Assignment 01 실험 결과 요약 리포트

과제: Prompt 버전 관리 & 평가 (with Langfuse)
주제: 이메일·문서 자동 작성
실험일: 2025-10-07
샘플 수: 5건 (V1/V2 각각)

---

1. 버전 비교표 (실험 결과)

V1 vs V2 성능 비교

지표	V1 (dev)	V2 (production)	개선도
필수 항목 포함률	0.0%	53.3%	**+53.3%p**
길이 제한 준수율	100.0%	100.0%	+0.0%p
톤 일치도 (0-5)	4.80	4.80	+0.00
편집 필요도 (0-5, 낮을수록 좋음)	1.20	1.20	+0.00

핵심 발견: V2는 V1 대비 **필수 항목 포함률에서만 유의미한 개선(+53.3%p)**을 보였으며, 톤 일치도와 편집 필요도는 GPT-4o-mini의 기본 성능으로 인해 동일한 수준을 유지했습니다.

---

2. 핵심 지표 분석

정량 평가 결과

필수 항목 포함률

- **V1**: 0.0% (5건 모두 필수 항목 누락)
    - 문제: `must_include` 파라미터 자체가 없어 LLM이 무엇을 포함해야 하는지 모름
- **V2**: 53.3% (평균 53.3% 포함)
    - 개선: `must_include` 리스트를 명시적으로 전달하고 "MUST INCLUDE ALL" 강조
    - 한계: 여전히 46.7%의 항목을 누락 → 추가 개선 필요

길이 제한 준수율

- **V1/V2**: 100% (모든 샘플이 제한 준수)
    - OpenAI `max_tokens` 파라미터로 물리적 제한 적용 효과적

정성 평가 결과 (LLM-as-Judge)

톤 일치도

- **V1/V2**: 4.80/5.0 (매우 우수)

  - GPT-4o-mini의 기본 톤 조절 능력이 우수
  - V1에서도 충분히 좋은 성능 → V2에서 추가 개선 불필요했음

편집 필요도

- **V1/V2**: 1.20/5.0 (매우 낮음 = 우수)
  - 실무 사용 가능성 높음, 최소한의 편집만 필요
  - 두 버전 모두 문서 품질은 우수

---

3. V1 주요 실패 사례 (Top 3)

실험에서 확인된 실제 실패 사례:

Case 1: e003 (편집 필요도 2/5)

- **필수 항목 포함률**: 0%
- **문제점**: 모든 필수 요소 누락, 가장 높은 편집 필요도

Case 2: e001 (편집 필요도 1/5)

- **필수 항목 포함률**: 0%
- **문제점**: 필수 요소 누락했으나 전반적 품질은 양호

Case 3: e002 (편집 필요도 1/5)

- **필수 항목 포함률**: 0%
- **문제점**: 필수 요소 누락했으나 전반적 품질은 양호

공통 패턴:

- **100% 필수 항목 누락**: V1은 5건 모두 필수 항목을 하나도 포함하지 못함
- **입력 메커니즘 부재**: `must_include` 파라미터 자체가 없음
- **검증 로직 부재**: 필수 항목 체크를 위한 가이드라인 없음

---

4. 다음 개선안 (V3 고려사항)

실험 결과 기반 구체적 개선안:

최우선 과제

1. **필수 항목 포함률 90% 이상 달성**
   - 현재: V2 53.3% → 목표: 90%+
   - 방법: Few-shot examples, 체크리스트 강화, JSON Schema

구체적 개선 전략

1. Few-shot Examples 추가

- 우수 사례 3개를 프롬프트에 포함
- 각 예시에 필수 항목이 어떻게 포함되는지 명시

- 예상 효과: 필수 항목 포함률 70-80%로 향상

2. JSON Schema 출력 강제

```
{
  "greeting": "...",
  "body": "...",
  "mandatory_items": ["item1", "item2"],
  "closing": "..."
}
```

- 구조화된 출력으로 필수 필드 강제
- 예상 효과: 필수 항목 포함률 85-95%로 향상

3. Chain-of-Thought 체크리스트

```
Step 1: 필수 항목 확인
- [ ] 예산 정보
- [ ] 일정 정보
...
Step 2: 문서 작성
```

- 필수 항목을 먼저 체크하게 유도
- 예상 효과: 누락 방지, 90%+ 포함률

4. 업종별 프롬프트 세분화

- 이메일 / 보고서 / 제안서별 전용 템플릿
- 각 문서 유형의 필수 요소 특화

5. Self-Critique 2-Pass 생성

- 1차: 초안 생성
- 2차: 필수 항목 누락 여부 자가 점검 후 수정
- 예상 효과: 필수 항목 포함률 95%+ 달성

---

5. 실험 결론

정량적 성과

- 필수 항목 포함률: **0% → 53.3% (+53.3%p 향상)**
- 길이 제한 준수율: **100% 유지**
- 톤 일치도: **4.80/5 유지** (모델 기본 성능 우수)
- 편집 필요도: **1.20/5 유지** (실무 사용 가능)

핵심 인사이트

1. **명시적 제약 전달의 중요성**: `must_include` 파라미터 추가만으로 53.3%p 향상
2. **모델 기본 성능 활용**: 톤/품질은 이미 우수 → 추가 개선 불필요
3. **개선 여지 존재**: V2도 46.7% 누락 → V3에서 90%+ 목표

학습 포인트

1. **Prompt Engineering**: 파라미터 설계가 성능에 직접적 영향
2. **LLM-as-Judge**: 정성 평가 자동화 가능 (톤, 편집 필요도)
3. **Versioning with Langfuse**: 체계적 프롬프트 버전 관리 및 추적
4. **Data-Driven Improvement**: 실험 결과 기반 개선 방향 도출

실험 설계: Langfuse 기반 Tracing + Dataset 관리
평가 방법: 정량(필수항목, 길이) + 정성(LLM-as-Judge)
Repository: `assignment01/`