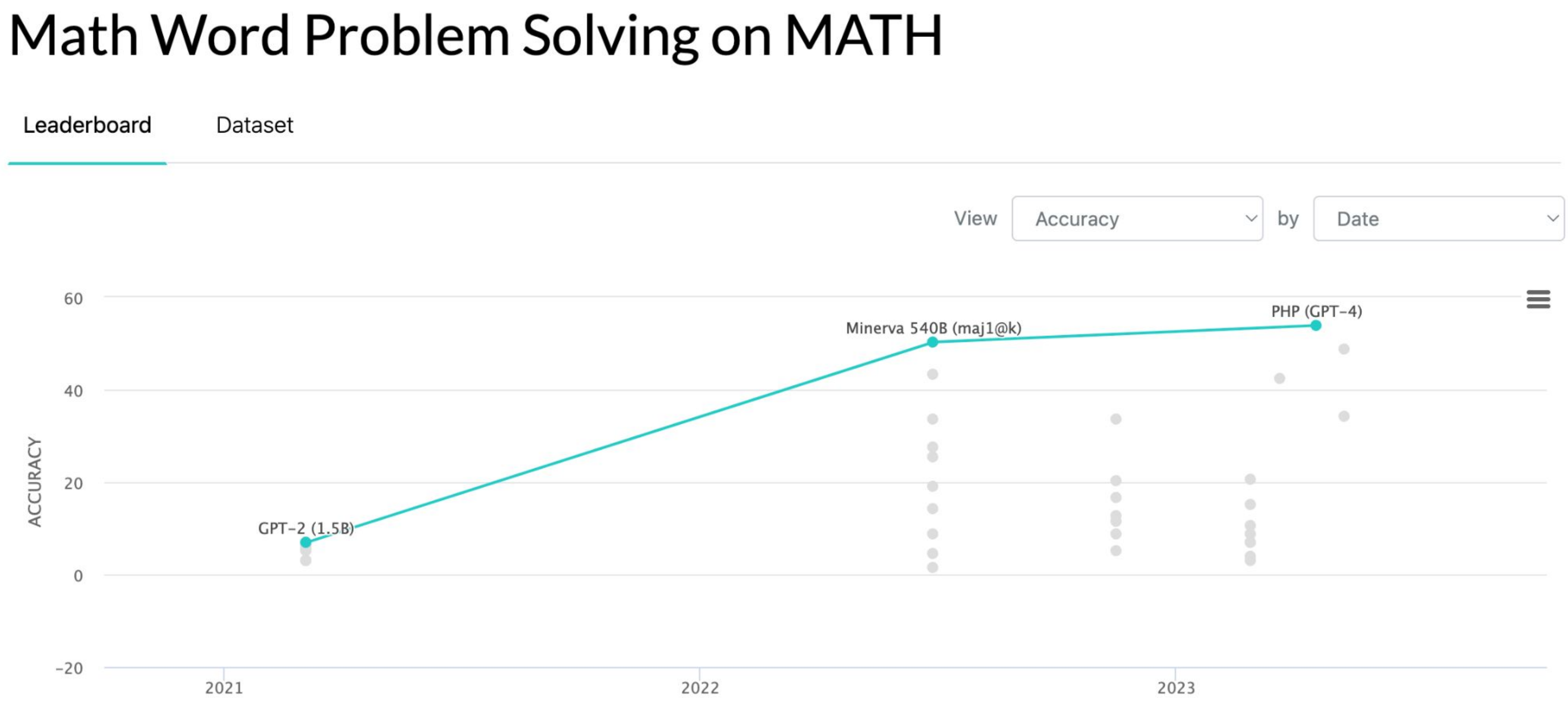
13 But this is negative, so I need to add 23 to get a positive value.

14 So another possible value of t is $t = -6 + 23 = 17$.

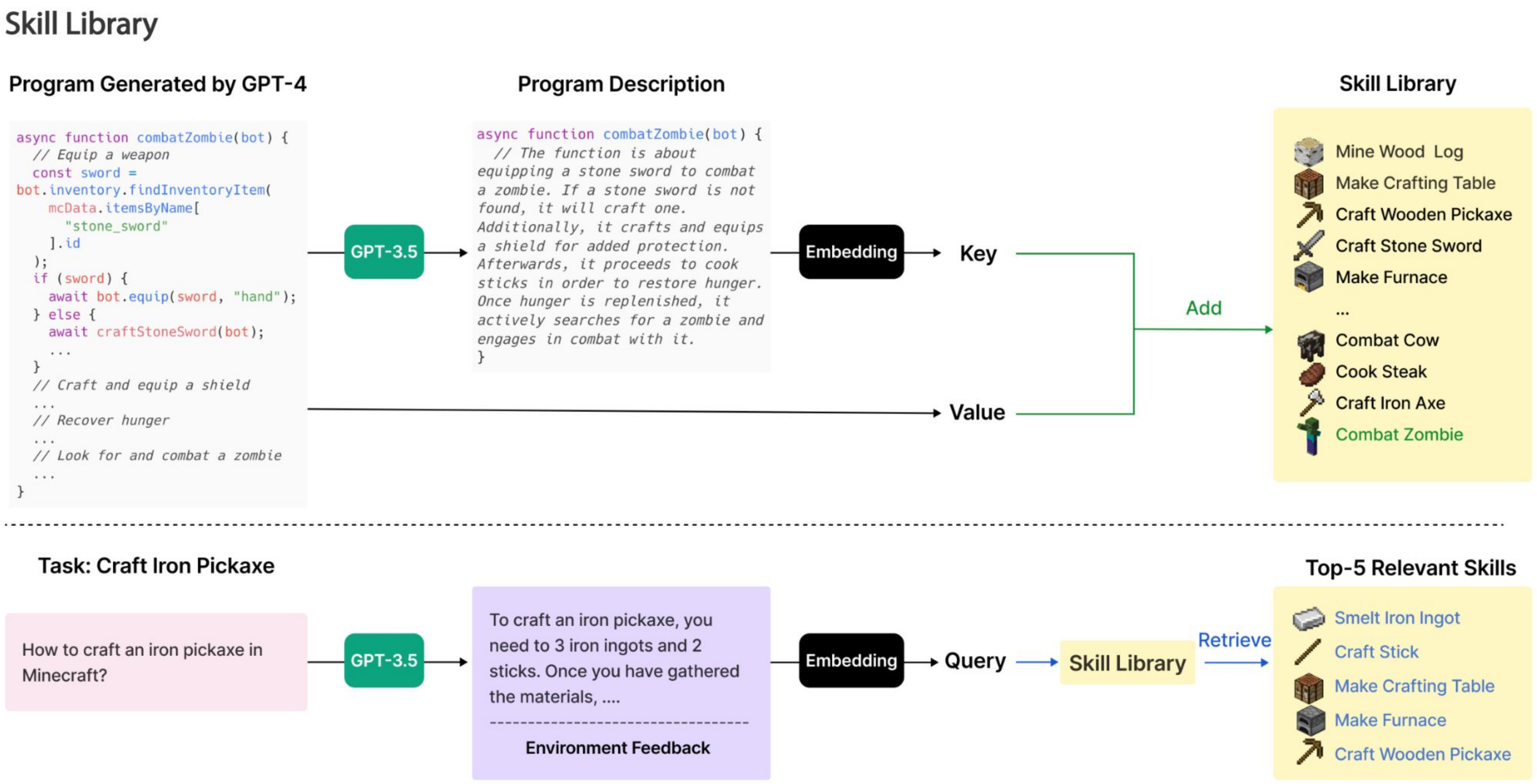
15 This is odd and satisfies the original congruence, so this is the unique solution.

16 Answer: 17

Structured CoTs: *Let's Verify Step By Step* → 78%



CoTs to build a knowledge library?



Parallel CoTs and verifiers

- Can LLMs without CoT solve SAT? No (unless P=NP)
- Can Transformers multiply long numbers? No (unless ...)

- Short CoTs don't change SAT and long CoTs are hard to learn
- But what if we allow a lot of parallel CoTs and learn a verifier?
  - can train to solve SAT from examples (though brutally)

assignment important?

formula to check assignment important?

assignment important? answer

Beyond Transformers: ask computational complexity

The next major improvement in DL*

- *Recurrence in depth*
  - *hard to train but new methods appear*

Chain of thought learning

  - *improved RL*
  - *stepwise decomposition*
  - *libraries and delegation*
  - *parallelism with verifiers*

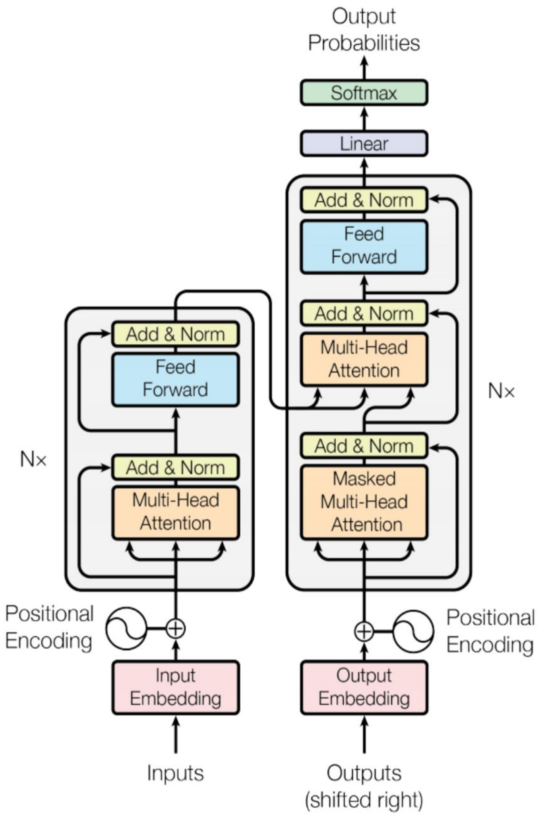


Figure 1: The Transformer - model architecture.

Computational complexity

Helps deep learning