Stanford | ONLINE

# Agentic AI

## XAG329A

## Course Syllabus

### Course Description

Learn how to design and build modern AI agents with the same graduate-level curriculum taught by Adjunct Professor Chowdhery and Assistant Professor Mirhoseini at Stanford, adapted for working professionals. This agentic AI course guides you through end-to-end agent workflows, covering the latest techniques in self-improving AI agent tool use, reasoning, and planning.

Simple agents fall apart the moment the task gets messy: they can't check their own work, recover when something goes wrong, or carry context across sessions. This course teaches you how agents do all three. Drawing on Stanford research and recent advances in the field, you'll learn both the theoretical foundations and practical engineering skills needed to design, evaluate, and deploy robust AI agents for real-world applications..

**You Will Learn To:**

- Design agentic systems capable of handling complex, multi step real world applications
- Practice test time scaling techniques to improve LLM performance within agentic systems
- Understand state of the art self improvement methods, including verifiers, feedback loops, reinforcement learning, and search
- Analyze agents that effectively use tools and actions
- Augment LLMs with retrieval mechanisms and long term memory systems
- Develop planning capabilities and support multi step reasoning in agents
- Design and conduct robust evaluation frameworks for agent performance

### Instructors

Aakanksha Chowdhery
Adjunct Professor of Computer Science, Stanford University

Azalia Mirhoseini
Assistant Professor of Computer Science, Stanford University

### Prerequisites

This course is built for technical professionals who want to move past basic LLM usage and build systems that reason, plan, and act. You should come in with:

- Python programming experience
- Basic familiarity with LLMs, including how to call LLM APIs
- Basic understanding of machine learning concepts

## Course Structure

The course is delivered in three parts. Each part builds on the one before it, and the assignments carry forward so that the work you do early in the course becomes the foundation for the agent you build at the end.

**Part 1: LLM and Agent overview**
Large language models from pre training through post training, and the design patterns behind basic agents.

**Part 2: AI Agent Test-Time Scaling and Self-Improvement Techniques**
Test time scaling, verification and reward modeling, and the methods agents use to improve from feedback.

**Part 3: Agent Tool-Use, Memory, Planning/Reasoning, and Robust Evaluations**
Planning and multi step reasoning, agentic frameworks for software engineering, memory, long horizon evaluation, and open research directions.

## Course Topics at a Glance

**Part 1, Module 1: Introduction**
Meet Professor Chowdhery and Professor Mirhoseini, and get oriented to the evolving LLM landscape, key applications in coding and personal AI assistants, and the foundational components of LLM development.

**Part 1, Module 2: LLM Overview**
The foundational principles of large language model architecture and the post training lifecycle, including how models compress data during pre training and how base models become specialized through supervised fine tuning and alignment.

**Part 1, Module 3: RLHF and RLVR Overview**
How a language model is post trained: supervised fine tuning, reinforcement learning from human feedback and reward model preparation, Constitutional AI for automated preference data, and reasoning models trained with verifiable rewards.

**Part 1, Module 4: Basic AI Agent**
Agent fundamentals, the ReAct pattern, and representative workflows, illustrated through coding agents and research agents, with lessons from the production agent Claude Code and the open problems in building self improving agents.

**Part 2, Module 1: Test Time Scaling**
How allocating more compute at inference improves performance, methods for generating and selecting high quality outputs, and how to optimize the balance between computational cost and response quality.

**Part 2, Module 2: Self Improvement, Rewards and Verification**

Output and process verification and why each improves performance in training and inference, the difference between output and process rewards, and why an ensemble of weak verifiers can outperform a single large model.

**Part 2, Module 3: Self Improvement Techniques**
How agents combine reasoning with external feedback, how they ground outputs by interacting with tools and environments, and how reinforcement learning uses execution feedback and verification signals to improve performance.

**Part 2, Module 4: Course Conclusion**
A synthesis of the key ideas around test time compute and self improvement, leading into the Part 2 final assignment.

**Part 3, Module 1: Reasoning and Self Improving Agents**
How agents plan and reason across multiple steps through tree search and parallel execution, how they are trained on synthetic data for reasoning and tool use, and how LLMs are being used for scientific discovery.

**Part 3, Module 2: Agentic Frameworks for Software Engineering and Augmenting LLMs with Memory**
Agents applied to software development, including CUDA kernel development and compiler optimization, and how to extend agent memory using ideas from MemGPT and Cartridges.

**Part 3, Module 3: Long Horizon Tasks and Agentic Evaluations**
How to measure what agents can reliably complete as task complexity and duration increase, why they struggle with certain categories of work, and how economic value and task duration give different views of progress.

**Part 3, Module 4: Next Frontiers in Agentic AI**
A recap of the course, four recent papers on self improving agents, and future research directions including the self improvement loop and intelligence efficiency as inference costs rise.

## Coursework and Requirements

**Coursework**

This course contains a combination of activities such as videos, reading materials, written exercises, multiple choice exercises, and hands on coding assignments. Your responses to the exercises will be kept within the learning platform. However, you will not have access to the responses you submitted after your course access has expired. If you would like to keep a copy for your records, please do so separately.

**Live Sessions**

Live video sessions run every six weeks, so your access window always includes two sessions regardless of when you enroll. Sessions alternate between two formats: a look at what is new in agentic AI led by an industry speaker or Stanford affiliate, and an application based showcase led by a course facilitator. Attendance is optional.

## Course Requirements

Pass all three parts to earn your certificate. Each part ends with a single assessment.

| Assessment | Format | Passing Score |
|---|---|---|
| Part 1 Final Exam | Multiple-choice exam | 85% or higher |
| Part 2 Final Assignment | Graded coding assignment, about 3 hours | 80% or higher |
| Part 3 Final Assignment | Graded coding assignment, about 5 hours | 80% or higher |

Part 2 and Part 3 assignments are graded on a twice weekly cycle, with results posted in Moodle. Your certificate of completion is issued after your final result is posted.

## Part 1 Final Exam

The Final Exam consists of multiple-choice questions derived from a combination of the course lectures and knowledge checks. In order to pass the course, you must achieve a score of 85% or higher. You may take the exam more than once.

## Part 2 and Part 3 Coding Assignments

Coding begins in Part 2. The Part 2 assignment covers the core techniques for test time scaling and self improvement. The Part 3 assignment has you build a self improving method using agent tools and then extend it into a deep research agent, so you finish the course with a complete working agentic system.

## Required Materials and Additional Costs

Each of Part 2 and Part 3 has a private GitHub repository. You will get a link to the repository inside Moodle once you start that part.

The Part 2 and Part 3 coding assignments call external model APIs, so you will need to set up your own OpenRouter account and API key. Budget approximately $100 in API usage on top of the course fee.

## Course Materials

All course materials are provided within the course learning platform. These include videos, handouts and assignments.

This course features real-world examples to provide the learner with relevant examples of the concepts. Case studies and examples involving specific persons or companies do not imply any affiliation of those persons or companies with Stanford, nor do they imply Stanford's endorsement of those persons or companies or their actions.

Stanford | ONLINE

The course learning platform is available for 60 days after the date of enrollment via your mystanfordconnection account.

**Online Conduct and Honor Code**

We take academic integrity and misconduct very seriously. Please refer to the CGOE Terms of Service, which includes information on our Honor Code and Rules for Online Conduct. Failure to comply with these policies may result in disciplinary action, including removal from the course/program.