Core Concepts of AI Technology, General-Purpose Tools, and AI Capabilities and Boundaries

Presenter:** Gu Liping, Researcher

2026-07-15

ChinaRxiv

Abstract

This report, presented by Researcher Gu Liping of the National Science Library, Chinese Academy of Sciences, targets the exchange librarian training program within the library and information system and systematically expounds the core concepts, general-purpose tools, capability boundaries, and compliant applications of generative artificial intelligence (GenAI) in the field of library and information science. The report is divided into four parts. The first part reviews 20 core technical concepts, including AI, AGI, complex systems, emergent capabilities, the Transformer architecture, MoE, Token, and RLHF, and constructs a systematic cognitive framework for large language models. The second part, taking platforms such as DeepSeek, OpenClaw, and Coze as examples, introduces the operating modes of AI tools, seven patterns of agent design, eight memory management strategies, and localized deployment solutions. The third part focuses on the technical limitations of AI, including hierarchical theories of scientific research, Token cost optimization, vector database retrieval, seven formulas for prompt engineering, the three-layer architecture of Agent Skill, and security protection. In conjunction with laws and regulations such as the Law of the People' s Republic of China on Scientific and Technological Progress and the Interim Measures for the Management of Generative Artificial Intelligence Services, it emphasizes scientific and technological ethics and research integrity. The fourth part summarizes the implications for library and information work and proposes five recommendations: actively embrace AI, maintain clear awareness, use it in compliance with regulations, continue learning, and return to the essence of the profession. The report emphasizes that AI is an auxiliary tool for improving efficiency rather than a means of replacing professional judgment. Library and information professionals should make good use of technology, strictly uphold bottom lines, and give full play to irreplaceable human intelligence in knowledge organization.

Full Text

Core Concepts of AI Technology, General-Purpose Tools, and AI Capabilities and Boundaries

Presenter: Gu Liping, Researcher

Affiliation: Documentation and Information Center, Chinese Academy of Sciences

Occasion: Training Program for Exchange Librarians in the Documentation and Information System of the Chinese Academy of Sciences

Time: 10:00–11:40 (100 minutes)

Report Outline (including third-level headings and time annotations)

I. Opening Remarks and Report Overview (10:00–10:05)

1. Remarks and Background Introduction

1.1 Background of the Training Program and Significance of the Report

1.2 Report Topic Structure and Expected Objectives

II. Core Concepts of AI Technology (10:05–10:35)

2. Basic Concepts of Generative Artificial Intelligence (GenAI)

2.1 Artificial Intelligence (AI) and Artificial General Intelligence (AGI)

2.2 Complex Systems, Emergent Capabilities, and World Models

2.3 Foundation Models: From General-Purpose Platforms to Specialized Applications

3. Model Architecture of Generative Artificial Intelligence

3.1 Large Language Models (LLM) and the Transformer Architecture

3.2 Mixture-of-Experts Models (MoE) and Distributed Models

4. Key Technologies of Generative Artificial Intelligence

4.1 Self-Attention Mechanism and Word Embeddings

4.2 Tokenization, Parameters, and Context Length

5. Training Methods for Generative Artificial Intelligence

5.1 Scaling Laws and Pretraining

5.2 Fine-Tuning, RLHF, and Few-Shot Learning

III. General AI Tools (10:35–11:05)

6. Large-Language-Model Application Tools—Using DeepSeek as an Example

6.1 Core Differences Between Fast Mode and Expert Mode

6.2 Collaborative Use of Intelligent Search and Document Upload

6.3 Report Generation and Typeset Output

7. AI Agents and Workflows

7.1 Seven Core Agent Design Patterns

7.2 Differences Between AI Agents and Agentic AI

7.3 Eight Memory-Management Strategies

8. Agent Deployment Platforms and Tool Ecosystems

8.1 OpenClaw: An Open-Source AI Agent Gateway

8.2 Building Workflows with Coze

8.3 iFlow, MaxClaw, AstronClaw, WorkBuddy, and other platforms

8.4 Local deployment of OpenClaw and integration with Ollama

IV. AI Capabilities and Boundaries (11:05–11:35)

9. The Boundaries of AI Capabilities and Technical Limitations

9.1 Layered Theory of Scientific Research and the Positioning of AI Applications

9.2 Token Consumption Mechanisms and Cost Optimization

9.3 Vector Databases and Knowledge-Retrieval Capabilities

10. Prompt Engineering and Skill Development Practice

10.1 Seven Prompt-Questioning Formulas

10.2 The Three-Layer Architecture of Agent Skills and Iterative Optimization

10.3 Security Protection: Skill Vetter and Lobster Butler

Main Text of the Speech

I. Opening Remarks and Overview of the Report (10:00–10:05)

Colleagues, and participants in the exchange-librarian training program, good morning.

I am Gu Liping from the Documentation and Information Center of the Chinese Academy of Sciences. It is a great honor to be here today in this training program to discuss with you a topic that is profoundly changing the way we work —"Core Concepts of AI Technology, General-Purpose Tools, and the Capabilities and Boundaries of AI."

We are at a special historical juncture. Over the past two or three years, generative artificial intelligence has penetrated, at an astonishing speed, into research, education, documentation and information services, and many other fields. As documentation and information professionals, we are both guardians of knowledge and providers of knowledge services. We therefore have an even greater

responsibility to be among the first to understand this technology, master it, and at the same time recognize clearly its boundaries and limitations.

Today' s report will last 100 minutes and is divided into four parts. The first part will address the core concepts of AI technology, helping everyone build a systematic understanding of generative artificial intelligence. The second part will introduce general-purpose AI tools, including mainstream platforms such as DeepSeek, OpenClaw, and Coze. The third part will discuss the capabilities and boundaries of AI, covering technical limitations, laws and regulations, and research ethics. Finally, there will be a brief summary and outlook. I hope this report will lay a solid conceptual foundation for everyone' s subsequent hands-on practice sessions.

II. Core Concepts of AI Technology (10:05-10:35)

2. Fundamental Concepts of Generative Artificial Intelligence (GenAI)

Let us begin with the most basic concepts. Many colleagues may already be frequent users of various AI tools, but still lack a systematic understanding of the underlying concepts. I organize them into six core concepts.

2.1 Artificial Intelligence (AI) and Artificial General Intelligence (AGI)

The first is artificial intelligence (AI). AI is a branch of computer science aimed at developing systems capable of performing tasks that require human intelligence. This concept is not new; it was proposed as early as the 1950s. But what has truly brought AI into millions of households is the emergence of large language models in recent years.

The second is artificial general intelligence (AGI). AGI refers to systems possessing intelligence comparable to, or surpassing, that of humans, and capable of learning and handling complex tasks across domains. At present, all the AI tools we use—including ChatGPT and DeepSeek—are still not AGI; they are the culmination of specialized AI. But AGI is the goal and direction pursued by the industry, and understanding this distinction is very important.

2.2 Complex Systems, Emergent Capabilities, and World Models

The third is complex systems. A complex system is composed of many interconnected components, and its overall behavior cannot be derived simply. A large language model is a complex system—it contains hundreds of billions or

even trillions of parameters, and we cannot explain each of its outputs through a simple chain of cause and effect.

The fourth is emergent capability. This refers to new capabilities that suddenly appear once an AI model reaches a certain scale threshold. For example, when model parameters increase from the tens-of-billions level to the hundreds-of-billions level, the model suddenly acquires capabilities it did not previously possess at all, such as multi-step reasoning, code generation, and cross-language translation. This phenomenon of "quantitative change leading to qualitative change" is emergence. Just as a single human neuron is not intelligent, but when 86 billion neurons are combined, human intelligence is produced.

The fifth is the world model. A world model is an AI system' s internal understanding of the operating mechanisms of the real world. After a large model reads massive amounts of text during training, it not only learns the statistical ...

statistical patterns, and also, to some extent, "understands" the logic by which the world operates. Of course, this understanding is incomplete and sometimes even erroneous; this is also the fundamental reason why AI produces hallucinations.

2.3 Foundation Models: From General-Purpose Platforms to Professional Applications

The sixth concept is foundation models. A foundation model is a general-purpose AI model pretrained on a large-scale dataset; it can provide basic capabilities for specific tasks. By way of comparison, a foundation model is like a university graduate who has received a general education and possesses a broad base of knowledge, but still needs job-specific training before being competent in a particular position. The fine-tuning and few-shot learning discussed later are precisely the processes of "job training" that enable a foundation model to adapt to specific tasks.

The path from a general-purpose platform to professional applications starts from the foundation model and, through methods such as fine-tuning, prompt engineering, and retrieval augmentation, enables the model to develop professional capabilities in a specific domain. Much of the work of literature and information centers operates at this level—how to make general-purpose AI models better serve specialized tasks such as literature retrieval, knowledge organization, and research evaluation.

3. Model Architectures of Generative Artificial Intelligence

3.1 Large Language Models (LLMs) and the Transformer Architecture

Next, we discuss model architectures. The seventh concept is the large language model (LLM). An LLM is a deep neural network with an enormous number of parameters, used to understand and generate human language. DeepSeek, ChatGPT, and GLM, which we use in everyday contexts, are all essentially LLMs.

The eighth concept is the Transformer architecture. This is the foundational architecture of all current large language models. Based on the self-attention mechanism, it thoroughly transformed methods for sequence modeling. The 2017 Google paper *Attention Is All You Need* proposed the Transformer, after which the entire field of natural language processing was reshaped. The core advantage of the Transformer is that it can process all positions in a sequence in parallel, unlike earlier RNNs that processed text word by word; this greatly improves training efficiency.

3.2 Mixture-of-Experts Model (MoE) and Diffusion Models

The ninth is the Mixture-of-Experts model (MoE). MoE improves task accuracy and efficiency through a sparse activation strategy. A traditional model activates all parameters during each inference, whereas MoE activates only some of its "expert" modules. This is like a hospital in which not every doctor sees every patient; instead, patients are triaged to different specialties according to their conditions. DeepSeek adopts the MoE architecture, which is also one of the key reasons it can control costs while maintaining strong capabilities.

The tenth is the diffusion model. Diffusion models generate data by simulating the process of noise diffusion, and are mainly used for image generation. The principle is to gradually add noise to an image until it becomes pure noise, and then learn the reverse process, progressively recovering a clear image from the noise. CogView by Zhipu and OpenAI' s DALL-E both use diffusion models at the underlying level.

4. Key Technologies of Generative Artificial Intelligence

4.1 Self-Attention Mechanism and Word Embeddings

The eleventh is the self-attention mechanism. This is the core of the Transformer architecture and can process all positions in a sequence in parallel. In simple terms, the self-attention mechanism allows the model, when processing one word, to simultaneously "attend to" all the other words in the sentence and determine which words are more closely related to the current word. For example, when processing the word "Apple," if the context contains "phone," the model will tend to understand it as referring to Apple Inc., rather than to the fruit.

The twelfth is word embeddings. Word embeddings map words into a low-dimensional vector space, so that words with similar meanings are close to one another in that vector space. For example, the distance between "cat" and "dog" in the vector space will be much smaller than that between "cat" and "car." This ability to transform semantics into mathematical representations is the foundation for machines to "understand" language.

4.2 Tokenization, Parameters, and Context Length

The thirteenth is tokenization. A token is the smallest unit by which a large language model processes text.

A tokenizer segments text into meaningful units. For Chinese, a single Chinese character may be one Token, or several characters may be combined into one Token, depending on the tokenizer' s statistical rules. What requires particular attention is that Chinese is more "Token-expensive" than English, because Chinese tokenization is more fine-grained. This is also an important factor to consider when controlling costs in our use of AI tools.

The fourteenth is parameters. Parameters are learnable variables inside the model, used for prediction or classification. Parameter scale is the core indicator for measuring model size—7B means 7 billion parameters, and 70B means 70 billion parameters. The more parameters there are, the stronger the model' s capabilities usually are, but the higher the costs of training and inference also become.

The fifteenth is context length. Context length is the maximum number of Tokens that a model can process in a single inference. Early models had a context length of only 2,048 Tokens; now there are already models that support 128K or even longer contexts. Context length directly determines how long a document the model can process and how many rounds of dialogue history it can remember.

5. Training Methods for Generative Artificial Intelligence

5.1 Scaling Laws and Pretraining

The sixteenth is scaling laws. Scaling laws describe the relationship between model performance and training scale—in simple terms, the larger the model, the more data, and the stronger the computing power, the better the performance. This seemingly simple rule has been the core driving force behind the rapid development of AI over the past few years.

The seventeenth is pretraining. Pretraining is the process of conducting self-supervised learning on large-scale unlabeled data. By predicting the next word, the model learns linguistic patterns and world knowledge from massive amounts of text. Pretraining is the stage in which the model receives its "general education" ; it is extremely costly and can usually be undertaken only by large institutions.

5.2 Fine-Tuning, RLHF, and Few-Shot Learning

The eighteenth is fine-tuning. Fine-tuning is based on a pretrained model and uses task-specific data to pro···

perform targeted training. For example, fine-tuning a general-purpose model with a medical literature dataset can produce a specialized model that performs better in the medical field.

The nineteenth is RLHF, that is, reinforcement learning from human feedback. RLHF optimizes model behavior through human preference data. The specific process is: have the model generate multiple answers, let human annotators rank them, then use these preference data to train a reward model, and finally use a reinforcement-learning algorithm to optimize the main model. RLHF is a key technology that enables a model to move from "being able to speak" to "knowing how to speak" ; it makes the model' s responses more consistent with human expectations.

The twentieth is few-shot learning. Few-shot learning is the ability of a model to adapt quickly to new tasks using only a small amount of labeled data. In practical use, when we give the model a few examples in the prompt, it can follow them and do the task accordingly; this is an expression of few-shot learning.

The above are the 20 core concepts of AI technology. These concepts constitute the basic framework for our understanding of AI technology. The tools and applications discussed later are all built on these concepts.

III. General AI Tools (10:35-11:05)

6. Large Language Model Application Tools—Taking DeepSeek as an Example

Having finished the concepts, let us look at specific tools. Taking DeepSeek as an example: it is currently one of the most widely used large language model applications in China. I will focus on three aspects.

6.1 Core Differences Between Fast Mode and Expert Mode

DeepSeek has two modes: fast mode and expert mode. Fast mode is positioned for everyday dialogue and instant response. Its core advantages are fast response and support for multimodal file uploads; up to 50 files can be uploaded, and the model architecture is a lightweight optimized version. Expert mode is positioned for complex problems and deep reasoning. Its core advantages are rigorous chains of reasoning and strong long-text processing capability; it adopts a V3.2 plus R1 fusion architecture.

The recommendation for choosing is simple: use Fast Mode for everyday quick tasks, such as simple queries, chatting, and processing images and PDF files;

use Expert Mode for in-depth professional tasks, such as complex mathematics, programming, and logical analysis. After the April 2026 update, Expert Mode also supports file uploading and intelligent search. The functional gap between the two modes is narrowing, but differences in reasoning depth remain.

6.2 Collaborative Use of Intelligent Search and Document Uploading

DeepSeek' s intelligent search function is an important feature. It is not simple web scraping, but real-time information retrieval with contextual awareness. Once enabled, DeepSeek will actively analyze the question, determine whether it needs to go online to obtain the latest information, and seamlessly integrate the information found through search into its reasoning chain.

Even more powerful is that document uploading and intelligent search can be used at the same time. This means that AI can simultaneously process your private documents and online search results, providing integrated insights. For example, you can upload a company annual report while enabling intelligent search to obtain the latest industry research reports, allowing the AI to perform cross-comparison and effectively distinguish the authenticity of information. You can also upload a full-length novel and use intelligent search to call up relevant literary criticism, thereby obtaining a more comprehensive analysis of the work.

In my actual work, I once uploaded a series of documents on "open science" while enabling intelligent search, allowing the AI to combine the latest information from the internet with China' s participation in UNESCO' s definition of open science, and to analyze the gains, losses, and measures in China' s open-science work. This capacity to integrate "private knowledge + real-time information" is an important aid to literature and intelligence work.

6.3 Report Generation and Typeset Output

DeepSeek can also directly generate Word documents with formatting. By specifying formatting requirements in the prompt—for example, using SimHei size 3 for the main title, bold KaiTi size small 3 for subtitles, and FangSong size 4 for the body text—and then generating a Word document in HTML format with a download button. This function is highly practical for literature and intelligence workers who need to write reports regularly.

In practice, a complex prompt can be written in full: first specify the search task, then the analytical framework, then the document structure, and finally the formatting requirements. After the AI completes the task, download the report, and then revise it manually or with AI assistance. The entire process—from research to drafting—is far more efficient than traditional methods.

7. AI Agents (Agent) and Workflows

7.1 Seven Core Design Patterns for AI Agents

AI agents are a frontier direction in current AI applications. There are seven core design patterns that need to be understood:

First, **Prompt Chaining**: decompose a complex problem into multiple sub-problems and solve them step by step, thereby avoiding context overload and hallucination problems in large models.

Second, **Routing**: dynamically select the most appropriate tool or sub-agent to handle a user query through conditional logic.

Third, **Parallelization**: identify subtasks that have no dependency relationships and execute them concurrently to improve efficiency.

Fourth, **Reflection**: adopt a "producer–critic" model to evaluate and revise generated results.

Fifth, **Tool Use**: agents obtain real-time data or perform operations by invoking external tools, such as database queries and API calls.

Sixth, **Planning**: decompose complex tasks into multi-step workflows and dynamically adjust the execution path.

Seventh, **Multi-Agent**: assign complex tasks to multiple specialized agents to complete collaboratively.

These seven patterns are not isolated; in practical applications, they are often used in combination. For example, a scientific research

As an auxiliary intelligent agent, it may first decompose the research task using a planning mode, then execute it step by step with a chain of prompts, use a reflection mode for self-checking in the middle, and finally invoke a database through a tool-use mode to retrieve literature.

7.2 Differences Between AI Agents and Agentic AI

Here, two concepts need to be distinguished: AI Agents and Agentic AI.

AI Agents are single-entity systems driven by large language models and used to complete specific tasks. Their features are autonomy, task specificity, responsiveness, and adaptability. Application scenarios include customer support, meeting scheduling, data summarization, and so on.

Agentic AI is a system composed of multiple specialized agents that collaborate to achieve complex goals. Its features are goal decomposition, distributed collaboration, and dynamic strategy adjustment. Application scenarios include automation of scientific research, coordination of robot swarms, advanced medical decision support, and so on.

From the perspective of technological evolution, generative AI laid the technical foundation for AI Agents, but it lacks autonomy and the ability to interact with the external environment. By introducing architectural layers such as tool-calling APIs, memory buffers, and reasoning chains, AI Agents have achieved a transformation from passive content generation to active task execution. Agentic AI further develops into a multi-agent system; through the integration of specialized agents, advanced reasoning and planning, persistent memory architectures, and orchestration layers, it addresses the problem of collaborative execution of complex tasks.

The key differences between the two lie in the following: in terms of scope, AI Agents focus on narrow tasks, whereas Agentic AI handles complex open-ended problems; in terms of architecture, AI Agents have a single-agent architecture, whereas Agentic AI has a multi-agent architecture; in terms of collaboration mode, AI Agents rely on individual intelligence, whereas Agentic AI relies on collaborative orchestration.

Of course, both face challenges. AI Agents lack causal reasoning capabilities and inherit the hallucination tendencies and contextual sensitivity of large language models. The causal-reasoning capability of Agentic AI is deficient

losses are amplified, leading to task redundancy and error propagation; system debugging and responsibility tracing become difficult; and unified architectural standards and communication protocols are lacking.

7.3 Eight Memory-Management Strategies

Memory management for intelligent agents is a core technical issue. There are eight common strategies:

First, **full memory**: record all dialogue content and provide the complete historical context at each inference step. This is simple, but the context length is limited and the cost is high.

Second, **sliding window**: retain only the most recent several rounds of dialogue, while content beyond that range is forgotten. This is simple and efficient, but it is highly prone to forgetting.

Third, **relevance filtering**: retain key information based on importance scoring and discard unimportant content. This is intelligent, but the scoring mechanism is complex.

Fourth, **summary compression**: generate summaries of the dialogue content and store the summaries in place of the original content. This saves space, but details may be omitted.

Fifth, **vector database**: vectorize and store the dialogue content, then retrieve relevant memories through semantic search. Its semantic retrieval capability is strong, but the computational cost is high.

Sixth, **knowledge graph**: structurally store entities, relations, and other elements in the dialogue, and obtain memories through graph queries and reasoning. This supports complex reasoning, but construction costs are high.

Seventh, **hierarchical memory**: combine short-term memory and long-term memory, and store important information in long-term memory through a promotion mechanism. This is flexible, but complex to implement.

Eighth, **OS-like memory management**: simulate the memory management of an operating system by "swapping" information that exceeds the context into external storage and "swapping it back" when necessary. The structure is clear, but well-designed triggering and splicing logic are required.

These eight strategies each have advantages, disadvantages, and applicable scenarios, and developers can choose according to actual needs. In literature and information work, the two strategies of vector databases and knowledge graphs particularly merit attention.

attention, because they are directly related to the core businesses of knowledge organization and information retrieval.

8. Agent Deployment Platforms and Tool Ecosystem

8.1 OpenClaw: An Open-Source AI Agent Gateway

OpenClaw is currently the most closely watched open-source AI agent platform. It was once named Clawdbot, was later renamed Moltbot, and was then renamed OpenClaw because of a trademark dispute with Anthropic.

In essence, it is an open-source, "agentic AI" that can be deployed locally or in the cloud. It uses large models such as Claude and ChatGPT as the brain, directly controlling the computer—including the mouse, keyboard, file system, and browser.

OpenClaw turns large models into "hands-on" digital employees. Through chat software, it can operate an entire computer on your behalf. Its highlights include: fully open source, no vendor lock-in, and the ability to run offline; system-level permissions, enabling it to send emails, manage calendars, and manipulate files; the ability to issue commands remotely through everyday chat software; and persistent memory, where long-term context makes responses more personalized.

Typical use cases include: automatically cleaning up mailboxes, summarizing unread emails, and handling check-in tasks; checking tickets, comparing prices, and booking restaurants with a single sentence; batch-organizing download directories, automatically renaming invoices and archiving them; reading health data to generate daily reports; and, for developers, automating debugging, GitHub management, cross-platform script migration, and so on.

However, risks must be emphasized: OpenClaw has full control over the computer, so the potential security risk is extremely high. For initial trials, priority

should be given to an old computer, a cloud host, or a sandbox environment; do not hand over the main machine directly.

8.2 Building Workflows with Kouzi (Coze)

Kouzi is a workflow platform launched by ByteDance, suitable for building structured AI processing workflows. The creation process is: create a new project → create a blank application → configure the workflow → add nodes.

nodes. For each node, large-model tasks, plug-in functions, and input/output parameters can be configured. After configuration is complete, the "trial run" function can be used for testing and log inspection, and finally a web interface can be added and published.

The advantage of Coze is that it is visual and has a low threshold, making it suitable for users who do not write code to quickly build AI workflows. In literature and information scenarios, Coze can be used to build an automated process of "literature abstract generation → keyword extraction → classification and archiving."

8.3 iFlow, MaxClaw, AstronClaw, WorkBuddy, and other platforms

In addition to OpenClaw and Coze, several other platforms are also worth attention:

iFlow platform: It requires first installing the Node.js environment, then installing through the command line. After selecting an API Key, it can be used. It supports multiple large models such as DeepSeek, GLM, and Qwen.

MaxClaw (MiniMax): It provides a cloud-based service for using OpenClaw. Combined with the "Exploration Expert" Skill, it can enable multiple agents to handle complex problems, such as generating interactive quiz papers.

AstronClaw (iFlytek Xingchen): A one-click deployment solution based on OpenClaw, emphasizing that "no API or environment configuration is required; clicking one-key launch completes all configuration." Deployment can be completed in just over one minute. It comes with a large number of preset skills, has top domestic open-source models built in, and uses Sandbox isolation to ensure data security.

WorkBuddy (Tencent WeChat): A desktop agent launched by Tencent, compatible with OpenClaw Skills, supporting local operation, and able to connect to enterprise office tools such as WeChat, Feishu, and DingTalk. This marks a further step in the implementation of AI Agents in enterprise office scenarios.

Longxia Guanjia (Tencent): A security protection solution integrated into Tencent PC Manager, providing OpenClaw with functions such as Skills security detection, file security protection, network access protection, system security protection, and operation log recording. It moves from the two extremes of "no protection at all" or "complete isolation"

···extreme, shifting toward more reasonable "selective openness" permission management.

8.4 Local Deployment of OpenClaw and Integration with Ollama

For colleagues who need local deployment, OpenClaw can be integrated with Ollama to achieve fully localized, free "Token freedom."

Deployment is divided into four stages: in the first stage, install Ollama and configure the model storage path; in the second stage, download and run a large model that supports tool-calling capabilities, such as qwen3.5; in the third stage, modify the OpenClaw configuration file and set Ollama as the model provider; in the fourth stage, adjust the timeout period and restart the service.

The core value lies in: cost control—completely eliminating Token consumption; data security—all data and computation processes remain local; network independence—true offline availability. However, note that the model must support tool-calling capabilities in order to integrate properly with OpenClaw; this is a key screening criterion when selecting a model.

IV. AI Capabilities and Boundaries (11:05-11:35)

9. The Capability Boundaries and Technical Limits of AI

9.1 Theory of Research Stratification and the Positioning of AI Applications

To understand the capability boundaries of AI, one must first understand the theory of research stratification. Scientific research can be divided into three levels:

Invention (0 to 1): This is the most difficult original breakthrough—creating something new from nothing. AI currently cannot replace this stage, but it can assist with literature research and stimulate thinking.

Expansion (1 to 10): Further extending and optimizing the results of an invention, turning a rough prototype into a practical product. AI can play a relatively large role at this stage, for example in parameter optimization and experiments

design assistance.

Application (10 to 100): broadly apply research results and transform them into practical applications. AI has its greatest role at this stage, assisting with paper writing, data analysis, visualization, report generation, and more.

AI also has application value in academic conferences: before the conference, it can be used to analyze the conference handbook, screen reports worth attention,

and plan a listening route; during the conference, it can conduct in-depth background research on speakers of interest and generate "rapid conference briefing cards," including academic profiles, core tools, and preset questions; based on research-report presentation topics, it can analyze whether a speaker's techniques can address pain points in one's own research.

However, researchers must be clear: AI is an assistive tool, not a substitute. Researchers should return to deep thinking about academic knowledge and, through the approach of "going slowly in order to go far," improve research quality and innovation capacity. AI can help you improve efficiency, but it cannot help you generate insight.

9.2 Token Consumption Mechanism and Cost Optimization

A token is the "information building block" of a large language model. The model generates coherent text by computing relationships among tokens. Large-language-model companies charge by the number of tokens, because the more tokens processed, the greater the computation required and the more computing power consumed.

Many people mistakenly think AI charges by the number of conversation rounds, but the actual consumption is far higher than that. Before every conversation, the complete memory must be loaded—including user settings, historical conversations, and so on. These memories are stored in the `memory.md` file and continually accumulate. When the memory reaches hundreds of thousands of characters, even a simple "hello" will trigger traversal of a massive amount of memory, causing token consumption to surge.

There are three key techniques for reducing costs:

First, make good use of slash commands to interact directly with the program, consuming no tokens at all. `/new` starts a new conversation and clears the context; `/restart` restarts the program; `/stop` immediately stops the current task; `/compress` compresses memory, enabling the AI to "remember the big things and forget the small things."

Second, if a task can be handled with a program, do not use a large model for it. Hand repetitive, procedural, and programmable tasks over to scripts, and let the large model focus on core mental work such as judgment, reasoning, and creation. Checking email, querying the weather, pulling data, scheduled checks, and the like can all be handled in advance by scripts; call the large model only after results are available. AI is your brain, not your hands. Let it think; do not make it do manual drudgery.

Third, use different models for different tasks. Although top-tier models are powerful, they are expensive. Many simple tasks can be handled adequately by domestic large models, at only a few dozenths of the cost of top-tier models. Use top-tier models for complex tasks such as writing code and deep creative

work; use inexpensive models for repetitive or simple tasks such as compiling briefings, conducting information research, and generating documents.

For the same tool, different ways of using it may lead to a tenfold difference in cost. High expenses often arise from making AI do too many things it "should not be doing."

9.3 Vector Databases and Knowledge Retrieval

Vector databases are a core technology for AI knowledge retrieval. They are database systems specifically designed for efficiently storing, indexing, and retrieving high-dimensional vector data. They can transform unstructured data —text, images, audio, and so on—into vectors through embedding models, and enable fast similarity search.

The technical features of vector databases include: efficient storage and indexing (using advanced indexing algorithms such as HNSW and IVF-PQ); approximate nearest-neighbor search (quickly finding the closest set of vectors in massive datasets); support for multimodal data; hybrid retrieval capabilities (combining structured fields with vector similarity search); distributed architectures that support horizontal scaling; and real-time updates and incremental synchronization.

Compared with traditional databases: traditional databases store structured data in tabular form and are based on exact matching and conditional filtering; vector databases store vector representations of unstructured data, bas

Similarity is computed based on the distance between vectors. Traditional databases are suitable for transaction processing and data analysis, while vector databases are suitable for recommendation systems, semantic search, and multimodal integration.

For literature-information work, the significance of vector databases lies in the fact that they provide a technical foundation for implementing semantic-level literature retrieval. Traditional keyword matching can only find documents that contain specific terms, whereas vector retrieval can find documents that are semantically similar but use different wording. This is of revolutionary significance for cross-language retrieval and cross-domain discovery.

10. Prompt Engineering and Skill Development Practice

10.1 Seven Prompt-Questioning Formulas

Prompt engineering is a core skill for using AI tools. There are seven practical questioning formulas:

First, state the background and specific requirements: "Who I am + what I want to do + what conditions must not be exceeded." For example: "I am a library administrator planning to purchase 10 e-book readers for in-library

lending, with a budget ceiling of 5,000 yuan. Could you recommend several cost-effective products?" Tell AI your identity, task, and constraints, and it will be able to make precise recommendations.

Second, break the problem down and proceed in multiple steps: "what to do first → what to do next → what to do last." Split a large problem into small steps, and the AI' s answer will be clearer.

Third, specify formatting and template constraints: directly state the formatting requirements, such as requiring a table with three columns, so the AI will not improvise randomly.

Fourth, obtain suggestions in specific scenarios: request advice within a concrete scenario rather than asking in a vague and general way.

Fifth, explain professional concepts with everyday analogies: "use everyday things as comparisons." For example, use a "library book-borrowing system" to explain cloud computing, turning abstract concepts into examples close at hand.

Sixth, set money-saving and time-saving constraints: "budget/time + desired effect + must not exceed how much."AI will prioritize options that fit the budget and time constraints.

Seventh, help you identify errors through reverse validation: "let AI play the nitpicking critic and find loopholes." Have the AI pretend to be a senior review expert, point out the aspects of your plan that may cause problems, and help you avoid pitfalls in advance.

10.2 The Three-Layer Architecture and Iterative Optimization of Agent Skill

Agent Skill is the encapsulation of skills for AI agents. Its core innovation lies in a three-layer structure that is loaded on demand:

The first layer is the metadata layer (Meta), which is always loaded. It uses only two lines of text to describe the skill name and purpose, saving Tokens.

The second layer is the instruction layer (Instruction), which loads the complete process only when triggered, enabling precise matching.

The third layer is the resource layer (Resources), which includes reference materials, scripts, and assets. These are called only when needed, avoiding interference.

This design solves the problem of "overly long prompts causing Token waste and AI confusion." To use the analogy of "a chef cooking a dish" : the process is the cooking steps; the recipe is the temperature and ingredient ratios; the tools are the wok, spatula, and stove; and the materials are the ingredients and secret sauces. Corresponding to AI: `skill.md` is the process and recipe, `references`

are reference materials, `scripts` are executable scripts, and `assets` are asset resources.

A Skill is not simply a prompt. Although at the underlying level it is indeed a prompt, the essence of a Skill lies in the three-layer architectural design of "on-demand loading." Its value does not lie in technical complexity, but in transforming personal experience into reusable automated workflows—handing over the three to five most frequent tasks to AI, thereby achieving an efficiency leap in which "one sentence completes half an hour of work."

Practical advice: do not blindly download ready-made Skills from the internet. Instead, turn your own high-frequency repetitive work—writing weekly reports, preparing lessons, reviewing contracts, creating illustrations for articles, and so on—into Skills. A Skill must be based on a methodology that you yourself have validated, rather than on a generic template.

Skill development requires iterative optimization. The first version is almost never perfect; it requires a cycle of "create-test-analyze-optimize." The key to testing lies in: a single optimization objective (optimize only one dimension at a time), concrete evaluation criteria (do not say "good quality,"but rather "whether it matches my writing style"), and a unified testing method (run multiple tests with the same input). A good Skill usually requires 5 to 10 iterations, but with tools, this process can be compressed from several days to a few hours.

10.3 Security Protection: Skill Vetter and Lobster Butler

The security issues of AI agents must not be ignored. The National Computer Network Emergency Response Technical Team/Coordination Center of China (CNCERT) has issued a risk warning, noting that OpenClaw' s default security configuration is weak, and that malicious Skills can steal keys and deploy Trojans.

A real case is alarming: in the official ClawHub store, all 314 Skills published by the user hightower6eu were found upon inspection to be malicious plugins; not a single one was harmless. These plugins disguised themselves as normal tools such as cryptocurrency analysis and financial tracking, while in fact downloading and executing unknown code in the background.

The solution is to install Skill Vetter—a "security-auditing Skill" that functions similarly to antivirus software. Before installing any Skill, it automatically performs an audit, generates a security report, and checks the risk level, permission scope, and suspicious code patterns. The usage method is to set it as a mandatory pre-installation review, so that all Skills must pass inspection before they can be installed.

Tencent' s Lobster Butler provides more comprehensive security protection: Skills security detection (blocking malicious plugins), file security protection (specifying protected areas), network access protection (preventing port expo-

sure), system security protection (preventing unauthorized system modifications), and operation logging (fully recording all operations for rollback).

Security principles: do not blindly trust download counts—high download volume does not mean safety; security auditing is essential; test high-risk Skills in an isolated environment. If you are an OpenClaw user, be sure to install Skill Vetter as a mandatory security-audit tool before installing any Skill.

11. Laws, Regulations, and Research Ethics

11.1 The *Science and Technology Progress Law* and the *Interim Measures for the Administration of Generative Artificial Intelligence Services*

When using AI tools, one must understand the relevant laws and regulations.

Article 104 of the *Science and Technology Progress Law of the People' s Republic of China* clearly stipulates that no organization or individual may fabricate or falsify scientific research achievements; publish or disseminate false scientific research achievements; or engage in the buying and selling, ghostwriting, or proxy-submission of academic papers, their experimental research data, application and acceptance materials for science and technology plan projects, and the like. Article 112 sets out the legal consequences for violating these provisions, including warnings, public criticism through notification, fines, confiscation of illegal gains, and, in serious cases, revocation of permits or licenses.

The *Interim Measures for the Administration of Generative Artificial Intelligence Services* define generative artificial intelligence as models and related technologies with the capacity to generate content such as text, images, audio, and video. They clarify the definitions of service providers and service users, encourage the innovative application of generative artificial intelligence technologies across various industries and fields, and support collaboration among industry organizations, enterprises, educational and research institutions, public cultural institutions, and others in technological innovation, data-resource development, application transformation, and risk prevention.

As workers in literature and information services, we are users of generative artificial intelligence services, and we may also become service providers within knowledge-service platforms. We therefore need to pay attention simultaneously to the responsibilities and obligations of both sides.

11.2 The *Opinions on Strengthening the Governance of Science and Technology Ethics* and the *Guidelines for Responsible Research Conduct (2023)*

The *Opinions on Strengthening the Governance of Science and Technology Ethics* require guiding scientific and technological personnel to consciously comply with the requirements of science and technology ethics, strengthen their

awareness of science and technology ethics, consciously practice the principles of science and technology ethics, and uphold the bottom line of science and technology ethics. In particular, they mention the need to formulate science and technology [[unclear: text continues beyond the visible page]] in key fields such as life sciences, medicine, and artificial intelligence.

technical ethics norms, guidelines, and the like.

The *Guidelines for Standards of Responsible Research Conduct (2023)* set out explicit requirements for the use of generative artificial intelligence:

First, for content generated using generative artificial intelligence—especially key content involving facts, viewpoints, and the like—the generation process should be clearly labeled and explained, so as to ensure truthfulness and accuracy and to respect the intellectual property rights of others.

Second, where other authors have already labeled content as having been generated by artificial intelligence, it should generally not be cited as an original source; if citation is indeed necessary, an explanation should be provided.

Third, generative artificial intelligence must not be listed as a co-contributor to research outcomes. The main ways and details of using generative artificial intelligence should be disclosed in relevant sections such as the research methods or appendices.

Fourth, where generative artificial intelligence is used in review activities, prior consent should be obtained from the organizers of the review activity, and measures should be taken during operation to prevent leakage of review content.

These norms are directly related to our day-to-day work in literature and information services. In particular, when using AI in research evaluation, academic search, and knowledge services, they must be strictly observed.

11.3 *Boundary Guidelines for the Use of AIGC in Academic Publishing* and Research Integrity

The *Boundary Guidelines for the Use of AIGC in Academic Publishing* reveal a serious problem: improper use of AIGC can increase the "productivity" of paper mills and accelerate academic misconduct. Specific manifestations include:

Fabricated research—presenting research as having been conducted when none was actually carried out; for example, in the field of biomedicine, misconduct such as fabricated immunoblots, duplicated images, or errors in experimental reagent sequences may occur.

Tortured phrases—unusual combinations of words, possibly produced by using AI software to rewrite plagiarized text in order to avoid detection.

Cross-language plagiarism—directly translating text from other languages without proper citation.

Content generated from standardized templates—causing papers to contain sections that would not originally have appeared.

Selling co-author slots and author slots—marketed at the accepted-manuscript stage, with ghostwriters behind the scenes writing customized papers for the "authors."

Undermining the integrity of peer review—manipulating the peer-review process and colluding with journal editors or special-issue guest editors.

Citation rings—authors pay to purchase citations.

These problems pose new challenges for literature and information work. In literature retrieval, citation analysis, and research evaluation, we need to be alert to these new forms of academic misconduct spawned by AI. Identifying awkward phrasing, detecting cross-language plagiarism, discovering citation rings, and the like all require us to update our tools and methods.

V. Summary and Outlook (11:35–11:40)

12. Summary and Outlook

12.1 Review of Key Points

Looking back over today' s 100 minutes, we covered four parts:

First, the core concepts of AI technology. Starting from 20 core concepts such as artificial intelligence, AGI, complex systems, and emergent capabilities, we built a systematic understanding of generative artificial intelligence. The key point is: large language models are complex systems based on the Transformer architecture; they acquire capabilities through pretraining and fine-tuning, and their capabilities originate from emergent phenomena driven by scaling laws.

Second, general-purpose AI tools. We introduced DeepSeek' s fast mode and expert mode, intelligent search and document collaboration, and report-generation functions; seven agent design patterns and eight memory-management strategies; deployment platforms such as OpenClaw, Coze, Xinliu, MaxClaw, AstronClaw, and WorkBuddy; as well as connecting OpenClaw with Ollama to implement localized deployment.

the solution.

Third, AI capabilities and boundaries. We discussed the positioning of AI within the theory of stratified scientific research; token consumption and cost optimization; the retrieval capabilities of vector databases; the seven prompt-questioning formulas and the three-layer architecture and iterative method of Agent Skills; Skill Vetter and Lobster Butler for safety protection; as well as laws, regulations, and ethical norms such as the *Law on Scientific and Technological Progress*, the *Interim Measures for the Administration of Generative Artificial Intelligence*

Services, the *Opinions on the Governance of Science and Technology Ethics*, the *Guidelines for Responsible Research Conduct*, and the *Guidelines on the Boundaries for the Use of AIGC in Academic Publishing*.

12.2 Implications and Prospects for Literature Intelligence Work

For those of us engaged in literature intelligence work, I have several prospects to offer:

First, embrace it proactively. AI is not a threat; it is a tool. Literature intelligence professionals are naturally endowed with the capacity for information organization and knowledge services, and this is precisely the human wisdom that AI applications need most. Learning to use AI tools is not an elective; it is a required course.

Second, maintain clear awareness. AI has capability boundaries: it can produce hallucinations, fabricate literature, and generate content that appears plausible but is not. Our professional value lies in identification, verification, and organization—these are things AI cannot replace. Vector databases can enable AI to conduct semantic retrieval, but evaluating the quality of retrieval results still requires professional judgment.

Third, use it compliantly. Laws, regulations, and research ethics are the bottom line. Content generated with AI must be labeled; AI must not be listed as the person who completed the research results; and the use of AI in papers requires consent. Literature intelligence professionals should, even more, become models of compliant use.

Fourth, keep learning. AI technology is iterating extremely rapidly; today' s tools and methods may be outdated in half a year. But the core concepts are stable—once one understands foundational knowledge such as Transformer, tokens, vector retrieval, and agent design patterns, one can quickly adapt to new tools.

Fifth, return to the essence. Technology is a means, not an end. The essence of literature and information work lies in knowledge organization, knowledge discovery, and knowledge services. AI can help us complete these tasks more efficiently, but the deep thinking embodied in "slowing down in order to go farther," the judgment of academic value, and insight into user needs will always remain our core competitiveness.

In the subsequent courses, you will also learn how OpenClaw empowers literature retrieval, literature summarization and analysis, data collection and processing, data-analysis modeling and visualization, assistance in paper writing, custom Skills development, and related topics. Today' s general report is the foundation for these hands-on courses. I hope that, on the basis of understanding the concepts, everyone will boldly experiment and carefully verify in the

subsequent practical sessions, and truly put AI tools to use—using them well and using them safely.

Thank you all.

(End of full text)

Note: Figure translations are in progress. See original paper for figures.