

SK 네트워크 Family AI 과정 5기

데이터 전처리 인공지능 학습 결과서

□ 개요

- 산출물 단계 : 데이터 전처리
- 평가 산출물 : 인공지능 학습 결과서
- 제출 일자 : 2025-02-13
- 깃허브 경로 : <https://github.com/SKNETWORKS-FAMILY-AICAMP/SKN05-FINAL-4TEAM.git>
- 작성 팀원 : 박보람

개요	<ol style="list-style-type: none">1. 비교 목적<ol style="list-style-type: none">a. 일반 GPT 모델과 프롬프트 최적화 모델의 성능 비교b. 프롬프트 엔지니어링이 응답 품질에 미치는 영향 분석2. 평가 기준<ol style="list-style-type: none">a. 정확성(BLEU, ROUGE)b. OPEN AI GPT 모델의 응답 검토3. 실험 방법<ul style="list-style-type: none">. 동일한 데이터 세트를 사용하여 두 모델을 평가
답변 요약 모델	<p>텍스트 보정 비교</p> <ul style="list-style-type: none">• 예제 데이터 세트• “저는 프로젝트 클라우드 인프라 담당했습니다 기존 온프레미스에서 클라우드로 옮기는 과정에서 aws 써서 아키텍처 설계 비용 최적화 위해 오토 스케일링 서버 리스 도입했고요 테라폼으로 인프라 코드로 배포 자동화 진행 배포 시간 70 프로 줄었고 비용도 30 프로 낮아짐 추가로 ci 씨디 구축해서 배포 주기 단축함 2 주에서 3 일로요 마이그레이션 때 데이터 무결성 유지 위해 백업 전략 했고 모니터링도 도입 장애 최소화 협업해서 데브옵스 문화 정착”• 출력 비교: 두 모델이 생성한 텍스트 비교 (문장 구조, 의미 보존, 가독성) <ol style="list-style-type: none">1. 일반 GPT 모델 (Zero-shot)<ul style="list-style-type: none">• 저는 프로젝트에서 클라우드 인프라를 담당했습니다. 기존의 온프레미스에서 클라우드로 이전하는 과정에서 AWS 를 이용해 아키텍처를 설계하고, 비용 최적화를 위해 오토 스케일링과 서버 리스를 도입했습니다. 또한, 테라폼을 사용하여 인프라를 코드로 배포하는 자동화를 진행했고, 이로 인해 배포 시간이 70% 단축되었으며 비용도 30% 줄었습니다. 추가로, CI/CD 를 구축하여 배포 주기를 2 주에서 3 일로 단축했습니다. 마이그레이션 과정에서 데이터의 무결성을 유지하기 위해 백업 전략을 세우고, 장애를 최소화하도록 모니터링도 도입했습니다. 이 모든 과정을 통해 협업을 통한 데브옵스 문화를 정착시키는데 기여했습니다.2. 프롬프트 최적화 모델<ul style="list-style-type: none">• 저는 프로젝트의 클라우드 인프라를 담당했습니다. 기존의 온프레미스에서 클라우드로 전환하는 과정에서 AWS 를 사용하여 아키텍처를 설계하고, 비용 최적화를 위해 오토 스케일링과 서버리스를 도입했습니다. 또한, 테라폼을 사용하여 인프라를 코드로 관리하고

	<p>배포 자동화를 진행했는데, 이로 인해 배포 시간이 70% 줄었고 비용도 30% 절감되었습니다. 추가로, CI/CD 를 구축하여 배포 주기를 2 주에서 3 일로 단축시켰습니다. 데이터 마이그레이션 과정에서 데이터 무결성을 유지하기 위한 백업 전략을 수립하였고, 모니터링 시스템을 도입하여 장애를 최소화하였습니다. 이 모든 과정을 통해 협업을 강화하고 데브옵스 문화를 정착시키는 데 기여하였습니다.</p> <p>3. 분석 포인트</p> <ul style="list-style-type: none"> -프롬프트 최적화 모델이 보다 구조적인 문장을 생성 - 숫자 및 성과 강조 - 의미가 더 명확하게 전달됨 <p>초압축 요약 비교</p> <ul style="list-style-type: none"> • 입력 데이터 : 보정된 텍스트 사용 • 출력 비교: 일반 GPT 와 프롬프트 최적화 모델의 요약 비교 <p>1. 일반 GPT 모델 (Zero-shot)</p> <ul style="list-style-type: none"> • 프로젝트에서 클라우드 인프라를 담당해 AWS 를 활용, 오토 스케일링과 서버 리스 도입으로 비용 최적화를 이루었습니다. 테라폼을 이용한 자동화로 배포 시간과 비용을 감소시키고, CI/CD 구축으로 배포 주기를 단축하였습니다. <p>2. 프롬프트 최적화 모델</p> <ul style="list-style-type: none"> • AWS 와 테라폼으로 클라우드 전환, 배포 시간 70% 단축, 비용 30% 절감 <p>3. 분석 포인트</p> <ul style="list-style-type: none"> -프롬프트 최적화 모델이 더 구조적이고 핵심이 강조됨 - 숫자 및 구체적 성과 포함 - 음성체 스타일 유지 (일관성)
평가 및 점수	<p>BLEU 점수 비교</p> <ul style="list-style-type: none"> - 답변 보정 모델 <ul style="list-style-type: none"> 1. 일반 GPT 모델 : 0.1592 2. 프롬프트 최적화 모델 : 0.1703 - 답변 요약 모델 <ul style="list-style-type: none"> 1. 일반 GPT 모델 : 0.0000 2. 프롬프트 최적화 모델 : 0.5211 <p>ROUGE 점수 비교</p> <ul style="list-style-type: none"> - 답변 보정 모델 <ul style="list-style-type: none"> 1. 일반 gpt 모델:(precision=1.0, recall=0.777778, fmeasure=0.875) 2. 프롬프트 최적화 모델: (precision=1.0, recall=1.0, fmeasure=1.0)} - 답변 요약 모델 <ul style="list-style-type: none"> 1. 일반 GPT 모델 (precision=0.66667, recall=0.5, fmeasure=0.571429) } 2. 프롬프트 최적화 모델 (precision=0.66667, recall=0.5, fmeasure=0.571429) } <p>사용자 만족도 평가</p> <ul style="list-style-type: none"> - 평가 방식: OPEN AI GPT 모델의 질문 및 응답 검토. - 점수(5 점 만점): - 답변 보정 모델 <ul style="list-style-type: none"> 1. 일반 GPT 모델 : 총 4.5 점(명확성 4.5, 간결성 4, 일관성 4.5, 전달력 4.5, 전문성 5)

	<p>2. 프롬프트 최적화 모델: 총 5 점 (명확성 5, 간결성 5, 일관성 5, 전달력 5, 전문성 5)</p> <ul style="list-style-type: none"> - 프롬프트 최적화 모델이 더 간결하면서도 명확하게 핵심 내용을 전달하고 있습니다. 특히 불필요한 단어를 줄여 가독성이 향상되었으며, 문장 간 연결이 매끄러워 일관성이 뛰어납니다. 또한, 결과를 강조하는 방식도 더 효과적이라 전달력이 높아졌습니다. - 답변 요약 모델 <ul style="list-style-type: none"> 1. 일단 GPT 모델: 총 4.3 점 (명확성 4.5 , 간결성 4, 일관성 4.5, 전달력 4.5, 전문성 4.5) 2. 프롬프트 최적화 모델: 총 4.4 점(명확성 4.2, 간결성 5, 일관성 4, 전달력 4.6, 전문성 4.3) - 프롬프트 최적화 모델이 압축적인 전달이 필요한 경우 더 효과적이에요!
--	--