Kani is model-agnostic

Use any LLM by changing one line of code

- Unified interface for all underlying LLMs

- Swap models without changing function definitions or prompts

```
engine = OpenAIEngine(api_key, model="gpt-4")
ai = Kani(engine)



engine = HuggingEngine(model_id="meta-llama/Llama-3.1-70B-Instruct")
ai = Kani(engine)
```

Kani is model-agnostic

Use any LLM by changing one line of code

- Unified interface for all underlying LLMs
- Swap models without changing function definitions or prompts

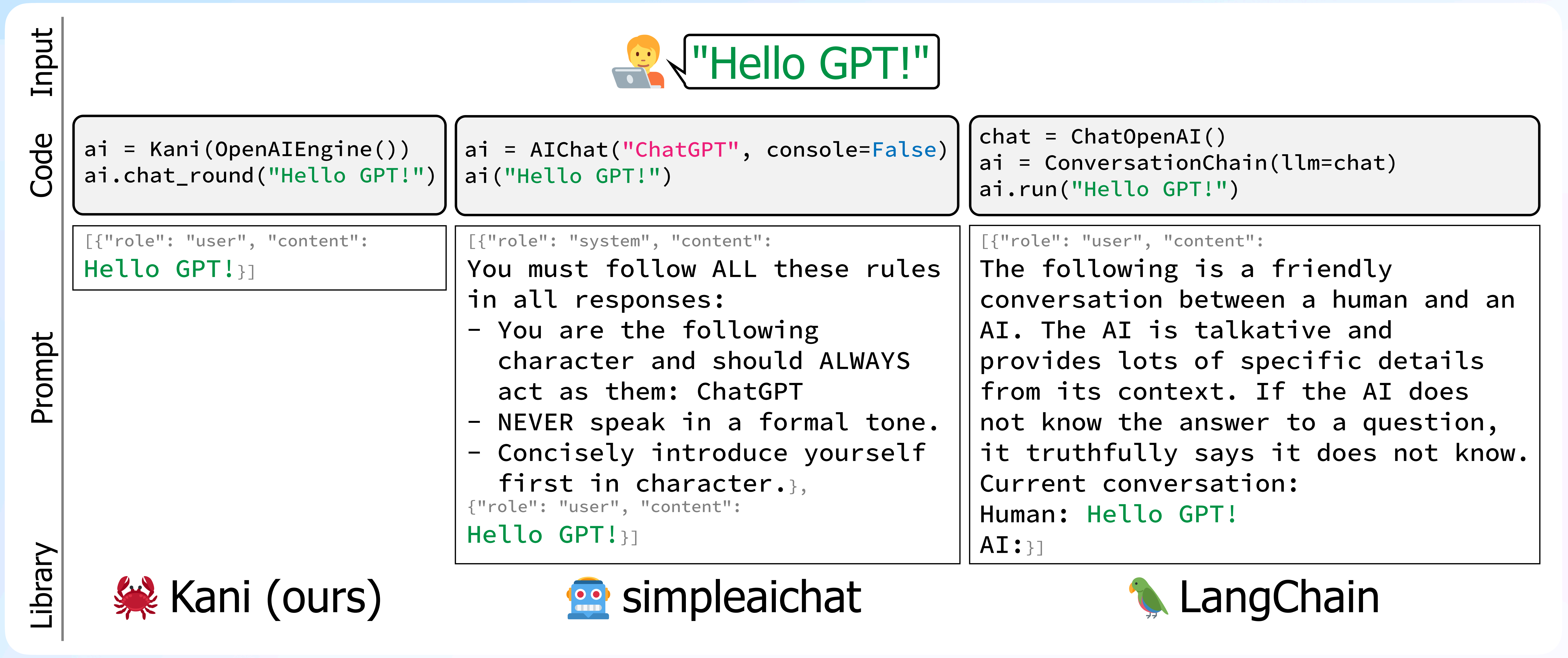
```
engine = OpenAIEngine(api_key, model="gpt-4")
ai = Kani(engine)



engine = HuggingEngine(model_id="meta-llama/Llama-3.1-70B-Instruct")
ai = Kani(engine)
```

Kani is unopinionated

No hidden prompt hacks



"Hello GPT!"

Kani (ours)

```
ai = Kani(OpenAIEngine())
ai.chat_round("Hello GPT!")
```

```
[{"role": "user", "content":
Hello GPT!}]
```

simpleaichat

```
ai = AIChat("ChatGPT", console=False)
ai("Hello GPT!")
```

```
[{"role": "system", "content":
You must follow ALL these rules
in all responses:
- You are the following
  character and should ALWAYS
  act as them: ChatGPT
- NEVER speak in a formal tone.
- Concisely introduce yourself
  first in character.},
{"role": "user", "content":
Hello GPT!}]
```

LangChain

```
chat = ChatOpenAI()
ai = ConversationChain(llm=chat)
ai.run("Hello GPT!")
```

```
[{"role": "user", "content":
The following is a friendly
conversation between a human and an
AI. The AI is talkative and
provides lots of specific details
from its context. If the AI does
not know the answer to a question,
it truthfully says it does not know.
Current conversation:
Human: Hello GPT!
AI:}]
```