

Dacon : 난독화된 한글 리뷰 복원 AI 경진대회

읽것뉘 햅뉘햅븃쓱쥬 | 윤시호, 이유진, 강준석, 은지현

CONTENTS

01

데이터 소개 및
문제 정의

02

해독 전략 및
모델 설명

03

결과





01. 문제 정의 및 테이터 소개

1-1 문제 정의 및 데이터 소개

DACON: 난독화된 한글 리뷰 복원 AI 경진대회

식별하기 어렵게 쓴 한글 리뷰를 원래 한글 리뷰로 복원하는 AI 알고리즘 개발

INPUT

별 한 게토 앓갹땀. 왜 싸람듯릭 펄 1캐를 준눈징 킅꺅폰
싸람믄룣섞 맏룩 섹멍핼자닐 텃꿔롸눈 너무 킬교...
야믈툰 둠 변 닛씨 각꺄 싹흔 곳. 캬뵙을 20여 년 닌꺅본
곶 중 제윳 킅푼 낙꺃땡곶.



OUTPUT

별 한 개도 아깝다. 왜 사람들이 별 1개를 주는지 꺅어본
사람으로서 말로 설명하자니 댓글로는 너무 길고...
아무튼 두 번 다시 가기 싫은 곳. 캬핑을 20여 년 닌꺅본
곶 중 제일 기분 나뻘땡곶.

1-2 평가 산식

1. 일치 문자 개수: num_same=순서가 같은 위치에서 일치한 문자 개수

2. 정밀도: $\text{Precision} = \frac{\text{num_same}}{\text{예측한 문자 수}}$

3. 재현율: $\text{Recall} = \frac{\text{num_same}}{\text{정답 문자 수}}$

4. F1 Score: $F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$



02. 해독 전략 및 모델 설명

2-1 해독 전략

- 한번에 seq2seq 모델링을 하기에는 노이즈가 많음
- 문자 수준으로 모델링해서 먼저 노이즈 제거
- 노이즈 제거 이후, seq2seq 모델링
- seq2seq 모델이 복원한 문장을 후처리하여 최종 문장 완성도를 향상

INPUT	OUTPUT
절테 간면 았 되는 곳 멍뭍	절대 가면 안 되는 곳 메모
편난흰 잘 식곳 왔썅닝다. 준윙에 만집도 만학씩 좋흙 겐 갓따용.	편안히 잘 쉬고 왔습니다. 주위에 맛집도 많아서 좋은 것 같아요.
넴묵 멋직고 콩귀 좋습니닷. 췌곡웁니다!	너무 멋지고 공기 좋습니다. 최고입니다!

2-2 모델 설명



2-3 EDA 및 1차 해독: MLP를 통한 문자 수준의 해독

1. 데이터 전처리

- 난독화된 한글을 초성-중성-종성으로 자모 분리한 후, 어절 내 상대적 위치 정보 계산

2. 빈도 기반 확률 행렬 생성

- 초성, 중성, 종성의 조건부 빈도를 저장하는 행렬 생성

3. 정규화된 추론 적용

- $\mathbb{P}_{model}(a|cho, jung, jong, position)$: 클래스 a 에 대한 모델의 예측 확률

$\mathbb{P}_{data}(cho'|cho)$: 데이터에 의해 관측된 조건부 확률

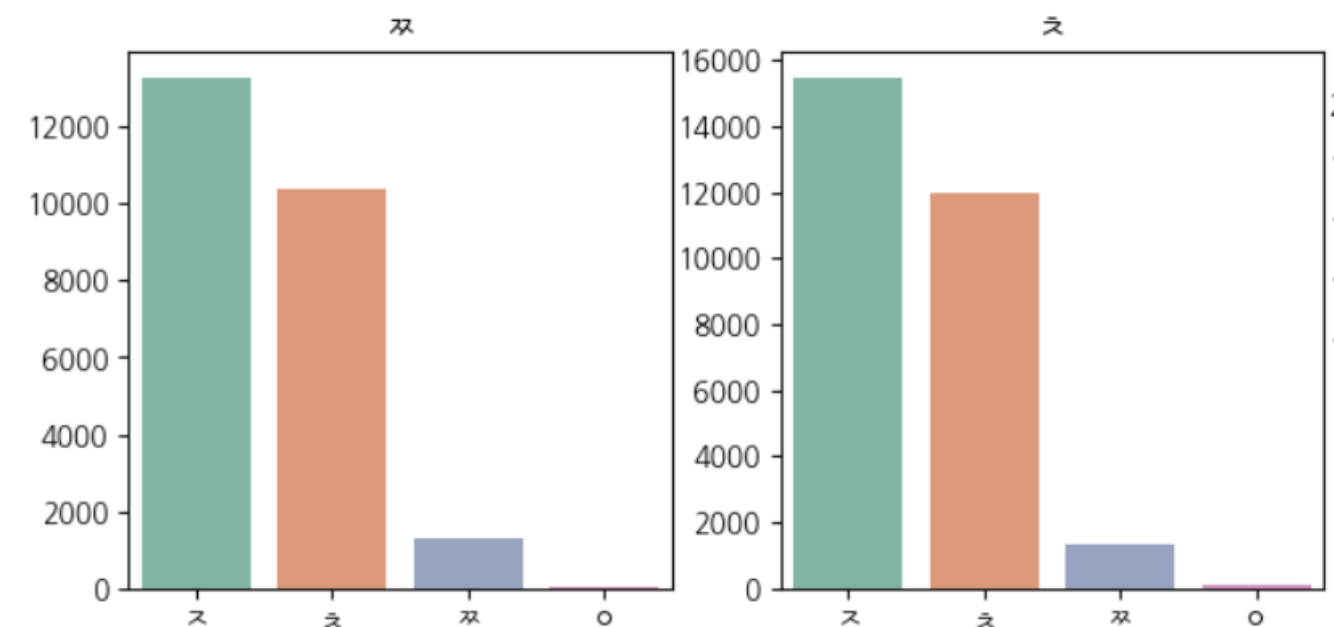
$$a_1 = \underset{a}{\operatorname{argmax}} \mathbb{P}_{model}(a|cho, jung, jong, position)$$

$$a_2 = \underset{a \neq a_1}{\operatorname{argmax}} \mathbb{P}_{model}(a|cho, jung, jong, position)$$

$$odds_{model} = \frac{\mathbb{P}_{model}(a_1|cho, jung, jong, position)}{\mathbb{P}_{model}(a_2|cho, jung, jong, position)}$$

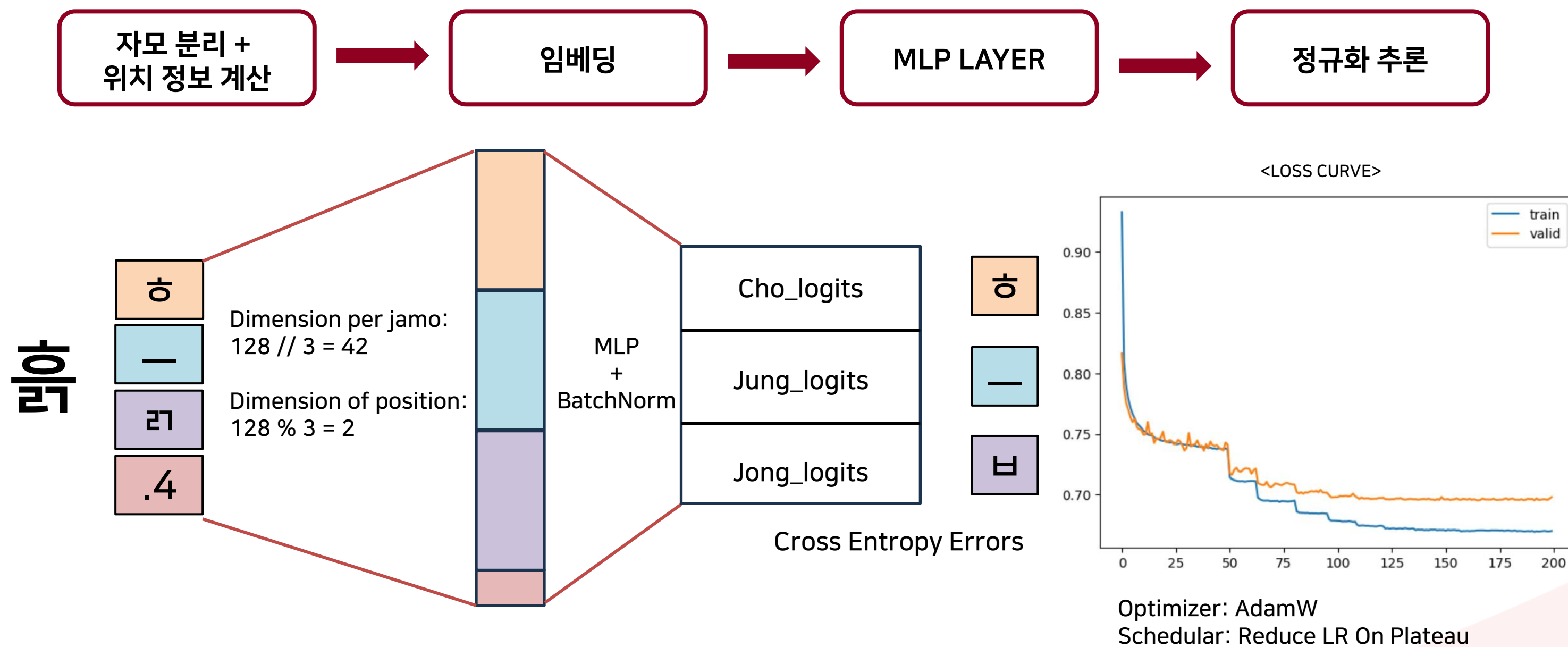
$$odds_{data} = \frac{\mathbb{P}_{data}(a_1|cho)}{\mathbb{P}_{data}(a_2|cho)}$$

$\mathbb{P}_{model}(a_2|cho, jung, jong, position) > threshold$ 이면,
 a_2 에 대한 검증 실시, 아니면 a_1 으로 의사결정



[예시] 초성 ㅈ, ㄷ

2-3 EDA 및 1차 해독: MLP를 통한 문자 수준의 해독



불 맞짚~~ 글런데 방음잃 뭉흠파네용.

불 맞지~~ 그런데 방음이 미흐파네요.

2-4 2차 해독: ELECTRA

1. 전처리, Chunking

- 특수문자 및 불완전한 한글(ㅋㅋㅋ, ㅎㅎ, ㅌㅌ) 제거
- 종결하는 문장부호(. , ! , ?)나 괄호 등의 기준으로 분리

2. SentencePiece 기반 Custom Tokenizer 학습

- SentencePiece 학습기에 corpus를 입력으로 주면, 해당 corpus의 통계에 기반한 토큰나이저 학습
- Character Coverage를 1로 제한하여, BERT나 GPT와 같은 일반적인 언어모델에서 사용하는 sub-word tokenizing이 아닌, Character level tokenizing

3. KoELECTRA 기반 문장 복원 모델 학습

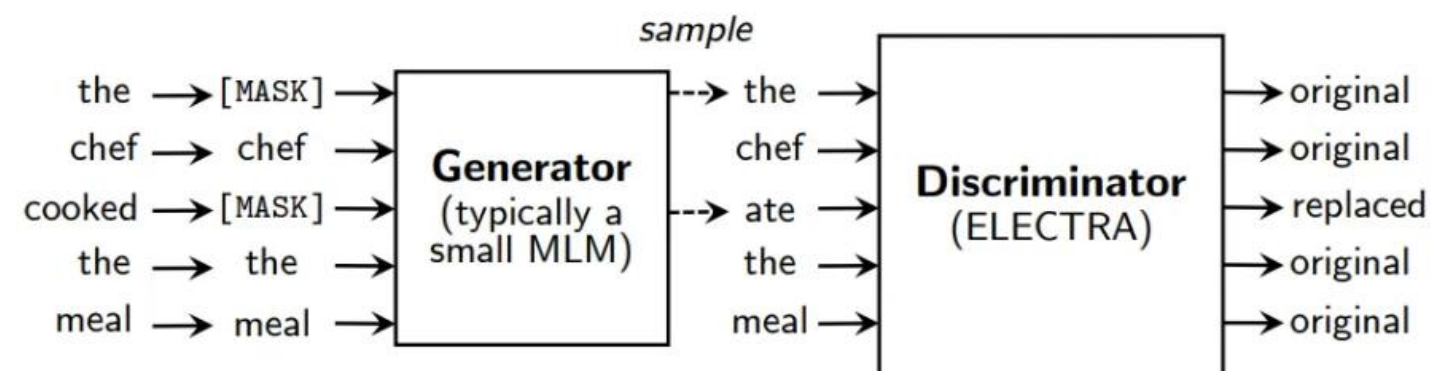
- 부자연스러운 문장을 복원하는 데에 특화된 Transformer 인코더 기반 모델

2-4 2차 해독: ELECTRA

ABOUT ELECTRA

- Generator (G) 학습 방식: Transformer Encoder 기반 모델
 - 입력 시퀀스에서 15%를 MASK 처리
 - G가 마스킹된 부분을 예측
 - 예측된 확률 분포에서 토큰 샘플링 → 새로운 시퀀스 생성 (corrupt sequence)
- Discriminator (D) 학습 방식 : Transformer Encoder 기반 모델
 - G가 생성한 corrupt sequence를 입력으로 받아 각 토큰이 진짜인지 가짜인지 이진 분류 수행
 - Loss Function: Binary Cross Entropy 사용

Replaced Token Detection(RTD)



2-4 2차 해독: ELECTRA

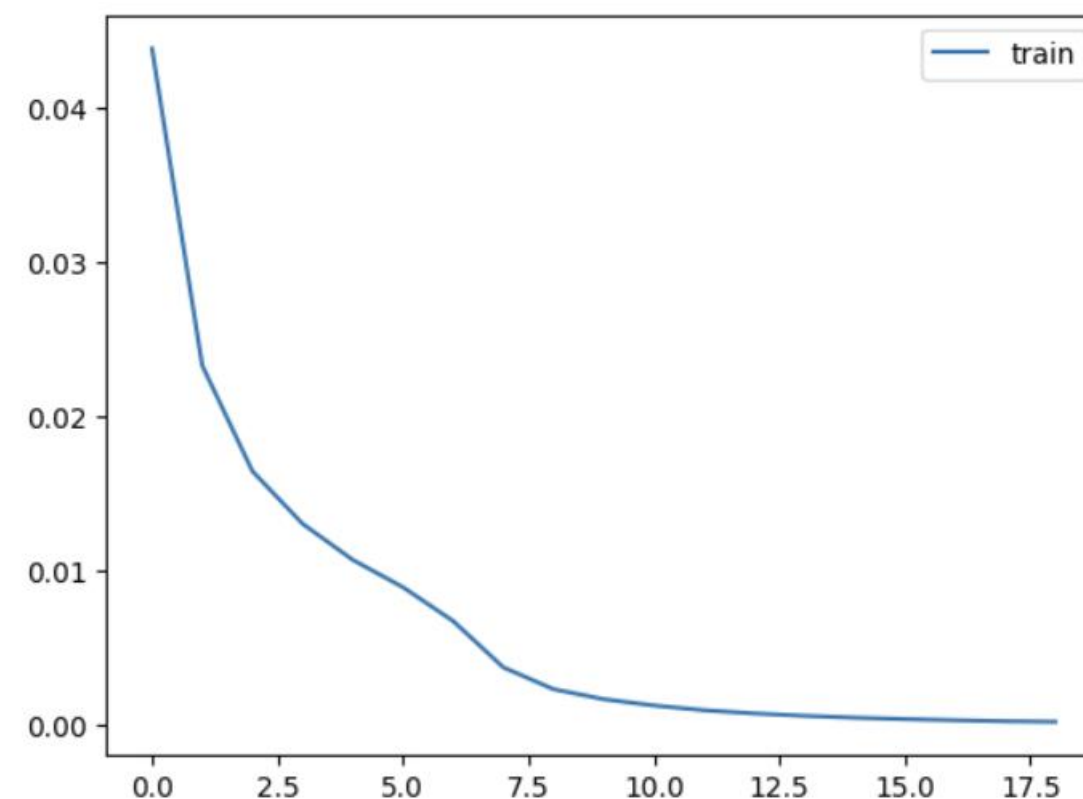
1. Training Config

- Loss / Optimizer : Cross Entropy Error / AdamW
- Scheduler: Linear Scheduler with warmup (warm-up ratio 0.1)
 - Transformer 기반의 언어모델에서 자주 사용하는 Scheduler
 - Warmup step동안 학습률이 선형적으로 증가한 후, 다시 종료 step까지 선형적으로 감소

2. Result

- Training Loss : 0.0002 (전체 학습 20 epochs 기준) / Validation Loss : 0.0179
- TEST DATA: 조금씩 오차가 있지만, 인간은 충분히 읽고 이해할 수 있을 만큼 해독됨

<LOSS CURVE>



블 맞지~~ 그런데 방음이 미흐파네요.



뷰 맛집~~ 그런데 방음이 미후하네요.

2-4 2차 해독: KoGemma

1. 데이터 셋 준비

- ELECTRA로 해독된 데이터는 이미 성능이 높아 추가 훈련이 불필요 → MLP까지 해독된 데이터를 훈련 데이터셋으로 사용
- 너무 긴 문장 → 여러 청크로 분할, 문장부호 제거 (불필요한 변형 방지)

2. beomi/gemma-ko-7b 모델 활용

- PEFT (Parameter-Efficient Fine-Tuning)
 - LoRA, 8-bit Quantization, AMP
 - Prompting:
Your task is to transform the given obfuscated Korean review into a clear, correct sentence.
The number of words and letters per word must be observed.



- 전체 19209 문장 배치 사이즈 4, 그라디언트 누적 스텝 8 -> 1에폭 600스텝, 총 3에폭

2-4 2차 해독: KoGemma

3. 결과

- LLM(KoGemma)이 의미는 더 잘 반영하지만, 원본 문장 구조를 유지하지 않음

어절 글자 수 변화

- 원본: 컴뽀툴뤄쳐룸메 뭉겼는데 숨막뜨 TV업 몬선 뽕툼를 위웅핵섯 편힌 실렘꼬 선탄간 꺼잔야욕
- Electra: 콤포트레저룸에 묵었는데 스마트 TV에 모션 베드를 이용해서 편히 쉬었고 **선택한 거잖아요**
- koGemma: 콤포트레저룸에 묵었는데 스마트 TV에 모션 베드를 이용해서 편히 쉬었고 **선택하길 잘했어요**

임의로 어절 추가 및 어절 삭제

- 원본: 특힌 객실 1012혼눈 싹뭇실 뒤편 예열컨 싹웁귀가 삼열 종때료 토엹한 삭막한 푸가 쉬야위 **절바늘** 짜찌합늬답
- Electra: 특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 **초대로 도염한 싹막한** 뷰가 시양이 **절반을** 차취합니다
- koGemma: 특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 **창틀 쪽으로 돌출한 삭막한** 뷰가 시야를 **[삭제됨]** 차지합니다

Goal: 특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 / **초대로 도염한** / **삭막한** 뷰가 시양이 / **절반을** / **차지합니다**

- 원본 문장의 구조(어절 개수 및 각 어절의 글자 수)를 기준으로, 두 변형 문장 중에서 더 적절한 어절을 선택해 최종 문장을 만드는 것이 목표

2-5 앙상블: Seq matcher

Lcs, Levenshtein sequence matcher

- 원본, Electra, koGemma를 공백 기준 어절로 분리 (토큰화)
- difflib.SequenceMatcher로 Electra와 koGemma 간 opcode(op: equal, delete, insert, replace) 도출
 - Equal: koGemma 의 어절 그대로 사용
 - Delete: Electra 의 어절 유지
 - Replace:
 - 동일 길이:
한글 텍스트를 h2j를 통해 자모 분해 후 rapidfuzz의 fuzz.ratio (Levenshtein 기반)로 유사도 평가
원본과 더 유사한 토큰 선택
 - 길이 불일치:
DP로 LCS 계산, fuzzy_equal로 유사한 토큰 매칭 (동일 글자 수 & 유사도 임계치 이상 확인)
백트래킹 후 'equal', 'delete', 'replace' opcode 도출

2-5 앙상블: Seq matcher

- 예시)

원본:	특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 초대로 도염한 싹막한 뷰가 시야가 절반을 차지합니다
Electra:	특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 초대로 도염한 싹막한 뷰가 시야가 절반을 차지합니다
koGemma:	특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 창틀 쪽으로 돌출한 싹막한 뷰가 시야를 차지합니다
Electra:	특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 초대로 도염한 싹막한 뷰가 시야가 절반을 차지합니다
koGemma:	특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 창틀 쪽으로 돌출한 싹막한 뷰가 시야를 차지합니다
결과:	특히 객실 1012호는 사무실 뒤편 에어컨 실외기가 샤워 초대로 도염한 싹막한 뷰가 시야가 절반을 차지합니다

- Seq match 이전 koGemma 점수 0.9083 → Electra+koGemma Seq match 점수 0.9766 (Electra 단독, 0.9643)

2-5 앙상블: Perplexity 활용 최적 문장 선택

Perplexity 활용 최적 문장 선택 (llama-3-Korean-Blossom-8B)

1. ELECTRA와 KoGemma로부터 복원된 두 문장에서, 각 어절별 비교
2. 다른 어절들로부터 가능한 모든 조합의 후보 문장 생성(2의 지수적으로 증가 → 후보 수에 따라 제어)
3. 후보 문장들을 LLM의 입력으로 했을 때의 perplexity 계산 및 가장 낮은 perplexity를 가지는 문장 최종 선택

<Sentences for comparison>

ELECTRA: 직원인지 사장인지 **체크이**할 때부터 친절함 1도 **없었**구요.

KoGemma: 직원인지 사장인지 **체크인**할 때부터 친절함 1도 **없었**구요.

<Candidates>

1. 직원인지 사장인지 **체크이**할 때부터 친절함 1도 **없었**구요.
2. 직원인지 사장인지 **체크인**할 때부터 친절함 1도 **없었**구요.
3. 직원인지 사장인지 **체크이**할 때부터 친절함 1도 **없었**구요.
4. 직원인지 사장인지 **체크인**할 때부터 친절함 1도 **없었**구요.

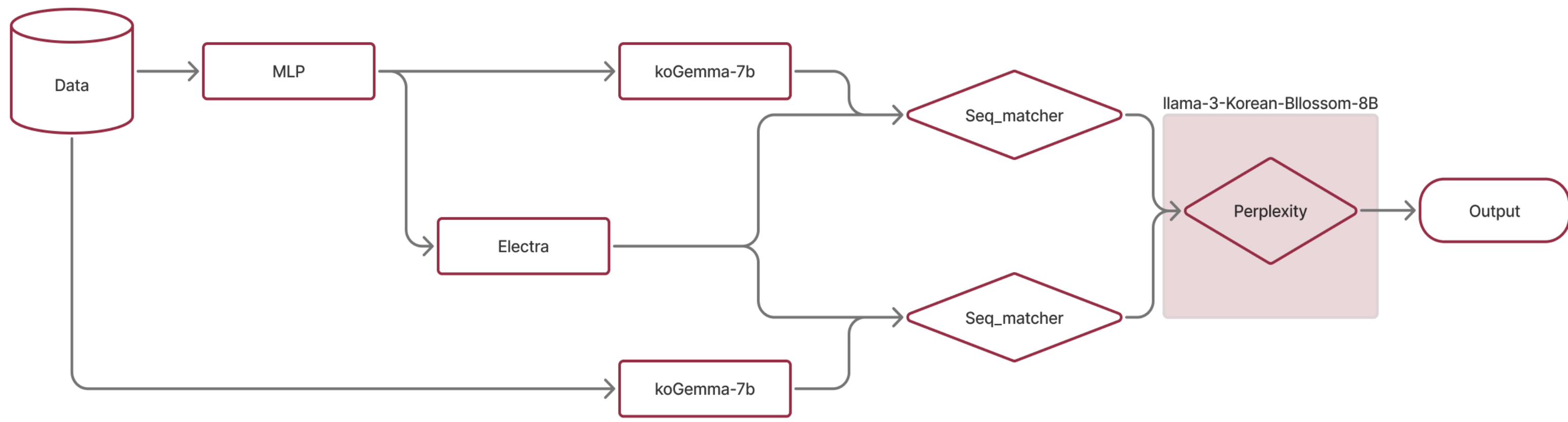
* Perplexity(혼란도)란?

언어 모델의 성능을 평가하는 지표 중 하나로, 모델이 주어진 문장을 얼마나 자연스럽게 예측하는 지에 대한 척도
값이 1에 가까울수록 모델이 확신도를 가지고 예측을 수행한 것이고, 커질수록 예측에 어려움을 겪은 것으로 해석가능



03. 결과

3-1 결과: flow chart



3-2 결과: TEST.CSV

INPUT

불 맛있~~ 그런데 방음없 뭐흠파네용.
층간 소음광 팔코닛가 이중장임 아니랏섯
팜메 파툏 쏜릴, 약침에 깔맏귀원치
카마긋원집 개속 울엇써 잠을 못 잤어요ㅠ
크런텔 뷰는 너무 좋음용~~~

DEC1 (MLP)

불 맞지~~ 그런데 방음이 미흐파네요.
층간 소음과 발고니가 이중장이 아이라서
밤에 파도 소리, 아침에 깔매기인지
가마기인지 계속 유어서 잠을 못 잤어요ㅠ
그런텔 뷰는 너무 좋아요~~~




DEC2 (ELECTRA / KoGemma)

뷰 맛집~~ 그런데 방음이 미후하네요.
층간 소음과 발코니가 이중장이 아니라서
밤에 파도 소리, 아침에 카매이인지
칸마키인지 계속 웃어서 잠을 못 잤어요ㅠ
그런데 뷰는 너무 좋아요 ~~~

Ensemble (ELECTRA + KoGemma)

뷰 맛집~~ 그런데 방음이 미흡하네요.
층간 소음과 발코니가 이중창이 아니라서
밤에 파도 소리, 아침에 깔매기인지
갈마귀인지 계속 울어서 잠을 못 잤어요ㅠ
그런데 뷰는 너무 좋아요~~~

3-3 결과: DAICON

#	팀	팀 멤버	점수	제출수	등록일
1	워것둑 햏둑햏븑햏쑤	da 자띠 시호 js	0.98178	45	6분 전
1	워것둑 햏둑햏븑햏쑤	da 자띠 시호 js	0.98178	45	6분 전
2	CSU_AI	 	0.97782	67	2일 전
3	응통	hy Ar 응통	0.97749	47	3일 전
4	파이썬초보만		0.97337	83	하루 전



Thank You