Local LLaMA with Function Calling on Raspberry Pi

Alfred Long

Why Edge Computing?

- Privacy
- Low latency
- High availabilty

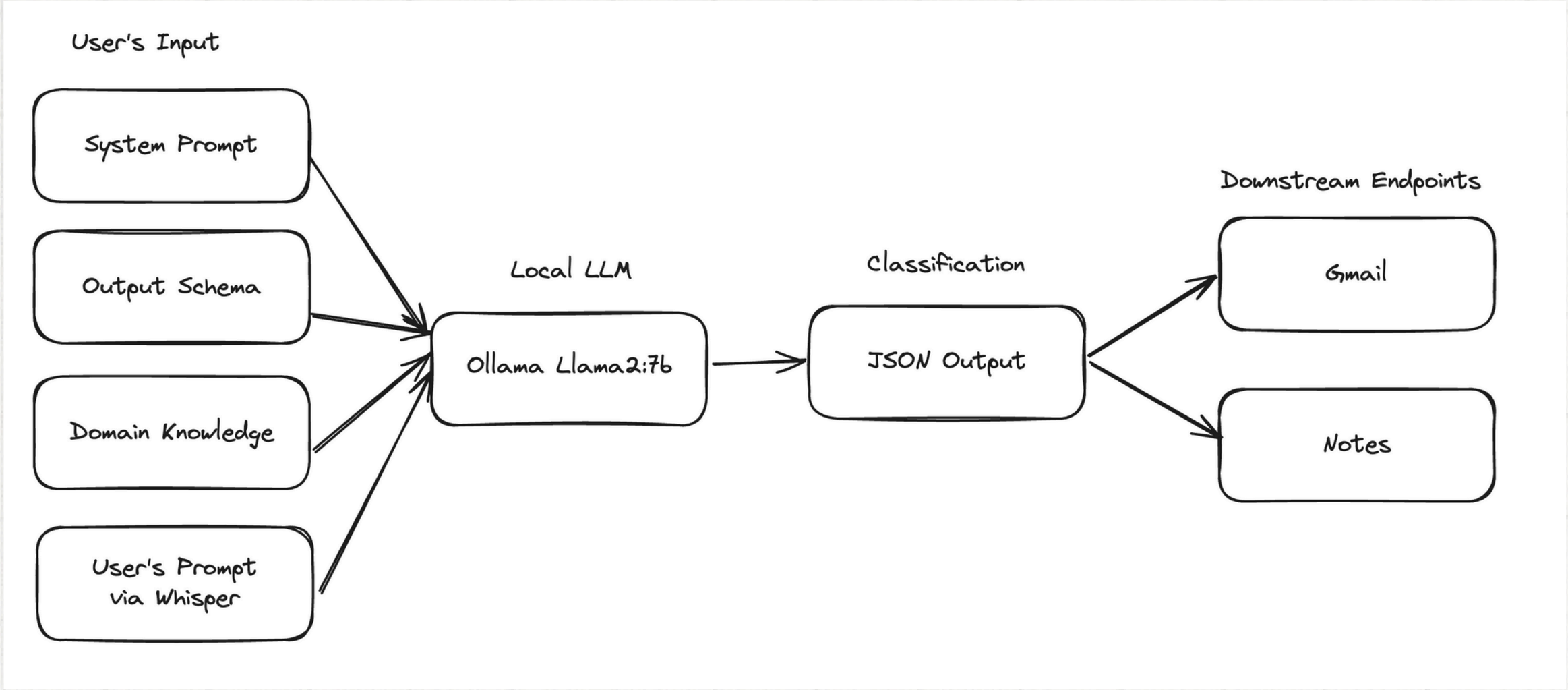
Project vision and mission

01. Platform for high school kids to explore with LLMs

02. Introduce ideas about edge computing and locally deployed servers

03. By exposing API endpoints and system prompts, students can play around

Tech Framwork



Sample Showcase

```
email_mapping = {
    "Alfred": "hlong25@wisc.edu",
    "Amber": "bdeng34@wisc.edu",
    "Suman": "suman@cs.wisc.edu",
}
```

```
schema_emails = {
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "person": {
                "type": "string",
                "description": "The name of the person to receive the email"
            },
            "message": {
                "type": "string",
                "description": "The message to be sent in the email"
            },
            "title": {
                "type": "string",
                "description": "The title of the email"
            },
            "address": {
                "type": "string",
                "description": "The address of the person to receive the email"
            }
        }
    }
}
```

Sample Showcase (ctn'd)

```
payload = {
    "model": "llama2",
    "messages": [
        {"role": "system", "content": f"""
        You are a helpful AI assitant.
        The user will input a message indicating what he wants to do.
        It could be either of the two cases (or both):
        1. Send an email to a person with a message and title.
        2. Make himself a note so that he won't forget important things to do.
        If it falls into the first category:
        Send an email to the corresponding person with the correct information using the emails mapping provided to you here: {email_mapping}
        If there are multiple emails to send, just output every email in the same format.
        Output in JSON format using the schema defined here: {schema_emails}.
        If it falls into the second category:
        Create a new note in default folder of default account using the macnotesapp.
        Output in JSON format using the schema defined here: {schema_note}.
        Note that each thing could falls into both categories. In that case, you should output both two kinds under the corresponding schema.
        """
```

```
{'emails': [{'person': 'Suman', 'message': 'I have a presentation tomorrow at 10 am and then I still have to make the tonight to present will have down or just a.', 'title': 'Reminder for pr
esentation and dinner reservation', 'address': 'some address'}, {'person': 'Amber', 'message': 'We need to cancel our reservation on the rear stick due to lack of table reserved.', 'title':
'Cancellation of dinner reservation', 'address': 'some address'}], 'notes': [{'name': 'Presentation reminder', 'body': 'I have a presentation tomorrow at 10 am and then I still have to make
the tonight to present will have down or just a.'}, {'name': 'Dinner reservation cancellation', 'body': 'We need to cancel our reservation on the rear stick due to lack of table reserved.'}]
}
```

```
"""
======EXAMPLE=======
{'emails': [{'person': 'some person', 'message': "some message", 'title': 'some title', 'address': 'some address'}, {...}],
'notes': [{'name': 'some name', 'body': 'some body'}, {...}]}
======EXAMPLE=======
```

DEMO!

Future Exploration

01

Better utilize the GPU on RPi

02

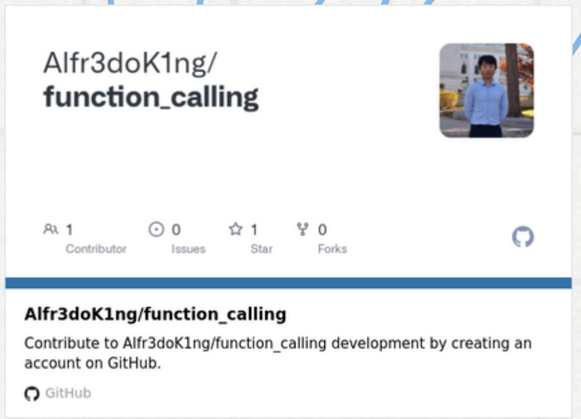
Break different parts on different machines

03

Incorporate RAG for domain knowledge

04

Connected with real world API, e.g. a switch or a lightbulb



Thank you very much!

Any question is appreciated!