

# 캡스톤 디자인 | 최종결과 보고서

프로젝트 제목(국문): LLM 기반 한국어 문법 오류 교정 서비스

프로젝트 제목(영문): LLM-based Korean grammar error correction service

프로젝트 팀(장): 학번: 20191120 이름: 이용빈

프로젝트 팀(원): 학번: 20191780 이름: 육정훈

## 1. 중간보고서의 검토결과 심사위원의 '수정 및 개선 의견'과 그러한 검토의견을 반영하여 개선한 부분을 명시하시오.

1-1) LLaMA2 기반 Bllossom 모델은 13b의 파라미터를 가지고 있어 매우 큰 모델인데, 어떻게 학습할 것인가?

- LLaMA2의 token embedding, lm head layer를 제외한 선형 레이어에 LoRA Adapter 부착, rank를 조정하여 학습한다.
- Quantization을 활용하여 4bit 양자화를 이용하여 모델의 파라미터 용량을 줄여 학습한다.

위의 두 과정을 통해 단일 A6000gpu에서 학습 및 추론 가능하다.

## 2. 기능, 성능 및 품질 요구사항을 충족하기 위해 본 개발 프로젝트에서 적용한 주요 알고리즘, 설계방법 등을 기술하시오.

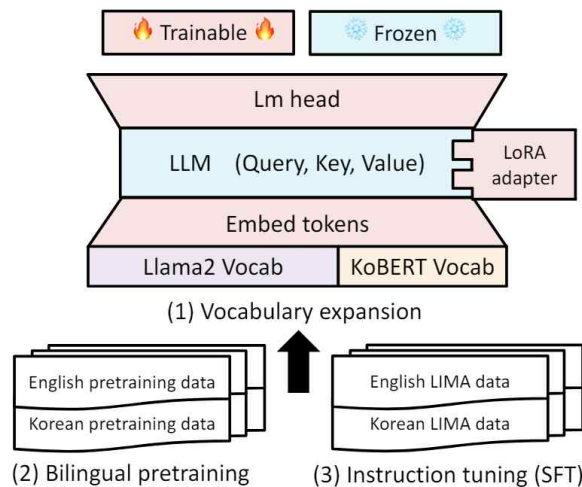


그림 1 Bllossom 모델의 구조 및 학습 절차

### 2-1) 한국어에 특화된 Bllossom 13b 모델 사용

- 기존 LLaMA2의 학습 데이터에는 한글이 0.06%밖에 포함되지 않아 Tokenizer에 한국어 Token이 부족하다. 그렇기 때문에 한글이 제대로 토큰화되지 않으며, 한국어를 공부하는 외국인이 작성한 데이터인 L2의 경우 정도가 심한 문법 오류가 존재해 이러한 단점이 더욱 부각된다. KoBERT의 Vocab을 이용하여 Vocab Extension을 진행한 모델인 Bllossom 모델을 선택하여 모델이 다양한 맞춤법 오류에 대응할 수 있도록 하였다.

### 2-2) L1, L2, L1+L2 SFT

- 한국인이 작성한 글의 문법 오류가 존재하는 L1, 외국인이 작성한 한국어 글에 문법 오류가 존재하는 L2, 이를 합친 L1+L2의 두 가지 오류 유형, 세 가지 데이터셋에서 모델을 학습하였다.
- Instruction의 형태로 모델에게 입력을 주어 Instruction Tuning의 형태로 학습하였다.

- ◆ 입력은 문법 오류가 존재하는 문장이 포함된 Instruction과 해당 오류가 고쳐진 문장을 합쳐 구성되며, 이때의 Instruction 부분은 loss 계산 과정에서 제외하여 모델의 잘못된 문법 학습을 방지한다.

“[INST] <<SYS>>  
 You are a helpful assistant.  
 <</SYS>>

“아래의 한국어 글에 대해 문법을 교정한 문장만 출력  
 해줘.  
 외야하면 그때 첫사랑을 만났기 때문이다.” [/INST]”

그림 2. 모델에 입력되는 Instruction의 예시

#### 2-3) AWS SFT

- ◆ 외국인이 작성한 L2 글에 대해 등급과 점수를 예측한다.
- ◆ Bllossom 70b 모델을 이용하여 글쓰기의 구조, 의미 등 다양한 방면에서의 교정을 출력할 수 있도록 학습하였다.
- ◆ Bllossom 13b 모델을 이용하여 글이 입력되면 등급과 점수를 출력할 수 있도록 학습하였다.

#### 2-4) 컨테이너 기반 웹 서비스 구축

- ◆ 애플리케이션의 각 구성 요소를 Docker 컨테이너로 패키징하였다. 웹 서버, 데이터베이스, 모델 서버 등을 각각의 컨테이너로 구성하여 모듈화하고, 독립적으로 배포 및 관리할 수 있도록 설계하였다. 추후 트래픽에 대응하여 확장성있는 프로젝트 구축이 가능해졌다.
- ◆ Docker Compose를 사용하여 여러 컨테이너로 구성된 멀티 컨테이너 애플리케이션을 정의하였다. 이로 인해 서비스의 설정을 하나의 파일에서 관리할 수 있으며, 하나의 명령어로 모든 서비스를 일괄적으로 실행하거나 중지할 수 있다.
- ◆ nginx를 리버스 프록시로 사용하여 각 컨테이너 간의 보안 연결을 설정하고 백엔드 서버, 모델 서버가 외부에 직접 노출되지 않도록 한다. 또한 모델 서버 증축 시 로드 밸런서 역할도 수행하여 트래픽을 적절히 분산하고 안정적인 서비스를 가능하게 한다.
- ◆ mlflow를 모델 중앙 저장소로 활용하여 모델 버전 관리, 실험 설정 관리 등을 효과적으로 수행한다. 또한 각 모델의 학습 및 배포를 체계적으로 관리하여, 협업을 촉진한다.

3. 요구사항 정의서에 명시된 기능 및 품질 요구사항에 대하여 최종 완료된 결과를 기술하시오.

3-1) 전체 파이프라인

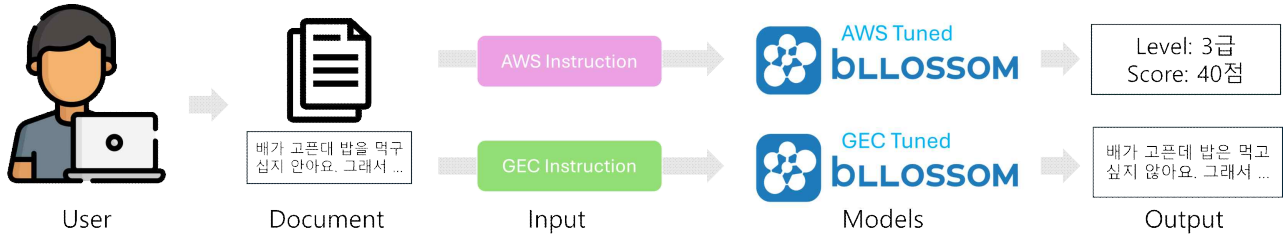


그림 3 모델 추론 파이프 라인

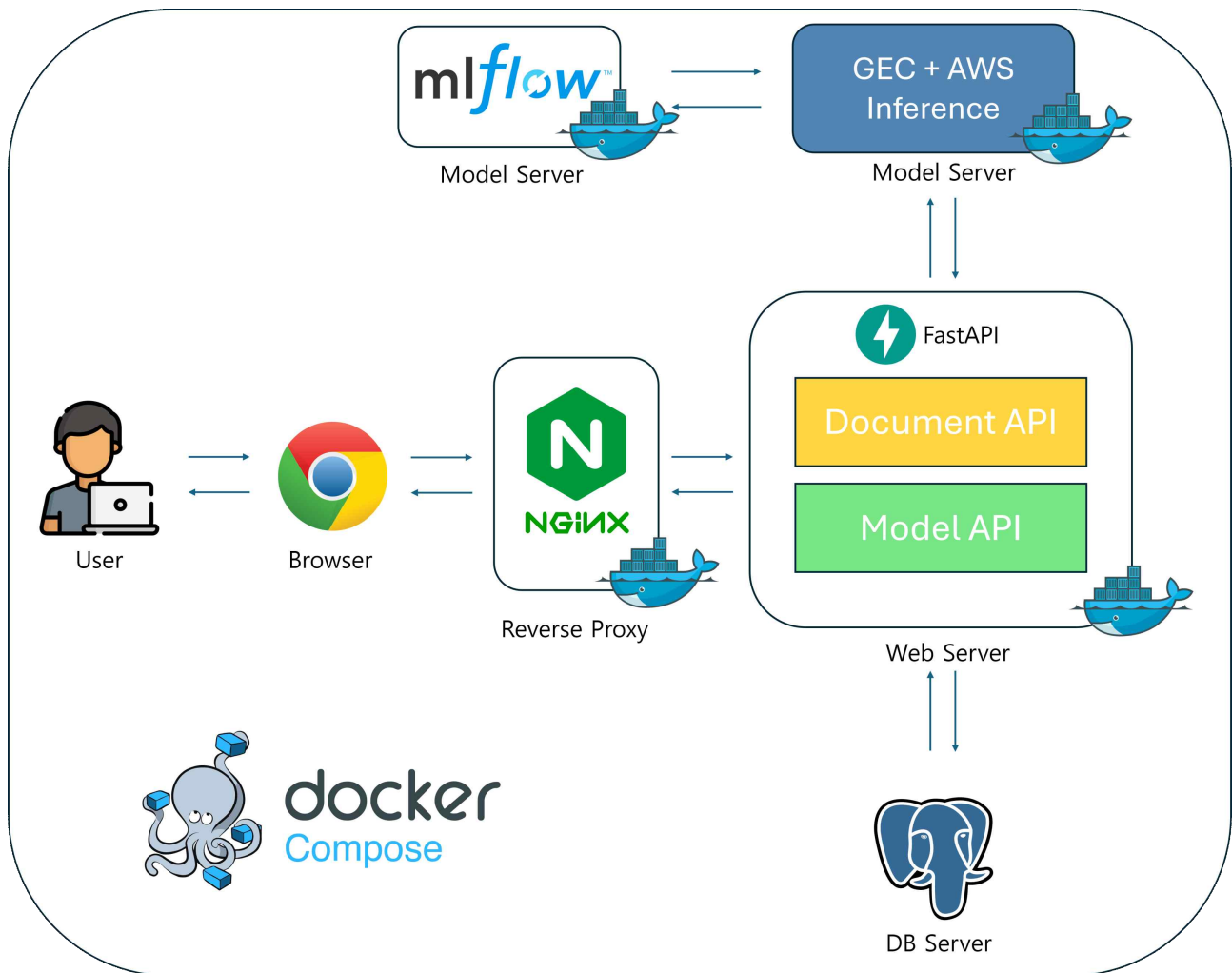


그림 4 서비스 아키텍처

### 3-2) 정성적 결과



그림 5 2-3의 Bllossom-70b 모델 추론 결과

### 3-3) 정량적 결과

#### ◆ 1. 기존 kaist kobart와 성능 비교

Model	GLEU	M <sup>2</sup>		
		Pre.	Rec.	F <sup>0.5</sup>
KoBART	67.24	75.34	55.95	70.45
BLLOSSOM	82.96	90.27	77.76	87.46

표 1. L1 유형 데이터셋에서 학습된 모델의 성능 비교

Model	GLEU	M <sup>2</sup>		
		Pre.	Rec.	F <sup>0.5</sup>
KoBART	45.06	43.35	24.54	37.58
BLLOSSOM	54.47	55.21	37.11	50.3

표 2. L2 유형의 데이터셋에서 학습된 모델의 성능 비교

Model	GLEU	M <sup>2</sup>		
		Pre.	Rec.	F <sup>0.5</sup>
KoBART	33.7	44.75	14.64	31.7
BLLOSSOM	64.34	70.12	50.03	64.91

표 3. L1, L2 유형의 데이터가 혼합된 데이터셋에서 학습된 모델의 성능 비교

세 데이터셋 모두에서 최소 9.41%, 최대 33.21%의 성능 향상을 보였으며, 두 오류 유형이 혼합된 데이터셋에서도 준수한 성능을 보임

#### ◆ 2. AWS ACC, QWK 결과

ACC(Level)	QWK(Score)
0.9774	0.5709

표 4. BLLOSSOM 모델의 AWS 실험 결과

#### 4. 구현하지 못한 기능 요구사항이 있다면 그 이유와 해결방안을 기술하시오,

최초 요구사항	구현 여부(미구현, 수정, 삭제 등)	이유(일정부족, 프로젝트 관리미비, 팀원변동, 기술적 문제 등)
		해당 사항 없음

#### 5. 요구사항을 충족시키지 못한 성능, 품질 요구사항이 있다면 그 이유와 해결방안을 기술하시오.

분류(성능, 속도 등) 및 최초 요구사항	충족 여부(현재 측정결과 제시)	이유(일정부족, 프로젝트 관리미비, 팀원변동, 기술적 문제 등)
		해당 사항 없음

#### 6. 최종 완성된 프로젝트 결과물(소프트웨어, 하드웨어 등)을 설치하여 사용하기 위한 사용자 매뉴얼을 작성하시오.

##### 6-1) 데이터 및 코드 준비

- ◆ L1, L2, L1+L2 Dataset, m2 scorer, glue:  
[https://github.com/soyoung97/Standard\\_Korean\\_GEC](https://github.com/soyoung97/Standard_Korean_GEC)
- ◆ git clone [https://github.com/Swalbak/Capstone\\_Design\\_GEC.git](https://github.com/Swalbak/Capstone_Design_GEC.git)
- ◆ pip install -r requirements.txt

##### 6-2) 모델 학습 방법

main.py

- python3 main.py --[Option]

- train\_data\_path: str

- 훈련 데이터 경로

- eval\_data\_path: str

- 검증 데이터 경로

- n\_epoch: int (default: 10)

- 훈련 에폭 수

- batch\_size: int (default: 1)

- 훈련 배치 크기

- output\_dir: str (default: ./)

- 모델 저장 경로

- do\_eval: bool (default: False)

- True일 시 모델의 학습이 끝난 후 train, eval dataset의 추론 후 .pkl 형식으로 저장한다.

### 6-3) 모델 추론 방법

make\_predict.py

- python3 make\_predict.py --[Option]
  - model\_path: str
    - 추론을 진행할 모델 저장 경로
  - data\_path: str
    - 추론 대상 데이터 저장 경로
  - save\_path: str (default: ./)
    - 추론 결과 저장 경로

### 6-4) 웹 서비스 이용 방법



그림 6 문서 편집 화면 데모

웹 사이트의 주소로 접근하여 로그인 후 문서 편집 창을 열고 문서를 작성한 뒤 check 버튼을 누르면 실시간 문법 교정 서비스를 제공받을 수 있다.

## 7. 캡스톤디자인 결과의 활용방안

### 7-1) 사회적 파급효과

- **한국어 글쓰기 교육의 혁신:** 본 프로젝트를 통해 개발된 한국어 문법 오류 교정 서비스는 학생, 외국인 학습자, 직장인 등 다양한 사용자층이 한국어 글쓰기를 보다 정확하게 할 수 있도록 돕는다. 특히, 외국인 학습자들이 한국어를 배우는 과정에서 중요한 보조 도구로써 한국어 학습의 장벽을 낮추는 데 기여할 수 있다.
- **의사소통 효율성 향상:** 보다 정확한 문법과 맞춤법을 사용한 글쓰기를 통해 일상생활, 비즈니스, 학술 연구 등 다양한 분야에서 의사소통의 질을 높일 수 있다. 이는 개인 간의 의사소통에서 오해를 줄이고, 명확하고 효율적인 커뮤니케이션을 가능하게 한다.

### 7-2) 기술적 파급효과

- **한국어 자연어 처리 기술 발전:** LLM 기반의 한국어 문법 오류 교정 모델 개발을 통해 한국어 자연어 처리(NLP) 기술의 발전에 기여할 수 있다. 본 프로젝트는 향후 다양한 한국어 NLP 응용 프로그램 개발에 중요한 기술적 기반이 될 수 있다.
- **기술 확장성:** 본 프로젝트에서 구축된 MLOps 인프라는 향후 다른 LLM 기반의 프로젝트에서도 활용 가능하며, 모델의 지속적인 개선과 자동화된 배포를 통해 효율적인 인공지능 기반 서비스 운영 환경을 제공한다.

### 7-3) 경제적 파급효과

- **산업계 응용 가능성:** 한국어 문법 오류 교정 서비스는 교육, 출판, 미디어, 기업 문서 작성 등 다양한 산업에서 응용될 수 있다. 기업에서는 보고서, 이메일, 제안서 작성 시 오류를 줄이고 문서의 품질을 향상시켜 생산성을 높일 수 있다.
- **서비스 상품화:** 본 프로젝트를 통해 개발된 웹 어플리케이션은 상용 서비스로 발전할 가능성이 있다. 구독 서비스나 API 판매 등으로 수익을 창출할 수 있으며, 이를 통해 지속 가능한 비즈니스 모델을 구축할 수 있다.
- **인공지능 산업 발전:** 한국어 특화 AI 서비스의 개발은 국내 인공지능 산업의 경쟁력을 높이는 데 기여할 수 있다. 이를 통해 국내 AI 스타트업과 기업들이 새로운 시장 기회를 창출하고, 글로벌 시장에서도 경쟁력을 갖추게 될 수 있다.

### 7-4) 기대효과

- **사용자 경험 향상:** 사용자는 본 서비스의 도움을 받아 보다 정확하고 명확한 글쓰기를 할 수 있게 되며, 이는 전반적인 사용자 경험을 향상시키는 결과를 가져올 수 있다.
- **지속적 개선 및 확장 가능성:** MLOps를 통한 지속적 학습과 모델 개선이 가능하므로, 서비스의 품질이 지속적으로 향상되고, 사용자 피드백을 반영한 빠른 업데이트가 가능하다. 또한, 향후 다른 언어로의 서비스 확장도 가능하여 글로벌 서비스로 성장할 잠재력이 있다.
- **학술적 기여:** 본 프로젝트의 결과는 학술 논문으로 작성되어 관련 학회나 학술지에 게재될 예정이다. 이를 통해 자연어 처리 분야에 학문적 기여를 하고, 후속 연구의 기반을 제공할 수 있다.