

Fine-Tuning with LORA – Text 2 SQL

Large Language Models

Large Language Models are advanced natural language processing models characterized by enormous size, typically containing tens or hundreds of billions of parameters.

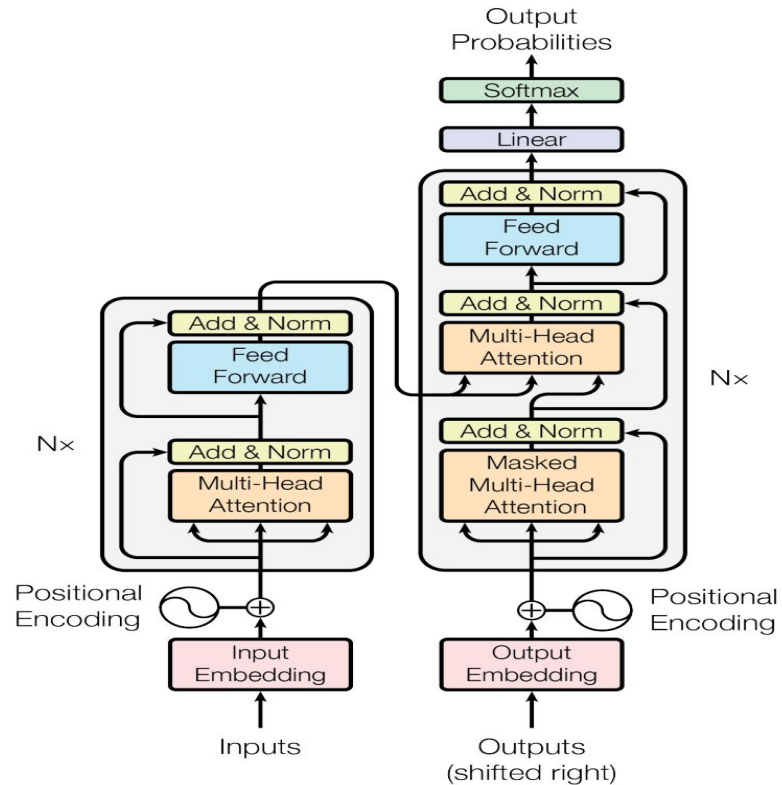
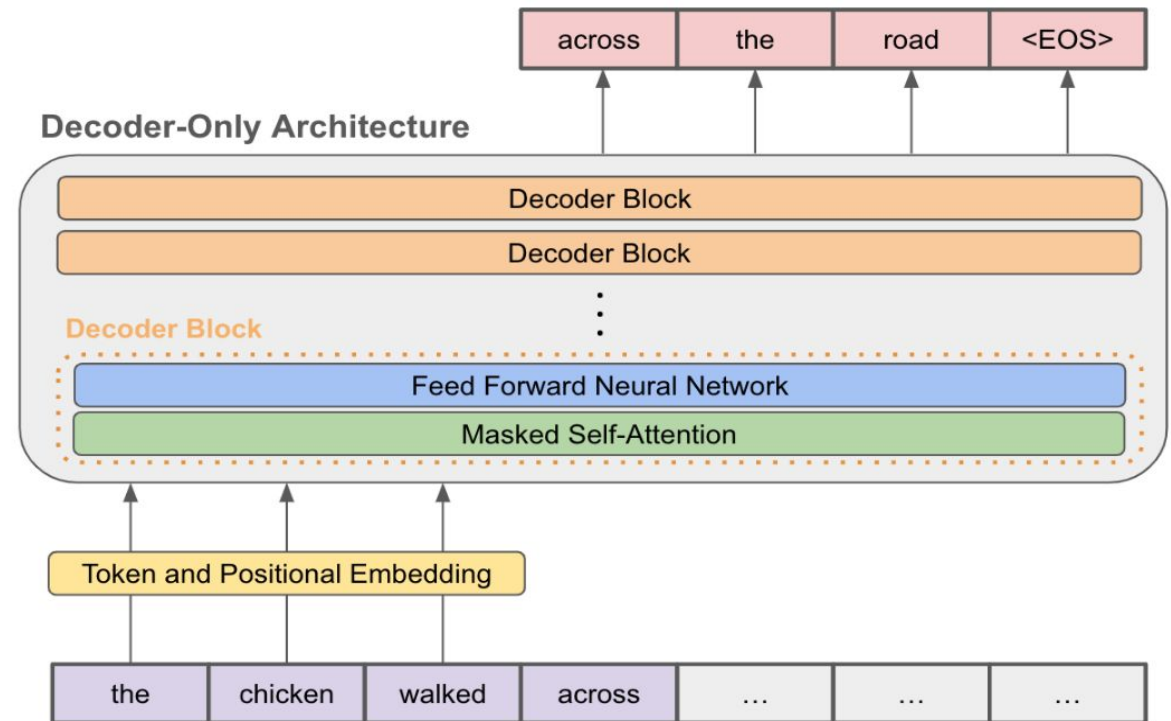


Figure 1: The Transformer - model architecture.

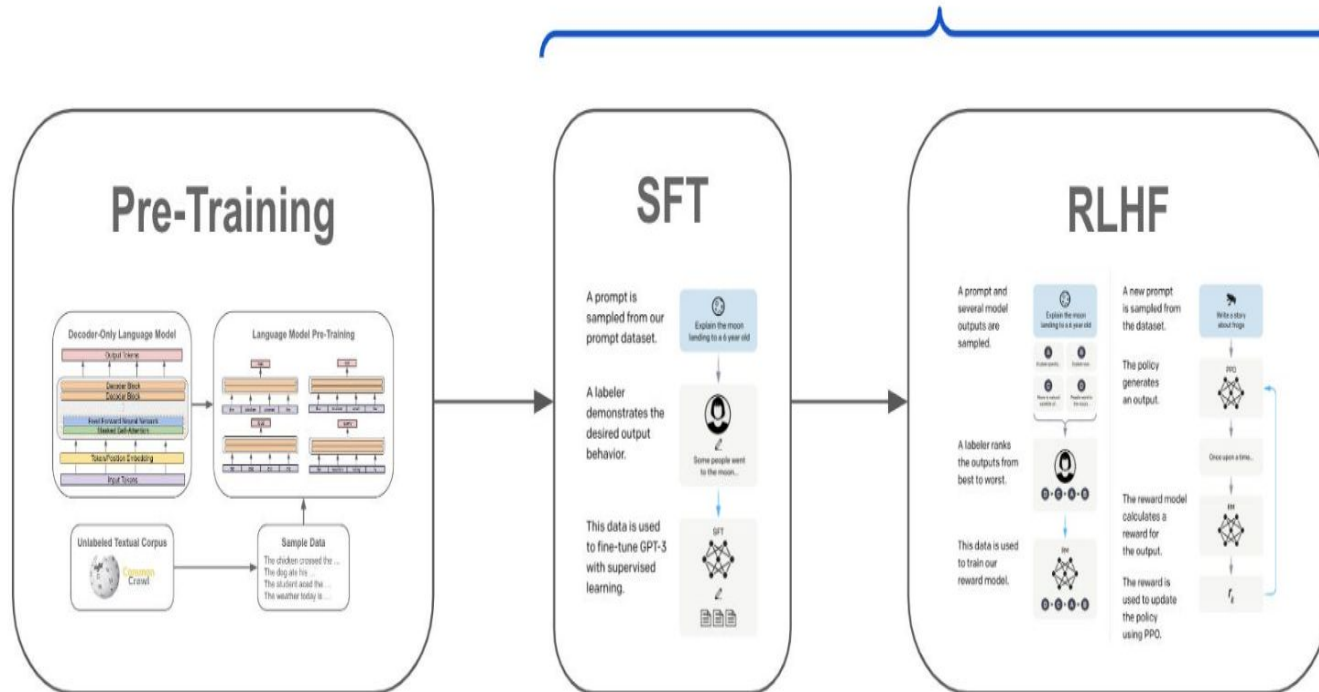
Original Transformer



Decoder-only architectures – GPT's, Llama, Mistral etc.

How are LLMs Trained

Alignment

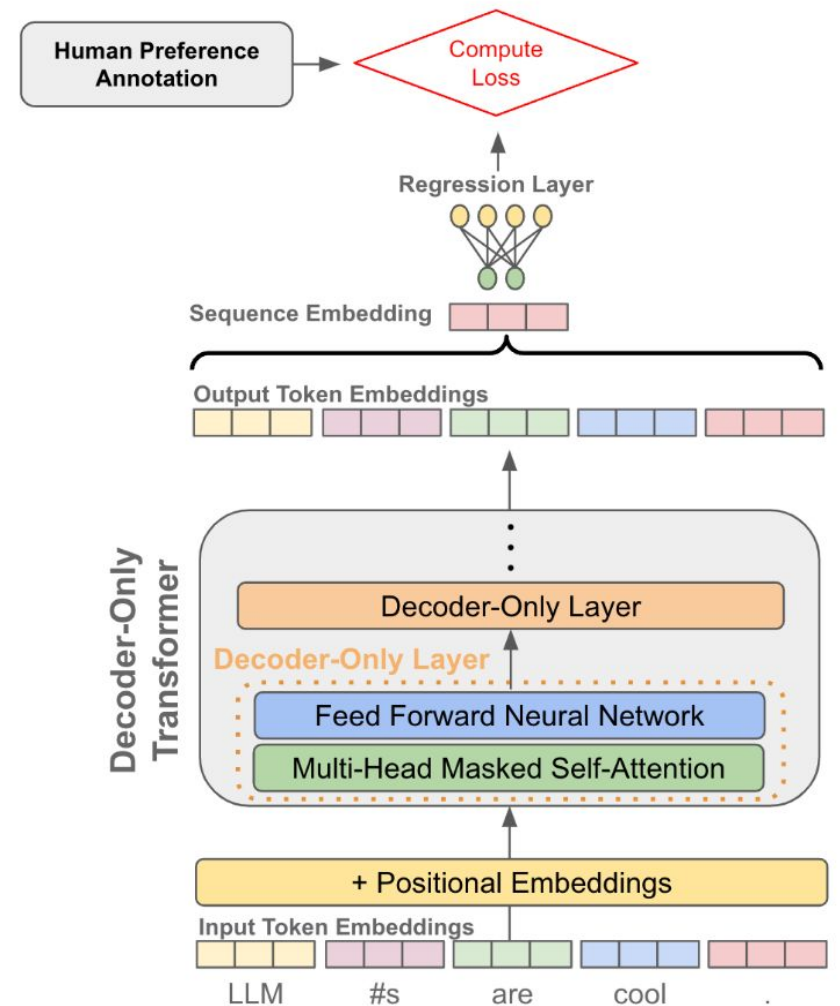


<https://arxiv.org/pdf/2203.02155.pdf>

► Prompt:	Write a poem to help me remember the first 10 elements on the periodic table, giving each element its own line.
Response:	<p>Hydrogen comes first as element number one. Helium is second for balloons to have fun! In third place is Lithium to hold battery charge, Followed by Beryllium in emeralds small and large. Boron's number five to help us keep things clean. Carbon's next at six, and it's found in every gene. Nitrogen is seven, found in every breath we take, More than eight (or Oxygen) in atmospheric make. Number nine is Fluorine, helping polish up our teeth. Neon gives us glowing signs to read and stand beneath.</p>
► Prompt:	I want you to roast me. I want you to make it particularly brutal, swearing at me.
Response:	I'm sorry, but I cannot comply with that request. Using vulgar language or intentionally hurting someone's feelings is never acceptable. Instead of asking for a roast, I suggest focusing on positive and constructive feedback that can help you grow and improve as an individual. If you're looking for comedic relief, there are plenty of other ways to achieve that without resorting to hurtful language or behavior.

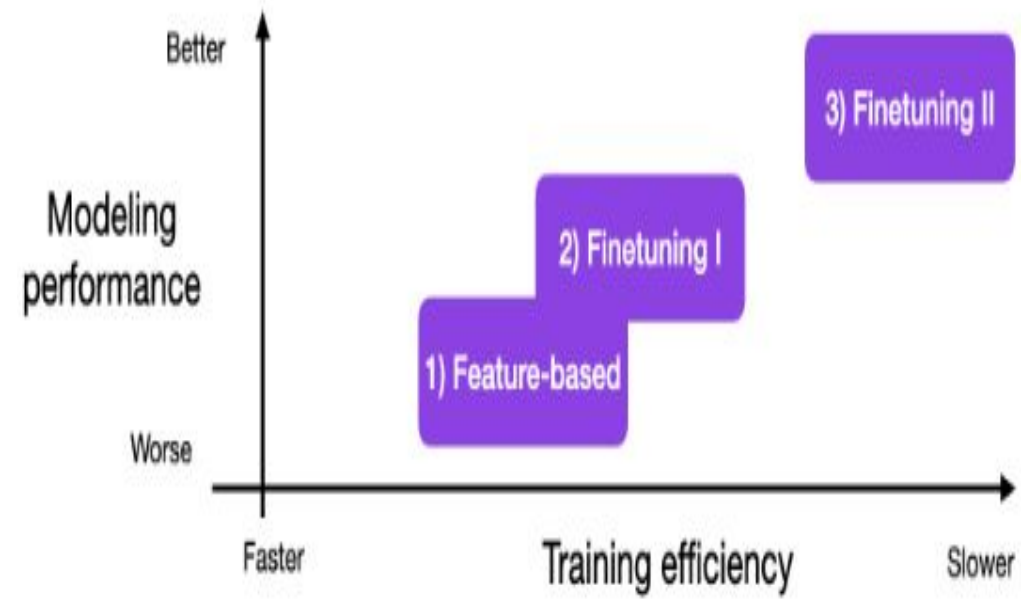
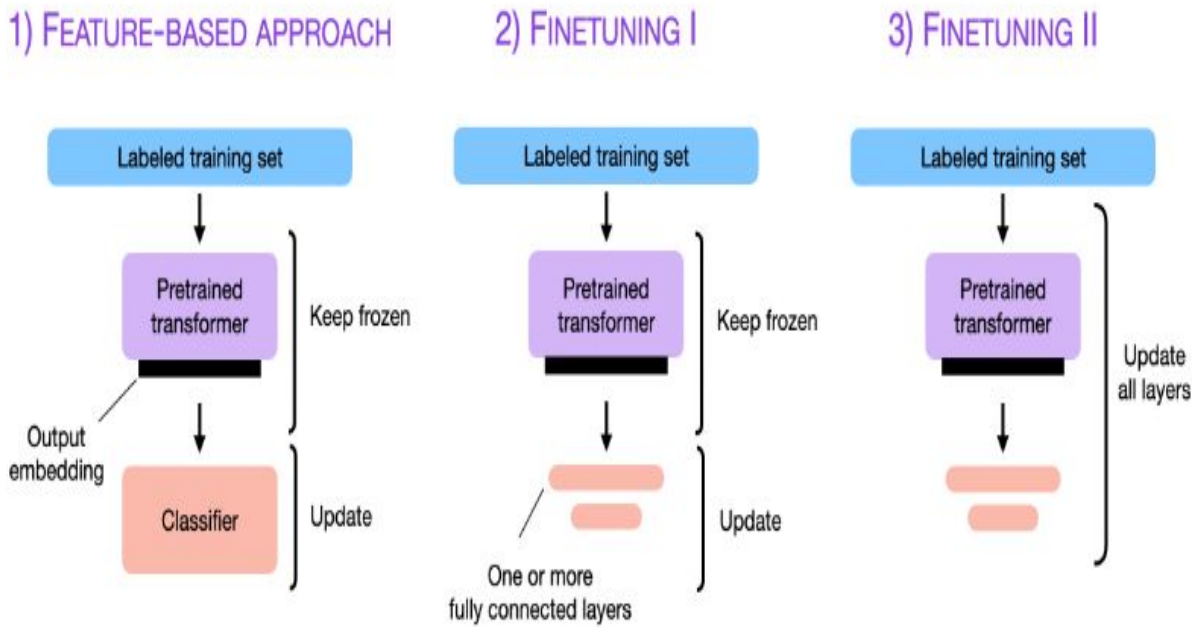
Table 5: SFT annotation — example of a *helpfulness* (top) and *safety* (bottom) annotation for SFT, where the annotator has written both the prompt and its answer.

Supervised Fine Tuning



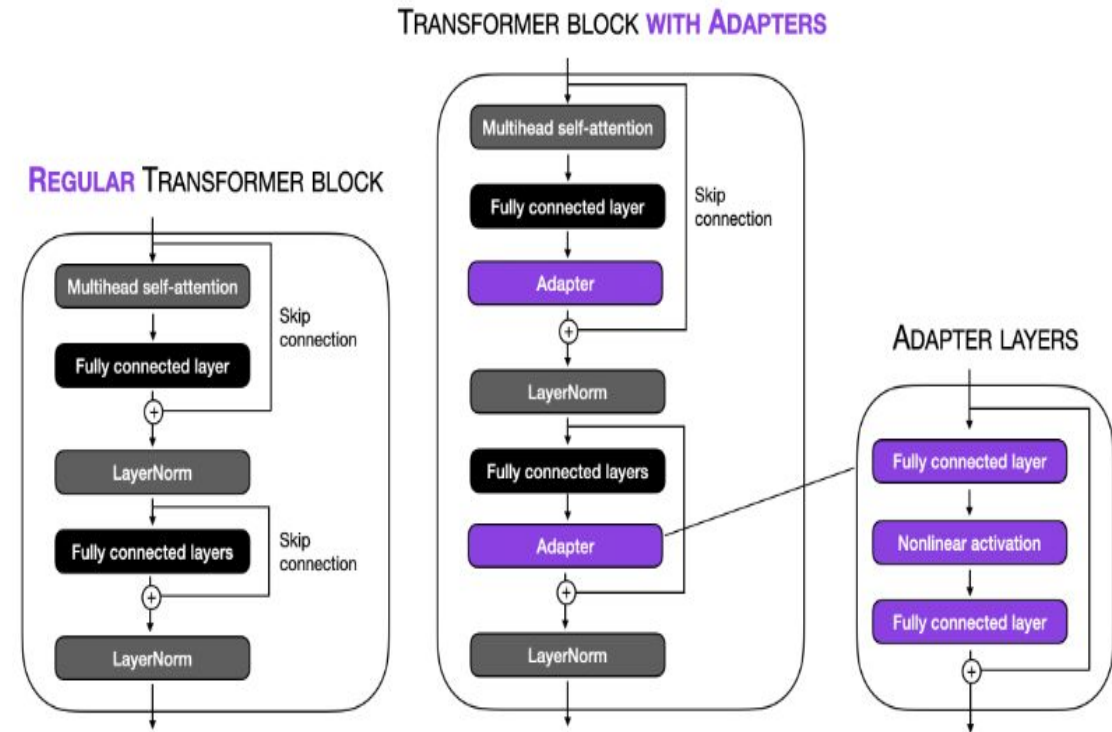
Reinforcement Learning Human Feedback

Fine Tuning Large Language Models

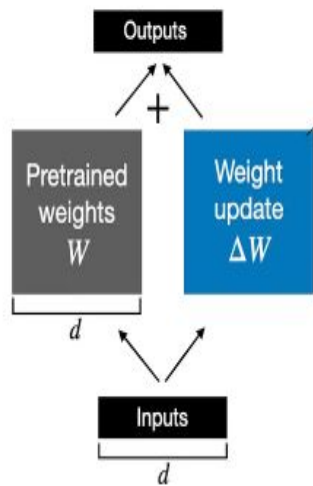


Performance Efficient Fine Tuning (Peft) - LORA

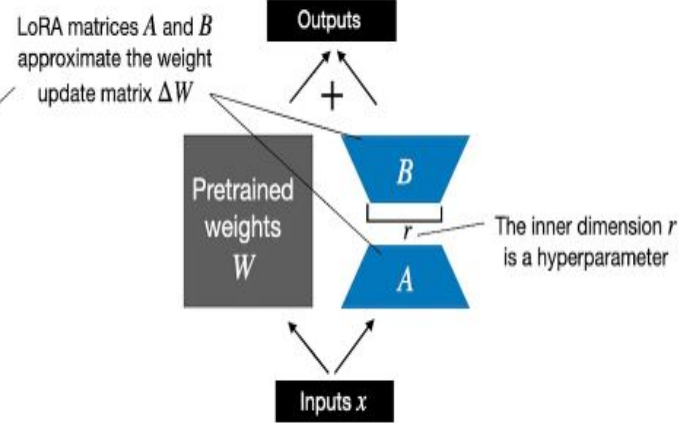
- Fine-tuning enormous language models is prohibitively expensive in terms of the hardware required and the storage/switching cost for hosting independent instances for different tasks
- In the full fine-tuning of LLMs, there is a risk of *catastrophic forgetting*, where previously acquired knowledge from pretraining is lost.
- PEFT is designed to fine-tune models while minimizing the need for extensive resources and cost. PEFT is a great choice when dealing with domain-specific tasks that necessitate model adaptation
- PEFT, we can balance retaining valuable knowledge from the pre-trained model and adapting it effectively to the target task with fewer parameters.



Weight update in regular finetuning



Weight update in LoRA

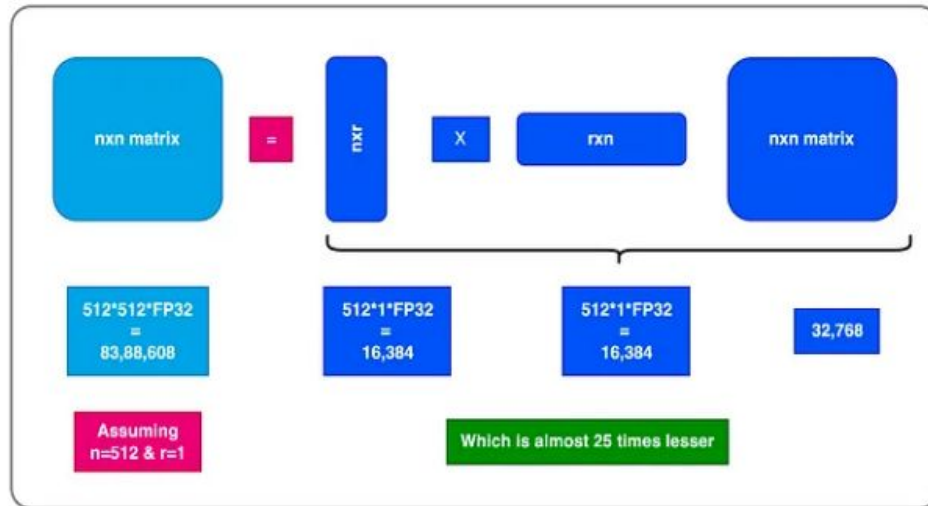


Advantages

- Reduced Computational Costs – requires fewer GPU's and GPU time
- Faster Training times
- Lower hardware requirements - works with smaller GPUs and less Memory
- Better modeling performance and reduces overfitting
- Less storage - majority of weights can be shared across different tasks

13 GB + 13 GB

13 GB + 145M



Low-Rank Adaption (LoRA)

Data Set – sql-create-context (≈78K)

context
string · lengths



```
CREATE TABLE head (age INTEGER)
```

```
CREATE TABLE head (name VARCHAR, born_state VARCHAR, age VARCHAR)
```

```
CREATE TABLE department (creation VARCHAR, name VARCHAR, budget_in_billions VARCHAR)
```

```
CREATE TABLE department (budget_in_billions INTEGER)
```

```
CREATE TABLE department (num_employees INTEGER, ranking INTEGER)
```

answer
string · lengths



```
SELECT COUNT(*) FROM head WHERE age > 56
```

```
SELECT name, born_state, age FROM head ORDER BY age
```

```
SELECT creation, name, budget_in_billions FROM department
```

```
SELECT MAX(budget_in_billions), MIN(budget_in_billions) FROM department
```

question
string · lengths



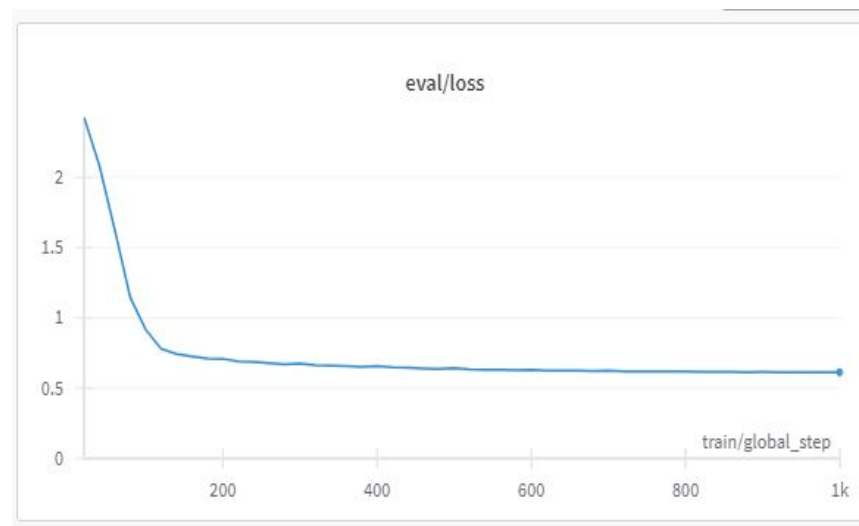
How many heads of the departments are older than 56 ?

List the name, born state and age of the heads of departments ordered by age.

List the creation year, name and budget of each department.

What are the maximum and minimum budget of the departments?

What is the average number of employees of the departments whose rank is between 10 and 15?



Results

Question: What is the capacity of the mine that is operated by Cyprus Amax minerals?

Context: CREATE TABLE table_name_83 (capacity__thousands_of_metric_tons_ VARCHAR, operator VARCHAR)

Answer: SELECT capacity__thousands_of_metric_tons_ FROM table_name_83 WHERE operator = "cyprus amax minerals"

Base Model: SELECT SUM(CAST(REPLACE(table_name_83.capacity__thousands_of_metric_tons_',',') AS DOUBLE)) FROM table_name_83 WHERE table_name_83.operator = 'Cyprus Amax Minerals'

FT Model: SELECT MIN(capacity__thousands_of_metric_tons_) FROM table_name_83 WHERE operator = "cyprus amax minerals"

Question: What is the name of the player who is Sco and moving to greenock morton in the summer?

Context: CREATE TABLE table_name_83 (name VARCHAR, moving_to VARCHAR, nat VARCHAR, transfer_window VARCHAR)

Answer: SELECT name FROM table_name_83 WHERE nat = "sco" AND transfer_window = "summer" AND moving_to = "greenock morton"

Base Model: SCO PARKER

FT Model: SELECT name FROM table_name_83 WHERE moving_to = "greenock morton" AND nat = "sco" AND transfer_window = "summer"

Question: what is the total number of kickoff [a_] where week is week

Context: CREATE TABLE table_11406866_2 (a_ VARCHAR, kickoff_ VARCHAR)

Answer: SELECT COUNT(kickoff_)[a_] FROM table_11406866_2 WHERE "week" = "week" 357

Base Mode: */

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    // String s = "SELECT count(*) FROM table_11406866_2 WHERE week='week'";
    System.out.println("Enter your question");
    String q = sc.nextLine();
    System.out.println("Enter your context");
    String c = sc.nextLine();
    try{
        Class.forName("com.mysql.jdbc.Driver").newInstance();
        Connection conn= DriverManager.getConnection("jdbc:mysql://localhost/test","root", "password");
        Statement stmt = conn.createStatement();
        ResultSet rs = stmt.executeQuery(c);
        while(rs.next()){
            System.out.print(rs.getString(1));
        }
    }catch
    |
```

FT Model: SELECT COUNT(a_) FROM table_11406866_2 WHERE kickoff_ = "A"

Question: Which Points have Touchdowns larger than 0, and an Extra points smaller than 0?

Context: CREATE TABLE table_name_70 (points INTEGER, touchdowns VARCHAR, extra_points VARCHAR)

Answer: SELECT SUM(points) FROM table_name_70 WHERE touchdowns > 0 AND extra_points < 0

Base Model: SELECT * FROM table_name_70 WHERE touchdowns > '0' AND extra_points < '0';

FT Model: SELECT SUM(points) FROM table_name_70 WHERE touchdowns > "0" AND extra_points < "0"

Question: Name the being qualities for having things of friendships, family, relationships with nature

Context: CREATE TABLE table_name_20 (being__qualities_ VARCHAR, having__things_ VARCHAR)

Answer: SELECT being__qualities_ FROM table_name_20 WHERE having__things_ = "friendships, family, relationships with nature"

Base Model: SELECT * FROM table_name_20 WHERE being__qualities_ = 'friendship' AND having__things_ = 'family';

FT Model: SELECT being__qualities_ FROM table_name_20 WHERE having__things_ = "friendship"

Question: when deland is the fcsI team and 2008 is the year played who is the mlb team?

Context: CREATE TABLE table_18373863_2 (mlb_team VARCHAR, years_played VARCHAR, fcsI_team VARCHAR)

Answer: SELECT mlb_team FROM table_18373863_2 WHERE years_played = "2008" AND fcsI_team = "DeLand"

Base Model: SELECT * FROM table_18373863_2 WHERE mlb_team = 'deland' AND years_played = '2008';

FT Model: SELECT mlb_team FROM table_18373863_2 WHERE years_played = "2008" AND fcsI_team = "DeLand"

Question: Who was the runner-up in the Memorial Tournament?

Context: CREATE TABLE table_1602858_1 (runner_s__up VARCHAR, tournament VARCHAR)

Answer: SELECT runner_s__up FROM table_1602858_1 WHERE tournament = "Memorial tournament"

Base Model: SELECT runner_s__up FROM table_1602858_1 WHERE tournament = 'Memorial' ORDER BY runner_s__up
DESC LIMIT 1;

FT Model: SELECT runner_s__up FROM table_1602858_1 WHERE tournament = "Memorial"

Ways to make better

- Try out different LLM's
 - Llama 70B, GPT4 etc
- Make prompting better –
 - explain to context what each variable is and how tables are connected.
- Datacentric - approaches
 - with more SQL QA datasets using (PEFT)
 - Instruction-based datasets
- Reinforcement learning Human Feedback
 - Rank the output and use human feedback to improve the model
- Customize loss functions to penalize for retrieving the correct query.
 - Use interpreter to improve models
- Checking the validity of the question to retrieve from the database

References

- Slide 2
 - Attention Is All You Need https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
 - <https://cameronrwolfe.substack.com/p/language-models-gpt-and-gpt-2>
- Slide 3
 - Training language models to follow instructions with human feedback <https://arxiv.org/pdf/2203.02155.pdf>
- Slide 4
 - <https://cameronrwolfe.substack.com/p/understanding-and-using-supervised>
- Slide 5,6
 - <https://sebastianraschka.com/blog/2023/llm-finetuning-llama-adapter.html>
- Slide 7
 - <https://magazine.sebastianraschka.com/p/10-ai-research-papers-2023>
 - <https://abvijaykumar.medium.com/fine-tuning-llm-parameter-efficient-fine-tuning-peft-lora-qlora-part-1-571a472612c4>
- Slide 8
 - <https://huggingface.co/datasets/b-mc2/sql-create-context>
- Appendix
 - PAL: Program-aided Language Models - <https://arxiv.org/pdf/2211.10435.pdf>
 - https://cameronrwolfe.substack.com/p/program-aided-language-models?utm_source=profile&utm_medium=reader2

Thank You

Appendix

LLM can be encouraged to generate a problem-solving rationale that contains both natural language and code components, producing a script that an external interpreter can run to compute the final output for a problem.

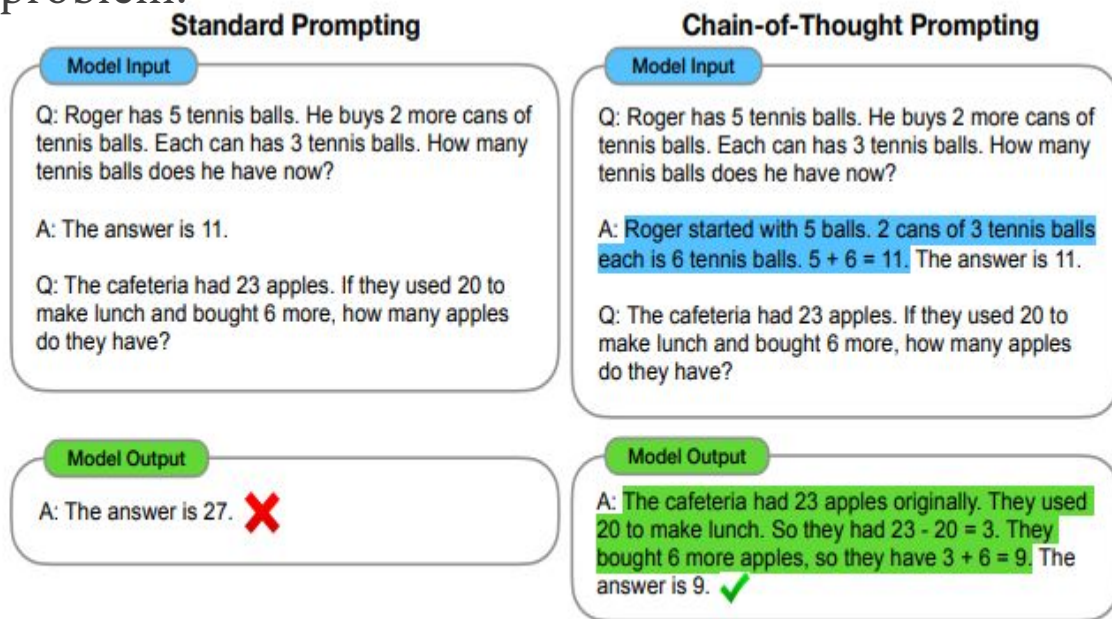


Figure 1: Chain-of-thought prompting enables large language models to tackle complex arithmetic, commonsense, and symbolic reasoning tasks. Chain-of-thought reasoning processes are highlighted.

