COLLEAGUE.SKILL: Automated AI Skill Generation via Expert Knowledge Distillation

Tianyi Zhou* **Dongrui Liu*** **Leitao Yuan** **Jing Shao** **Xia Hu**
Shanghai Artificial Intelligence Laboratory
{zhoutianyi, liudongrui, yuanleitao, shaojing, huxia}@pjlab.org.cn

Abstract

Automated skill generation—the ability to distill human expertise into structured, callable AI skills—is an emerging paradigm under open standards such as AgentSkills. We apply this paradigm to a critical knowledge-management problem: much of an expert's expertise—technical standards, decision patterns, institutional context, and communication style—exists as tacit knowledge that is difficult to transfer through conventional documentation. We present COLLEAGUE.SKILL (`https://github.com/titanwings/colleague-skill`), an open-source skill that converts an expert's digital traces (chat logs, documents, emails, screenshots) into a persistent, callable AI skill. The generated skill faithfully reproduces both the expert's work capabilities (technical standards, code review criteria, security practices, system ownership, workflows) and their persona (communication style, decision logic, interpersonal behavior) through a structured two-part architecture. COLLEAGUE.SKILL supports automatic data collection from Feishu, DingTalk, and Slack, and includes an incremental evolution mechanism that improves the skill over time via new material or conversational corrections.

1 Introduction

The emergence of AI coding agents has given rise to a new abstraction: the *skill*—a structured, callable unit that encapsulates domain expertise and can be invoked on demand within an agent framework. Open standards such as AgentSkills formalize this abstraction, enabling portable, composable skill definitions across platforms. This paradigm opens the door to *automated skill generation*: rather than hand-authoring skills, we can mine and distill them from existing digital artifacts—effectively performing *domain knowledge mining* over unstructured organizational data.

A natural and high-value application lies in capturing the expertise of individual team members. In any software organization, a significant portion of practical knowledge—coding conventions, review criteria, security practices, debugging heuristics, and communication norms—is tacit. It is embedded in chat histories, code review comments, internal documents, and daily interactions rather than in any formal specification. This latent domain knowledge represents a rich but largely untapped source for skill construction. Conventional knowledge transfer methods (handover documents, wiki pages, onboarding sessions) capture only the explicit surface, leaving the nuanced judgment, decision heuristics, and behavioral patterns buried in scattered digital traces.

Large language models can approximate human expertise, but generic models lack the specificity required to faithfully replicate an individual's working style and technical standards. COLLEAGUE.SKILL addresses this gap by combining structured domain knowledge extraction with persona modeling. Given an expert's digital traces—chat logs from Feishu, DingTalk, or Slack, internal documents, emails, and screenshots—the system automatically mines domain-specific patterns and distills them into a two-part AI skill that encodes both *what* the expert knows and *how* they communicate. The resulting skill can be invoked interactively, operating with the target expert's vocabulary, decision criteria, and interpersonal behavior.

*Project co-lead

2 System Design

2.1 Two-Part Architecture

COLLEAGUE.SKILL generates a two-part skill for each expert:

Part A — Work Skill. Encodes domain-specific technical knowledge: the systems and services the expert owned, their coding conventions (naming, style, review criteria), API design standards, security review practices (input validation, sensitive data masking, injection prevention), incident-handling procedures, and a curated experience knowledge base (e.g., "always encrypt user IDs before external exposure" or "never expose auto-increment primary keys").

Part B — Persona. Models the expert's behavioral profile through a five-layer structure:

- **L1. Hard rules** — Inviolable behavioral constraints that override all other layers.
- **L2. Identity** — Role, company, level, MBTI, and corporate culture affiliation.
- **L3. Expression style** — Catchphrases, sentence length, response latency patterns, emoji usage, and templated dialogue examples.
- **L4. Decision & judgment** — Priority ordering (e.g., data > feasibility > business logic), conditions for pushing back, and strategies for declining requests.
- **L5. Interpersonal behavior** — Differentiated conduct toward superiors, peers, and junior experts, plus behavior under pressure.

At runtime, the skill follows the pipeline: *receive task* → *Persona decides attitude* → *Work Skill executes* → *output in their voice*.

Figure 1 illustrates the end-to-end automated construction pipeline.

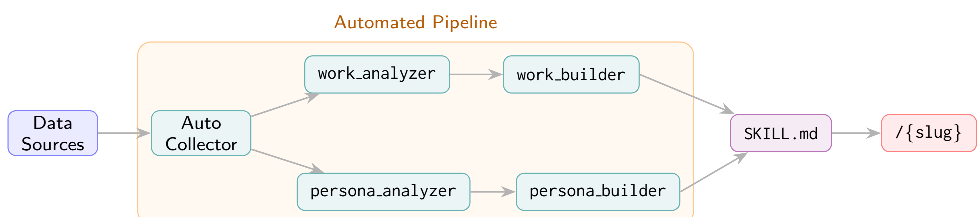

Figure 1: End-to-end pipeline of COLLEAGUE.SKILL. Source material is automatically collected from enterprise platforms (Feishu, DingTalk, Slack, etc.), analyzed in parallel along Work and Persona tracks, and compiled into an invokable `SKILL.md`. All data is processed and stored *locally*; no raw content leaves the user's machine.

2.2 Data Collection Pipeline

COLLEAGUE.SKILL accepts source material through multiple channels, summarized in Table 1.

Table 1: Supported data sources.

Source	Method	Messages	Docs
Feishu	Full API auto-collect	✓	✓
DingTalk	Browser + API	(browser)	✓
Slack	Bot API	✓	—
WeChat	SQLite export	✓	—
Email	`.eml`/`.mbox` parser	✓	—
PDF / Image	Claude native read	✓	✓
Paste / MD	Direct input	✓	✓

For Feishu, the collector authenticates via tenant or user access tokens and automatically retrieves group chat messages sent by the target, private conversation history, and all documents and wikis authored or edited by the target. DingTalk falls back to browser automation for message history due to API restrictions. All collected material is persisted locally under `knowledge/{slug}/` before analysis.

2.3 Skill Generation and Evolution

Generation. The intake prompt collects three lightweight inputs from the user: an alias, a one-sentence professional profile (company, level, role), and a personality sketch (MBTI, tags, cultural affiliation). These, combined with collected source material, are passed through two parallel analysis prompts—`work_analyzer` and `persona_analyzer`—and then rendered via `work_builder` and `persona_builder` into structured Markdown files. The final `SKILL.md` is a merged document that can be invoked directly as `/{slug}` within Claude Code.

Incremental evolution. The skill supports three update pathways:

- **Append new material**: Additional files or messages are analyzed for *incremental* information only; new conclusions are merged without overwriting prior ones.
- **Conversational correction**: The user can say *"he wouldn't do that, he should be X"*, which the `correction_handler` prompt converts into a typed correction record appended to the skill file with immediate effect.
- **Version control**: Every update archives the previous version via `version_manager.py`, enabling rollback to any historical state.

3 Usage

COLLEAGUE.SKILL is installed as a Claude Code skill and invoked with `/create-colleagues`. After creation, the expert is accessible via `/{slug}` (full skill), `/{slug}-work` (work capabilities only), or `/{slug}-persona` (persona only). Management commands include `/list-colleagues`, `/colleague-rollback`, and `/delete-colleague`.

The following exchanges illustrate the skill in action, generated from a ByteDance L2-1 backend engineer profile tagged as *security-conscious* and *impact-driven*:

Scenario 1: Code Review

```
User    ▷ Can you review this API design?
expert  ▷ The user ID in the response is a raw auto-increment PK --- encrypt it before
        exposing.  Phone number needs masking.  Also, N+1 query, fix it.  Use {code,
        message, data}. That's the spec.
```

Scenario 2: Security Review

```
User    ▷ This endpoint fetches a URL the user provides.
expert  ▷ That's a textbook SSRF. Whitelist the allowed domains, block internal IPs,
        and don't follow redirects. No exceptions.
```

4 Conclusion

COLLEAGUE.SKILL demonstrates that structured knowledge extraction combined with layered persona modeling can produce AI skills that credibly replicate an individual expert's working style. By combining automatic data collection from enterprise communication platforms with incremental evolution mechanisms, the system provides a practical path for preserving and reusing institutional knowledge across teams. COLLEAGUE.SKILL is released under the MIT License.