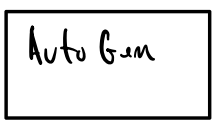
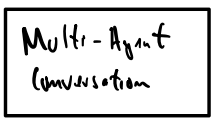
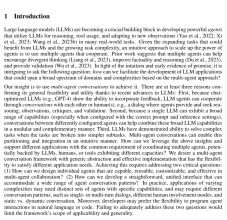
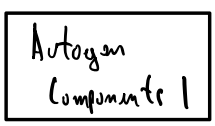
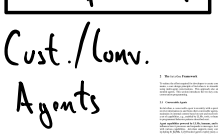


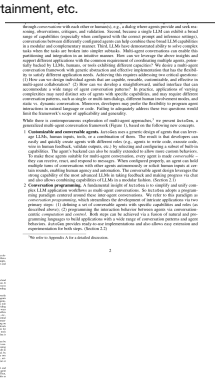


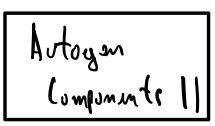


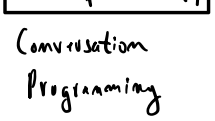
AutoGen2 is an open-source framework that allows developers to build LLM ap- plications via multiple agents that can converse with each other to accomplish tasks. AutoGen agents are customizable, conversable, and can operate in vari- ous modes that employ combinations of LLMs, human inputs, and tools. Using AutoGen, developers can also flexibly define agent interaction behaviors. Both natural language and computer code can be used to program flexible conversation patterns for different applications. AutoGen serves as a generic framework for building diverse applications of various complexities and LLM capacities. Em- pirical studies demonstrate the effectiveness of the framework in many example applications, with domains ranging from mathematics, coding, question answer- ing, operations research, online decision-making, entertainment, etc.

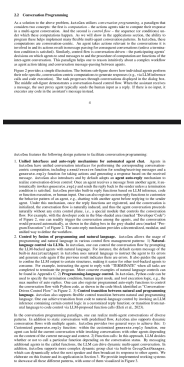


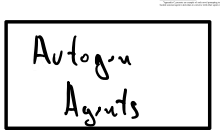


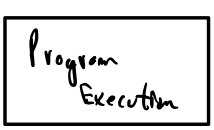












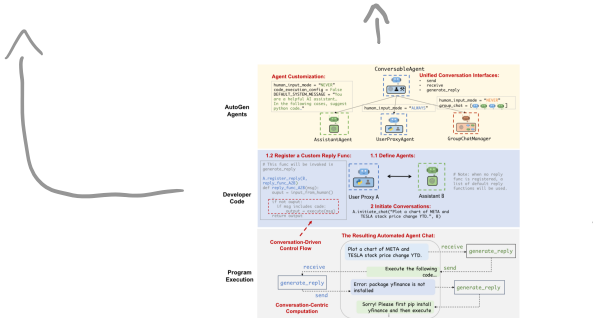


Figure 2: Illustration of how to use AutoGen to program a multi-agent conversation. The top sub-figure illustrates the built-in agents provided by AutoGen, which have unified conversation interfaces and can be customized. The middle sub-figure shows an example of using AutoGen to develop a two-agent system with a custom reply function. The bottom sub-figure illustrates the resulting automated agent chat from the two-agent system during program execution.

By allowing custom agents that can converse with each other, conversable agents in AutoGen serve as a useful building block. However, to develop applications where agents make meaningful progress on tasks, developers also need to be able to specify and mold these multi-agent conversations.

Autogen Applications

Autogen Applications

Autogen Applications

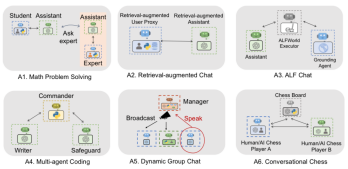


Figure 3: Six examples of diverse applications built using AutoGen. Their conversation patterns show AutoGen's flexibility and power.

Demo

Q&A

Break

Practice

Feedback

Autogen + Open AI

Tools / Function Call

Demo / Practice

(Search Agent)

Q&A / Break

Autogen Pers. Assist.

Demo / Practice

(Personal Assistant)

Autogen Mus. Agent

Q&A / Break

Next?

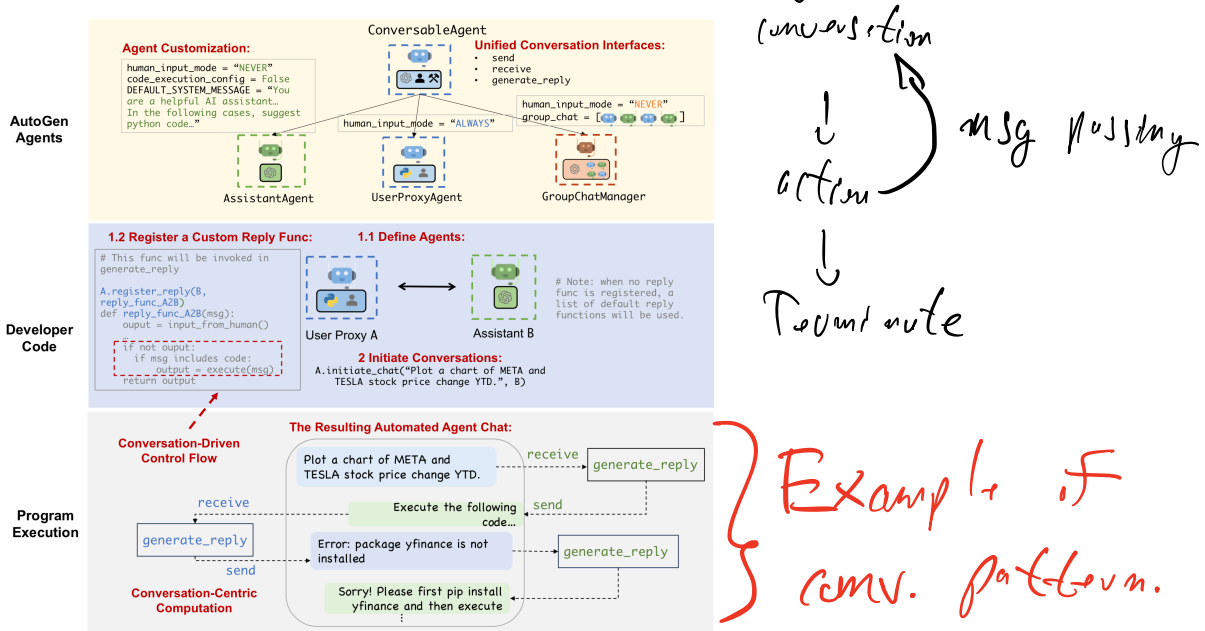
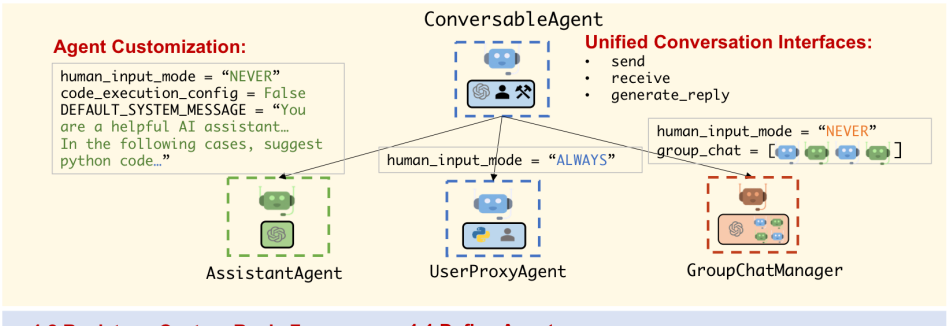


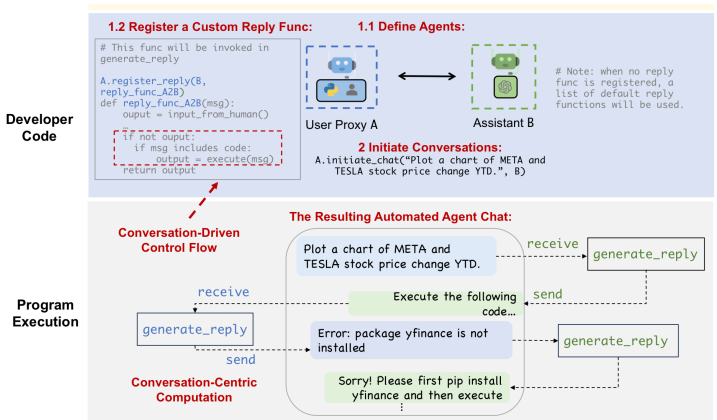
Figure 2: Illustration of how to use AutoGen to program a multi-agent conversation. The top sub-figure illustrates the built-in agents provided by AutoGen, which have unified conversation interfaces and can be customized. The middle sub-figure shows an example of using AutoGen to develop a two-agent system with a custom reply function. The bottom sub-figure illustrates the resulting automated agent chat from the two-agent system during program execution.

1.



"Unified interface & auto. reply for autom. chat"

2.



Control = Nat. lang + Programming

Conversational Patterns

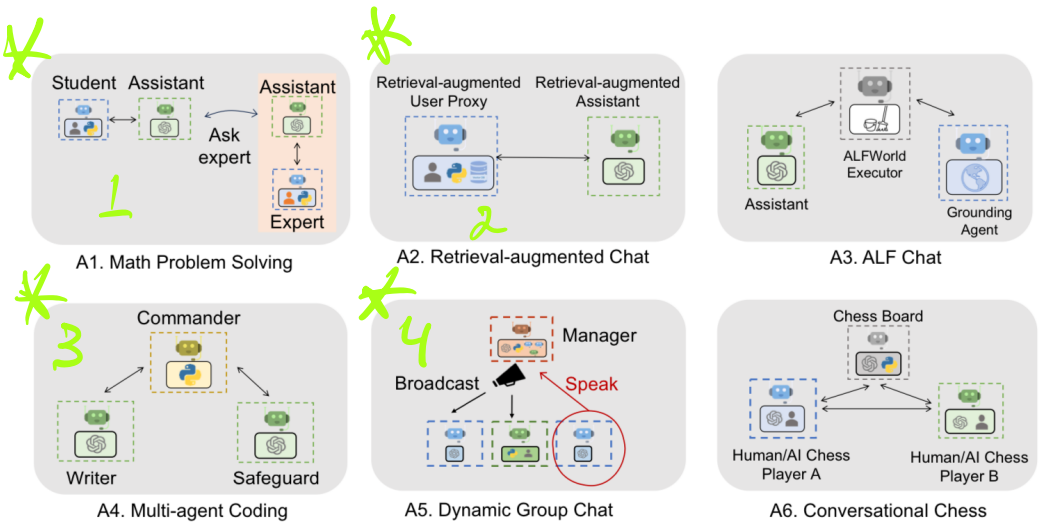


Figure 3: Six examples of diverse applications built using AutoGen. Their conversation patterns show AutoGen's flexibility and power.

Pattern := how you set up the interac-
tions. (** if I add these, must add
code examples)

1. Student - Assistant - Expert (Meta problem solving)

2. Retr. Aug. Chat

3. Writer - Commander - Safeguard (multi-agent coding)

4. Dynamic - group chat