Text-to-text transformers

Course structure

1. The Transformer: motivation, original architecture and attention mechanism.
2. Transformer-based Encoders. Masked language models based on the Transformer architecture. BERT and related models.
3. Classification and sequence tagging with Transformers. Using encoders to generate feature representation for various NLU tasks.
4. Transformer-based Decoders. Generation of text based on the Transformer architecture. GPT and related decoders. Text generation methods. Prompt tuning.
5. Towards ChatGPT. Instruction tuning, Reinforcement learning from Human Feedback (RLHF) and main modern LLMs.
6. Efficient Transformers
7. Sequence to sequence tasks: machine translation, text detoxification, question answering, dialogue.
8. Uncertainty estimation for Transformers and NLP tasks.
9. Multilingual Language Models based on Transformer architecture.
10. Multimodal and vision Transformers.
11. Transformers for event sequences.
12. Transformers for tabular data.
13. RAG with Transformers. Introduction to AGI agents.

Outline

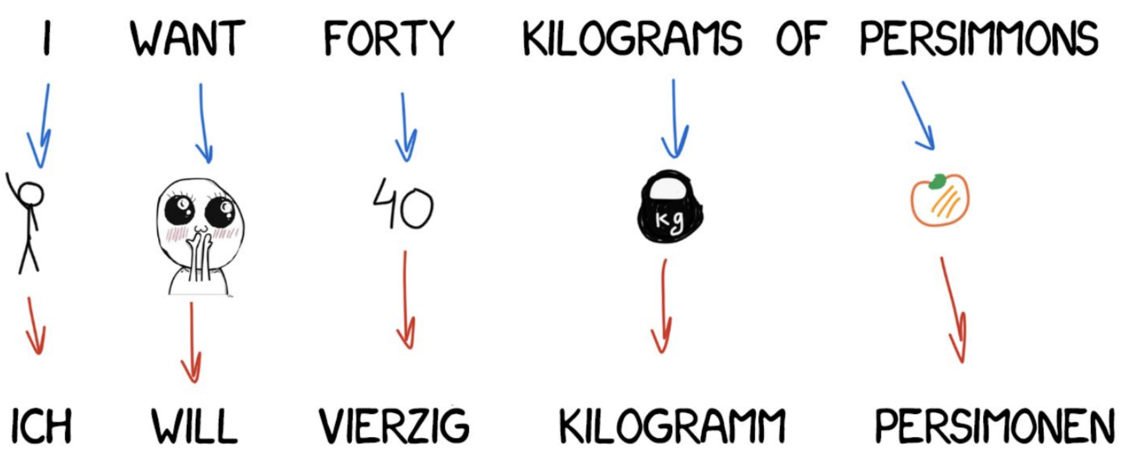
- Sequence-to-sequence tasks: machine translation, text detoxification, question answering, dialogue response generation
- Pretrained sequence-to-sequence models
- Technical tricks for training and inference: infrastructure and performance.

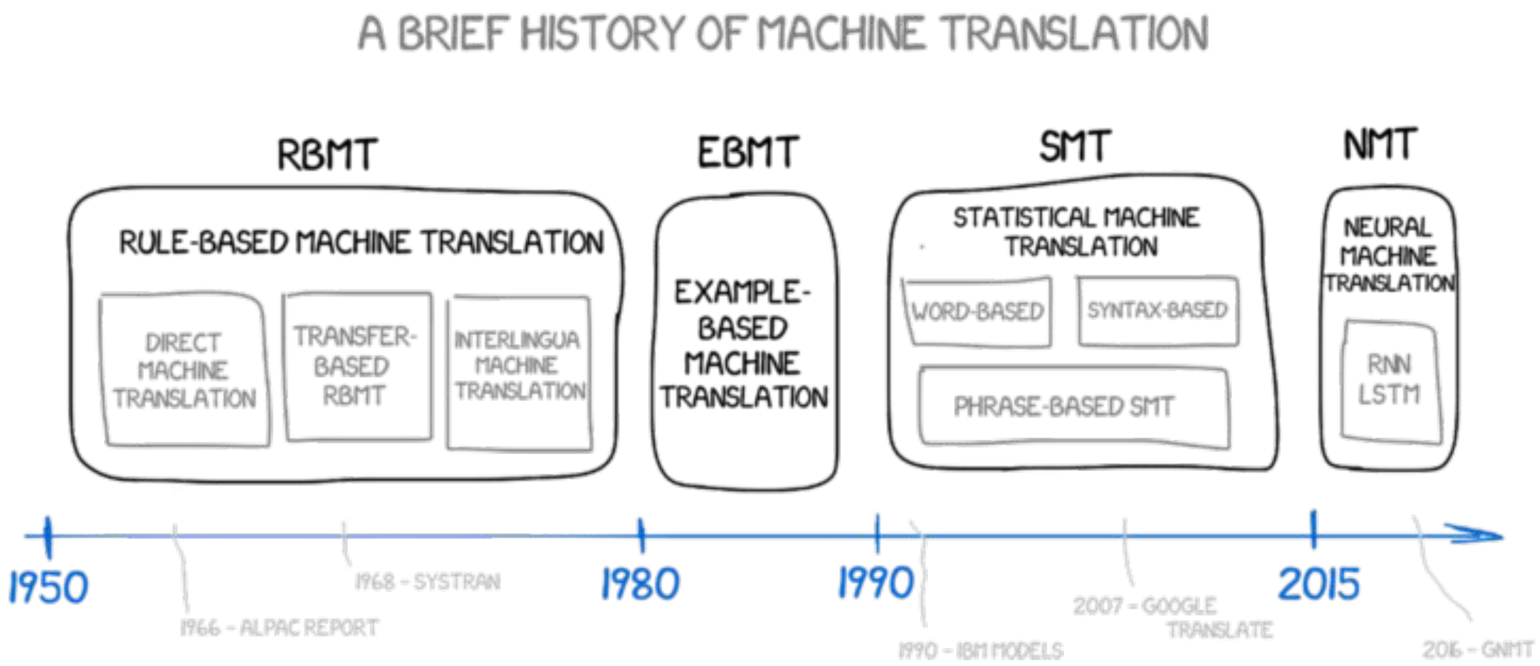
Seq2seq tasks

What seq2seq tasks do you know?

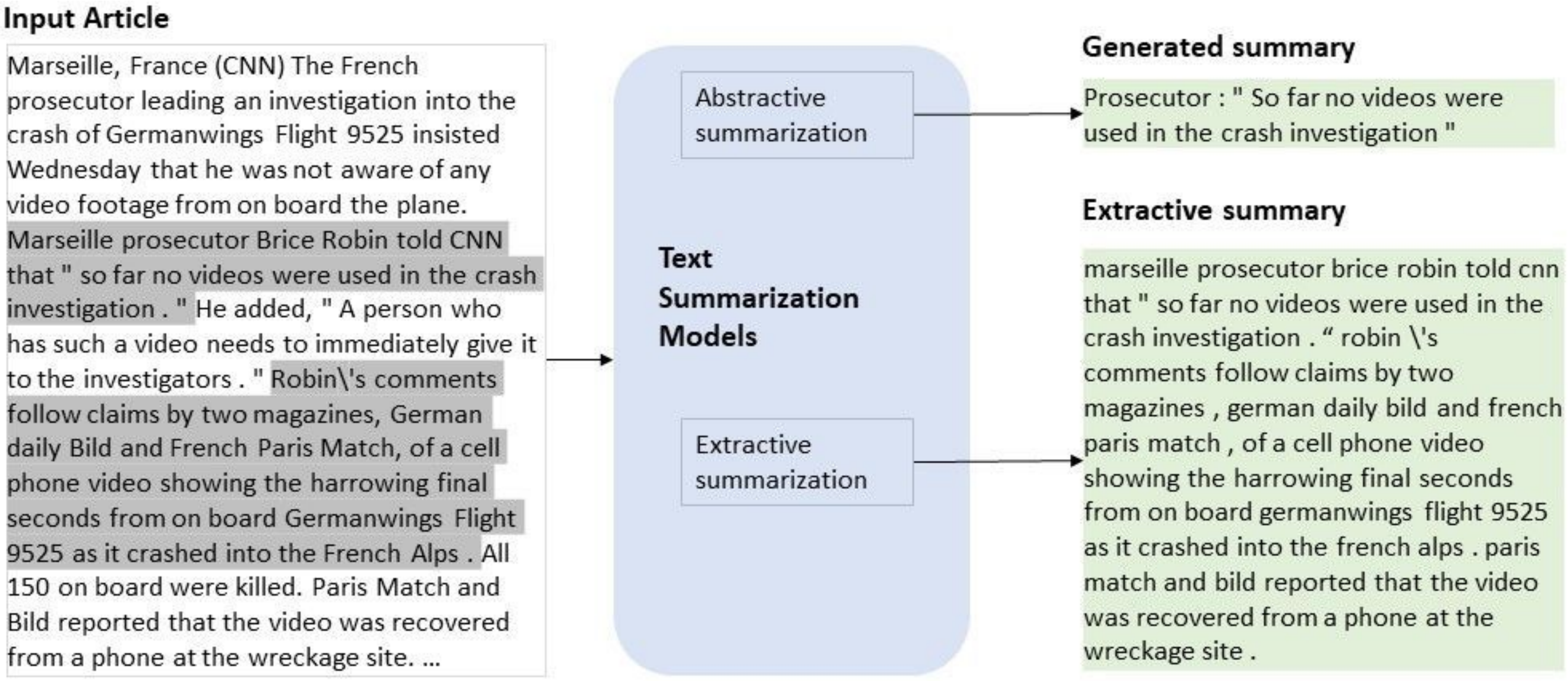
- Machine translation
- Summarization
- Paraphrasing
  - Text style transfer
- Question answering
- Dialogue response generation
- Spelling correction
- Data-to-text generation
  - From structured data
  - Image captioning
  - Automatic speech recognition

Machine Translation





Summarization



Paraphrasing

Original: I need to go to the grocery store to buy some food.

Paraphrased: I have to visit the supermarket in order to purchase groceries.

Original: The cat is sleeping on the mat.

Paraphrased: The feline is resting peacefully on the floor covering.

Original: She is a talented pianist with excellent musical skills.

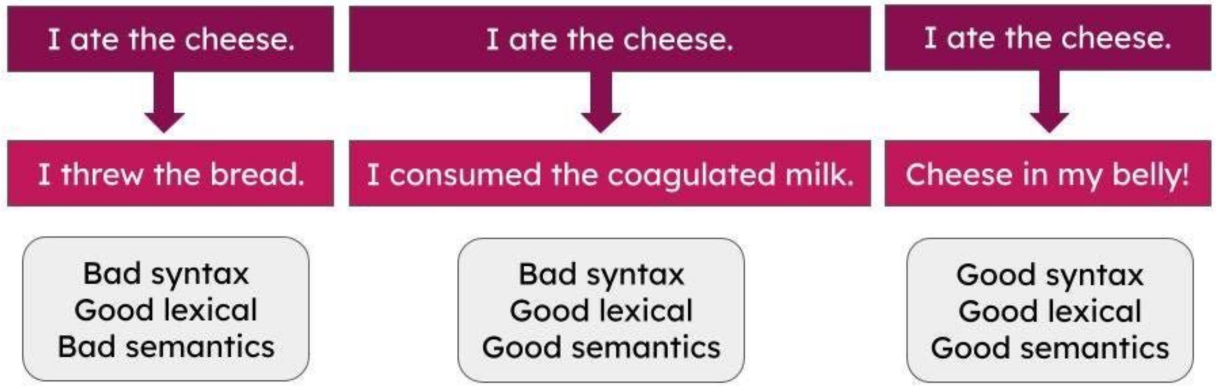
Paraphrased: Her musical abilities are exceptional, particularly in playing the piano.

Original: The teacher explained the concept in a clear and concise manner.

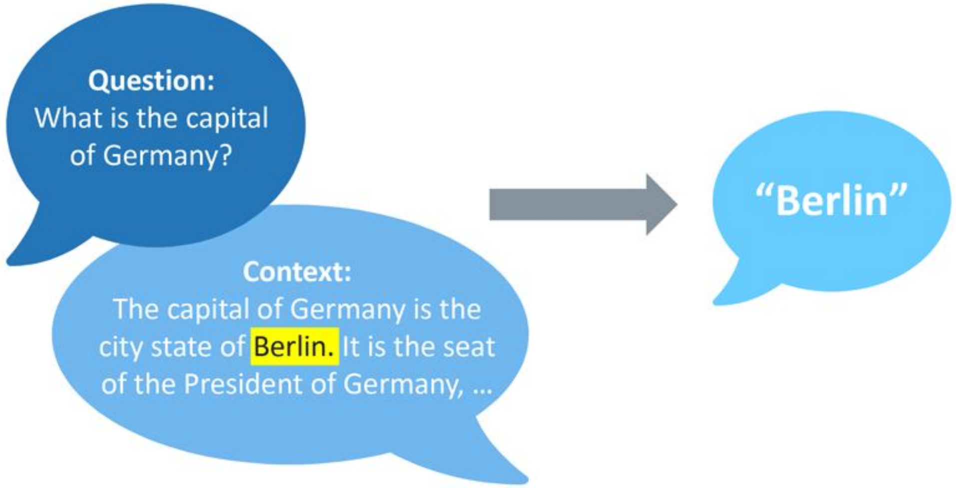
Paraphrased: The instructor provided a lucid and succinct explanation of the concept.

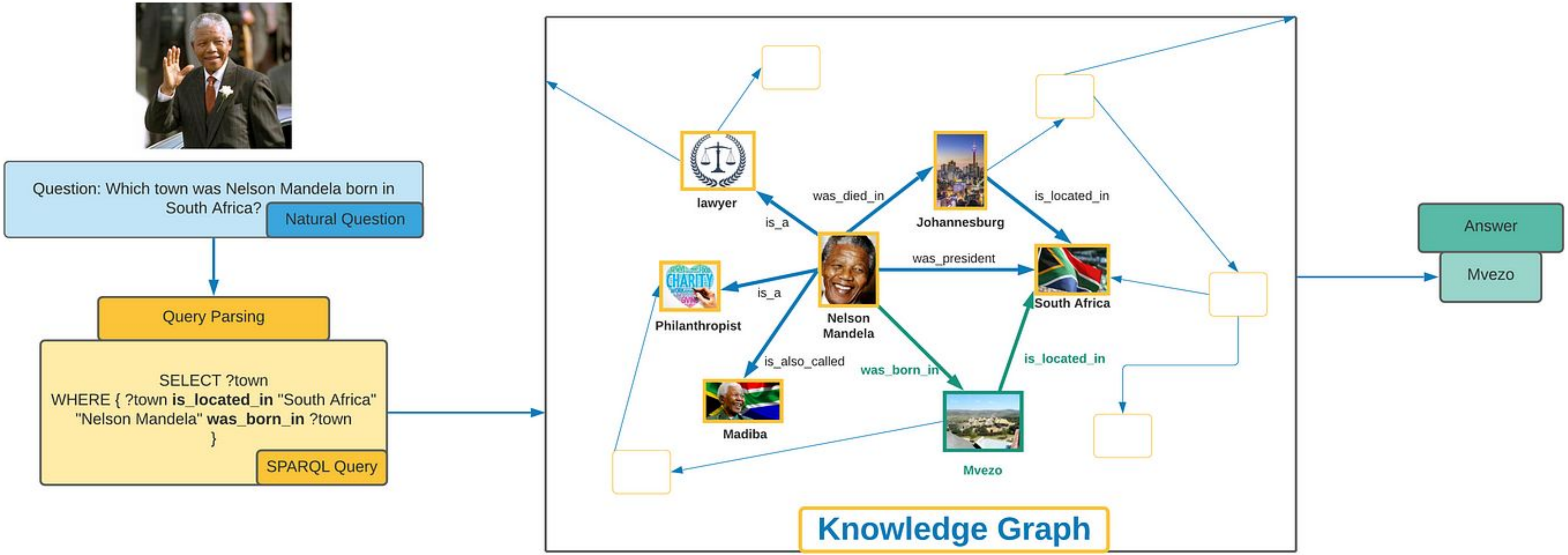
Original: I want to travel and explore different cultures around the world.

Paraphrased: My desire is to journey and discover diverse cultures across the globe.

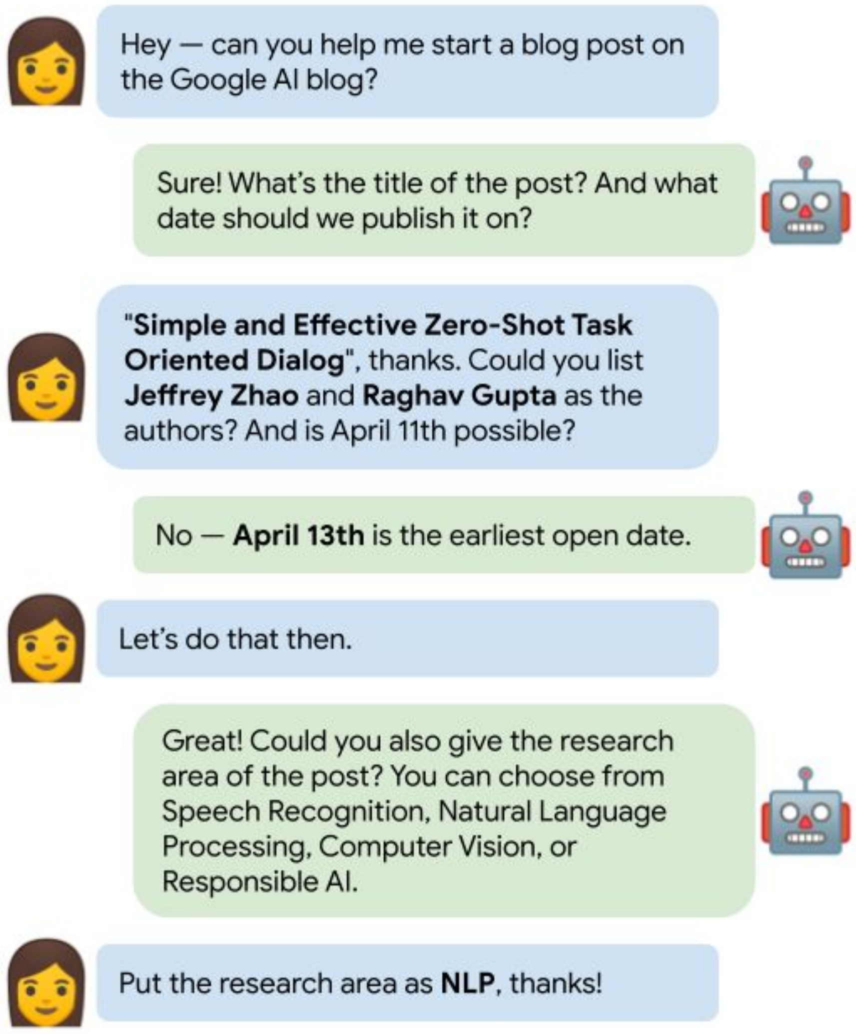


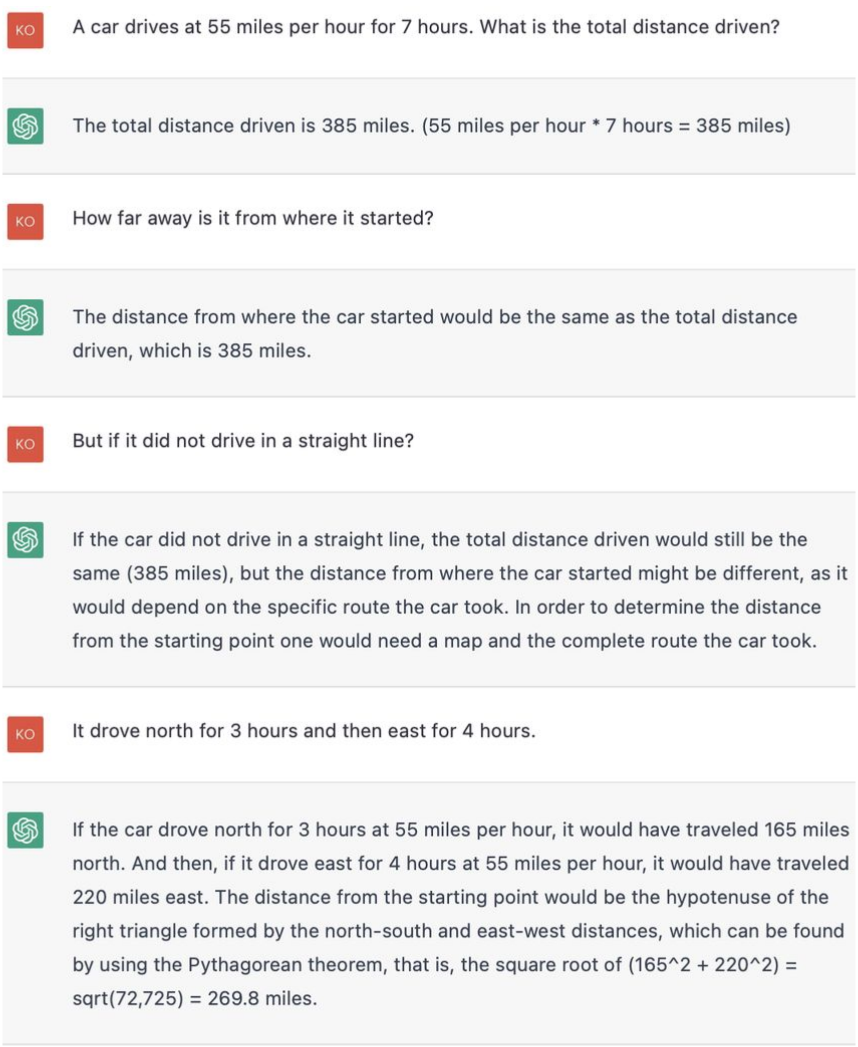
Question Answering





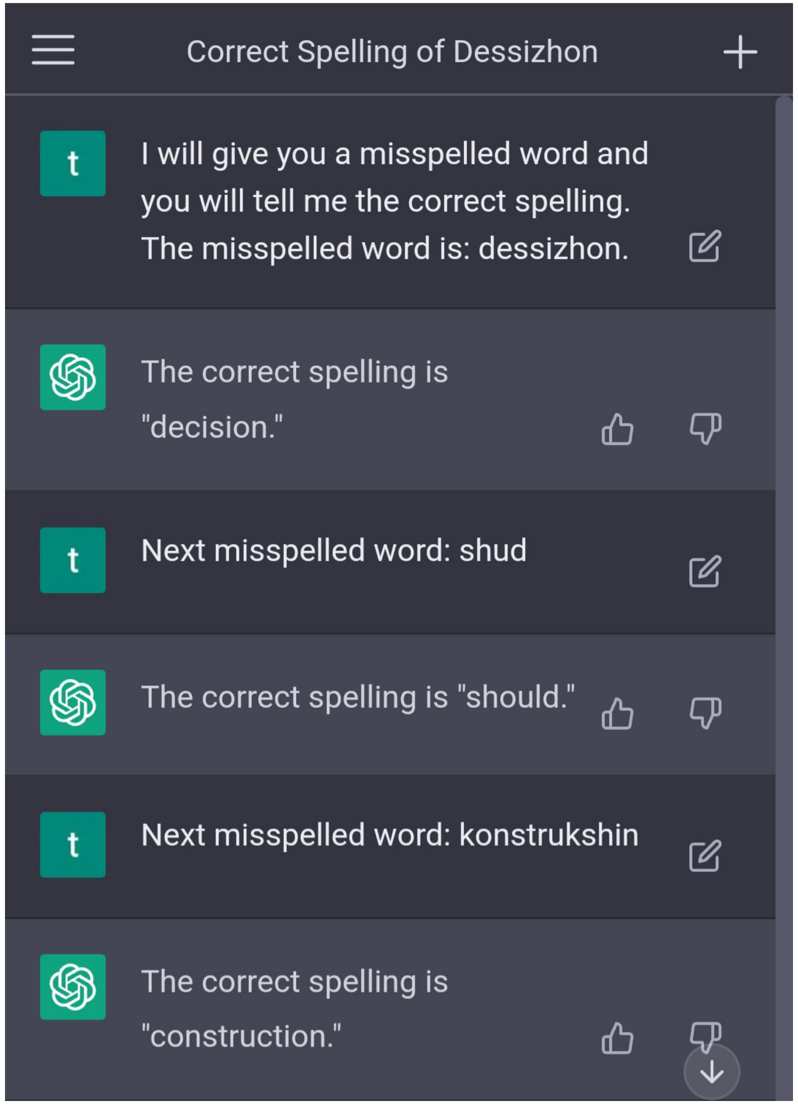
Dialogue response generation



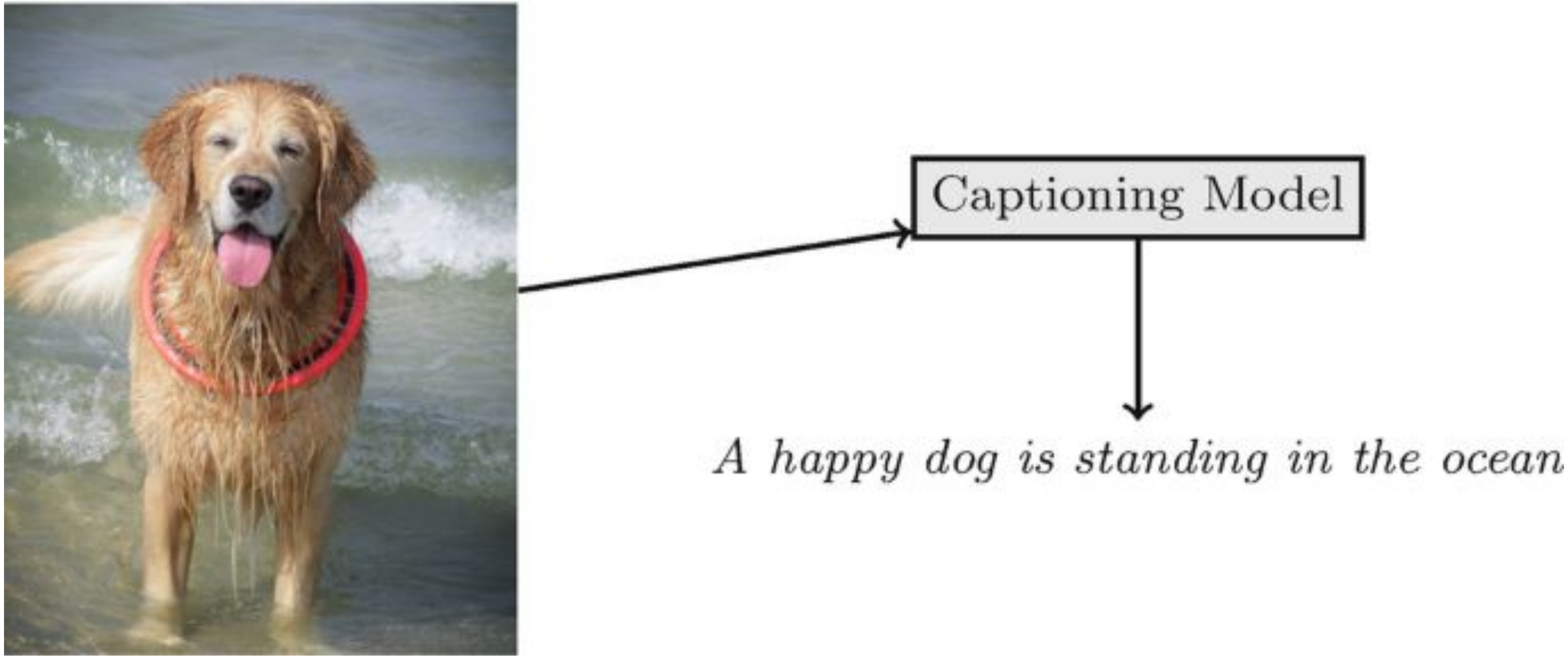


Spelling correction

wrong sentence	correct sentence	misused tokens		Type of Similarity
他派了很多照片。 He sent a lot of photos.	他拍了很多照片。 He took a lot of photos.	(派&拍)	→	phonological (sounds similar)
他门才开始考试。 His door starts the exam.	他们才开始考试。 They start the exam.	(门&们)	→	visual (looks similar)



Data-to-text generation



Are these different tasks?

- All these tasks can be solved by:
  - fine-tuning a transformer on a parallel corpus
  - predicting the answer by sampling or beam search decoding
- What should be different?
  - Pretraining data and objectives
    - E.g. back-translation for translation
    - E.g. sentence gaps for summarization
  - Decoding methods
    - Balancing diversity vs correctness

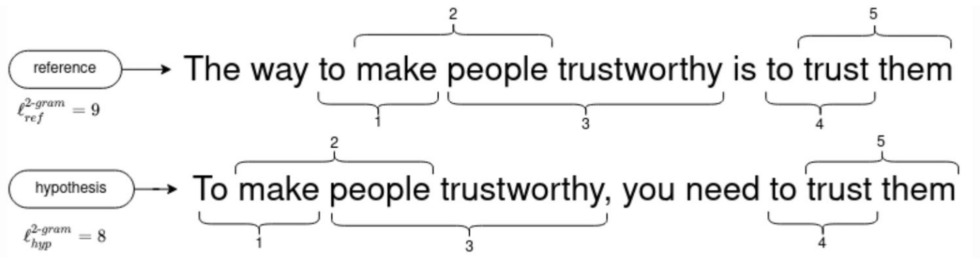
Evaluating seq2seq models

- What we care about:
    - Fluency/naturalness/grammaticality
    - Consistence with the input (“correctness”)
    - Relevance to the input
    - Informativeness
    - Coherence (good structure)
    - Consistency with the desired style
    - Diversity
- What we typically evaluate:
    - Proportion of common words with a reference text
        - E.g. BLEU, ROUGE, ChrF

BLEU

$$Bleu\ (N) = Brevity\ Penalty \cdot Geometric\ Average\ Precision\ Scores\ (N)$$

$$\begin{aligned} Geometric\ Average\ Precision\ (N) &= exp(\sum_{n=1}^{N} w_n\ log\ p_n) \\ &= \prod_{n=1}^{N} {p_n}^{w_n} \\ &= (p_1)^{\frac{1}{4}} \cdot (p_2)^{\frac{1}{4}} \cdot (p_3)^{\frac{1}{4}} \cdot (p_4)^{\frac{1}{4}} \end{aligned}$$



The following table details the precisions for 4 n-grams.

n-gram	1-gram	2-gram	3-gram	4-gram
p_n	$\frac{7}{9}$	$\frac{5}{8}$	$\frac{3}{7}$	$\frac{1}{6}$

$$Brevity\ Penalty = \begin{cases} 1, & \text{if } c > r \\ e^{(1-r/c)}, & \text{if } c <= r \end{cases}$$

ROUGE

Consider the reference R and the candidate summary C:

- R: The cat is on the mat.
- C: The cat and the dog.

$$RECALL = \frac{Overlapping\, number\, of\, n\text{-}grams}{Number\, of\, n\text{-}grams\, in\, the\, reference}$$

$$PRECISION = \frac{Overlapping\, number\, of\, n\text{-}grams}{Number\, of\, n\text{-}grams\, in\, the\, candidate}$$

ROUGE-1 recall = 3/6 = 0.5

ROUGE-1 precision = 3/5 = 0.6

ROUGE-1 F1-score = 2 * (precision * recall) / (precision + recall) = 0.54

Chr-F1

$$\text{CHRF}\beta = (1+\beta^2)\frac{\text{CHRP}\cdot\text{CHRR}}{\beta^2\cdot\text{CHRP}+\text{CHRR}}$$

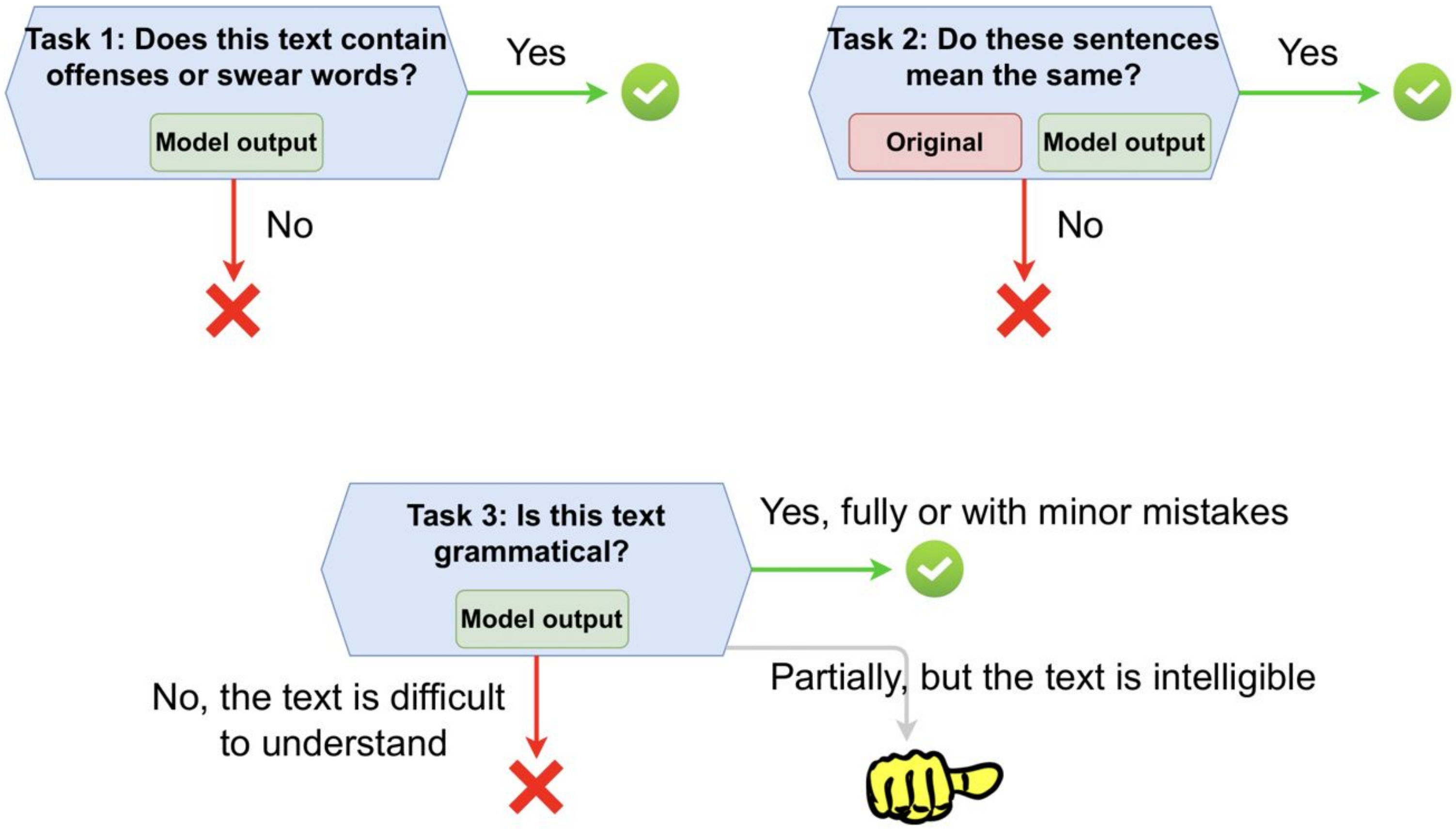
where CHRP and CHRR stand for character n-gram precision and recall arithmetically averaged over all n-grams:

- CHRP
  percentage of n-grams in the hypothesis which have a counterpart in the reference;

- CHRR
  percentage of character n-grams in the reference which are also present in the hypothesis.

and β is a parameter which assigns β times more importance to recall than to precision – if $\beta = 1$, they have the same importance.

Example: text style transfer evaluation



Example: text style transfer evaluation

- The task: paraphrase a text while changing it style in a specific way
- Example metrics for Multilingual Text Detoxification (TextDetox) 2025 challenge:
  - ***style transfer accuracy*** (STA): For this, specifically fine-tuned xlm-roberta-large for toxicity binary classification is used.
  - ***content preservation*** (SIM): We evaluate semantic similarity as cosine similarity of the LaBSE sentence encoder.
  - ***fluency*** (FL): To estimate the adequacy of the text and its similarity to the human-written detoxified references, we calculate **ChrF1** measure.

- **Joint score** (J): an averaged sentence-level multiplication of STA, SIM, and FL:

  J = (STA * SIM * FL). This metric will be used **for ranking models during the automatic evaluation.**

https://pan.webis.de/clef25/pan25-web/text-detoxification.html

These automatic metrics are not very good* =/

Automatic Evaluation

Team Name	STA	SIM	FL	J	ChrF
gleb_shnshn	**0.975**	**0.935**	**0.959**	**0.873**	0.529
orzhan	**0.982**	0.860	**0.969**	**0.822**	0.550
FRC CSC RAS	0.945	0.855	**0.967**	**0.784**	0.571
SomethingAwful	**0.948**	0.819	0.911	0.709	**0.573**
Mindful Squirrel	0.933	0.798	0.885	0.659	0.564
king_menin	0.942	0.728	0.889	0.614	0.497
T5 (baseline)	0.796	0.827	0.837	0.560	**0.573**
team_ruprompts	0.804	0.804	0.829	0.542	0.563
Ruprompts (baseline)	0.811	0.793	0.804	0.528	0.547
Barracudas	0.852	0.758	0.785	0.523	0.532
Human References	0.846	0.716	0.783	0.494	**0.773**
NSU team	0.830	0.756	0.757	0.483	0.505
anzak	0.569	**0.892**	0.910	0.441	0.536
Delete (baseline)	0.558	**0.887**	0.852	0.406	0.529

Human Evaluation

Team Name	STA	SIM	FL	J
Human References	**0.888**	**0.824**	0.894	**0.653**
SomethingAwful	0.794	**0.872**	**0.903**	**0.633**
T5 (baseline)	0.791	0.822	**0.925**	**0.606**
FRC CSC RAS	0.734	**0.865**	**0.918**	0.598
Mindful Squirrel	**0.824**	0.791	0.846	0.582
team_ruprompts	0.778	0.809	0.903	0.568
orzhan	0.805	0.782	0.869	0.565
Barracudas	0.790	0.718	0.782	0.505
king_menin	**0.808**	0.697	0.897	0.501
Ruprompts (baseline)	0.803	0.703	0.866	0.493
NSU team	0.767	0.721	0.825	0.455
anzak	0.433	0.624	0.791	0.171
Delete (baseline)	0.387	0.705	0.726	0.162
gleb_shnshn	0.249	0.128	0.238	0.016

* At least, when they are directly optimized when developing the models

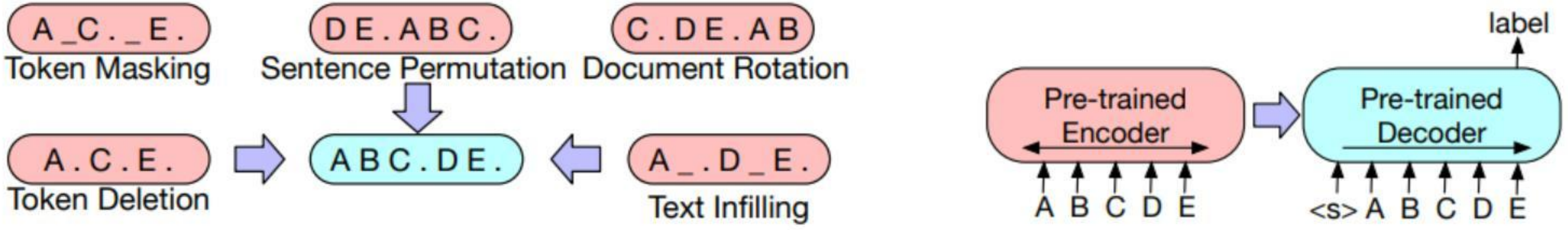
Alternative training objectives

- Text2text models can be fine-tuned with reinforcement learning
  - Sample several generated texts from the model given the same input
  - Score the texts with your automatic quality metrics
  - Make a gradient update to increase the likelihood of the best texts
- In most cases, this simple procedure achieves very high scores
  - However, the model often does this by "hacking" the metrics
  - An adversarial objective might help, but GANs for text are not well developed

Pretrained seq2seq transformers

BART: Bidirectional and Auto-Regressive Transformers

- Same architecture as in (Vaswani, 2017)
  - Replaced ReLU with GeLU and slightly changed initialization
- Pretrained with denoising (reconstructing corrupted text)



- The model is fine-tuned for downstream tasks with task-specific heads on top of the decoder
- SOTA in summarization, question answering, and some other tasks

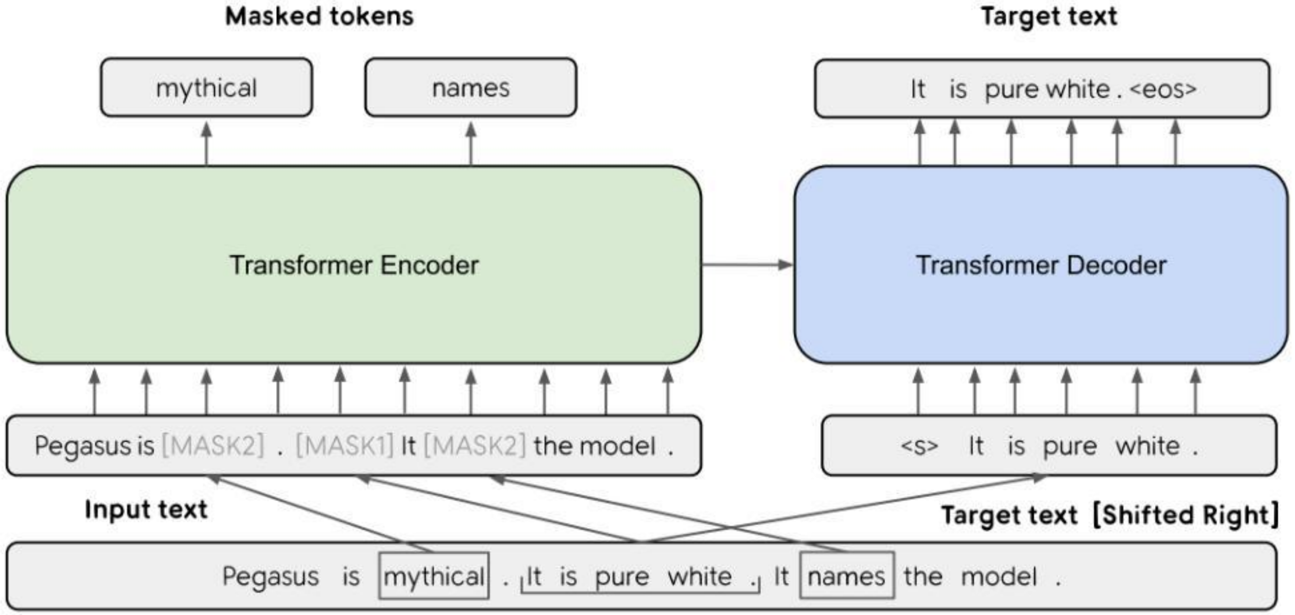
https://arxiv.org/abs/1910.13461

BART: Bidirectional and Auto-Regressive Transformers

Source Document (abbreviated)	**BART Summary**
The researchers examined three types of coral in reefs off the coast of Fiji ... The researchers found when fish were plentiful, they would eat algae and seaweed off the corals, which appeared to leave them more resistant to the bacterium Vibrio coralliilyticus, a bacterium associated with bleaching. The researchers suggested the algae, like warming temperatures, might render the corals' chemical defenses less effective, and the fish were protecting the coral by removing the algae.	Fisheries off the coast of Fiji are protecting coral reefs from the effects of global warming, according to a study in the journal Science.
Sacoolas, who has immunity as a diplomat's wife, was involved in a traffic collision ... Prime Minister Johnson was questioned about the case while speaking to the press at a hospital in Watford. He said, "I hope that Anne Sacoolas will come back ... if we can't resolve it then of course I will be raising it myself personally with the White House."	Boris Johnson has said he will raise the issue of US diplomat Anne Sacoolas' diplomatic immunity with the White House.
According to Syrian state media, government forces began deploying into previously SDF controlled territory yesterday. ... On October 6, US President Donald Trump and Turkish President Recep Tayyip Erdoan spoke on the phone. Then both nations issued statements speaking of an imminent incursion into northeast Syria ... . On Wednesday, Turkey began a military offensive with airstrikes followed by a ground invasion.	Syrian government forces have entered territory held by the US-backed Syrian Democratic Forces (SDF) in response to Turkey's incursion into the region.

PEGASUS

- Pre-training with Extracted Gap-sentences for Abstractive Summarization
- Two pretraining objectives: MLM and predicting missing sentences.
  - *Principal sentences* to mask are selected by ROUGE w.r.t. the remaining text
- Result: SOTA on 12/12 summarization tasks



https://arxiv.org/abs/1912.08777

T5: Text-to-Text Transfer Transformer

All tasks are formulated as text-to-text tasks

E.g. classification: predict “ True” or “False”, any other string ---> penalty

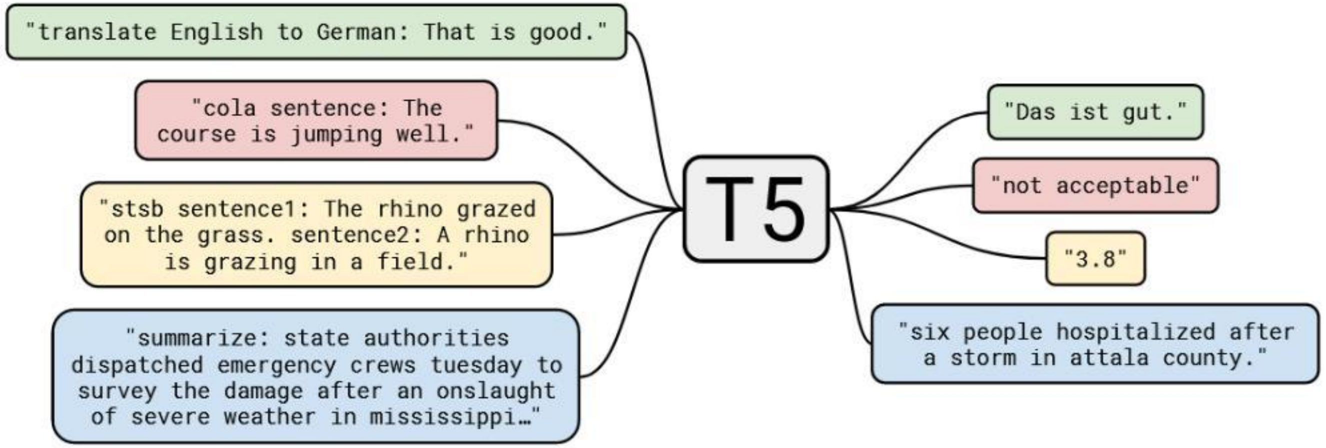


Figure 1: A diagram of our text-to-text framework. Every task we consider—including translation, question answering, and classification—is cast as feeding our model text as input and training it to generate some target text. This allows us to use the same model, loss function, hyperparameters, etc. across our diverse set of tasks. It also provides a standard testbed for the methods included in our empirical survey. “T5” refers to our model, which we dub the “Text-to-Text Transfer Transformer”.

https://arxiv.org/abs/1910.10683

T5 tasks

Trained on several datasets for 18 different tasks which majorly fall into 8 categories.

1. Text Summarization
2. Question Answering
3. Translation
4. Sentiment analysis
5. Natural Language Inference
6. Coreference Resolution
7. Sentence Completion
8. Word Sense Disambiguation

https://arxiv.org/abs/1910.10683

Task Name	Explanation
CoLA	Classify if a sentence is grammatically correct
RTE	Classify whether if a statement can be deducted from a sentence
MNLI	Classify for a hypothesis and premise whether they contradict or contradict each other or neither of both (3 class).
MRPC	Classify whether a pair of sentences is a rephrasing of each other (semantically equivalent)
QNLI	Classify whether the answer to a question can be deducted from an answer candidate.
QQP	Classify whether a pair of questions is a rephrasing of each other (semantically equivalent)
SST2	Classify the sentiment of a sentence as positive or negative
STSB	Classify the sentiment of a sentence on a scale from 1 to 5 (21 Sentiment classes)
CB	Classify for a premise and a hypothesis whether they contradict each other or not (binary).
COPA	Classify for a question, premise, and 2 choices which choice the correct choice is (binary).
MultiRc	Classify for a question, a paragraph of text, and an answer candidate, if the answer is correct (binary)
WiC	Classify for a pair of sentences and a disambiguous word if the word has the same meaning in both sentences.
WSC/DPR	Predict for an ambiguous pronoun in a sentence what it is referring to.
Summarization	Summarize text into a shorter representation.
SQuAD	Answer a question for a given context.
WMT1	Translate English to German
WMT2	Translate English to French
NQ	Closed Book Answering on Natural Questions(nq) Corpus

T5 architecture

- Mostly a vanilla encoder-decoder transformer, but:
  - No bias in layer normalization (only scale)
  - Layer normalization before attention and FFN (outside the residual path)
  - Instead of position embeddings, incorporate relative positions into attention

Vanilla transformer attention

$$\alpha_{ij} = \frac{\exp e_{ij}}{\sum_{k=1}^{n} \exp e_{ik}} \qquad e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T}{\sqrt{d_z}}$$

Attention with relative positions

$$e_{ij} = \frac{x_i W^Q (x_j W^K)^T + x_i W^Q (a_{ij}^K)^T}{\sqrt{d_z}}$$

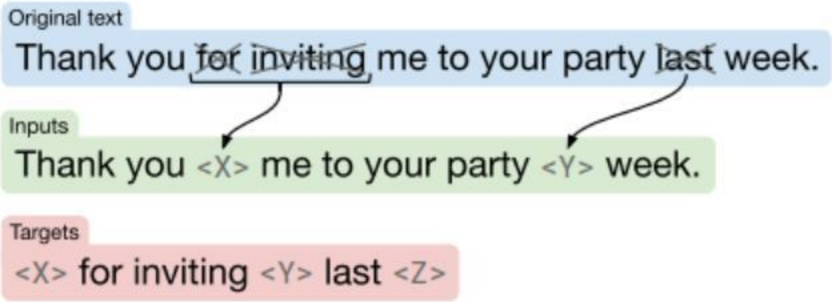
$$a_{ij}^K = w_{\mathrm{clip}(j-i,k)}^K$$
$$a_{ij}^V = w_{\mathrm{clip}(j-i,k)}^V$$
$$\mathrm{clip}(x,k) = \max(-k, \min(k,x))$$

- Why relative positions?
  - When absolute positions embeddings are added on the first layer, the model can “forget” about them in the following layers
  - With this approach, the model is not limited by sequence length

https://arxiv.org/abs/1910.10683

T5 pretraining

- C4 dataset: Colossal Clean Crawled Corpus
- Unsupervised objective: generalized masked language modelling



On downstream tasks, this objective is better than LM.

Objective	GLUE	CNNDM	SQuAD	SGLUE	EnDe	EnFr	EnRo
Prefix language modeling	80.69	18.94	77.99	65.27	**26.86**	39.73	**27.49**
BERT-style (Devlin et al., 2018)	**82.96**	**19.17**	**80.65**	**69.85**	**26.78**	**40.03**	**27.41**
Deshuffling	73.17	18.59	67.61	58.47	26.11	39.30	25.62

Objective	GLUE	CNNDM	SQuAD	SGLUE	EnDe	EnFr	EnRo
BERT-style (Devlin et al., 2018)	82.96	19.17	**80.65**	69.85	26.78	**40.03**	27.41
MASS-style (Song et al., 2019)	82.32	19.16	80.10	69.28	26.79	**39.89**	27.55
★ Replace corrupted spans	83.28	**19.24**	**80.88**	**71.36**	**26.98**	39.82	**27.65**
Drop corrupted tokens	**84.44**	**19.31**	**80.52**	68.67	**27.07**	39.76	**27.82**

Besides the objective type, the paper explores many more design choices (e.g. encoder + decoder vs decoder-only)

https://arxiv.org/abs/1910.10683

T5 training

- AdaFactor for optimization
- 2^19 = 524,288 steps on C4 before fine-tuning
- fine-tuned for 2^18 = 262,144 steps on all tasks
- Maximum sequence length of 512 and a batch size of 128 sequence

"Whenever possible, we "pack" multiple sequences into each entry of the batch so that our batches contain roughly 2^16 = 65,536 tokens."

In total, this batch size and number of steps corresponds to pre-training on 2^35 ≈ 34B tokens. This is considerably less than BERT (137B tokens), or RoBERTa (2.2T tokens)

- a reasonable computational budget
- still providing a sufficient amount of pre-training for acceptable performance

https://arxiv.org/abs/1910.10683

T5 results

- A single model fine-tuned for many diverse (English) NLP tasks
  - Later, multilingual T5 models have been released, but without fine-tuning
- SOTA on many of them
  - Question answering
  - Summarization
  - Some classification tasks

Model	GLUE Average	CoLA Matthew's	SST-2 Accuracy	MRPC F1	MRPC Accuracy	STS-B Pearson	STS-B Spearman
Previous best	89.4[a]	69.2[b]	97.1[a]	**93.6**[b]	**91.5**[b]	92.7[b]	92.3[b]
T5-Small	77.4	41.0	91.8	89.7	86.6	85.6	85.0
T5-Base	82.7	51.1	95.2	90.7	87.5	89.4	88.6
T5-Large	86.4	61.2	96.3	92.4	89.9	89.9	89.2
T5-3B	88.5	67.1	97.4	92.5	90.0	90.6	89.8
T5-11B	**90.3**	**71.6**	**97.5**	92.8	90.4	**93.1**	**92.8**

Model	QQP F1	QQP Accuracy	MNLI-m Accuracy	MNLI-mm Accuracy	QNLI Accuracy	RTE Accuracy	WNLI Accuracy
Previous best	74.8[c]	**90.7**[b]	91.3[a]	91.0[a]	**99.2**[a]	89.2[a]	91.8[a]
T5-Small	70.0	88.0	82.4	82.3	90.3	69.9	69.2
T5-Base	72.6	89.4	87.1	86.2	93.7	80.1	78.8
T5-Large	73.9	89.9	89.9	89.6	94.8	87.2	85.6
T5-3B	74.4	89.7	91.4	91.2	96.3	91.1	89.7
T5-11B	**75.1**	90.6	**92.2**	**91.9**	96.9	**92.8**	**94.5**

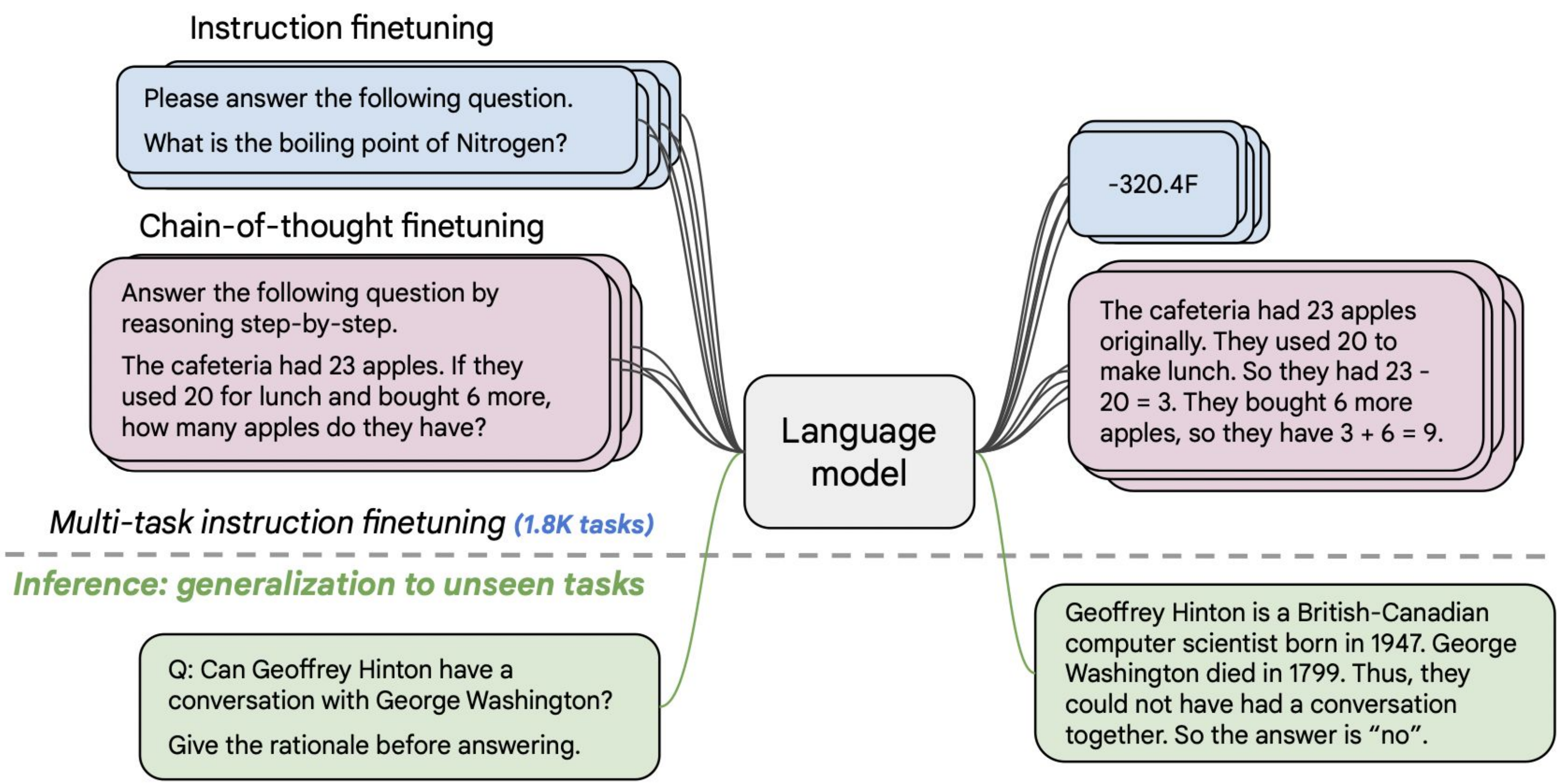
Model	SQuAD EM	SQuAD F1	SuperGLUE Average	BoolQ Accuracy	CB F1	CB Accuracy	COPA Accuracy
Previous best	90.1[a]	95.5[a]	84.6[d]	87.1[d]	90.5[d]	95.2[d]	90.6[d]
T5-Small	79.10	87.24	63.3	76.4	56.9	81.6	46.0
T5-Base	85.44	92.08	76.2	81.4	86.2	94.0	71.2
T5-Large	86.66	93.79	82.3	85.4	91.6	94.8	83.4
T5-3B	88.53	94.95	86.4	89.9	90.3	94.4	92.0
T5-11B	**91.26**	**96.22**	**88.9**	**91.2**	**93.9**	**96.8**	**94.8**

Model	MultiRC F1a	MultiRC EM	ReCoRD F1	ReCoRD Accuracy	RTE Accuracy	WiC Accuracy	WSC Accuracy
Previous best	84.4[d]	52.5[d]	90.6[d]	90.0[d]	88.2[d]	69.9[d]	89.0[d]
T5-Small	69.3	26.3	56.3	55.4	73.3	66.9	70.5
T5-Base	79.7	43.1	75.0	74.2	81.5	68.3	80.8
T5-Large	83.3	50.7	86.8	85.9	87.8	69.3	86.3
T5-3B	86.8	58.3	91.2	90.4	90.7	72.1	90.4
T5-11B	**88.1**	**63.3**	**94.1**	**93.4**	**92.5**	**76.9**	**93.8**

Model	WMT EnDe BLEU	WMT EnFr BLEU	WMT EnRo BLEU	CNN/DM ROUGE-1	CNN/DM ROUGE-2	CNN/DM ROUGE-L
Previous best	**33.8**[e]	**43.8**[e]	**38.5**[f]	43.47[g]	20.30[g]	40.63[g]
T5-Small	26.7	36.0	26.8	41.12	19.56	38.35
T5-Base	30.9	41.2	28.0	42.05	20.34	39.40
T5-Large	32.0	41.5	28.1	42.50	20.68	39.75
T5-3B	31.8	42.6	28.2	42.72	21.02	39.94
T5-11B	32.1	43.4	28.1	**43.52**	**21.55**	**40.69**

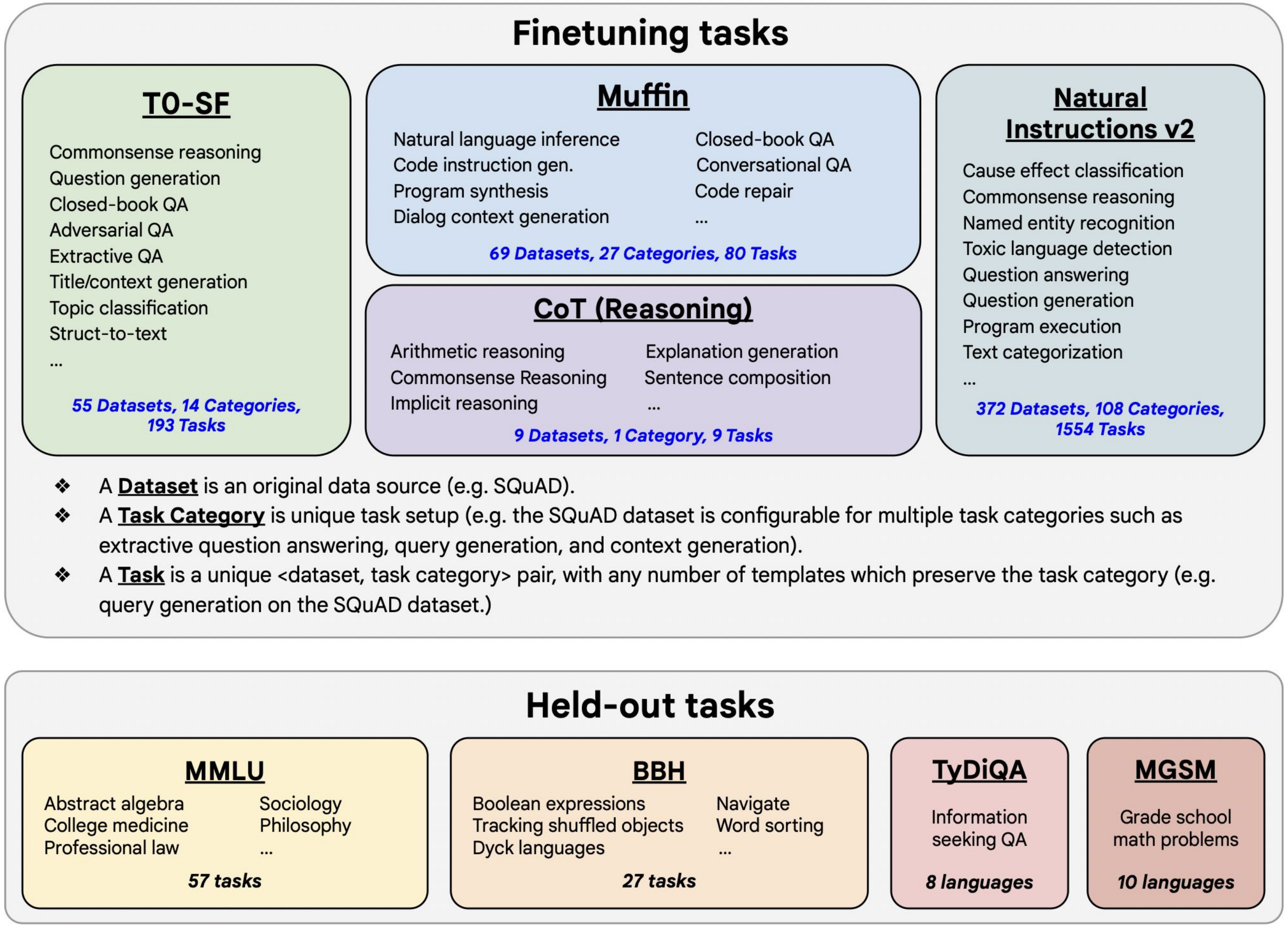
https://arxiv.org/abs/1910.10683

FLAN-PALM/T5: Scaling Instruction-Finetuned Language Models



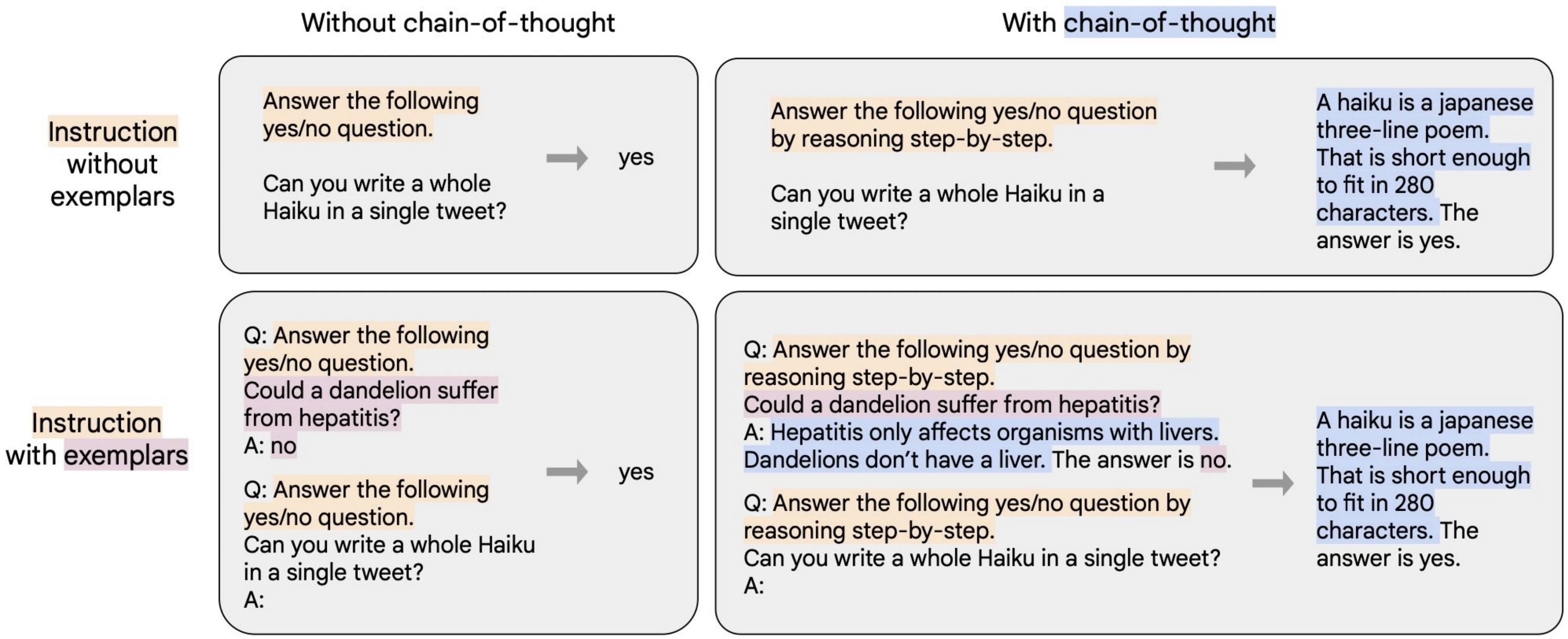
https://arxiv.org/pdf/2210.11416

FLAN-PALM/T5: Scaling Instruction-Finetuned Language Models



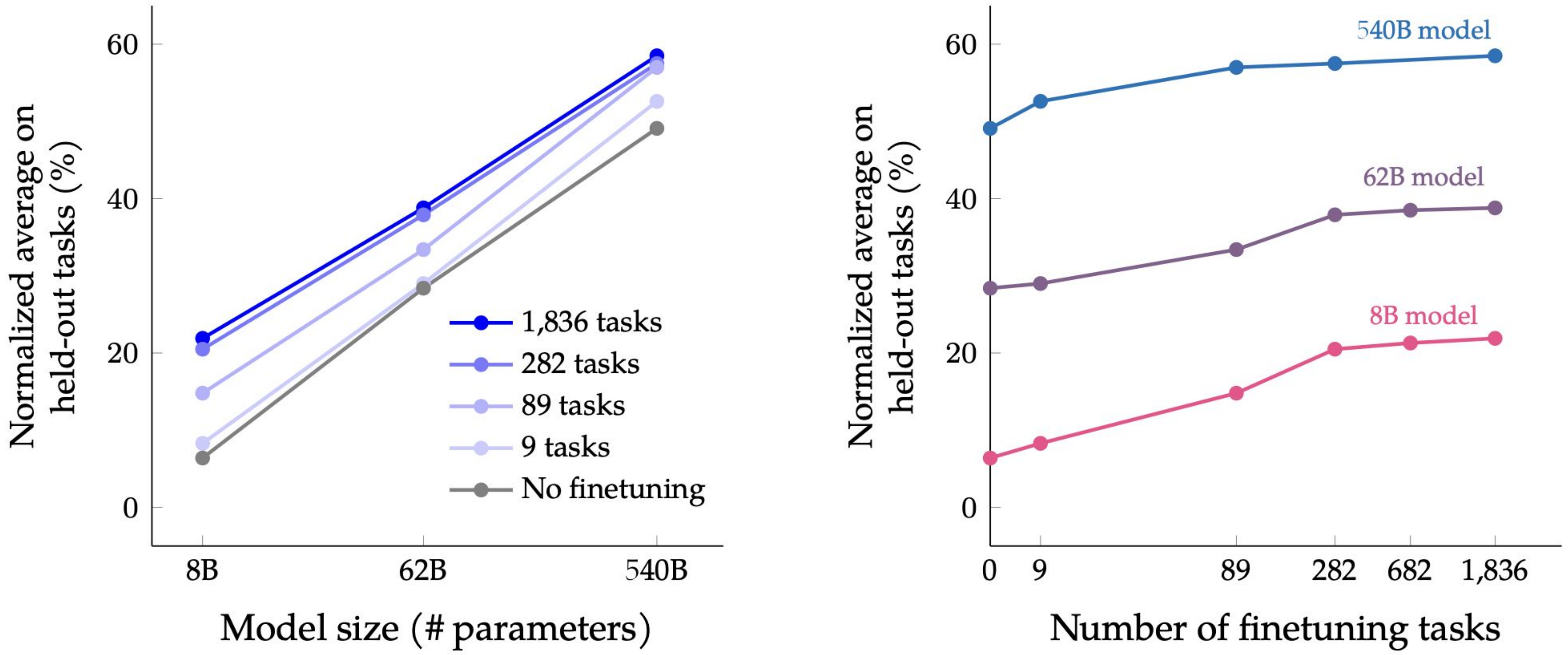
https://arxiv.org/pdf/2210.11416

FLAN-PALM/T5: Scaling Instruction-Finetuned Language Models



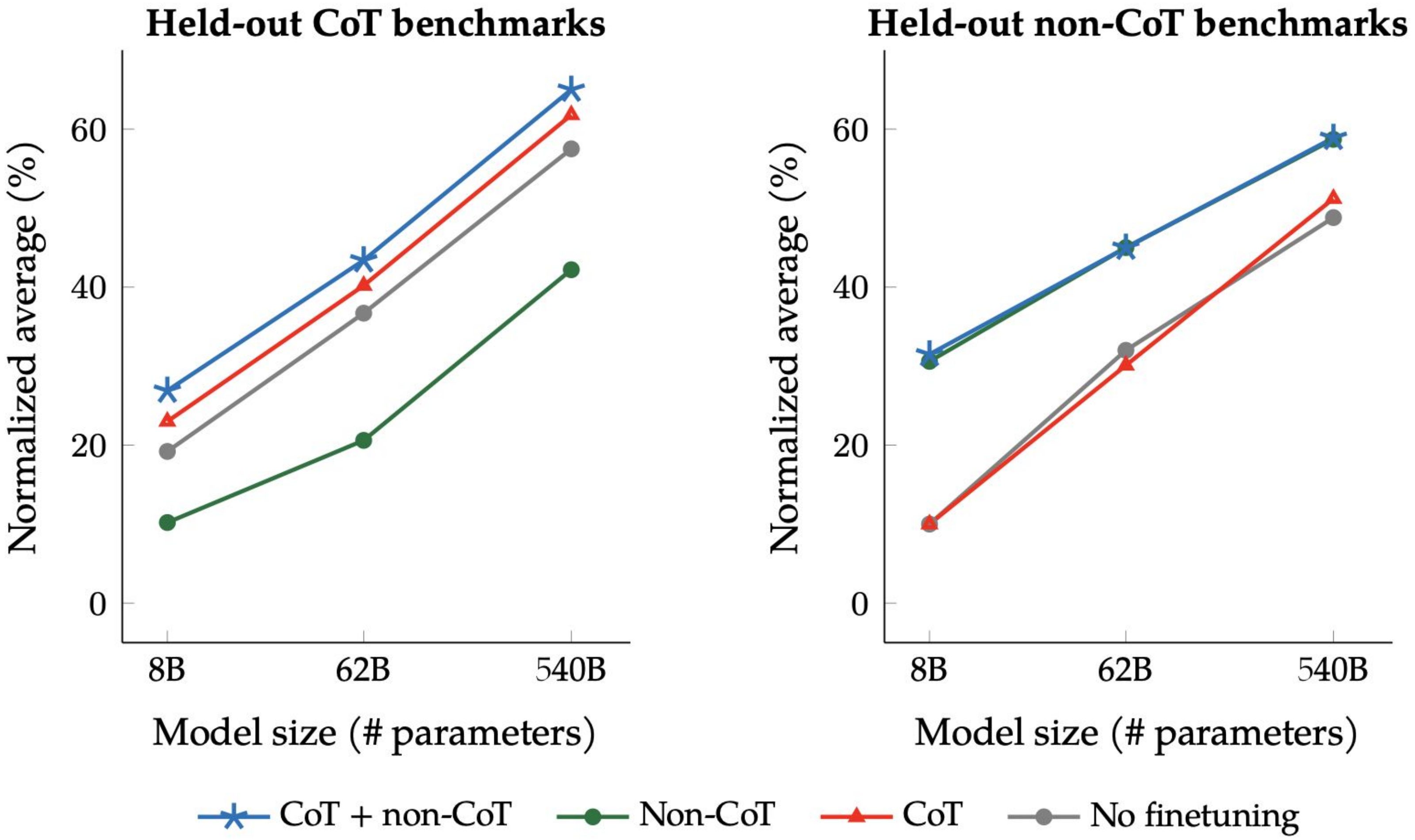
https://arxiv.org/pdf/2210.11416

FLAN-PALM/T5: Scaling Instruction-Finetuned Language Models



https://arxiv.org/pdf/2210.11416

FLAN-PALM/T5: Scaling Instruction-Finetuned Language Models



https://arxiv.org/pdf/2210.11416

FLAN-PALM/T5: Scaling Instruction-Finetuned Language Models

Model input (Boolean Expressions)

Q: (False or not False or False) is
A: Let's think step by step.

PaLM 540B output

(False or not False or False)
(False or False or False)
(False or False)
(False) ✖ **(wrong answer)**

Flan-PaLM 540B output

False or not False is True. True or False is True.
The answer: True. ✔

Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:
(A) They will discuss the reporter's favorite dishes
(B) They will discuss the chef's favorite dishes
(C) Ambiguous

A: Let's think step by step.

PaLM 540B output

The reporter and the chef will discuss their favorite dishes.
The reporter and the chef will discuss the reporter's favorite dishes.
The reporter and the chef will discuss the chef's favorite dishes.
The reporter and the chef will discuss the reporter's and the chef's favorite dishes.

✖ **(doesn't answer question)**

Flan-PaLM 540B output

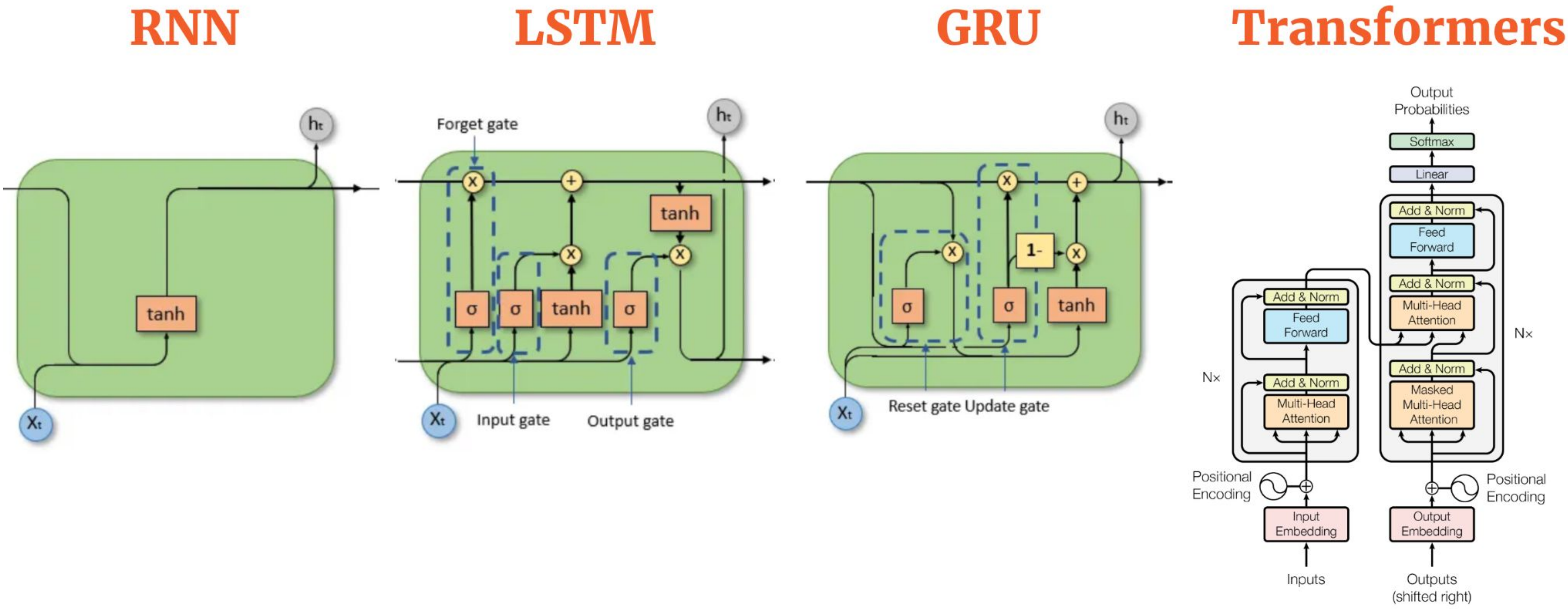
The reporter and the chef will discuss their favorite dishes does not indicate whose favorite dishes they will discuss. So, the answer is (C). ✔

https://arxiv.org/pdf/2210.11416

RWKV – Reinventing RNNs for the Transformer Era

https://arxiv.org/abs/2305.13048

RNN vs Transformer



RWKV

R: Receptance

W: Weight

K: Key

V: Value

Combines the efficient parallelizable training of **Transformers** with the efficient inference of **RNNs**

RWKV

- alleviate the memory bottleneck and quadratic scaling
- reformulate the attention mechanism in RWKV to introduce a variant of linear attention
- preserve the rich expressive properties that make a Transformer a dominant architecture
- ability to offer parallelized training
- uses WKV operator instead of attention

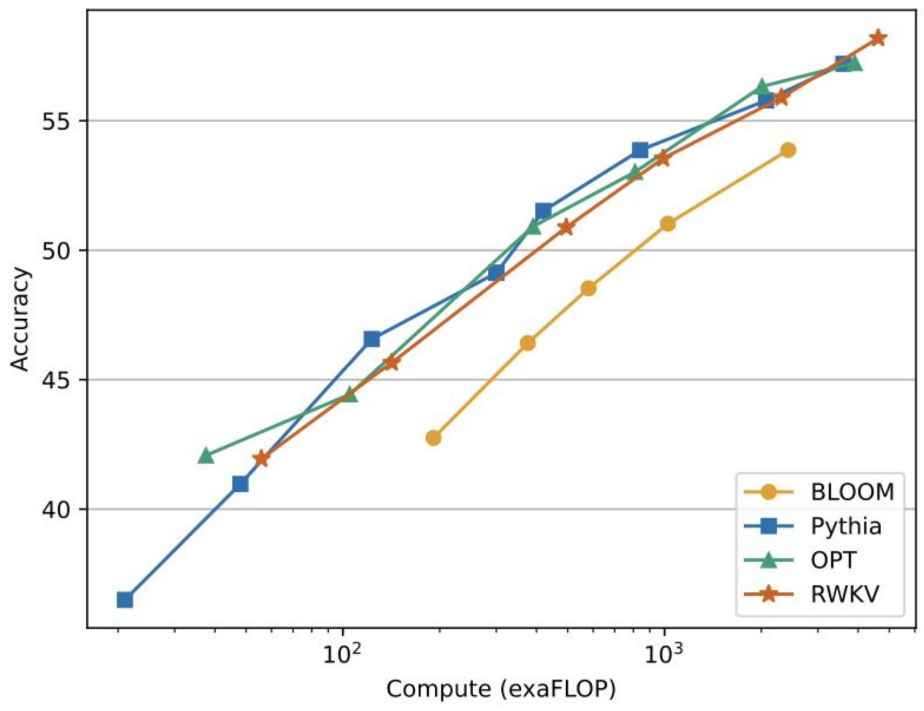


Figure 1: Average performance of RWKV models compared to transformers across twelve NLP tasks. For further details, see section 5.

Token shift & States shift

Linearly aggregate the past

No non-linearity in state

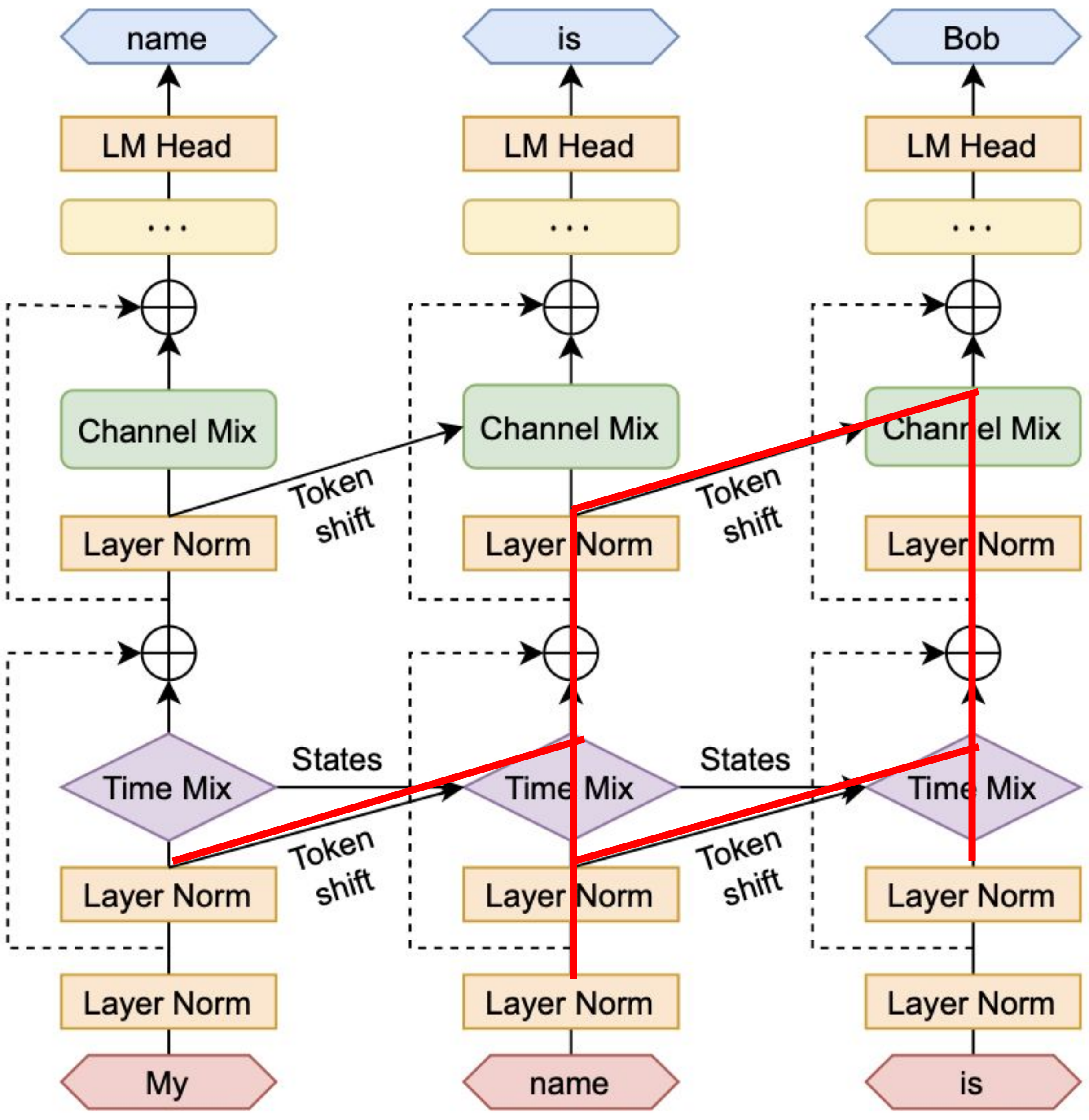


Figure 3: RWKV architecture for language modeling.

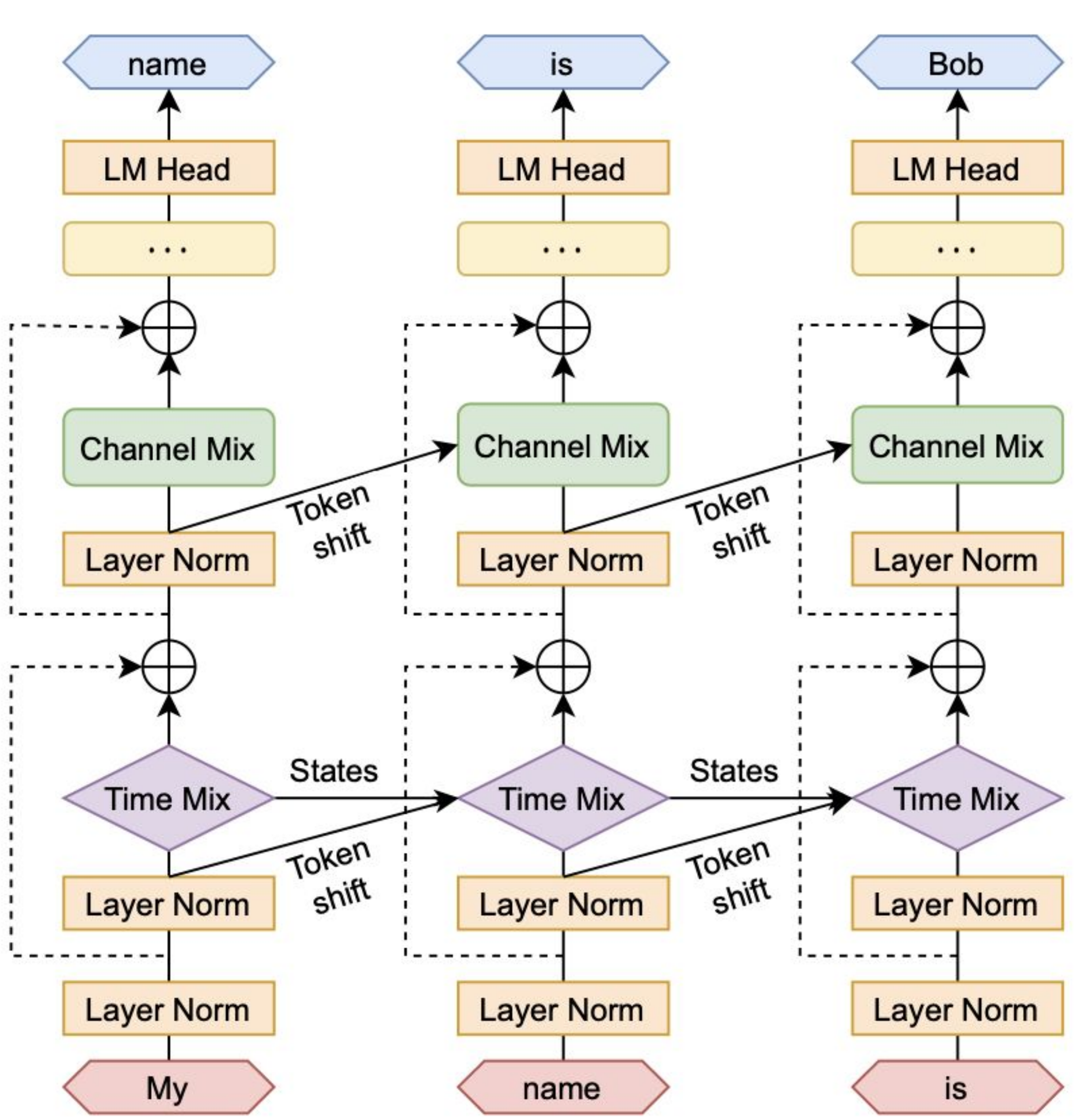


Figure 3: RWKV architecture for language modeling.

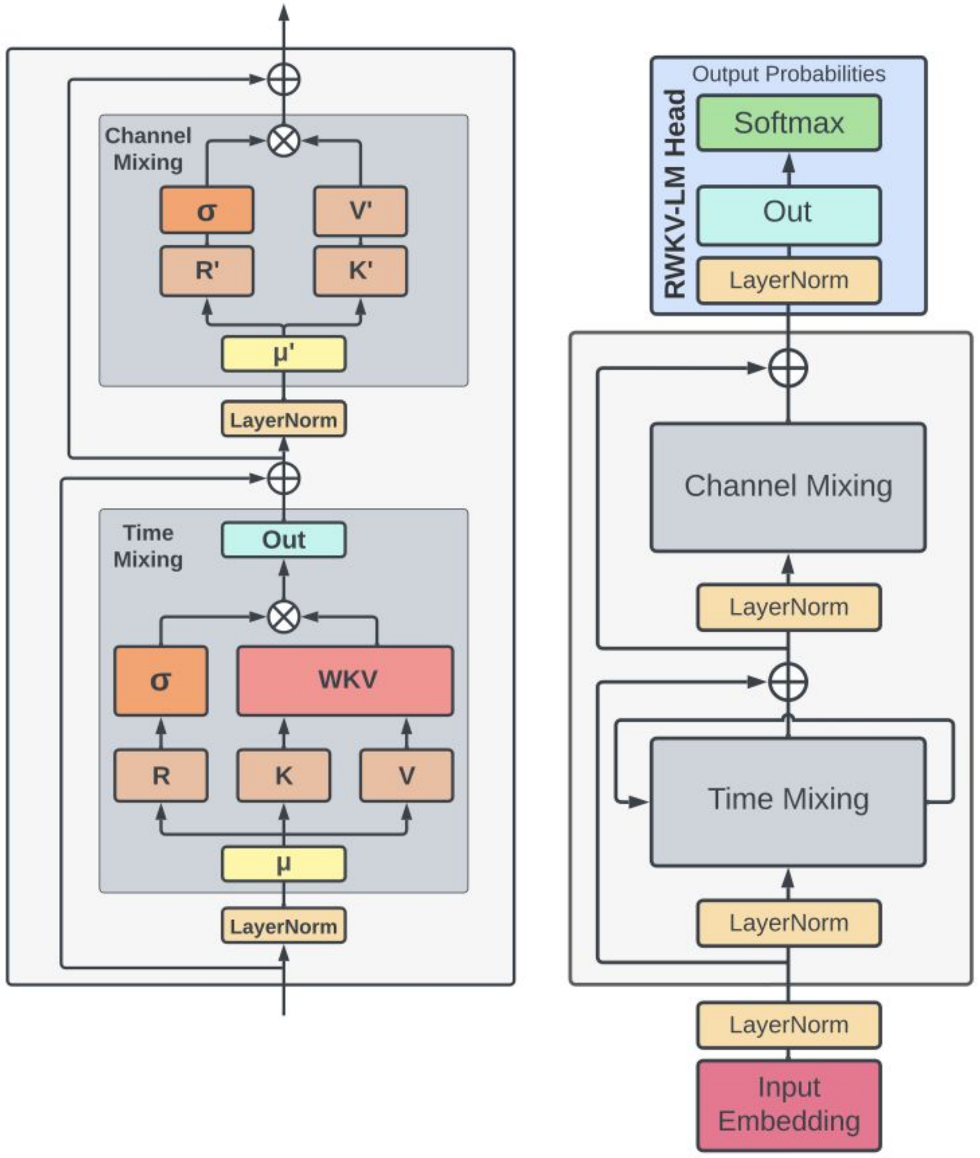
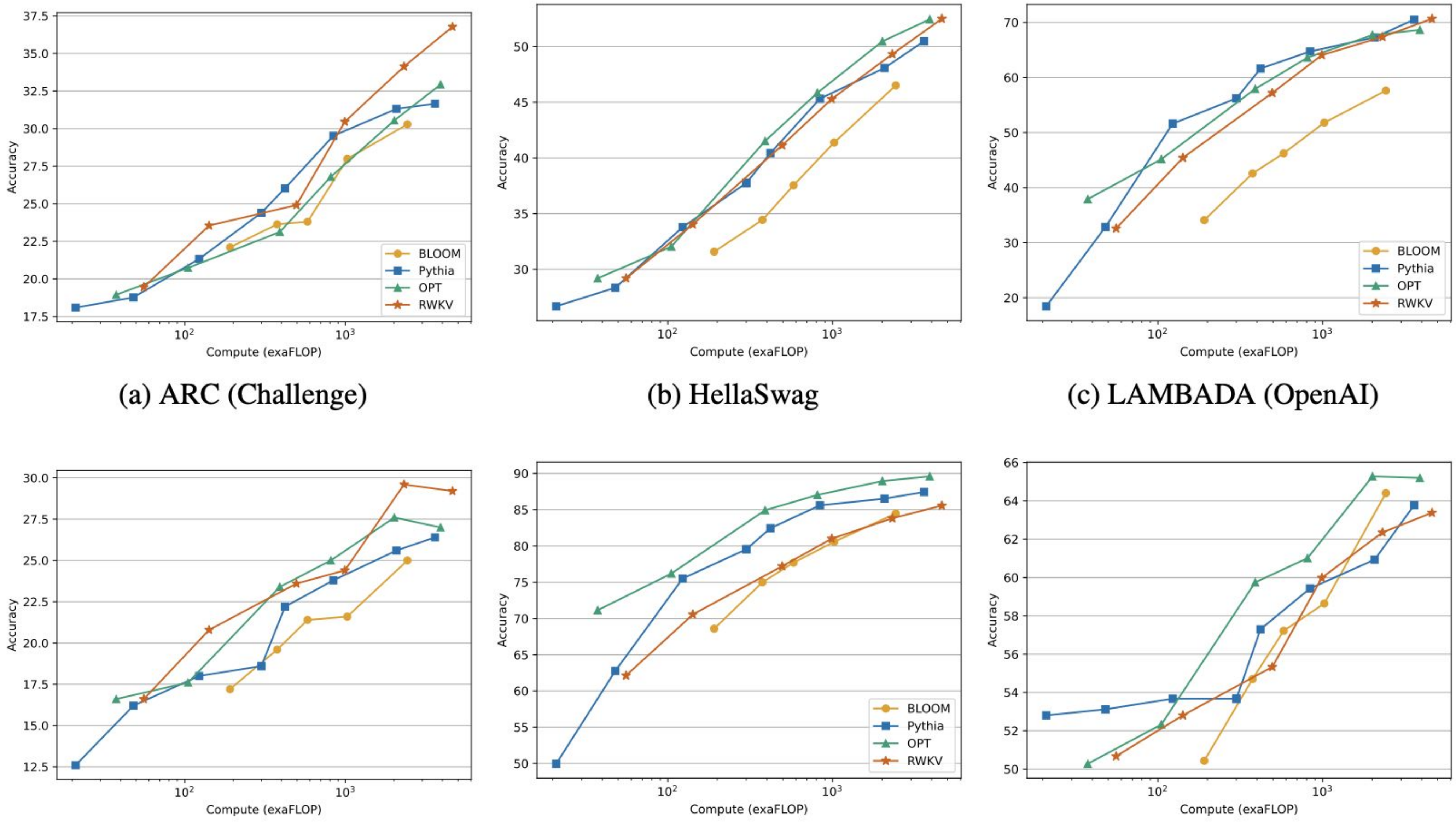


Figure 2: Elements within an RWKV block (left) and the complete RWKV residual block, equipped with a final head for language modeling (right).



(a) ARC (Challenge) (b) HellaSwag (c) LAMBADA (OpenAI)

(d) OpenBookQA (e) ReCoRD (f) Winogrande

Figure 5: Zero-Shot Performance of RWKV on common language modeling evaluation benchmarks. Additional plots can be found in Appendix J.

More results

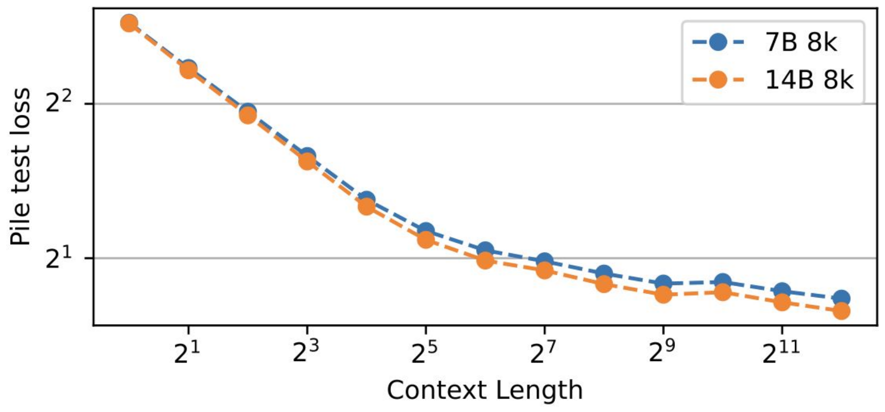


Figure 6: RWKV shows decreasing mean test loss as a function of context length on the Pile (Gao et al., 2020)

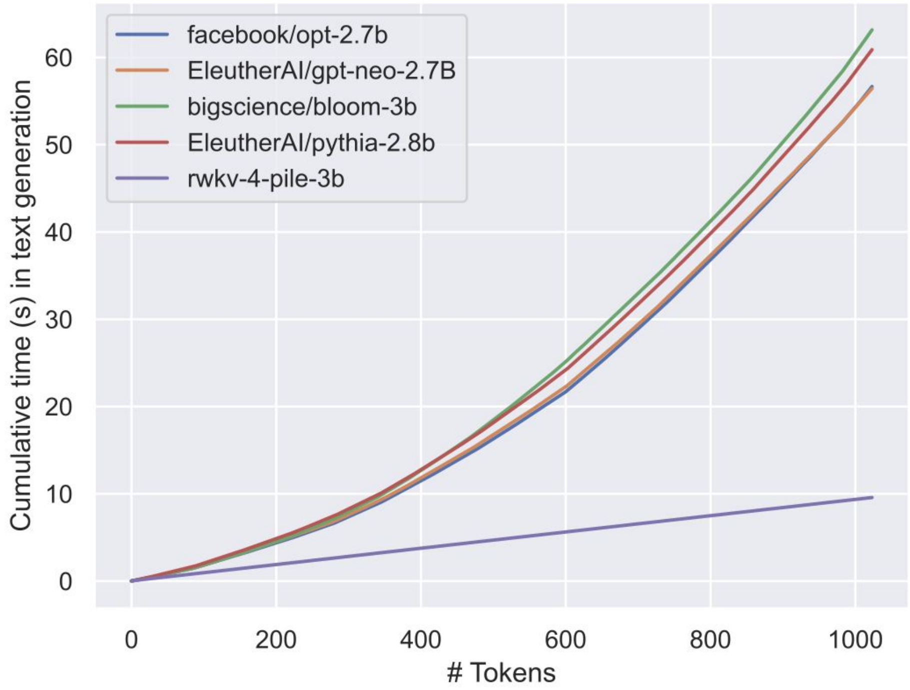


Figure 7: Cumulative time on text generation for LLMs. Unlike transformers, RWKV exhibits linear scaling.

Latest results

RWKV-7 World-v3 2.9B MMLU 54.56% (organic leap from RWKV-6 World-v2.1 3.1B MMLU 32.38%, no eval-maxxing, no HQ-annealing, no post-training)

lm-evaluation-harness	params	English	LAMBADA	PIQA	StoryCloze16	Hellaswag	WinoGrande	arc_challeng	arc_easy	headQA_en	openbookQA	sciq	ReCoRD	MultiLang	xLBD	xSC	xWG	xCOPA
model	B	avg%	acc	acc	acc	acc_norm	acc	acc_norm	acc	acc_norm	acc_norm	acc	em	avg%	acc	acc	acc	acc
RWKV-7 "Goose" World-v3	2.90	71.1%	73.0%	79.6%	79.4%	76.5%	72.4%	50.5%	79.5%	42.3%	46.6%	94.7%	87.7%	62.3%	47.2%	63.3%	76.0%	62.8%
Llama-3.2-3B (+emb = 3.61b)	3.61	68.7%	70.2%	76.5%	78.0%	73.6%	69.5%	46.3%	74.2%	39.8%	42.8%	95.4%	89.6%	57.3%	39.6%	59.2%	71.8%	58.6%
Qwen2.5-3B (+emb = 3.40b)	3.40	68.6%	66.9%	78.4%	77.2%	73.6%	68.1%	47.2%	77.4%	40.7%	43.0%	95.3%	87.1%	57.0%	37.8%	58.0%	73.2%	58.9%
stablelm-3b-4e1t	2.80	67.5%	70.6%	79.6%	76.9%	73.8%	66.5%	39.9%	72.3%	39.0%	40.4%	94.6%	88.9%	54.6%	39.2%	56.1%	66.7%	56.4%
RWKV-6 "Finch" World-v2.1	3.10	66.0%	71.4%	75.8%	78.6%	68.2%	65.4%	39.0%	71.1%	36.6%	40.6%	92.9%	86.3%	59.2%	44.6%	59.9%	71.8%	60.6%

RWKV-7 World-v3 1.5B MMLU 44.84% (organic leap from RWKV-6 World-v2.1 1.6B MMLU 26.34%, no eval-maxxing, no HQ-annealing, no post-training)

lm-evaluation-harness	params	English	LAMBADA	PIQA	StoryCloze16	Hellaswag	WinoGrande	arc_challeng	arc_easy	headQA_en	openbookQA	sciq	ReCoRD	MultiLang	xLBD	xSC	xWG	xCOPA
model	B	avg%	acc	acc	acc	acc_norm	acc	acc_norm	acc	acc_norm	acc_norm	acc	em	avg%	acc	acc	acc	acc
RWKV-7 “Goose” World-v3	1.52	67.7%	69.4%	77.2%	76.3%	70.8%	67.9%	42.5%	75.8%	39.0%	46.4%	94.4%	85.3%	58.6%	43.0%	59.7%	72.2%	59.7%
SmolLM2-1.7B (+emb = 1.81b)	1.81	67.9%	67.4%	76.7%	76.0%	71.5%	66.7%	47.0%	78.0%	39.8%	44.6%	93.4%	86.3%	49.2%	29.3%	52.0%	62.5%	53.1%
Qwen2.5-1.5B (+emb = 1.78b)	1.78	65.5%	62.2%	75.8%	73.8%	67.9%	63.5%	45.3%	75.4%	37.6%	40.2%	94.6%	83.9%	53.8%	34.1%	55.9%	68.1%	57.1%
stablelm-2-1_6b	1.64	65.1%	65.8%	76.3%	75.8%	68.9%	64.3%	39.9%	68.6%	34.3%	39.8%	95.3%	86.8%	52.3%	34.8%	53.8%	64.4%	56.3%
RWKV-6 "Finch" World-v2.1	1.60	62.2%	67.3%	73.9%	74.9%	61.1%	60.5%	34.2%	64.2%	35.7%	38.6%	90.3%	83.7%	56.2%	40.9%	56.7%	69.2%	58.1%
Llama-3.2-1B (+emb = 1.50b)	1.50	62.1%	62.2%	74.6%	72.4%	63.6%	60.5%	36.2%	65.7%	34.0%	37.4%	90.9%	85.9%	51.8%	33.0%	55.2%	63.7%	55.4%

lm-evaluation-harness	params	English	LAMBADA	PIQA	StoryCloze16	Hellaswag	WinoGrande	arc_challeng	arc_easy	headQA_en	openbookQA	sciq	ReCoRD	MultiLang	xLBD	xSC	xWG	xCOPA
model	B	avg%	acc	acc	acc	acc_norm	acc	acc_norm	acc	acc_norm	acc_norm	acc	em	avg%	acc	acc	acc	acc
RWKV7-G1 "GooseOne" 20250324	0.450	60.5%	59.3%	72.1%	70.3%	57.4%	59.3%	34.9%	68.3%	34.1%	40.0%	91.1%	78.6%	52.9%	34.5%	55.3%	65.9%	55.7%
SmolLM2-360M (+emb = 407m)	0.407	59.7%	53.4%	71.8%	68.2%	56.4%	59.0%	37.9%	70.2%	34.3%	37.8%	90.8%	77.4%	43.7%	18.7%	49.7%	54.5%	51.8%
RWKV-7 "Goose" Pile	0.421	55.8%	57.9%	69.2%	67.7%	48.1%	56.4%	27.6%	56.2%	32.1%	32.2%	85.9%	80.2%	47.6%	29.4%	50.7%	57.3%	52.9%
mamba-370m (+emb = 421m) Pile	0.421	54.7%	55.6%	69.5%	66.3%	46.5%	55.5%	27.9%	55.0%	32.3%	30.8%	84.9%	77.0%	47.2%	28.5%	50.5%	57.3%	52.4%

lm-evaluation-harness	params	English	LAMBADA	PIQA	StoryCloze16	Hellaswag	WinoGrande	arc_challeng	arc_easy	headQA_en	openbookQA	sciq	ReCoRD	MultiLang	xLBD	xSC	xWG	xCOPA
model	B	avg%	acc	acc	acc	acc_norm	acc	acc_norm	acc	acc_norm	acc_norm	acc	em	avg%	acc	acc	acc	acc
RWKV7-G1 "GooseOne" 20250307	0.191	52.8%	49.1%	68.0%	64.3%	42.9%	53.8%	28.2%	56.1%	30.3%	32.4%	85.8%	70.5%	47.6%	27.1%	51.9%	58.3%	53.4%
SmolLM2-135M (+emb = 163m)	0.163	53.0%	42.9%	68.2%	63.4%	43.2%	53.1%	29.7%	64.3%	31.1%	33.0%	83.8%	70.1%	41.7%	12.5%	49.4%	52.8%	52.0%
RWKV-7 "Goose" Pile	0.168	49.8%	45.6%	65.5%	61.8%	36.9%	52.3%	24.7%	47.9%	29.1%	30.0%	81.8%	71.9%	43.9%	21.7%	49.3%	52.0%	52.8%
mamba-130m (+emb = 168m) Pile	0.168	48.5%	44.2%	64.4%	60.4%	35.3%	52.4%	24.3%	48.1%	28.8%	28.6%	78.1%	68.9%	43.7%	20.1%	49.2%	53.2%	52.4%

Conclusions

- Transformers can be applied very naturally to many seq2seq tasks
  - However, their evaluation is a problem
- There are some good pretrained seq2seq transformers
- However, nowadays mostly Decoders are used
- New alternatives are coming, e.g. RWKV