A Comparative Look at miniGPT and chatGPT: A Preliminary Investigation into Logical Problem Handling and Formatted Output Generation

Xiao Hong

1 Abstract & Introduction

Based on the previous research, a significant problem utilizing GPT for robot operation is the computational cost associated with GPT-4. Xiaofan deployed the miniGPT-4 in our own server, providing access to a "small" LLM model at virtually no cost.

However, it's important to remember that the computational power and parameter scale harnessed by miniGPT remain significantly limited.

To evaluate miniGPT's capability in processing logical problems and generating formatted outputs, we designed two experiments for a comparative analysis with GPT3.5-turbo and GPT-4.

Relative to its limited computational power and parameters, miniGPT performs very well in terms of cost-effectiveness, but in terms of absolute capability, it lags significantly behind GPT3.5 and GPT4. For specific tasks, it might be possible to enhance miniGPT's capabilities in certain areas through model fine-tuning.

However, if a comprehensive enhancement of miniGPT's capabilities is desired, increasing computational power and parameter scale might still be necessary under current technological conditions.

The findings presented here are preliminary and cursory, the test cases may not be completely reasonable. However, they can serve as a reference to a certain extent.
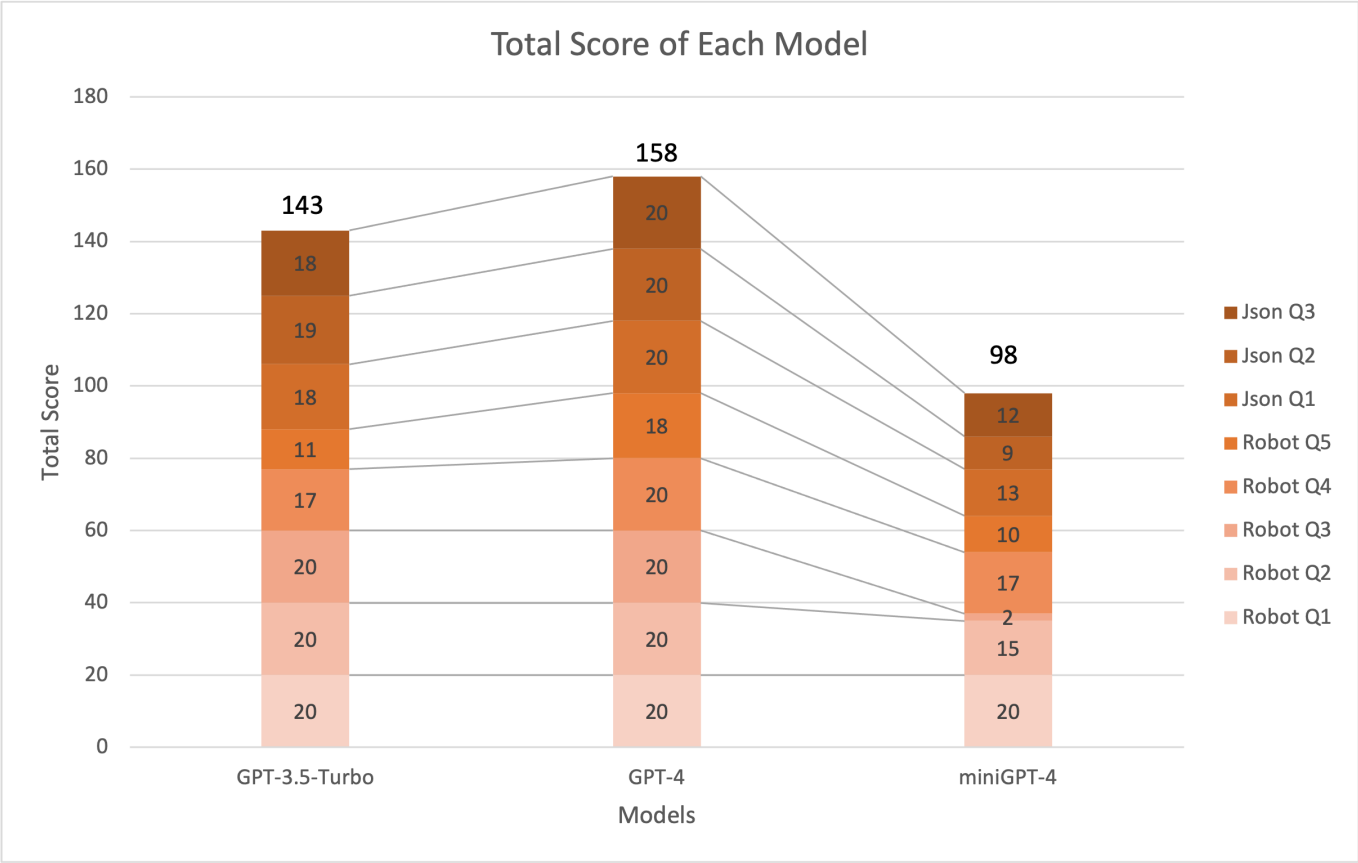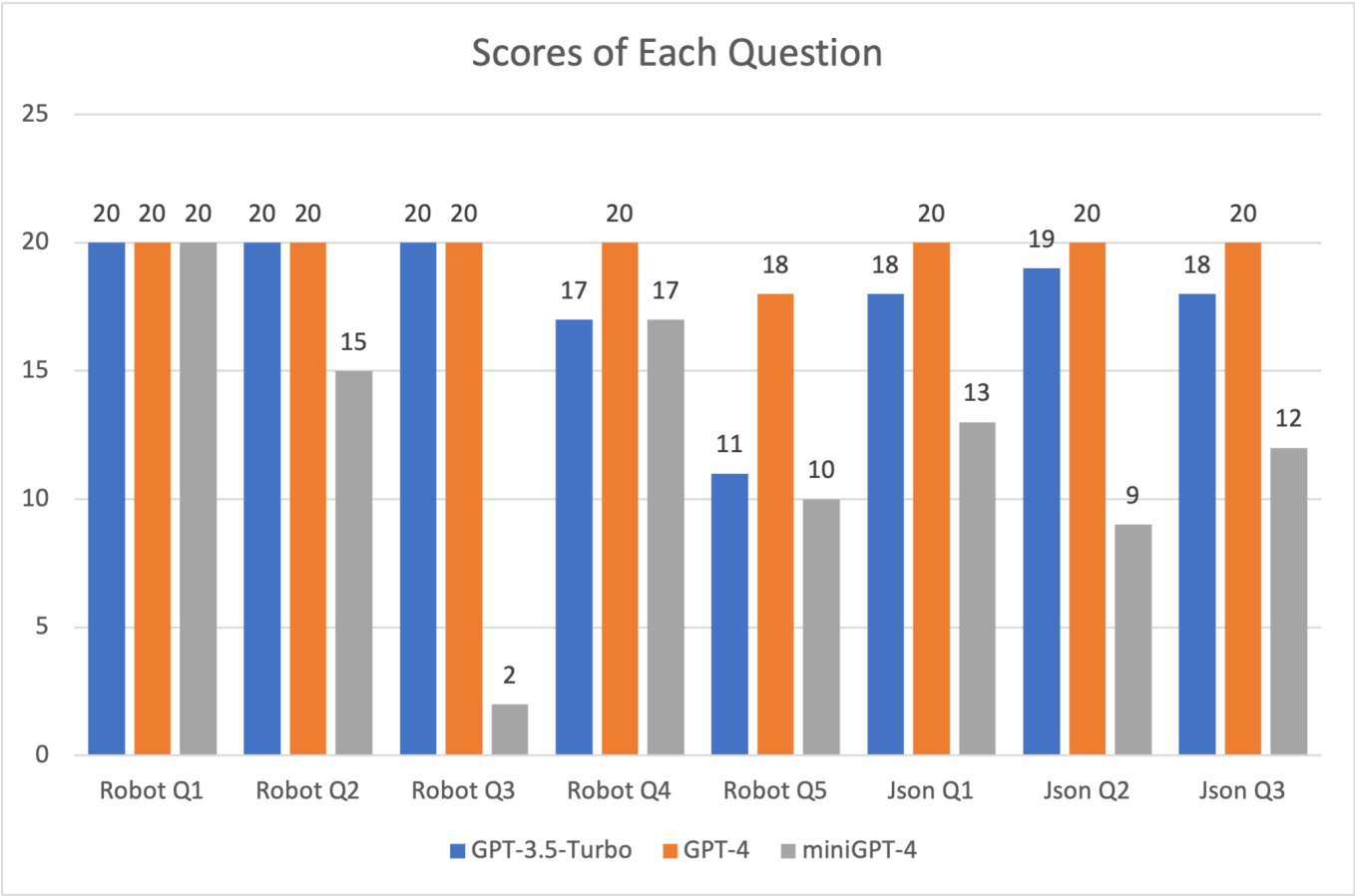
Fig.1 Scores of Each Model

Fig.2 Scores of Each Question

2 Method

We designed two experiments: asking the LLM to be a Robot command interpreter, and asking the LLM to answer questions via JSON format.

For each scenario, we set one system prompts to describe the requirements in detail, and set serval questions based on the requirements. Each question carries a weight of 20 points, culminating in a total of eight questions.

Upon receiving the responses from the LLMs, we assigned scores according to the quality of the answers, in alignment with our predetermined scoring rubric.

3 Test Cases Design

The test cases are some prompts that ask the LLM to give formatted output.

1. Robot Commands （DEMO）

System:

You are a robot command interpreter, responsible for translating the natural language to robot command script.

Here are the robot command script format:

1. Every command has two parts: command name and arguments. Use "::" between the command name and arguments. Use "," to split each argument.

2. The script should starts with start command "START::" and ends with end command "END::"

3. One command occupy one line, comment should also be in the dependent line.

4. Comment line should start with "//", just like single line comment in C and C++.

Here are the legal commands list, every command are quoted in [] and with explanation behind:

[START::] the start command, takes no argument, means the script starts here.

[END::] the end command, takes no argument, means the script ends here.

[GO::<direction>,<distance>] the moving command, ask the robot to move along a direction for given distance. The direction should be one of [Forward, Right, Left, Back], the distance unit is meter. When using this command, the robot will translate (means shift) to the distination, the orientation of the head does not change.

[TAG::] put a tag in current piosition, takes no argument.

[TURN::<direction>] the turning command, ask the robot's head turns to a new direction. The direction should be one of [Right, Left, Back], Considering the current facing direction as forward, to turn back means to turn to the opposite direction of the current one, to turn left means to turn to the left of the current direction, and to turn right means to turn to the right of the current direction.

You need to obey the rules:

1. Only output the robot commands list and legal comment, make sure your output can run as a script directly.

2. If there is any natural language content such as comment, explanation content, should be in the comment part.

3. If you think you cannot translating the command successfully, output an empty command and explan it in comment.

4. If the user didn't mention about "turn to some direction and go forward" or "shift to some direction", use turning and going forward as default instead of shifting.

Here are some input and output examples:

```

[User]:

Please go forward for 2 meters and turn back for 1.5 meters

[Output]:

START::

GO::Forward,2

//Turn back

TURN::Back

GO::Forward,1.5

END::

```

```

[User]:

Please shift right for 2 meters and shift back for 3 meters.

[Output]:

START::

//Go right for 2 meters

GO::Right,2

GO::Back,3

END::

```

User Prompts

1. Go forward for 1 meter, go back for 2 meters.

2. Go forward for 2 meters, turn left and go forward for 0.7 meters, and shift right for 2 meters.

3. Walk a square with 3-meter side length.

4. Turn left and walk 5 meters, then turn back and walk 2 meters, and repeat for 3 times.

5. Try walk a circle-like path, with radius of 3 meters.

2. JSON format

System

You are a JSON formatter AI assistant that helps people to answer questions with JSON format.

You should answer every quesion using JSON.

Here are the rules:

1. Your whole answer should be a JSON object.

2. Your answer should at least contains 2 parts: Answer and Explanation.

3. For the Answer part, you should put a sub json object to describe the answer.

4. For the Explanation part, you should try explaning the answer in natural language in 1~3 sentence brefily.

Some examples:

```

User: Tom has 3 apples and 2 oranges, Jimmy gives him 3 more apples and 2 penciles, how many of each kind fruits does Tom has now?

Assiatant: { "Answer": { "Apple":6, "Orange":2 }, "Explanation":"A pencile is not fruit, so pencile is not calculated in." }

```

```

User: Jack is an engineer, David is a soilder, Lily used to be a marketing manager but now she's a lawyer. what's the jobs now of each person above?

Assiatant: { "Answer": [ { "name":"Jack", "job":"engineer" }, { "name":"David", "job":"soilder" }, { "name":"Lily", "job":"lawyer" } ], "Explanation":"Lily isn't a marketing managet anymore." }

```

User prompts

1. Tom is 18 years old, Shawn is 5 years older than Tom, Tim is 3 years younger than Shawn. How old is each person above?

2. I have a Macbook pro, which values 3000 dollars. I use it for video editing and programming. I also have a iPad, I can do simple video editing work by iPad but cannot do programming work on it. What can I do with the two devices above?

3. Tony has a box, in the box there is a pen, a can of cola, an envelope, inside the envelope there is a card-pocket, inside the card-pocket there is a bank card, a credit card, a passport. There is two photos sandwiched in the passport. Describe what Tony have and the containing relationship.

4 Experiment Environment

We use GPT3.5-turbo, GPT-4 and miniGPT-4 deployed by Xiaofan to run the test cases.

The Model Tempature was set at 0.2 . For GPT3.5-turbo and GPT4, the top-p was set at 0.95 . For miniGPT, the top-p cannot be adjusted.

Given that we need to use the "system" prompt to set the preset instruction, and the chatGPT website version doesn't support the insertion the "system" prompt, the results of GPT 3.5 and 4 are executed by API. As mini-GPT lacks an API, we input the "system" prompt as a user prompt and appended the phrase "After you read and understood the instructions, say 'ok' to me."

Furthermore, miniGPT4 requires a picture before starting a conversation, we uploaded a blank 300*300 picture of pure white in order to minimize the influence from the picture.

During the experiments, we found that the miniGPT is not as stable as we thought. Occasionally, miniGPT would stop responding and become unresponsive.

5 Results

We set scoring rules for each question.

FULL Score 160	Robot Q1	Robot Q2	Robot Q3	Robot Q4	Robot Q5	Json Q1	Json Q2	Json Q3	Total
GPT-3.5-Turbo	20	20	20	17	11	18	19	18	143
GPT-4	20	20	20	20	18	20	20	20	158
miniGPT-4	20	15	2	17	10	13	9	12	98

Robot Commands

5 questions. Each question 20 points. For each question:

1. No syntax error: 5 points.

2. Correct motion: 10 points.

3. Good format and comment: 5 points.

Q1: Go forward for 1 meter, go back for 2 meters.

GPT3.5-Turbo

5 + 10 + 5 = 20 p

```
START::
// Move forward for 1 meter
GO::Forward,1
// Move back for 2 meters
GO::Back,2
END::
```

GPT-4

5 + 10 + 5 = 20 p

```
START::

GO::Forward,1

GO::Back,2

END::
```

miniGPT-4

5 + 10 + 5 = 20 p

```
START::
GO::Forward,1
GO::Back,2
END::
```

Q2: Go forward for 2 meters, turn left and go forward for 0.7 meters, and shift right for 2 meters.

GPT-3.5-Turbo

5 + 10 + 5 = 20 p

```
START::

GO::Forward,2
```

```
//Turn left

TURN::Left

GO::Forward,0.7

//Shift right for 2 meters

GO::Right,2

END::
```

GPT-4

5 + 10 + 5 = 20 p

```
START::

GO::Forward,2

TURN::Left

GO::Forward,0.7

GO::Right,2

END::
```

miniGPT-4

0 + 7 + 5 = 15 p

Command "SHIFT" doesn't exitst.

```
START::
GO::Forward,2
//Turn left
TURN::Left
GO::Forward,0.7
//Shift right
SHIFT::Right
END::
```

Q3: Walk a square with 3-meter side length.

GPT-3.5-Turbo

5 + 10 + 5 = 20 p

```
// Start at the first corner of the square facing forward
START::
// Move forward for 3 meters
GO::Forward,3
// Turn right to face the second corner of the square
TURN::Right
// Move forward for 3 meters
GO::Forward,3
// Turn right to face the third corner of the square
TURN::Right
// Move forward for 3 meters
GO::Forward,3
// Turn right to face the fourth corner of the square
TURN::Right
// Move forward for 3 meters
GO::Forward,3
// End at the first corner of the square facing forward
END::
```

GPT-4

5 + 10 + 5 = 20 p

```
START::

// Walk the first side of the square
GO::Forward,3

// Turn right to walk the second side of the square
TURN::Right
GO::Forward,3

// Turn right to walk the third side of the square
TURN::Right
GO::Forward,3

// Turn right to walk the fourth side of the square
TURN::Right
GO::Forward,3

END::
```

miniGPT-4

0 + 2 + 0 = 2 p

illegal comments, and wrong motions.

```
To walk a square with a 3-meter side length, you can use the following
commands:

START::
GO::Forward,3
GO::Right,3
GO::Forward,3
GO::Left,3
GO::Forward,3
GO::Right,3
GO::Forward,3
END::

This will make the robot walk along the sides of a square with a 3-meter
side length.
END::
```

Q4: Turn left and walk 5 meters, then turn back and walk 2 meters, and repeat for 3 times.

GPT-3.5-Turbo

5 + 8 + 5 = 17 p

repeated 1 more time.

```
START::

//Turn left
TURN::Left

//Walk 5 meters forward
GO::Forward,5

//Turn back
TURN::Back

//Walk 2 meters forward
GO::Forward,2

//Repeat for 2 more times
TAG::
GO::Left,5
TURN::Back
GO::Forward,2
TAG::
GO::Left,5
TURN::Back
GO::Forward,2
TAG::
GO::Left,5
TURN::Back
```

```
GO::Forward,2

END::
```

GPT-4

5 + 10 + 5 = 20 p

```
START::

// Repeat 3 times
// First iteration
TURN::Left
GO::Forward,5
TURN::Back
GO::Forward,2

// Second iteration
TURN::Left
GO::Forward,5
TURN::Back
GO::Forward,2

// Third iteration
TURN::Left
GO::Forward,5
TURN::Back
GO::Forward,2

END::
```

miniGPT-4

5 + 8 + 5 = 17 p

missed one time repetition.

```
START::
TURN::Left
GO::Forward,5
TURN::Back
GO::Forward,2
TURN::Left
GO::Forward,5
TURN::Back
GO::Forward,2
TURN::Left
END::
```

Q5: Try walk a circle-like path, with radius of 3 meters.

GPT-3.5-Turbo

5 + 1 + 5 = 11 p

Not a cicle.

```
START::

// Move forward for 3 meters
GO::Forward,3

// Turn right and move forward for 6.28 meters (2pi3)
TURN::Right
GO::Forward,6.28

// Turn right and move forward for 6.28 meters
TURN::Right
GO::Forward,6.28

// Turn right and move forward for 6.28 meters
TURN::Right
GO::Forward,6.28

// Turn right and move forward for 6.28 meters
TURN::Right
GO::Forward,6.28

END::
```

GPT-4

5 + 8 + 5 = 18 p

Actaually turn angle cannot be specificated, but the workflow is correct.

```
// A circle-like path cannot be accurately represented with the available
commands.
// However, we can approximate it by moving in small increments and
turning slightly.
// This script will approximate a circle with a radius of 3 meters.

START::

// Approximate circle with 36 segments
// Each segment length: 2 * pi * 3 / 36 = 0.5236 meters
// Each turn angle: 360 / 36 = 10 degrees

// Repeat 36 times
GO::Forward,0.5236
```

```
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
GO::Forward,0.5236
TURN::Right
```

```
    END::
```

miniGPT-4

5 + 0 + 5 = 10 p

Not a circle.

```
START::
GO::Forward,3
//Turn right
TURN::Right
GO::Forward,3
//Turn left
TURN::Left
GO::Forward,3
//Turn right
TURN::Right
GO::Forward,3
//Turn left
TURN::Left
GO::Forward,3
END::
```

JSON format

3 questions. Each question 20 points. For each question:

1. No syntax error and well formated: 6 points.
2. Correct information: 7 points.
3. Structure is clear and easy to understand: 7 points.

Q1: Tom is 18 years old, Shawn is 5 years older than Tom, Tim is 3 years younger than Shawn. How old is each person above?

GPT3.5-Turbo

6 + 7 + 5 = 18 p

```
{
    "Answer": {
        "Tom": 18,
        "Shawn": 23,
        "Tim": 20
    },
    "Explanation": "Shawn is 5 years older than Tom, and Tim is 3 years
younger than Shawn."
}
```

GPT-4

6 + 7 + 7 = 20 p

```
{
  "Answer": [
    { "name": "Tom", "age": 18 },
    { "name": "Shawn", "age": 23 },
    { "name": "Tim", "age": 20 }
  ],
  "Explanation": "Shawn is 5 years older than Tom (18+5=23), and Tim is 3
years younger than Shawn (23-3=20)."
}
```

miniGPT-4

4 + 4 + 5 = 13 p

Not well formated, Tim's age is wrong.

```
{ "Answer": { "Tom":18, "Shawn":23, "Tim":15 }, "Explanation":"Tom is 18
years old, Shawn is 5 years older than Tom, which is 23 years old, and Tim
is 3 years younger than Shawn, which is 15 years old." }
```

Q2: I have a Macbook pro, which values 3000 dollars. I use it for video editing and programming.
I also have a iPad, I can do simple video editing work by iPad but cannot do programming work
on it. What can I do with the two devices above?

GPT3.5-Turbo

6 + 6 + 7 = 19 p

Value is not necessary.

```
{
    "Answer": {
        "Macbook Pro": {
            "Value": 3000,
            "Usage": ["Video Editing", "Programming"]
        },
        "iPad": {
            "Value": "Unknown",
            "Usage": ["Simple Video Editing"]
        },
        "Recommendation": "Use Macbook Pro for video editing and
programming tasks, and use iPad for simple video editing tasks on-the-go."
    },
```

```
        "Explanation": "Macbook Pro is suitable for heavy-duty tasks like
  video editing and programming, while iPad can be used for simple video
  editing tasks when you are not at your desk."
  }
```

GPT-4

6 + 7 + 7 = 20 p

```
  {
    "Answer": {
      "Macbook Pro": ["Video Editing", "Programming"],
      "iPad": ["Simple Video Editing"]
    },
    "Explanation": "You can do video editing and programming on your Macbook
  Pro, while you can only do simple video editing on your iPad."
  }
```

miniGPT-4

4 + 2 + 3 = 9 p

Wrong information.

```
  { "Answer": { "Macbook Pro":3000, "iPad":1000 }, "Explanation":"A Macbook
  Pro is worth 3000 dollars and can be used for video editing and
  programming. An iPad can be used for simple video editing, but it is not
  suitable for programming work." }
```

Q3: Tony has a box, in the box there is a pen, a can of cola, an envelope, inside the envelope there is a card-pocket, inside the card-pocket there is a bank card, a credit card, a passport. There is two photos sandwiched in the passport. Describe what Tony have and the containing relationship.

GPT3.5-Turbo

4 + 7 + 7 = 18 p

```
  {
  "Answer": {
  "Box": {
  "Contains": {
  "Pen": 1,
  "Can of Cola": 1,
  "Envelope": {
  "Contains": {
  "Card-Pocket": {
```

```
"Contains": {
"Bank Card": 1,
"Credit Card": 1,
"Passport": {
"Contains": {
"Photos": 2
}
}
}
}
}
}
}
}
},
"Explanation": "Tony has a box containing a pen, a can of cola, and an
envelope. The envelope contains a card-pocket, which contains a bank card,
a credit card, and a passport. The passport contains two photos."
}
```

GPT-4

6 + 7 + 7 = 20 p

```
{
  "Answer": {
    "Box": {
      "Pen": {},
      "Cola": {},
      "Envelope": {
        "Card-pocket": {
          "Bank card": {},
          "Credit card": {},
          "Passport": {
            "Photos": 2
          }
        }
      }
    }
  },
  "Explanation": "Tony has a box containing a pen, cola, and an envelope.
Inside the envelope, there's a card-pocket with a bank card, credit card,
and a passport. The passport has two photos sandwiched in it."
}
```

miniGPT-4

2 + 7 + 3 = 12 p

Wrong brace, wired structure.

```
{
"Tony": {
"Box": {
"Pen": true,
"Can of Cola": true,
"Envelope": {
"Card—Pocket": {
"Bank Card": true,
"Credit Card": true,
"Passport": {
"Photos": [
{
"descriptions": "sandwiched in the passport"
}
]
}
}
}
}
}
```

6 Conclusion

Compared to the massive computational power required for the original versions of ChatGPT3.5 and GPT4, miniGPT4's ability to achieve a similar level of performance on a single machine with a single graphics card is quite remarkable. However, we must recognize that miniGPT exhibits a noticeable gap when handling logical problems and producing strongly formatted outputs compared to the original GPT models. This is not merely due to a lack of computational power. In fact, to suit the limited computational power and memory space of a single graphics card, the reduced parameter scale of miniGPT is likely the true reason for its inability to solve complex and formatting problems.

These findings should be seen as preliminary and rough; the experiments and use cases we designed may not entirely be accurate. However, they can serve as a useful point of reference.

If we think of the original GPT3.5 and GPT4's capabilities in executing logical and formatted tasks as the levels of undergraduate and graduate students, respectively, then miniGPT might be at the level of a middle school or high school student. Although this level of disparity is clear, it seems insignificant compared to the computational power gap. However, whether this gap is linear is something we believe requires further research. In other words, to bridge these seemingly small gaps, we may ultimately still need to upgrade the model to a larger one that can compete with GPT.

7 Future Work

Fine-tuning miniGPT for specific domains to enhance its capability in solving certain types of problems might be the most valuable approach. Although, this might make miniGPT a 'specialized' language model, rather than a general-purpose model.

8 Acknowledgements

10 References

1. OpenAI. API Reference. Accessed: 2023-06-07. 2023. URL: https://platform.openai.com/docs/api-reference

2. Deeplearning. ChatGPT Prompt Engineering for Developers. Accessed: 2023-06-07. 2023. URL: https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/

3. Microsoft. Azure OpenAI Service Documentation. Accessed: 2023-06-07. 2023. URL: https://learn.microsoft.com/en-us/azure/cognitive-services/openai/