회의 컨텍스트 비서 에이전트 생성 연구

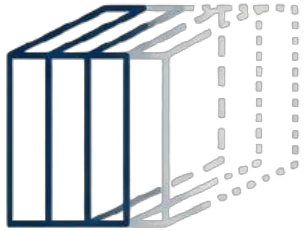
LLM 기반 자율 에이전트를 통한 회의 효율성 극대화 및 지식 자산화 전략

[redacted] 최우연

목차

1. 연구 배경

회의 과정에서 발생하는 지식 손실 구조

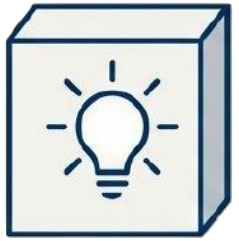


맥락의 단절
(Context Fragmentation)

회의 중 논의된 결정과 배경 정보가 시간이 지나면서 잊혀지고, 후속 회의에서 연속성이 깨지는 문제 발생

책임의 분산
(Responsibility Diffusion)

구두로 합의된 액션 아이템이 명확히 기록되지 않아 담당자와 마감일이 누락되거나 잊히는 문제 발생



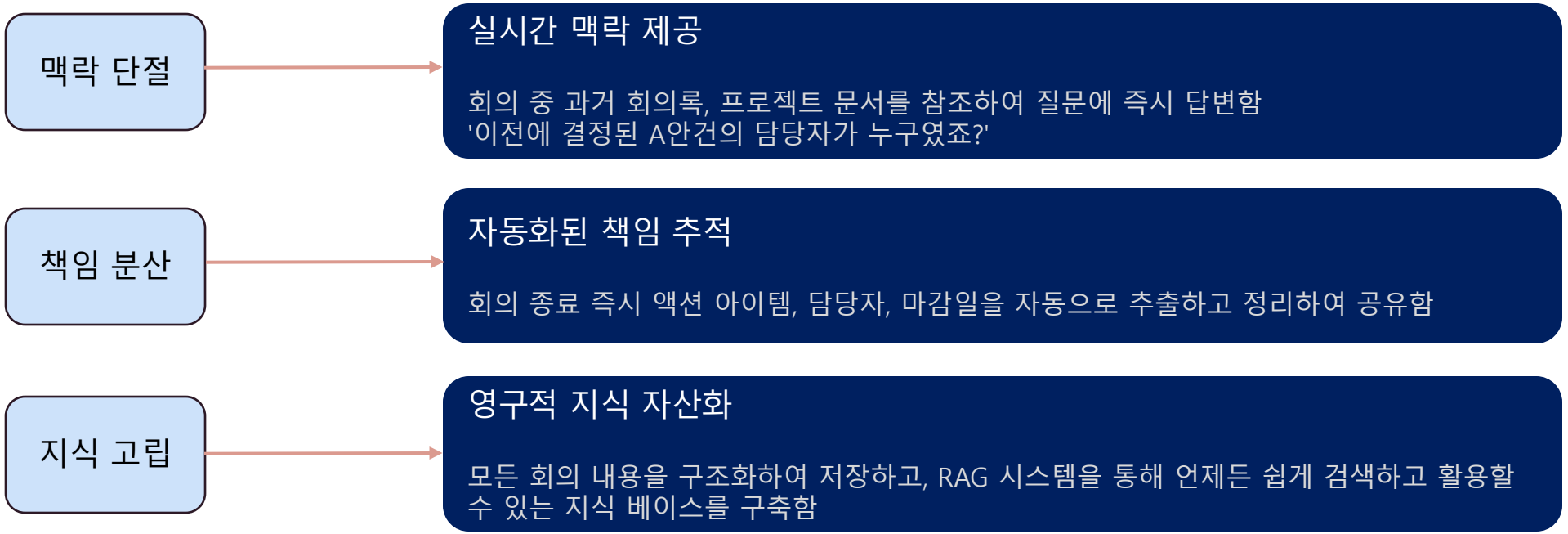
지식의 고립
(Information Silos)

중요한 논의 내용이 회의록에 제대로 담기지 않거나, 검색이 어려워 조직의 지식 자산으로 활용되지 못하고 사장되는 문제 발생

회의는 반복되지만 지식은 축적되지 않음 - 본 연구는 이 구조적 손실을 해결하기 위한 회의 컨텍스트 에이전트를 제안함

2. 연구 목표

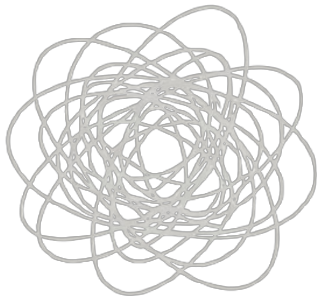
컨텍스트 비서 에이전트(CSA : Context-aware Service Agent)는 회의의 모든 순간에 참여하여 지식의 손실을 막고 가치를 더하는 지능형 파트너를 목표로 함



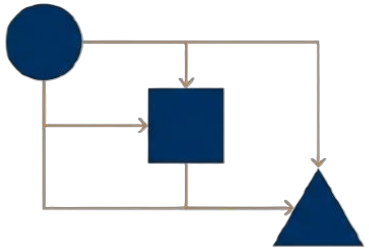
3. 연구 접근 방법: 왜 멀티 에이전트인가?

- ✓ 본 연구는 회의 과정의 복잡성과 맥락 의존성을 고려할 때, 단일 만능 에이전트보다 역할이 분화된 전문가 팀 기반 멀티 에이전트 구조가 더 높은 성능과 안정성을 제공할 수 있음
- ✓ 각 에이전트는 회의 맥락 이해, 기록, 지식 탐색, 구조화 등 명확히 구분된 역할과 책임을 가지며, 협업을 통해 회의 문제를 단계적으로 분해·해결가능하도록 구성(이러한 접근은 ChatDev, MetaGPT 등 최신 멀티 에이전트 연구에서 그 효과가 검증된 방법론 임)

하나의 만능 에이전트



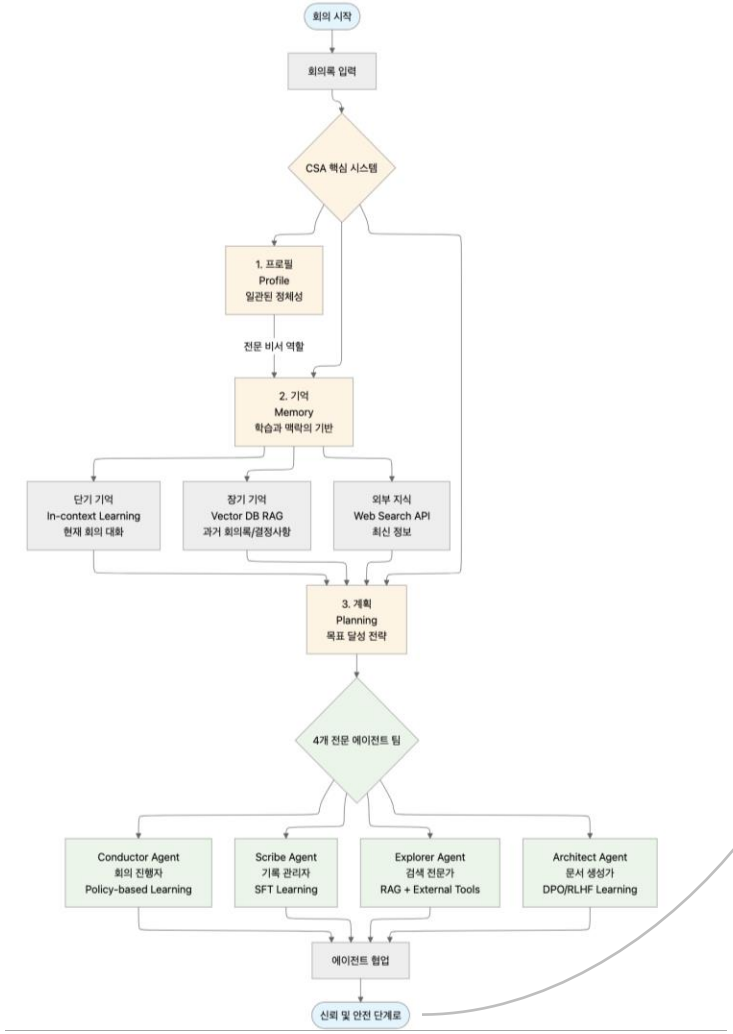
전문가 팀

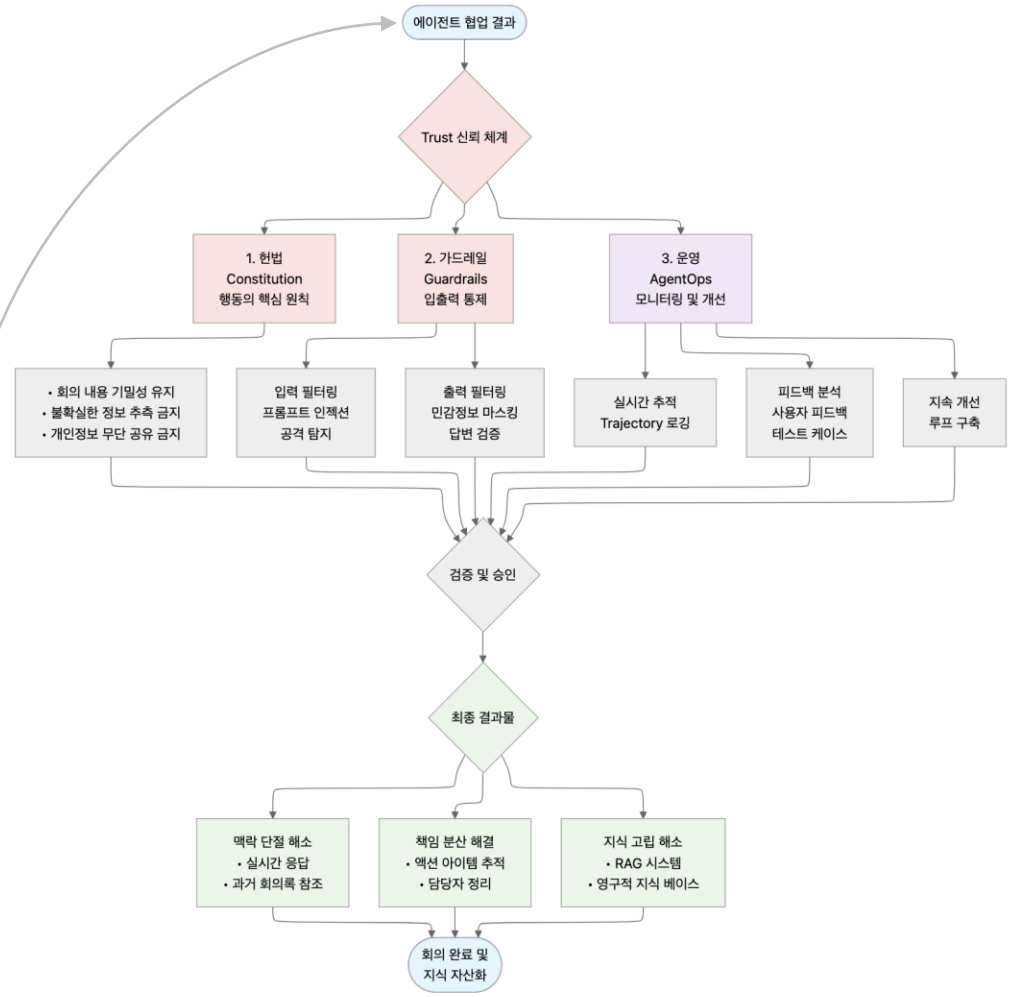


CSA의 협업 모델

협업 유형 : 협력적(Cooperative)	커뮤니케이션 구조 : 계층적(Layered)
✓ 모든 에이전트는 회의 맥락 보존과 지식 자산화라는 공통 목표를 공유하며, 병렬적·보완적으로 협력함	✓ 회의 진행자 에이전트가 상위 컨트롤 타워로서 사용자 요청과 회의 상태를 해석하고, 이를 기반으로 가장 적합한 전문 에이전트에게 작업을 분배·조정함

4. CSA 시스템 아키텍처

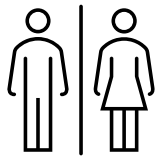




5. 인지 설계

협업 전략 선택의 기반

1.프로필(Profile) : 일관된 정체성



- **역할 정의** : "전문적이고 정확한 회의 비서"라는 명확한 역할과 성격(간결함, 맥락 우선)을 수작업으로 정의하여 초기 행동의 일관성을 확보
- 이를 통해 에이전트는 협업 상황에서도 자신의 역할 범위를 일관되게 유지함

2.기억(Memory) : 학습과 맥락의 기반

- 단기 기억(In-context), 장기 기억(Vector DB), 외부 지식(Web Search)을 통합하여 과거 경험을 바탕으로 현재의 맥락을 이해
- 공유된 기억 기반은 에이전트 간 정보 비대칭을 줄이고 협업 판단의 기준점으로 작용

3. 계획(Planning) : 목표 달성을 위한 전략 수립

- **ReAct 패턴**을 기반으로 목표를 분해하되, 협업 상황에서는 단순 실행뿐 아니라 토론·검증·재계획 전략을 선택적으로 수행
- **Reflexion**을 통해 자신의 행동뿐 아니라 협업 결과를 평가하며 계획을 지속적으로 조정

6. 다층 기억 구조 설계

유형	저장 내용	구현 방식	핵심 목적
단기 기억 (Short-Term Memory)	현재 회의의 실시간 대화, 최근 발언 맥락	In-context Learning(LLM의 컨텍스트 창)	실시간 응답의 일관성 및 맥락 유지
장기 기억 (Long-Term Memory)	이전 회의록, 결정 사항, 프로젝트 정보	Vector DB(RAG) (임베딩 기반 저장)	반복 회의·중복 논의 방지
조직 기억 (Organizational Memory)	결정 요약, R&R, 정책, 공식 결론	Vector DB(RAG) (임베딩 기반 저장)	담당자 변경 시 지식 유실 방지
외부 지식 (External Knowledge)	최신 기술 동향, 산업 뉴스, 실시간 정보	Web Search API (외부 도구 호출)	회의 판단의 현실성·시의성 확보

- ✓ 에이전트는 질의 맥락에 따라 최신성(recency), 관련성(relevance), 중요성(importance)을 기준으로 단기 기억·장기 기억·외부 지식 중 적절한 정보원을 동적으로 선택가능함
- ✓ 본 기억 구조는 단일 에이전트뿐 아니라, 멀티 에이전트가 동일한 맥락과 지식을 공유하기 위한 공통 인프라로 설계됨.

7. STT 통합 전략

STT 엔진 선택	후처리 파이프라인	신뢰도 관리
✓ Whisper, Google Speech-to-Text, Clova Speech ✓ 다국어 지원, 화자 분리(Speaker Diarization)	✓ 텍스트 정규화 (숫자, 날짜, 고유명사) ✓ 문장 분절 및 문맥 보정 ✓ LLM 기반 오류 수정	✓ Confidence Score 임계값 설정 ✓ 불확실한 부분은 사용자 확인 요청

STT는 회의의 소리를 조직의 자산으로 변환하는 첫 번째 관문임

8. 멀티 에이전트 구성

1. 회의 진행자 (Conductor Agent)

역할 : 전체 워크플로우 조율, 사용자 요청 분석 및 작업 라우팅, 계획 모듈 관리

핵심 기술 : 정책 기반(Policy-based) 학습을 통해 최적의 다음 행동/도구를 선택

2. 기록 관리자 (Scribe Agent)

역할 : 회의록 작성, 결정 사항 및 액션 아이템(담당자/마감일) 추출, 구조화된 출력 생성

핵심 기술 : 지도 학습(SFT)을 통해 정확한 정보 추출 포맷 학습

3. 검색 전문가 (Explorer Agent)

역할 : RAG 시스템을 통해 내부 문서(이전 회의록) 및 외부 웹 정보 검색

핵심 기술 : 외부 도구(벡터DB, 검색 API) 활용 능력 증강

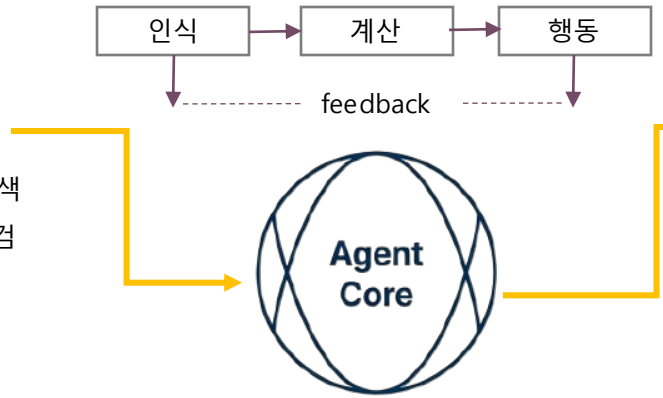
4. 문서 생성가 (Architect Agent)

역할 : 최종 요약문 생성, 공유 문서 포맷팅, 캘린더 등 외부 시스템 연동

핵심 기술 : DPO/RLHF를 통해 사용자가 선호하는 간결하고 명확한 스타일 학습

9. 외부 시스템 연동 구조

LLM 자체의 한계(실시간 정보 부재, 외부 시스템 제어 불가)를 극복하기 위해 외부 도구를 활용함



인식 (Perception)

회의 이력 검색기 (RAG) : Vector DB에서 과거 맥락 검색

웹 검색 도구 (Search API) : 최신 외부 정보를 실시간 검색

문서 접근 API : 공유 문서, 프로젝트 자료 조회

행동 (Action)

문서 생성기 : 회의록, 요약문을 지정된 포맷으로 생성

캘린더 연동 API : 추출된 액션 아이템을 담당자 캘린더에 자동 등록

알림 시스템 : Slack, Email로 리마인더 전송

계산 (Computation)

코드 인터프리터 : 복잡한 날짜 계산 등 논리적/수학적 연산 수행

✓ 실시간성: Web Search로 최신 정보 확보

✓ 맥락성: RAG로 과거 회의 내용 활용

✓ 실행력: API 연동으로 실제 업무 자동화

10. RAG 전략

검색 대상 (Source Documents)

- ✓ 모든 에이전트가 동일한 문서 풀(Pool) 접근
- ✓ **핵심 자료:** 이전 회의록
- ✓ **보조 자료:** 프로젝트 매뉴얼, 요구사항 정의서, 담당자 정보 등 신뢰도 높은 내부 문서

분할 전략 (Chunking Strategy)

- ✓ 청크 단위를 에이전트 간 참조 가능한 크기로 최적화 (512-1024 토큰)
- ✓ 의미 단위(안건별) 분할을 기본으로, 청크 간 10-20%의 내용이 겹치는 오버랩(Overlap) 기법을 적용하여 맥락 손실을 최소화.

검색 고도화 (Advanced Retrieval)

- ✓ 검색 결과 캐싱으로 중복 쿼리 방지
- ✓ 교차 검증으로 환각(Hallucination) 제거
- ✓ 단일 진실 출처(Single Source of Truth)
- ✓ Contextual Retrieval: 청크 저장 시 회의 날짜, 참석자 등 메타데이터를 함께 저장하여 검색 정확도 향상.
- ✓ HyDE (Hypothetical Document Embeddings): 각 에이전트가 자신의 관점으로 가상 답변을 먼저 생성하고, 이를 조합하여 다각도 검색 수행. 검색 결과는 모든 에이전트가 공유.

RAG는 단일 에이전트의 검색 기능이 아니라, 멀티 에이전트 협업의 공통 지식 기반임

11. 구조화된 출력

문제 정의

The Problem: 비구조화된 LLM 출력

❌ "김대리님이 보고서 작성..." → 시스템이 파싱할 수 없음
❌ "마감은 이번 주 중으로" → 캘린더 등록 불가
❌ 에이전트마다 다른 출력 형식 → 협업 시 데이터 변환 필요

해결 방안

The Solution: Pydantic 스키마

```python
from pydantic import BaseModel
from typing import List, Optional
from datetime import datetime

class ActionItem(BaseModel):
    task: str          # 할 일 내용
    assignee: str      # 담당자
    due_date: Optional[datetime] # 마감일

class MeetingMinutes(BaseModel):
    meeting_date: datetime
    attendees: List[str]
    action_items: List[ActionItem] # 중첩된 구조화된 데이터
```

활용 사례

Example: 회의 내용 → 구조화된 데이터

1. 입력

"김대리님이 SNS 기획안을 12월 20일까지 작성"

2. Pydantic 강제 변환

출력 (기계가 읽을 수 있는 형식):

{ "task": "SNS 기획안 작성", "assignee": "김대리", "due_date": "2024-12-20T00:00:00" }

3. 자동 검증

✓ 타입 확인: datetime 형식 맞음
✓ 필수 필드: task, assignee 존재
✓ 비즈니스 룰: due_date는 미래

4. 다른 시스템 연동

✓ 캘린더 API 자동 등록
✓ Slack 리마인더 설정
✓ Notion DB 저장

멀티 에이전트 협업

- 구조화된 출력 = 에이전트 협업의 공통어
- 기대효과 : 파싱 오류 감소, 에이전트 간 데이터 교환 성공률 향상, 후속 처리 개발 시간 단축 가능

12. 에이전트별 프롬프트 전략

Conductor Agent → Zero-Shot (역할 프롬프팅)

목적 : 예시 없이, 에이전트의 역할, 원칙, 페르소나를 정의하는 시스템, 프롬프트를 통해 기본 행동 정책을 설정

이유 : 회의 진행자는 상황에 따라 유연하게 대처해야 하므로 고정된 예시보다 원칙이 중요

Prompt Example

"당신은 전문적인 회의 진행자입니다. 간결하고 명확하게 답변하며, 불확실한 정보는 추측하지 말고 반드시 확인 후 응답하세요."

Scribe Agent + Explorer Agent → Few-Shot (In-Context Learning)

목적 : 몇 가지 구체적인 질의응답 예시를 프롬프트에 포함시켜, 모델이 원하는 출력 형식과 내용을 학습하도록 유도

이유 : 기록과 검색은 일관된 형식이 중요하므로 구체적 예시 필요

Prompt Example

"지난번 회의에서는 이렇게 정리했으니, 이번에도 동일한 형식으로 정리하세요."

Architect Agent → Plan-and-Solve (분해 프롬프팅)

목적 : 복잡한 과업을 여러 단계로 분해하여 지시함으로써 모델의 추론 과정을 명확하게 만들고 결과의 정확도를 높임

이유 : 문서 생성은 복잡한 과정이므로 단계별 분해가 필수

Prompt Example

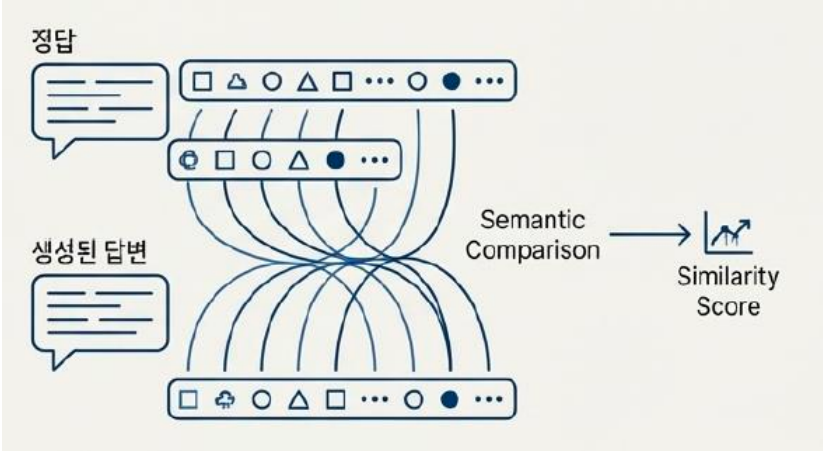
1. 주요 안건 목록 작성
2. 각 안건별 결정 사항 정리
3. 액션 아이템 추출
4. 최종 회의록 형식으로 통합

13. 출력 품질 검증

단순 키워드 매칭을 넘어, AI가 AI를 평가하는 지능형 품질 검증 시스템을 적용

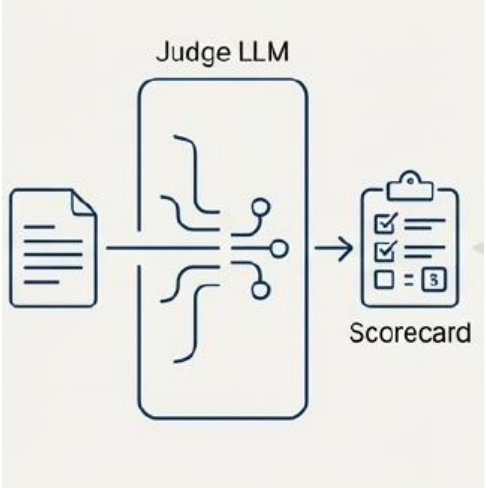
의미적 유사성 평가(BERTScore)

✓ 단어의 문맥적 의미를 이해하는 BERT 임베딩을 사용하여, 정답과 생성된 답변 사이의 의미적 유사도를 측정. 기존 n-gram방식의 한계를 극복



확장 가능한 종합 평가 (LLM-as-a-Judge)

✓ 사람의 평가와 유사한 판단을 내리도록 훈련된 다른 LLM을 평가자로 활용. 시간과 비용을 절약하며 일관된 평가 척도를 적용



Prompt Example

다음 회의 요약문을 3가지 기준으로 평가하세요(1~5점)

- 완전성(Completeness) : 모든 주요 결정 상황이 포함되었는가?
- 정확성(Accuracy) : 담당자와 일정이 정확하게 추출되었는가?
- 명확성(Clarity) : 누구나 이해할 수 있게 간결하게 작성되었는가?

14. 파인튜닝 프로세스

Phase1 : 기반 다지기 (Supervised Fine Tuning, SFT)

- ✓ **목표:** 회의 컨텍스트 이해를 위한 기본 능력 학습
- ✓ **데이터:**
  - 회의 전 과정(준비-진행-정리)에 대한 질문-정답 쌍
  - 회의 참석자 정보, 일정, 장소 정보 처리 예제
  - 컨텍스트 추출 및 의도 파악 태스크
- ✓ **학습 내용:**
  - 회의 도메인 지식 습득
  - 사용자 쿼리 이해 및 엔티티 추출
  - 기본 대화 패턴 및 응답 생성
  - API 호출 시점 판단
- ✓ **예상 결과:** 기본적인 회의 관련 질의응답이 가능한 모델

Phase 2: 선호도 정제 (Direct Preference Optimization, DPO)

- ✓ **목표:** 사용자 선호에 맞는 고품질 답변 스타일 학습
- ✓ **데이터:**
  - 동일 질문에 대한 [선호 답변 vs 비선호 답변] 쌍
  - 실제 사용자 피드백 기반 선호도 데이터
  - 컨텍스트 정확도, 간결성, 유용성 측면의 비교 데이터
- ✓ **학습 내용:**
  - 불필요한 정보 제거 (간결성)
  - 컨텍스트 정확도 향상
  - 사용자 의도에 맞는 우선순위 조정
  - 자연스러운 대화체 개선
- ✓ **예상 결과:** 사용자가 선호하는 스타일과 정확도를 갖춘 실용적 에이전트

15. 데이터 수집과 실험설계

데이터 수집

- ✓ Phase 1 (SFT):
  - 필요 데이터: 10,000 ~ 50,000 쌍의 질문-정답
  - 수집 방법:
    - 실제 회의록 크롤링 및 재가공 (2,000건)
    - 합성 데이터 생성 (GPT-4 활용, 5,000건)
    - 크라우드소싱 (3,000건)
  - 데이터 품질 검증: 전문가 리뷰 + 자동 필터링
- ✓ Phase 2 (DPO):
  - 필요 데이터: 5,000 ~ 10,000 쌍의 선호도 데이터
  - 수집 방법:
    - A/B 테스트를 통한 실사용자 선호도 수집
    - 전문가 annotation (회의 진행 경험자)
    - 선호도 기준: 간결성, 정확성, 실행 가능성

실험 설계

- ✓ 베이스라인
  - 단일 LLM (GPT-4, Claude)
  - 기존 회의 도구 (Otter.ai, Notion AI)
  - 인간 비서
- ✓ 실험 시나리오
  - 시나리오 1: 30분 정기 회의 (5-7명)
  - 시나리오 2: 2시간 전략 회의 (10-15명)
  - 시나리오 3: 다회차 연속 회의 (맥락 유지 검증)
- ✓ 평가 방법
  - 자동 평가: F1, ROUGE, BERTScore
  - 인간 평가: 5명의 평가자가 블라인드 테스트
  - 실사용 평가: 2주간 실제 팀에서 파일럿 테스트

16. 평가지표

Phase 1: SFT 평가 지표

1. 기본 성능 지표

- Perplexity: 언어 모델의 예측 정확도 (낮을수록 좋음), 목표: < 15
- Exact Match (EM): 정답과 정확히 일치하는 비율, 목표: > 70%
- F1 Score: 부분 일치를 고려한 정확도,목표: > 85%

2. 회의 도메인 특화 지표

- Entity Extraction Accuracy: 회의장소, 시간, 참석자 등 핵심 정보 추출 정확도, 목표: > 90%
- Intent Classification Accuracy: 사용자 의도(예약/조회/취소 등) 분류 정확도, 목표: > 95%
- Context Understanding Score: 복합 컨텍스트 이해도 (인간 평가), 목표: 5점 만점 중 > 4.0

Phase 2: DPO 평가 지표

1. 선호도 기반 지표

- Win Rate: A/B 테스트에서 DPO 모델이 선택된 비율, 목표: > 65% (vs SFT 모델)
- Preference Alignment Score: 인간 선호도와의 일치율, 목표: > 75%

2. 품질 지표

- Relevance Score: 질문과 답변의 관련성 (1-5점) 목표: > 4.5
- Conciseness Score: 간결성 평가 (불필요한 정보 제거) 목표: 평균 응답 길이 SFT 대비 20% 감소, 정보 손실 < 5%
- Helpfulness Score: 실제 도움 정도 (사용자 평가) 목표: > 4.3/5.0

통합 평가 지표

1. 사용자 만족도

- User Satisfaction Score (CSAT): 5점 척도 만족도 조사, 목표: > 4.0
- Net Promoter Score (NPS): 추천 의향, 목표: > 30

2. 실전 성능

- Task Success Rate: 실제 업무 완수율 (예약 성공, 정보 제공 등), 목표: > 90%
- Average Response Time: 평균 응답 시간, 목표: < 2초
- Multi-turn Success Rate: 복잡한 대화에서의 성공률, 목표: > 80%

17. 보안 및 운영 전략

헌법 : 행동의 핵심 원칙

AI가 반드시 지켜야 할 내부 원칙을 명문화하여 안정성을 확보

✓ 예시 :

- 회의 내용의 기밀성을 유지한다.
- 불확실한 정보는 추측하지 않는다.
- 개인정보를 무단으로 공유하지 않는다.

가드레일 : 입출력 통제

입력 필터링 : 프롬프트 인젝션 공격 및 업무 외 질문을 차단

출력 필터링 : 생성된 답변에서 민감 정보(개인정보)를 마스킹

통합 평가 지표

✓ 에이전트의 모든 행동-관찰 행위를 로깅

✓ 실패 사례 및 사용자 피드백을 분석하여 자동으로 테스트케이스를 생성하고, 시스템을 지속적으로 개선하는 루프 구축

18. 연구 한계 및 기대효과

기대 효과

✓ **효율성 증대**
회의 준비 및 정리 시간 획기적 단축

✓ **지식 자산화**
검색 가능한 지식 베이스 구축으로 조직의 집단 지성 강화

✓ **의사결정 품질 향상**
데이터 기반의 빠르고 정확한 맥락 공유

현재의 한계

✓ **환각(Hallucination) 위험**
RAG와 Self-Verification으로 완화하지만 완벽한 제거는 어려움

✓ **STT(Speech-to-Text) 의존성**
실시간 음성 인식 시스템의 정확도가 전체 성능에 큰 영향을 미침

다음 단계

✓ **시각화 기능 강화**
회의 중 논의된 프로세스를 Mermaid나 PlantUML 코드로 자동생성하여 순서도나 타임라인을 시각적으로 제공

CSA는 회의 내용을 실시간으로 기록·분석하여, 팀의 의사결정 속도를 높이고 업무 연속성을 확보함

19. 참고문헌

번호	저자	연도	논문제목	주요 내용	출처
1	Qian et al.	2023	Communicative Agents for Software Development	소프트웨어 개발을 역할 기반 다중 에이전트 협업으로 수행하는 프레임워크 제안	arXiv
2	Hong et al.	2023	MetaGPT	소프트웨어 개발 프로세스를 메타 프로그래밍 기반 다중 에이전트로 구조화	arXiv
3	Yao et al.	2023	ReAct	추론(Reasoning)과 행동(Acting)을 결합해 LLM의 문제 해결 능력 향상	ICLR
4	Lewis et al.	2020	RAG	외부 지식 검색과 생성을 결합해 지식 집약적 NLP 성능 향상	NeurIPS
5	Gao et al.	2023	RAG Survey	LLM 기반 RAG 구조, 학습 방식, 평가 방법을 체계적으로 정리	arXiv
6	Ouyang et al.	2022	InstructGPT	인간 피드백(RLHF)을 통해 지시 따르기 능력을 학습	NeurIPS
7	Rafailov et al.	2023	DPO	강화학습 없이 선호 데이터만으로 언어모델 정렬 수행	NeurIPS
8	Schick et al.	2023	Toolformer	언어모델이 스스로 도구 사용 데이터를 생성해 학습	arXiv
9	Zhang et al.	2020	BERTScore	BERT 임베딩을 활용한 의미 기반 텍스트 생성 평가 지표 제안	ICLR
10	Zheng et al.	2023	MT-Bench	LLM을 평가자로 활용해 챗봇 성능을 비교·평가하는 벤치마크 제안	NeurIPS

Thank you

AI 기반 컨텍스트 비서 에이전트를 통해 회의 생산성을 극대화하고, 실시간 회의 인사이트를 조직 지식 자산으로 전환하는 방안 제시