JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 31 No. 6, pp. 161-167, June 2026
https://doi.org/10.9708/jksci.2026.31.06.161

A Comparative Analysis of Skill Design Types in OpenClaw-Based Korean Office Task Automation

Youngmi Baek*, Jung Kyu Park*

*Professor, Division of AI Convergence, Major in Computer Science, Daejin University, Pocheon-si, Korea

[Abstract]

This paper examines how different skill design types affect task performance in a Korean office-task automation environment based on OpenClaw. We compare four SKILL.md design types: procedural, constraint, example, and checklist. Seven benchmark tasks are constructed, including reply drafting, identification of information to confirm, checklist generation, and structured summarization. The outputs are evaluated in terms of format compliance, inclusion of key information, handling of uncertainty, and avoidance of unsupported details. The results show that OpenClaw performs stably and accurately across most tasks. Differences among the four skill design types are generally small. Performance drops appear mainly in tasks involving scheduling uncertainty. Overall, task structure and uncertainty have a greater influence on performance than the instruction organization strategy of the skill template.

▸ **Key words**: OpenClaw, Korean office task automation, LLM agent, skill design, benchmark evaluation

[요 약]

본 논문에서는 OpenClaw 기반 한국어 업무자동화 환경에서 스킬 설계 유형이 과업 수행에 미치는 영향을 분석한다. 이를 위해 procedural, constraint, example, checklist의 네 가지 SKILL.md 설계 유형을 비교하였다. 회신 초안 작성, 확인 필요 정보 정리, 체크리스트 생성, 구조화된 요약 등 7개의 벤치마크 과업을 구성하였다. 출력은 형식 준수, 핵심 정보 반영, 불확실 정보 처리, 입력 외 정보 생성 여부를 기준으로 평가하였다. 실험 결과, OpenClaw는 대부분의 과업에서 안정적이고 높은 성능을 보였다. 스킬 설계 유형 간 차이는 전반적으로 크지 않았다. 반면 일정 조율과 같이 불확실한 정보를 포함한 과업에서는 입력에 없는 세부사항을 보완적으로 생성하는 경향이 일부 나타났다. 이러한 결과는 스킬 템플릿의 지시 구성 방식 자체보다 과업의 구조화 수준과 정보의 불확실성이 수행 결과에 더 큰 영향을 줄 수 있음을 보여준다.

▸ **주제어**: OpenClaw, 한국어 업무자동화, LLM 에이전트, 스킬 설계, 벤치마크 평가

• First Author: Youngmi Baek, Corresponding Author: Jung Kyu Park
*Youngmi Baek (ymbaek@daejin.ac.kr), Division of AI Convergence, Major in Computer Science, Daejin University
*Jung Kyu Park (jkpark@daejin.ac.kr), Division of AI Convergence, Major in Computer Science, Daejin University
• Received: 2026. 04. 13, Revised: 2026. 05. 15, Accepted: 2026. 05. 21.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

최근 대규모 언어모델(LLM)은 단순한 질의응답이나 텍스트 생성 수준을 넘어, 외부 도구를 활용하고 복합 과업을 수행하는 에이전트 형태로 빠르게 확장되고 있다. 특히 프롬프트 기반 학습과 프롬프트 엔지니어링은 모델의 출력을 통제하고 과업 적합성을 높이는 핵심 방법으로 주목받고 있으며, 이에 따라 프롬프트 설계 원리와 프레임워크를 체계적으로 정리하려는 연구도 활발히 이루어지고 있다 [1-3]. 또한 chain-of-thought prompting과 instruction-following 연구는 프롬프트 설계가 추론 과정과 지시 준수에 직접적인 영향을 줄 수 있음을 보여주었다 [4,5]. 이러한 흐름은 프롬프트를 단순한 입력 문장이 아니라, 에이전트의 행동 전략과 출력 방식을 결정하는 설계 요소로 사용되는 것을 보여준다.

이와 함께 LLM 기반 에이전트 자체를 하나의 연구 대상으로 다루는 논의도 빠르게 확대되고 있다. 특히 ReAct는 추론 과정과 외부 행동을 하나의 루프 안에서 결합함으로써, LLM 에이전트가 단순 응답 생성이 아니라 실제 과업 수행 단위로 확장될 수 있음을 보여주었다 [6]. 최근 해외 리뷰 연구들은 LLM 에이전트를 계획, 기억, 추론, 도구 사용, 상호작용을 포함하는 통합 구조로 설명하고 있으며, 단일 에이전트뿐 아니라 다중 에이전트 체계까지 포함한 응용과 평가 방식을 폭넓게 정리하고 있다 [7-9]. 또한 instruction tuning과 tool learning을 다룬 연구들은 모델의 지시 준수성과 외부 도구 활용 능력이 실제 업무 과업 수행과 밀접하게 연결되어 있음을 보여준다 [10,11]. 나아가 WorkArena, WebArena, MIND2WEB와 같은 최근 벤치마크 연구는 에이전트 평가가 실제 업무 및 웹 과업으로 확장되고 있음을 보여준다 [12-14].

그러나 기존 해외 연구의 상당수는 영어권 환경이나 범용적 응용 시나리오를 중심으로 이루어져 왔으며, 한국어 업무자동화 맥락에서 스킬 설계 방식이 실제 과업 수행에 어떤 차이를 만드는지를 실증적으로 비교한 연구는 상대적으로 부족하다. 특히 동일한 모델과 동일한 과업 환경에서 스킬 템플릿만을 달리하여 비교하는 벤치마크형 연구는 드문 편이다. 이러한 공백은 한국어 기반 업무자동화 환경에서 프롬프트 또는 스킬 설계 전략의 실제 효과를 검토할 필요성을 제시한다.

본 연구에서는 OpenClaw 기반 한국어 업무자동화 환경에서 SKILL.md 작성 방식의 차이가 과업 수행에 미치는 영향을 비교하고자 한다. 이를 위해 procedural, constraint, example, checklist의 네 가지 스킬 설계 유형을 구성하고, 회신 초안 작성, 확인 필요 정보 정리, 체크리스트 생성, 구조화된 요약과 같은 한국어 사무형 과업에 적용하였다. 본 연구는 스킬 설계 유형 간 평균 성능을 비교하는 동시에, 과업의 구조화 수준과 정보의 불확실성이 수행 결과에 미치는 영향을 분석함으로써 한국어 업무자동화 환경에서 스킬 설계 전략의 실제적 의미를 규명하고자 한다.

II. Related works

최근 LLM 에이전트 연구는 프롬프트 설계, 도구 사용, 자율적 의사결정을 중심으로 빠르게 확장되고 있다. 프롬프트 기반 학습과 프롬프트 프레임워크를 정리한 연구들은 프롬프트가 단순 입력 문장을 넘어 모델의 과업 수행 전략을 결정하는 핵심 요소임을 보여주었다 [1,2]. 또한 chain-of-thought prompting과 instruction-following 연구는 추론 과정과 지시 준수에 대한 프롬프트 설계의 영향을 보여주었으며 [4,5], ReAct는 reasoning과 acting의 결합을 통해 에이전트 실행 구조를 구체화하였다 [6]. 자율형 에이전트와 tool learning 관련 연구들은 외부 도구 활용과 상호작용 능력이 실제 업무 수행에서 중요함을 강조하였다 [7-11]. 최근 WorkArena, WebArena, MIND2WEB와 같은 벤치마크 연구는 이러한 평가를 실제 업무 및 웹 환경으로 확장하고 있다 [12-14].

본 연구에서 다루는 스킬 설계는 일반적인 프롬프트 엔지니어링과 동일한 개념이 아니다. 프롬프트 엔지니어링이 주로 단일 입력 프롬프트의 문장 구성과 지시 표현을 조정하는 접근이라면, 스킬 설계는 재사용 가능한 실행 지침, 금지 규칙, 예시, 점검 항목을 하나의 스킬 템플릿으로 구조화하여 에이전트 런타임에서 호출 가능하게 만든다는 점에서 더 상위의 추상화 수준을 가진다.

OpenClaw는 이러한 흐름을 실제 실행 환경으로 구현한 에이전트 플랫폼이라는 점에서 의미가 있다. 공식 문서에 따르면, OpenClaw의 스킬은 SKILL.md를 포함한 폴더 단위로 구성되며, 에이전트는 이를 바탕으로 사용할 수 있는 기능을 식별하고 필요한 경우 해당 지침을 읽어 과업 수행에 반영한다. 또한 skills watcher를 통해 세션 중간에도 스킬 목록이 갱신될 수 있고, 멀티에이전트 구조에서는 agent별로 skills, workspace, credentials가 분리되어 관리된다. 이러한 구조는 스킬 설계 방식 자체가 과업 수행 결과에 영향을 줄 수 있음을 보여주며, 본 연구가 SKILL.md 작성 유형을 비교 대상으로 설정한 근거가 된

다 [15].

한편 기존 연구들은 주로 범용 LLM 에이전트나 도구 학습의 전반적 구조를 논의하는 데 초점을 두고 있으며, 동일한 과업 환경에서 스킬 템플릿만을 달리하여 업무 수행 결과를 비교한 연구는 상대적으로 드물다. 특히 한국어 업무자동화 맥락에서 procedural, constraint, example, checklist와 같은 스킬 설계 유형을 직접 비교한 실증 연구는 거의 확인되지 않는다. 따라서 본 연구는 OpenClaw의 스킬 구조를 실험 단위로 구체화하여, 한국어 사무형 과업에서 스킬 설계 전략의 효과를 비교·분석한다는 점에서 기존 연구와 차별성을 가진다.

III. Research Method

1. Research Overview

본 연구는 OpenClaw 기반 한국어 업무자동화 환경에서 스킬 설계 유형이 과업 수행 결과에 미치는 영향을 비교하기 위해 실험을 수행하였다. 이를 위해 동일한 과업 환경과 동일한 실행 조건을 유지한 상태에서, SKILL.md 작성 방식만 달리한 네 가지 스킬 설계 유형(procedural, constraint, example, checklist)을 구성하였다. 각 유형은 동일한 업무 과업에 적용되었으며, 출력 형식 준수 여부, 핵심 정보 반영 정도, 불확실 정보 처리, 입력 외 정보 생성 여부 등을 기준으로 성능을 비교하였다.

본 실험은 참가자 기반 사용성 평가가 아니라, 사전에 설계한 한국어 사무형 과업을 반복 실행하는 비참가자형 벤치마크 실험으로 수행하였다. 초기 파일럿 과업을 포함하여 전체 과업을 점검한 후, 최종 비교 분석에는 task01을 제외한 task02~task08의 7개 과업을 사용하였다.

과업 실행은 OpenClaw CLI를 이용하여 수행하였다. 각 실행에서는 과업 파일과 해당 스킬 파일을 명시적으로 읽도록 지시하였으며, 최종 답변만 출력하도록 통제하였다. 출력 결과는 모두 텍스트 파일로 저장하여 후속 채점과 비교에 활용하였다.

2. Experimental Environment

실험은 리눅스 기반 환경에서 OpenClaw를 설치하여 수행하였다. OpenClaw 버전은 2026.3.25 (4329c93)였으며, 모든 실행은 main agent와 동일한 workspace에서 수행하였다. 기본 호출 모델은 Anthropic의 claude-sonnet-4-6이었고, 인증은 Anthropic API key를 사용하였다. 과업 파일은 benchmarks/email 폴더에 저장하였고, 스킬 파일은 skills/procedural, skills/constraint, skills/example, skills/checklist 폴더에 각각 SKILL.md 형태로 구성하였다. 또한 모든 실행에서 동일한 전역 지침을 적용하기 위해 workspace 루트에 AGENTS.md를 두고, 외부 발송·삭제·수정과 같은 실제 행동은 금지하였다.

3. Skill Design Types

본 연구에서는 OpenClaw의 SKILL.md 구조를 활용하여 다음 Table 1은 네 가지 스킬 템플릿의 대표 예시를 제시한다. 첫째, procedural 유형은 과업 수행 순서를 단계별 절차로 제시하는 방식이다. 둘째, constraint 유형은 금지 행동, 승인 조건, 추측 금지 등 제약 규칙을 우선적으로 제시하는 방식이다. 셋째, example 유형은 바람직한 출력 예와 바람직하지 않은 예를 함께 제시하여 수행 방식을 유도하는 방식이다. 넷째, checklist 유형은 작업 전·중·후에 확인해야 할 점검 항목을 중심으로 구성한 방식이다. 네 유형은 모두 동일한 과업을 대상으로 적용되었으며, 차이는 SKILL.md의 지시 방식에만 존재하도록 설계하였다.

Table 1. Representative Excerpts of the Four SKILL.md Templates

Procedural	1. Read task file → 2. Extract key information → 3. Produce requested format
Constraint	Do not send, delete, or modify; do not add unsupported facts
Example	Good example / Bad example
Checklist	Before / During / After task checklist

4. Benchmark Task Configuration

Table 2와 같이 최종 벤치마크는 한국어 기반 사무형 과업 7개(task02~task08)로 구성하였다. 각 과업은 실제 업무 상황을 단순화하여 설계하였으며, 회신 초안 작성, 확인 필요 정보 정리, 체크리스트 생성, 요약 보고, 우선순위 판단 등 한국어 업무자동화에서 자주 발생할 수 있는 작업을 포함하였다. 이러한 과업 구성은 실제 지식노동형 업무와 웹 기반 환경을 바탕으로 에이전트를 평가하려는 최근 벤치마크 연구의 흐름과도 연결된다 [12-14].

과업 설계 시에는 단순 요약이나 분류에 그치지 않도록 하기 위해, 불확실한 일정 정보, 누락된 입력 정보, 확정과 미확정 정보의 구분, 실제 행동 금지, 형식 제약 등의 요소를 포함하였다. 이를 통해 단순한 문장 생성 능력보다는 지시 준수, 정보 구조화, 보수적 판단, 과잉 추론 억제 능력이 드러날 수 있도록 하였다. 최종 벤치마크는 회신 초

Table 2. Benchmark Task Configuration

Task	Task Type	Core Objective	Key Evaluation Focus	Output Format
task02	Reply draft generation	Write a polite reply draft under scheduling constraints	Exclusion of unsupported time proposals, handling of uncertainty, format compliance	Email subject + email body draft
task03	Follow-up action organization	Identify required follow-up actions and missing information from an administrative request	Detection of confirmation needs, non-commitment to uncertain support conditions, action limit compliance	Requested items / information to confirm / follow-up actions
task04	Checklist generation	Create a checklist of missing or uncertain information before replying to a meeting request	Missing-information detection, avoidance of redundant items, concise checklist structure	4-6 bullet points
task05	Structured summary	Summarize a long notice email for reporting to a professor	Inclusion of key schedule and request, separation of confirmed vs. unconfirmed information	3-line summary
task06	Reply draft with clarification question	Write a cautious reply draft and include one follow-up question	Conservative interpretation of availability, no unsupported time generation, question inclusion	Email subject + email body draft
task07	Priority-based confirmation planning	Prioritize information that must be confirmed before processing a room reservation request	Ordering of confirmation priorities, avoidance of premature booking assumptions	Ranked checklist (1st-4th priority)
task08	Structured reporting summary	Report confirmed information, requests, uncertainty, and risks in a fixed format	Separation of confirmed/requested/unconfirmed/ risk information, no overcommitment	4-line structured summary

안 작성, 확인 필요 정보 정리, 체크리스트 생성, 구조화된 요약 등 이메일 기반 업무 상황을 중심으로 구성하였다. 본 연구에서 '불확실 정보'는 정확한 시간 미제시, 참석자 미확정, 지원 여부 보류, 후속 일정 미확정 등과 같이 과업 수행 시 추가 확인이나 보수적 해석이 필요한 정보를 의미한다. Table 3은 task06의 축약 예시를 보여준다.

Table 3. Abbreviated Prompt Example of task06

```
역할: 연구실 조교
과업: 학생에게 보낼 회신 초안 작성
제약: 실제 발송 금지, 구체 시간 임의 제안 금지, 미확정 일정 확정 금지
출력 형식: 메일 제목 + 본문 초안
요구: 마지막 문장에 확인 질문 1개 포함
```

5. Execution and Evaluation Procedures

각 과업은 네 가지 스킬 설계 유형에 대해 동일한 조건에서 실행하였다. 초기 1차 실행을 통해 과업의 변별력을 확인한 후, 최종 분석 대상 과업에 대해서는 반복 실행을 수행하여 결과의 일관성을 점검하였다. 실행 결과는 모두 텍스트 파일로 저장하였으며, 과업별 채점 기준에 따라 점수를 부여하였다. 최종 분석은 7개 과업(task02~task08), 4개 스킬 설계 유형, 각 조건당 3회 반복 실행으로 구성되었으며, 총 84회의 실행 결과를 비교하였다.

평가 점수는 과업별로 7점 만점 기준으로 산출하였다. 공통적으로는 출력 형식 준수 여부, 핵심 정보 반영 여부, 불확실 정보의 적절한 처리, 입력에 없는 정보 추가 여부, 실제 행동 지시 포함 여부 등을 평가 기준에 포함하였다. 예를 들어 회신 초안 작성 과업에서는 입력에 없는 구체 시간을 임의로 제안하지 않았는지, 요약 과업에서는 확정 정보와 미확정 정보를 구분했는지, 체크리스트 과업에서는 누락된 핵심 확인 항목을 포함했는지 등을 중점적으로 판단하였다.

모든 출력은 사전에 정의한 과업별 평가 기준표에 따라 2인의 연구자가 독립적으로 평가하였다. 평가 결과가 일치하지 않는 경우에는 논의를 통해 최종 점수를 확정하였다. 최종적으로는 과업별 점수와 스킬 유형별 평균 점수를 산출하여 비교하였으며, 점수 차이와 함께 어떤 과업에서 공통 오류가 발생했는지를 함께 분석하였다.

IV. Results

본 연구에서는 최종 벤치마크 과업으로 task02부터 task08까지 총 7개 과업을 사용하여, procedural, constraint, example, checklist의 네 가지 스킬 설계 유형을 비교하였다. 과업별 평가 결과를 종합한 전체 평균 점수는 7점 만점 기준 6.68점으로 나타났으며, 스킬 설계 유형별 평균은 procedural 6.64점, constraint 6.71점, example 6.71점, checklist 6.64점으로 집계되었다. Table 4에 요약한 것과 같이 전반적으로 네 유형 모두 높

Table 4. Mean Performance Scores by Task and Skill Design Type

Task	Procedural	Constraint	Example	Checklist	Task Mean
task02	5.0	5.5	5.5	5.0	5.25
task03	7.0	7.0	7.0	7.0	7.00
task04	7.0	7.0	7.0	7.0	7.00
task05	7.0	7.0	7.0	7.0	7.00
task06	6.5	6.5	6.5	6.5	6.50
task07	7.0	7.0	7.0	7.0	7.00
task08	7.0	7.0	7.0	7.0	7.00
Skill Mean	6.64	6.71	6.71	6.64	6.68

은 수행 수준을 보였으나, constraint와 example 유형이 procedural 및 checklist 유형보다 소폭 높은 평균을 나타냈다.

과업별 점수를 기준으로 Friedman 검정을 수행한 결과, 네 가지 스킬 설계 유형 간 차이는 통계적으로 유의하지 않았다($\chi^2(3)=3.00$, $p=.392$). 또한 pairwise Wilcoxon signed-rank 검정에서도 유의한 차이는 확인되지 않았다(all $p \geq .688$). 따라서 본 연구에서 관찰된 평균 점수 차이는 통계적으로 유의한 효과라기보다 서술적 차이로 해석하는 것이 타당하다.

Fig. 1의 과업별 평균 점수를 보면 task02가 5.75점으로 가장 낮았고, task06이 6.50점으로 그다음을 차지하였다. 반면 task03, task04, task05, task07, task08은 모든 스킬 설계 유형에서 7.00점으로 동일한 최고 수준의 점수를 기록하였다. 이는 본 벤치마크에 포함된 구조화된 이메일 기반 과업에서는 스킬 설계 방식과 관계없이 안정적인 출력이 가능했음을 의미한다. 반면 상대적으로 낮은 점수를 보인 task02와 task06은 모두 일정 조율이나 회신 초안 작성처럼 불확실한 정보를 해석해야 하는 과업으로, 다른 과업보다 보수적 판단과 제약 준수가 더 중요하게 작용하였다.

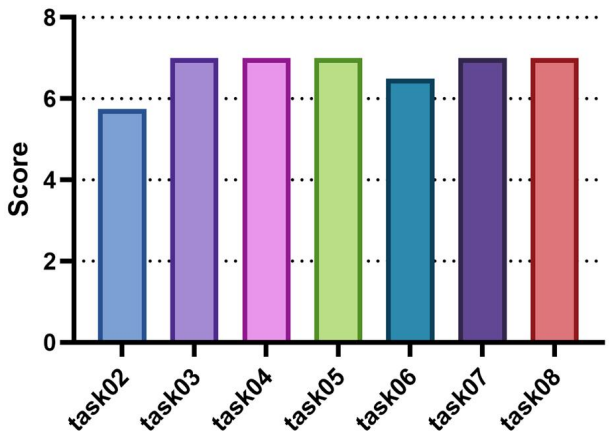

Fig. 1. Mean Performance Score by Task

낮은 점수가 나타난 두 과업의 오류 양상을 보면, task02에서는 입력에 명시되지 않은 구체적인 시간대를 제안하거나 불필요한 메타 문장을 포함하는 경우가 관찰되었고, task06에서는 학생의 가능 시간 범위를 다소 넓게 해석하는 확인 질문이 반복적으로 나타났다. 그러나 이러한 오류는 특정 스킬 설계 유형에만 국한되지 않고 네 조건 전반에서 유사하게 확인되었다. 이에 따라 Fig. 2에서 나타난 스킬 설계 유형 간 평균 차이는 존재하더라도 그 폭은 매우 제한적이며, 본 실험에서는 과업의 성격과 정보의 불확실성이 스킬 템플릿의 지시 구성 방식보다 성능 차이에 더 큰 영향을 준 것으로 해석할 수 있다.

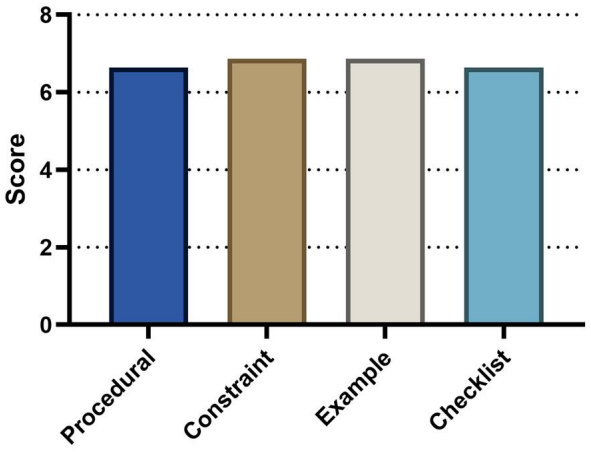

Fig. 2. Mean Score by Skill Design Type

constraint와 example 유형이 일부 과업에서 procedural 및 checklist보다 소폭 높은 평균을 보인 것은, 금지 규칙의 전면 배치와 예시 기반 유도가 입력 외 세부사항 보완이나 메타 문장 삽입을 일부 줄였기 때문으로 해석할 수 있다. 다만 이러한 차이는 task02와 같이 불확실한 일정 해석이 포함된 일부 과업에 제한적으로 나타났으며, 전체 벤치마크 수준에서는 통계적으로 유의한 차이로 확인되지는 않았다.

V. Discussion

본 연구 결과는 OpenClaw 기반 한국어 이메일 사무형 과업에서는 스킬 설계 유형의 영향이 제한적으로 나타남을 보여준다. 특히 본 연구에 포함된 과업 중 확인 필요 정보 정리, 체크리스트 작성, 구조화된 요약처럼 출력 형식이 명확한 이메일 기반 과업에서는 네 가지 스킬 유형이 모두 거의 동일한 결과를 생성하였다. 이는 과업의 구조가 충분히 명확할 경우, 스킬 템플릿의 지시 구성 방식보다 모델의 기본 추론 능력과 과업 지시 자체가 수행 결과를 더 강하게 좌우할 수 있음을 보여준다.

반면, 일정 조율이나 회신 초안 작성처럼 불확실한 정보를 포함한 과업에서는 부분적인 감점이 확인되었다. 특히 task02와 task06에서는 입력에 없는 구체 시간을 보완적으로 제안하거나, 사용자의 가능 범위를 다소 넓게 해석하는 경향이 공통적으로 나타났다. 이와 같은 결과는 스킬 설계 유형이 출력 형식의 정돈이나 제약 인식에 일부 도움을 줄 수는 있으나, 불확실한 일정 정보에 대한 과도한 보완 추론까지 완전히 억제하지는 못했음을 의미한다. 즉, 본 실험에서 관찰된 주요 오류는 개별 템플릿의 실패라기보다, 모델이 공통적으로 보이는 일정 정보 보완 경향에서 비롯된 것으로 볼 수 있다. 다만 본 연구의 벤치마크는 이메일 기반 사무형 과업에 한정되어 있어, 향후에는 일정 관리, 파일 정리, 웹 정보 수집과 같은 다른 업무 도메인으로의 확장이 필요하다. 따라서 본 연구는 OpenClaw 기반 한국어 이메일 사무형 과업을 대상으로 한 탐색적 벤치마크로 이해하는 것이 타당하다.

이러한 결과는 향후 한국어 업무자동화 벤치마크를 설계할 때, 단순한 분류나 요약 과업보다 불확실성 처리, 누락 정보 탐지, 확인 질문 생성, 확정·미확정 정보 구분과 같은 요소를 더 적극적으로 포함할 필요가 있음을 시사한다. 또한 실제 활용 관점에서는 스킬 설계 방식의 미세한 차이를 강조하기보다, 어떤 유형의 과업에서 모델이 입력 밖 세부사항을 보완하려는 경향을 보이는지 파악하고 이를 억제할 수 있는 과업 설계 및 검증 절차를 함께 마련하는 것이 중요하다.

VI. Conclusions

본 연구는 OpenClaw 기반 한국어 업무자동화 환경에서 procedural, constraint, example, checklist의 네 가지 스킬 설계 유형을 비교하였다. 이를 위해 7개의 한국어 이메일 사무형 과업을 구성하고 동일한 실행 환경에서 벤치마크 실험을 수행하였다.

실험 결과, OpenClaw는 대부분의 과업에서 안정적인 성능을 보였으며, 네 가지 스킬 설계 유형 간 차이는 전반적으로 크지 않았다. 반면 불확실한 정보를 포함한 일부 과업에서는 입력에 없는 세부사항을 보완적으로 생성하거나 가능 범위를 다소 확장 해석하는 경향이 관찰되었다. 이러한 결과는 스킬 설계 방식의 차이보다 과업의 구조화 수준과 정보의 불확실성이 수행 결과에 더 큰 영향을 줄 수 있음을 보여준다. 따라서 본 연구의 결과는 OpenClaw 기반 한국어 이메일 사무형 과업에 대한 초기 비교 결과로 해석하는 것이 타당하다. 향후에는 보다 다양한 업무 도메인과 실제 사용자 평가를 포함한 확장이 필요하다.

REFERENCES

[1] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing," ACM Computing Surveys, Vol. 55, No. 9, pp. 1-35. Jan. 2023. DOI: 10.1145/356081

[2] X. Liu, J. Wang, X. Yuan, J. Sun, G. Dong, P. Di, W. Wang, and D. Wang, "Prompting Frameworks for Large Language Models: A Survey," ACM Computing Surveys, Feb. 2026. DOI: 10.1145/3789253

[3] B. Chen, Z. Zhang, N. Langrené, and S. Zhu, "Unleashing the potential of prompt engineering for large language models," Patterns, Vol. 6, No. 6, pp. 101260, Jun. 2025. DOI: 10.1016/j.patter.2025.101260

[4] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E.H. Chi, Q.V. Le, and D. Zhou, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," Proceedings of the 36th International Conference on Neural Information Processing Systems, pp. 24824-24837, 2022.

[5] L. Ouyang, J. Wu, X. Jiang, *et al.*, "Training language models to follow instructions with human feedback," Proceedings of the 36th International Conference on Neural Information Processing Systems, pp. 27730-27744, 2022.

[6] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "ReAct: Synergizing Reasoning and Acting in Language Models," Proceedings of the 11th International Conference on Learning Representations(ICLR 2023), pp.1-33, 2023.

[7] L. Wang, C. Ma, X. Feng, *et al.*, "A survey on large language model based autonomous agents," Frontiers of Computer Science, Vol. 18, pp. 186345, Mar. 2024. DOI: 10.1007/s11704-024-40231-1

[8] X. Li, S. Wang, S. Zeng, Y. Wu, and Y. Yang. "A survey on LLM-based multi-agent systems: workflow, infrastructure, and

challenges," Vicinagearth, Vol. 1, pp. 9, Oct. 2024. DOI: 10.1007/s44336-024-00009-2

[9] S.S. Chowa, R. Alvi, S.S. Rahman, *et al.*, "From language to action: a review of large language models as autonomous agents and tool users," Artificial Intelligence Review, Vol. 59, pp. 71, Jan. 2026. DOI: 10.1007/s10462-025-11471-9

[10] S. Zhang, L. Dong, X. Li, *et al.*, "Instruction Tuning for Large Language Models: A Survey," ACM Computing Surveys, Vol. 58, No. 7, pp. 1-36, Jan. 2026. DOI: 10.1145/3777411

[11] C. Qu, S. Dai, X. Wei, *et al.*, "Tool learning with large language models: a survey, Frontiers of Computer Science, Vol. 19, pp. 198343, Jan. 2025. DOI: 10.1007/s11704-024-40678-2

[12] A. Drouin, M. Gasse, M. Caccia, *et al.*, "WorkArena: how capable are web agents at solving common knowledge work tasks?," Proceedings of the 41st International Conference on Machine Learning(ICML'24), pp. 11642– 11662, 2024.

[13] S. Zhou, F.F. Xu, H. Zhu, *et al.*, "WebArena: A Realistic Web Environment for Building Autonomous Agents." ICLR, pp. 1-24, 2024. Proceedings of the International Conference on Learning Representations, pp. 1-22., 2023

[14] A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, "Mind2Web: Towards a Generalist Agent for the Web," pp. 28091-28114, Proceedings of the the 37th International Conference on Neural Information Processing Systems, 2023.

[15] OpenClaw Documentation, https://openclaw.ai/

Authors

Youngmi Baek received her Ph.D. in Computer Engineering from Kyungpook National University in 2015. She began her research career at the POSTECH Information and Communication Research Laboratory (PIRL) in 2002 and joined the DGIST CPS Global Center in 2015. From February 2017, she served as a Research Professor at DGIST. She worked at Changshin University as an Assistant Professor from 2020 and joined Daejin University as an Associate Professor in 2026. Her research focuses on system modeling and network security, including cyberattack and anomaly detection, in automotive cyber-physical systems and autonomous manufacturing.

Jung Kyu Park received his Ph.D. degree in Computer Engineering from Hongik University in 2013. He was a Research Professor at Dankook University from 2014. From October 2015 to February 2017, he worked at UNIST. In 2018, he joined the Department of Computer Software Engineering at Changshin University as an Assistant Professor. Since August 2024, he has been serving as an Associate Professor in the Division of AI Convergence, Major in Computer Science, at Daejin University. His research interests include data storage, robotics, system software, embedded systems.