AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Awadallah, Ryen W. White, Doug Burger, Chi Wang

*Part of the slides borrowed from Chi Wang’s talk at UIUC*
An “Agent” is an entity with the capacity to act in a given environment for task solving.
Conversable Agents

A conversable agent is an entity with a specific role that can pass messages to send and receive information to and from other conversable agents, e.g., to start or continue a conversation.
Conversable Agents
A conversable agent is an entity with a specific role that can pass messages to send and receive information to and from other conversable agents, e.g., to start or continue a conversation.

Why can LLMs be used as Agents?
Conversable Agents
A conversable agent is an entity with a specific role that can pass messages to send and receive information to and from other conversable agents, e.g., to start or continue a conversation.

Why can LLMs be used as Agents?

**Autonomy:** Autonomy in agents refers to their ability to operate independently without human intervention, including following instructions, initiating actions, and adapting outputs based on environmental input. LLMs demonstrate autonomy by generating human-like text, engaging in conversations, performing tasks, and showcasing creativity.

**Reactivity:** Reactivity in agents refers to their ability to quickly respond to changes in the environment by perceiving and taking appropriate actions. Language models traditionally have limited perceptual and action spaces but can expand them using tool usage, i.e., querying APIs to perform tasks. LLM based agents can exhibit reasoning and planning abilities comparable to symbolic agents through techniques like Chain-of-Thought (CoT) and problem decomposition.
Question: Aside from the Apple Remote, what other device can control the program Apple Remote was originally designed to interact with?

Answer: iPod

Thought 1: I need to search Apple Remote and find the program it was originally designed to interact with.

Act 1: Search[Apple Remote]

Obs 1: The Apple Remote is a remote control introduced in October 2005 by Apple ... originally designed to control the Front Row media center program ...

Thought 2: Apple Remote was originally designed to control the Front Row media center program. I need to search Front Row next and find what other device can control it.

Act 2: Search[Front Row]

Obs 2: Could not find [Front Row]. Similar: ['Front Row Seat to Earth', 'Front Row Motorsports', 'Front Row (software)', ...]

Thought 3: Front Row is not found. I need to search Front Row (software).

Act 3: Search[Front Row (software)]

Obs 3: Front Row is a discontinued media center software ...

Thought 4: Front Row (software) is controlled by an Apple Remote or the keyboard function keys. So the answer is keyboard function keys.

Act 4: Finish[yes]
How to empower every developer to build conversable agents?

A programming framework for easily building multi-agent applications that could span a broad spectrum of domains and complexities

- Initially developed in FLAML (Nov 2022)
- Spined off to a standalone repo (October 2023)  aka.ms/autogen
How to empower every developer to build conversable agents?

Hugging Face 😊 for Agents

- Initially developed in FLAML (Nov 2022)
- Spined off to a standalone repo (October 2023)

aka.ms/autogen
**Aim:** Multi-agent conversation framework with generic abstraction and effective implementation that has the flexibility to satisfy different application needs.

- How can we design individual agents that are capable, reusable, customizable, and effective in multi-agent collaboration?
- How can we develop a straightforward, unified interface that can accommodate a wide range of agent conversation patterns?
AutoGens Key Concepts

Define agents: Conversable & Customizable

Get them to talk: Conversation Programming

Conversable agent

Multi-Agent Conversations

Agent Customization

Flexible Conversation Patterns
A conversable agent is an entity with a specific role that can pass messages to send and receive information (in the form of text) to and from other agents, e.g., to start or continue a conversation.
A conversable agent is an entity with a specific role that can pass messages to send and receive information (in the form of text) to and from other agents, e.g., to start or continue a conversation.
Conversation Programming

Involves defining agent capabilities and roles, and programming their interaction behaviors through conversation-centric computation and control.

**Computation** refers to the actions agents undertake to compute their responses within a multi-agent conversation, while control flow dictates the sequence or conditions under which these computations occur. AutoGen's approach focuses on conversation-centric computations, where agents engage in actions pertinent to their conversations, leading to message passing for subsequent interactions unless certain termination conditions are met.

**Control flow** in AutoGen is driven by conversations, meaning that agents' decisions on message recipients and computation procedures are influenced by the ongoing inter-agent dialogue. This paradigm facilitates intuitive reasoning about complex workflows through agent actions and conversation-driven message passing.
Conversation Programming

Plot a chart of META and TESLA stock price change YTD.

Execute the following code...

Error package yfinance is not installed

Sorry! Please first pip install yfinance and then execute the code

Installing...

Output:

$ vs. Month

No, please plot % change!

Here is the revised code ...

Output:

% vs. Month
Application Enabled by AutoGens

A1. Math Problem Solving
A2. Retrieval-augmented Chat
A3. Decision Making
A4. Multi-agent Coding
A5. Dynamic Group Chat
A6. Conversational Chess
Math Problem Solving

![Diagram showing Student Proxy and Student Assistant]

Enable Autonomous and Human-in-the-loop Problem Solving

![Bar chart showing success ratio for different methods]

Autonomous Problem-Solving Performance

Methods:
- AutoGen
- ChatGPT +Code
- ChatGPT +Plugin
- GPT-4
- Multi-Agent Debate
- LangChain ReAct

Success Ratio (%):
- AutoGen: 69.48%
- ChatGPT +Code: 48.33%
- ChatGPT +Plugin: 45.00%
- GPT-4: 55.18%
- Multi-Agent Debate: 26.67%
- LangChain ReAct: 23.33%
Retrieval Augmented Chat & Question Answering
Retrieval-Augmented Chat

- Retrieval Augmentation
- Interactive Retrieval

1. Question and Contexts
2. Satisfied Answers or "Update Context"
3. Terminate, feedbacks or "Update Context"
4. Satisfied Answers or Terminate
Retrieval-augmented Chat without vs. with *interactive retrieval*
AutoGen introduces an interactive retrieval feature where, if the retrieved context lacks information, the assistant responds with "Sorry, I cannot find any information about... UPDATE CONTEXT," prompting further retrieval attempts.
Multi-Agent Coding
Task - Optimization and Planning

(a) Problem setup.
What if the roasting cost is increased by 5% because of the potential salary increase?

(a) Problem setup.
Writer, combines the functions of a “Coder” and an “Interpreter.”

What if the roasting cost is increased by 5% because of the potential salary increase?

After increasing the roasting costs by 5%, I've recalculated the optimal coffee distribution solution for the supply chain. As a result, the new optimal total cost has become 2526.5, compared to the initial cost of 2470.0. This indicates that the increased roasting costs have caused the overall distribution cost to rise by 56.5 units.
What if the roasting cost is increased by 5% because of the potential salary increase?

After increasing the roasting costs by 5%, I’ve recalculated the optimal coffee distribution solution for the supply chain. As a result, the new optimal total cost has become 2526.5, compared to the initial cost of 2470.0. This indicates that the increased roasting costs have caused the overall distribution cost to rise by 56.5 units.

**Writer** combines the functions of a “Coder” and an “Interpreter.”

AutoGen supports 3 agent setups due to its customizable framework, additionally also allows dynamic switching between code and text.
What if the roasting cost is increased by 5% because of the potential salary increase?

- Safeguard performance

<table>
<thead>
<tr>
<th></th>
<th>Metrics</th>
<th>Percentage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>Multi-GPT4</td>
<td>96.00%</td>
</tr>
<tr>
<td></td>
<td>Single-GPT4</td>
<td>88.00%</td>
</tr>
<tr>
<td></td>
<td>Multi-GPT3.5</td>
<td>83.00%</td>
</tr>
<tr>
<td></td>
<td>Single-GPT3.5</td>
<td>48.00%</td>
</tr>
<tr>
<td>Recall</td>
<td>Multi-GPT4</td>
<td>98.00%</td>
</tr>
<tr>
<td></td>
<td>Single-GPT4</td>
<td>78.00%</td>
</tr>
<tr>
<td></td>
<td>Multi-GPT3.5</td>
<td>72.00%</td>
</tr>
<tr>
<td></td>
<td>Single-GPT3.5</td>
<td>32.00%</td>
</tr>
</tbody>
</table>

Increasing roasting costs by 5%, I've re-evaluated the coffee distribution solution for the high-traffic post-launch. The new optimal total cost has increased to the initial cost of 2470.0. The increased roasting costs have led to an overall solution cost to rise by 56.5 units.
Conversational Chess
Challenging your pawn in the center. Your move.

Developing my knight to a good square. Your move.
register_reply method supported by AutoGen agents to instantiate player agents and a board agent with custom reply functions, so players can only make moves while the chess board agent is restricted to only verify the legality of the move.
Player White: Your move
Player Black: Alright, let's start. I'll move my pawn from e7 to e5.
Here's how the board looks now:
```
  n b q k b n r
  p p p b p p p
  . . . . . . .
  . . . . . . .
  . . . . . . .
  P P P P P P P
  R N B Q K B N R
```
Your turn!

... "...

Player White: Yes, it certainly is! Let's keep this up. I'll move my knight from b8 to c6. Here's the updated board:
```
  n b q k b n r
  P P P b P P P
  . . . n . . .
  . . . p . . .
  . . . p . . .
  P P P P P P P
  R N B Q K B N R
```
Your move! You're giving me a tough time here.

Player White: Your move
Player Black: Alright, let's start. I'll move my pawn from e7 to e5.
Here's how the board looks now:
```
  n b q k b n r
  P P P b P P P
  . . . . . . .
  . . . . . . .
  . . . . . . .
  P P P P P P P
  R N B Q K B N R
```
Your turn!

Player White: Yes, it certainly is! Let's keep this up. I'll move my knight from b8 to c6. Here's the updated board:
```
  n b q k b n r
  P P P b P P P
  . . . n . . .
  . . . p . . .
  . . . p . . .
  P P P P P P P
  R N B Q K B N R
```
Your move! You're giving me a tough time here.

Board Agent: Your move is illegal. You changed the rock at all to knight and move it to c6. Please check your decision and re-make your move.

Player White: Sorry for the mistake. I'll move my knight from b8 to c6. Here's the updated board:
```
  n b q k b n r
  P P P b P P P
  . . . n . . .
  . . . p . . .
  . . . p . . .
  P P P P P P P
  R N B Q K B N R
```

"..."
## Summary of Benefits

<table>
<thead>
<tr>
<th>Multi-agent conversation</th>
<th>Improve over natural iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Divide &amp; conquer</td>
</tr>
<tr>
<td></td>
<td>Grounding &amp; validation</td>
</tr>
<tr>
<td>AutoGen</td>
<td>Modularity for ease of use &amp; extension</td>
</tr>
<tr>
<td></td>
<td>Natural human participation</td>
</tr>
<tr>
<td></td>
<td>Fast &amp; creative experimentation</td>
</tr>
</tbody>
</table>
Challenges

- **Finite context length**: The restricted context capacity limits the inclusion of historical information, detailed instructions, API call context, and responses. The design of the system has to work with this limited communication bandwidth, while mechanisms like self-reflection to learn from past mistakes would benefit a lot from long or infinite context windows. Although vector stores and retrieval can provide access to a larger knowledge pool, their representation power is not as powerful as full attention.

- **Challenges in long-term planning and task decomposition**: Planning over a lengthy history and effectively exploring the solution space remain challenging. LLMs struggle to adjust plans when faced with unexpected errors, making them less robust compared to humans who learn from trial and error. How many previous steps do we unroll?
Open Questions

● How to design optimal multi-agent workflows?
  ○ Evaluation
  ○ Optimization

● How to create highly capable agents?
  ○ Integration
  ○ Learning

● How to enable scale, safety and human agency?
  ○ Teaching
  ○ Interface
Future Works

AutoGen Studio: Interactively Explore Multi-Agent Workflows

- Declaratively define and modify agents and multi-agent workflows
- Create chat sessions with the specified agents and view results
- Explicitly add skills to your agents and accomplish more tasks
- Publish your sessions to a local gallery

MultiModal

Teachable Agents remember user teachings long-term

Learning Agents

Longer contexts

Evaluation

AgentEval - Assess the Utility of LLM-powered Applications

- November 20, 2023 - 10 min read

Júlia Kolesnik
Senior Researcher at Microsoft Research

Negar Arakzadeh
PhD student at the University of Waterloo

EcoAssistant - Using LLM Assistants More Accurately and Affordably

- Assistant is designed to solve user tasks more accurately and reliably
- The cost of using LLM models
- Retriever Hierarchy
- RAG (Retrieval-Augmented Generation) to improve the success rate of information retrieval
Future Works

Further works can investigate which strategies, such as agent topology and conversation patterns, lead to the most effective multi-agent conversations.

MultiModal

Teachable Agents remember user teachings long-term

AutoGen Studio: Interactively Explore Multi-Agent Workflows

- Declaratively define and modify agents and multi-agent workflows
- Create chat sessions with the specified agents and view results
- Explicitly add skills to your agents and accomplish more tasks
- Publish your sessions to a local gallery

Evaluation

AgentEval - Assess the Utility of LLM-powered Applications

- November 20, 2023 - 10 min read
- Julia Kietzmann, Senior Researcher at Microsoft Research
- Negar Arabzadeh, PhD student at the University of Waterloo

EcoAssistant - Using LLM Assistants More Accurately and Affordably

- Assistant is designed to solve user requests more accurately and cost-effectively
- The cost of using GPT models
- The idea of Retrieval-Augmented Generation (RAG) to improve the success rate validation estimation

Learning Agents

Longer contexts
Thank You!!