TUNiB DKTC

Korean Threatening Conversation Classification

다다익셋

팀장: 김종환

진민준 이치오 오수연

https://github.com/osy1223/DLthon_3team

Contents

팀 소개

김종환

데이터 전처리, 모델

진민준

데이터 증강, 전처리

이치오

데이터 증강, 모델

오수연

데이터 전처리, 모델

01

Introduction

01. 1. Project Introduction
01. 2. Project Planning

01. Project Introduction

문제 정의

DKTC (Dataset of Korean Threatening Conversations)

본 데이터셋은 TUNiB이 2021 인공지능 그랜드 챌린지 4차대회 음성인지 트랙에 참가하기 위해 자체적으로 제작한 데이터셋입니다.

여느 대회와는 달리 주최측이 샘플 외에 별도로 학습 데이터를 제공하지 않아 참가팀이 스스로 데이터를 만들어야 했습니다. TUNiB도 대회를 준비하는 과정에서 crowd sourcing을 통해 학습 데이터를 제작하였습니다.

NLU

인간의 언어를 분석하여 그 의미와 의도를 해석

1. 텍스트는 불필요한 요소(구두점 및 마침표 등)를 제거하기 위해 사전 처리
2. 시스템은 텍스트에서 엔티티, 키워드, 구문과 같은 주요 구성 요소를 식별
3. 문장 구조를 분석하여 단어와 개념 간의 관계를 이해
4. NLU 모델은 인식된 요소를 특정 의도 또는 목표에 매핑
5. NLU 엔진은 컨텍스트와 사용자 상호 작용 기록을 기반으로 이해를 개선

01. Project Introduction

문제 정의(분류)

train

협박 대화

갈취 대화

직장 내 괴롭힘 대화

기타 괴롭힘 대화

test

협박 대화

갈취 대화

직장 내 괴롭힘 대화

기타 괴롭힘 대화

일반 대화

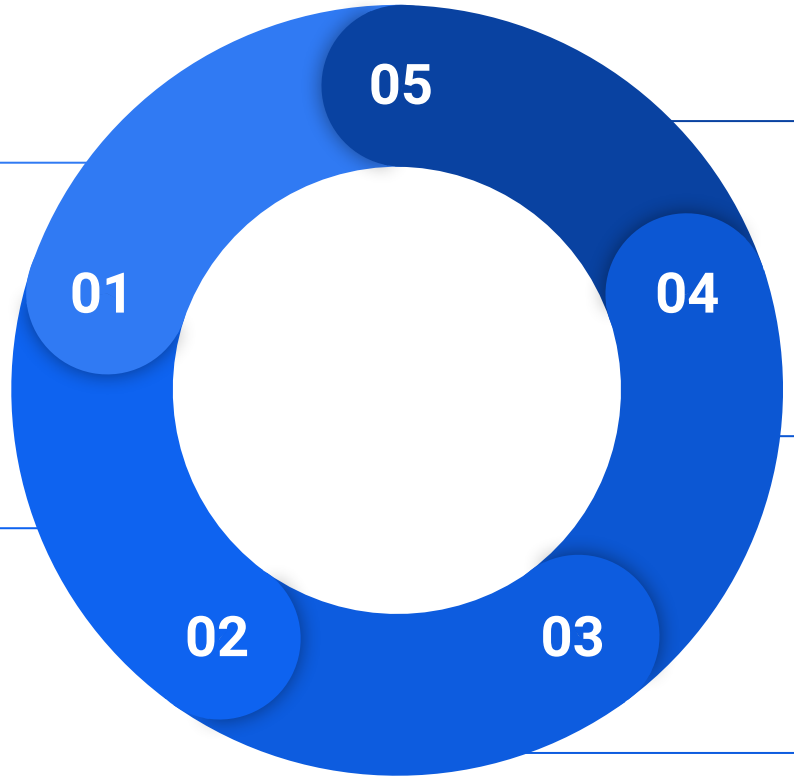
01.1. Project Planning

DATA

데이터 수집
데이터 증강
데이터 분석

Data Pre-Process

맞춤법 교정
정제
정규화



Insight

한국어 데이터 전처리
모델 튜닝

Evaluation

정성적 평가
정량적 평가

Model

Transformer
koBERT
koBERT + 1D conv
Transformer + Resnet
koBERT + Resnet

02

Data

02. 1. Dataset Directory
02. 2. Dataset Analysis
02. 3. Data stemming
02. 3. Data Augmentation

02. Data

TEXT Visualization

데이터 수집

- 기존 Train data set
- 일반대화 데이터
  AI-HUB – 기 확보
  GPT – 생성형 데이터

TEXT Correction

- 띄어쓰기 교정 (생략)
- 1차 맞춤법 교정
  (naver_api)
- 2차 맞춤법 교정 (kiwi)

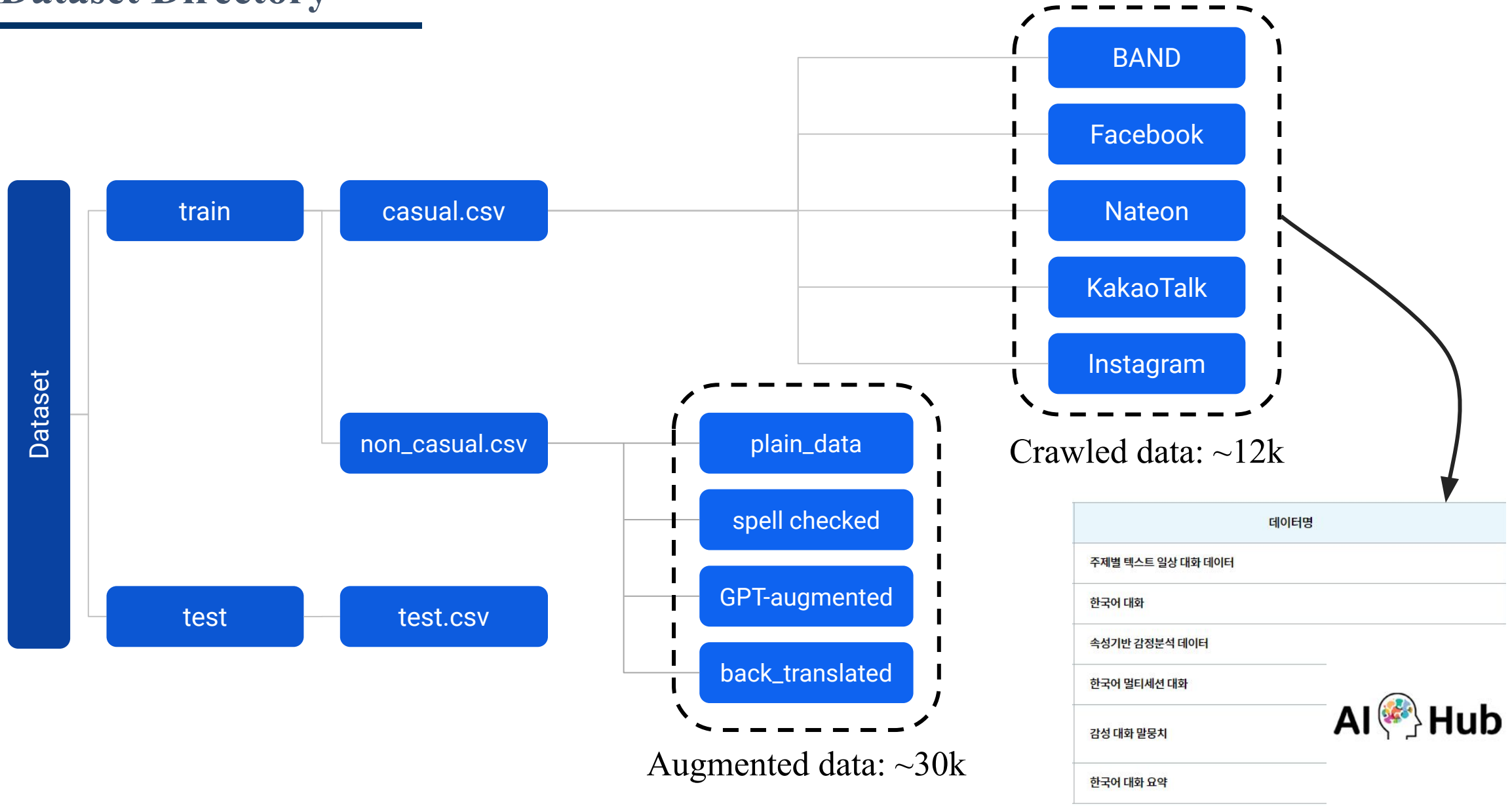
Data pre-process

- 정제, 정규화
- 불용어 처리
  (kiwi, custom)
- 토큰화
  +형태소 분석

Data augmentation

- Back-translation
- BERT-augmentation
- EDA (Easy Data Augmentation)

02.1. Dataset Directory

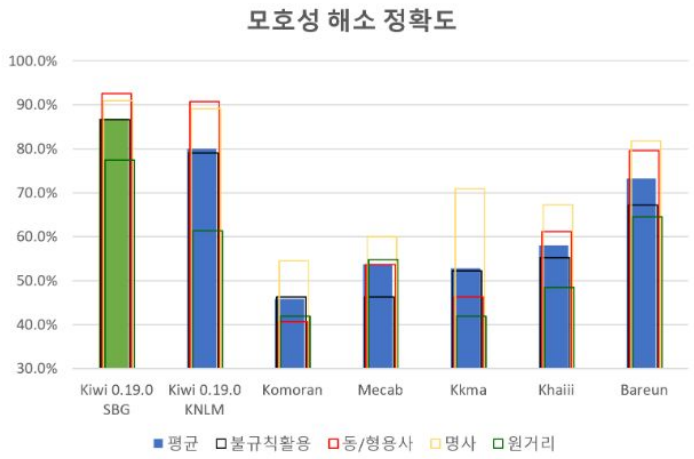


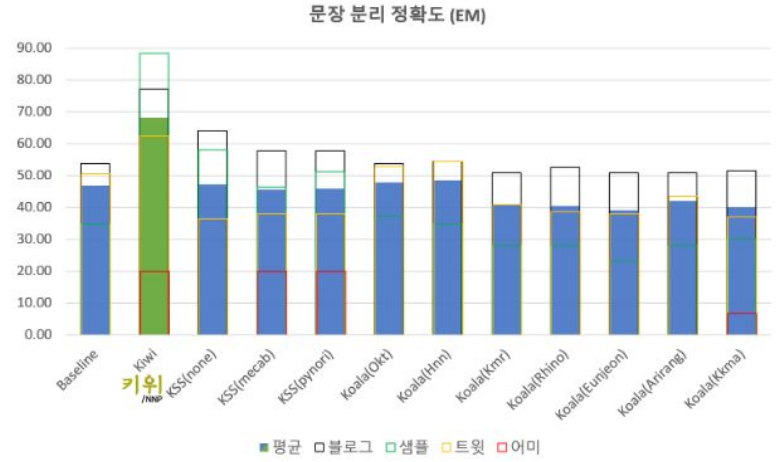
02.3. Data Stemming

형태소 분석기 비교(Mecab, Kiwi)

	4 epoch			속도비교
	f1-score		accuracy	
	0	1		
mecab	0.62	0.81	0.75	5초
khaiii	0.6	0.81	0.74	24초
kiwi*	0.62	0.8	0.74	44초

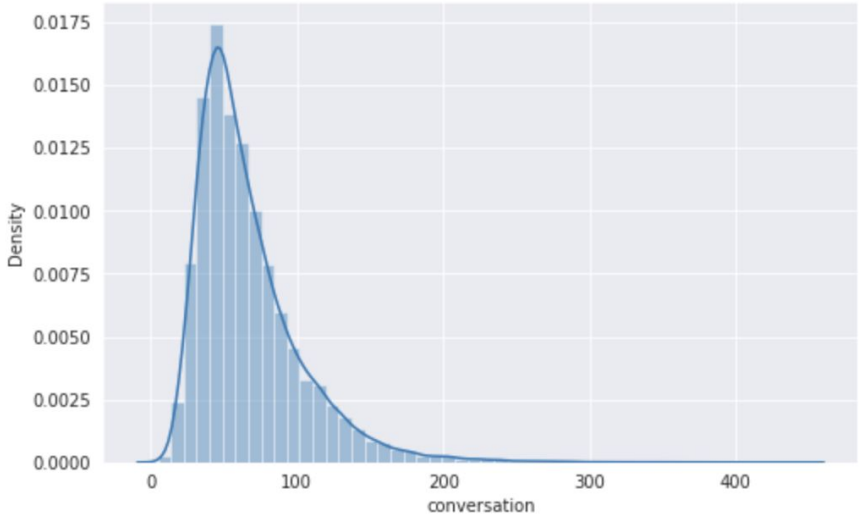
형태소 분석기	BERT						KR-BERT					
	1 epoch			4 epoch			1 epoch			4 epoch		
	f1-score		accuracy	f1-score		accuracy	f1-score		accuracy	f1-score		accuracy
	0	1		0	1		0	1		0	1	
mecab	0.47	0.81	0.72	0.62	0.81	0.75	0.68	0.85	0.79	0.69	0.84	0.79
khaiii				0.6	0.81	0.74	0.71	0.86	0.82	0.71	0.85	0.8
kiwi				0.62	0.8	0.74	0.73	0.86	0.82	0.71	0.86	0.81

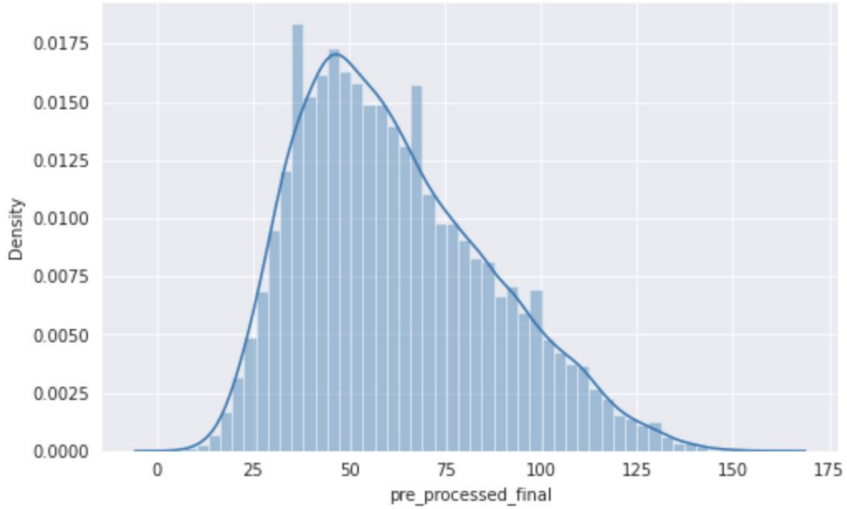




02.2. Dataset Analysis

불용어 제거를 통한 시퀀스 길이 감소 & 메모리 부하 감소





ex) 갈취 대화

02.4. Data Augmentation

- **SR (Synonym Replacement)**

  - x개의 특정 단어를 선택하여 동의어로 교체

  - English WordNet을 참고

- **RI (Random Insertion)**

  - 문장에서 임의의 동의어를 찾은 후, 문장 임의의 위치에 삽입하는 작업을 n번 수행

  - English WordNet을 참고

- **RS (Random Swap)**

  - 문장 내 특정 두 단어를 무작위로 선택하고, 위치를 바꾸는 작업을 m번 수행

- **RD (Random Deletion)**

  - 특정 단어를 p의 확률로 제거

- **BT(Back Translation)** -이미 번역된 문서를 원본 또는 원래 언어로 다시 번역(예를들어 영어 -> 한국어 -> 영어 or 영어 -> 일본어 -> 한국어)
- **BA(Bert Augmentation)** - BERT based 모델을 활용하여, 의미상 자연스러운 토큰을 삽입하거나 대체하는 형식으로 문장 augmentation을 수행
- **GPT Augmentation** - OpenAI gpt-4o API를 이용해 데이터를 4개의 톤(냉소체, 경어, 평어체, 간결체)으로 증강함 (15800 records ≈ $30)

EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks

02.4. Data Augmentation

Dataset experiment (= avg. of 3 trials):

Dataset	Validation F1-score	Least Validation Loss	Class Prediction Imbalance [0, 1, 2, 3, 4]
Plain	0.7920	0.903	104 77 128 112 79, std=19.57
Back translation	0.8138	0.827	75 109 60 209 47, std=58.3
GPT & BERT augmented	0.8170	0.623	106 119 114 93 68, std=18.25
Spell checked	0.8431	0.617	83 104 105 132 76, std=19.65
Merged	0.9265	0.305	113 97 113 105 69, std=16.32

03

Models

03. 1. Transformer

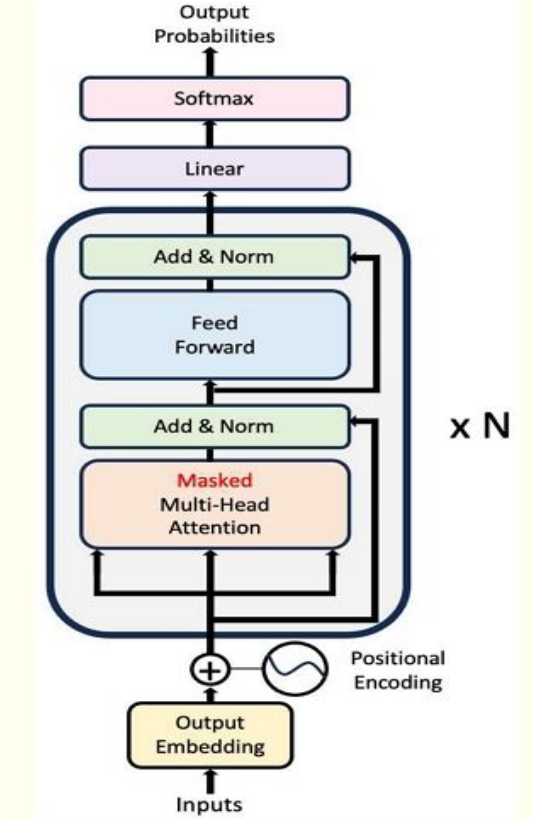
03. 2. KoBert

03. 3. KoBert Tuning

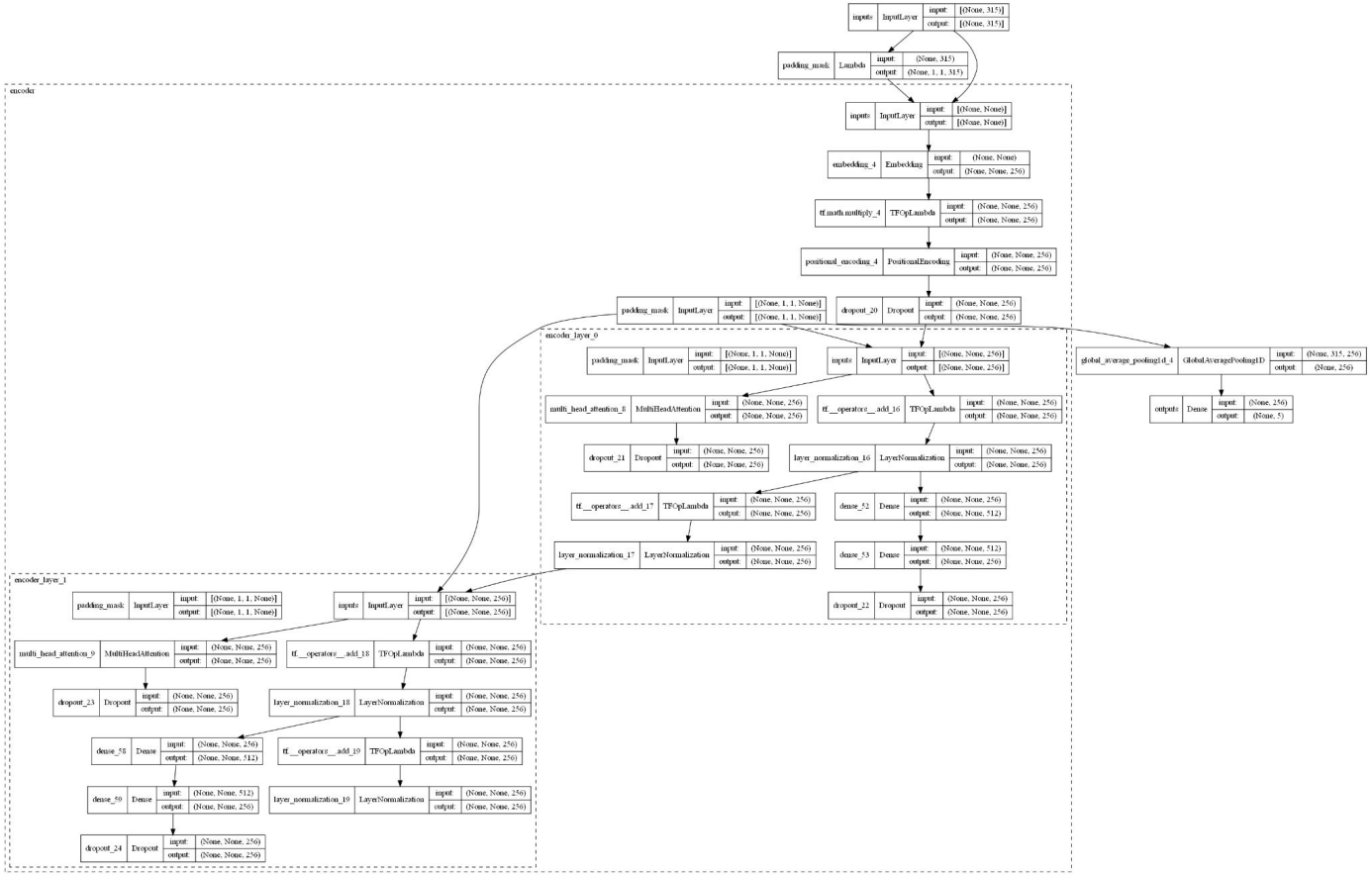
03. 4. Transformer Tuning

03.1. Transformer

Transformer:



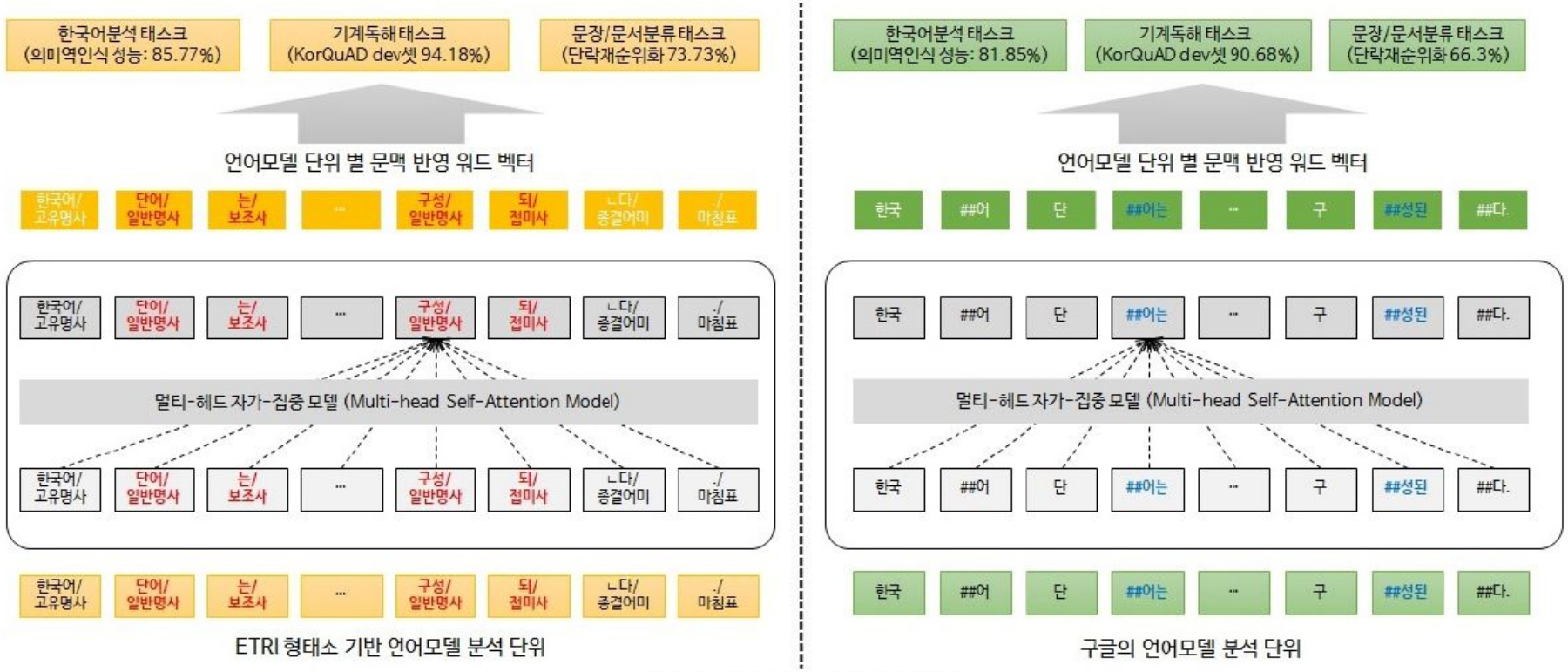
Structure of decoder



03.2. KoBERT

pre-trained koBERT:

기본 BERT의 한국어 성능 한계를 극복하기 위해 SKT Brain서 개발한 모델

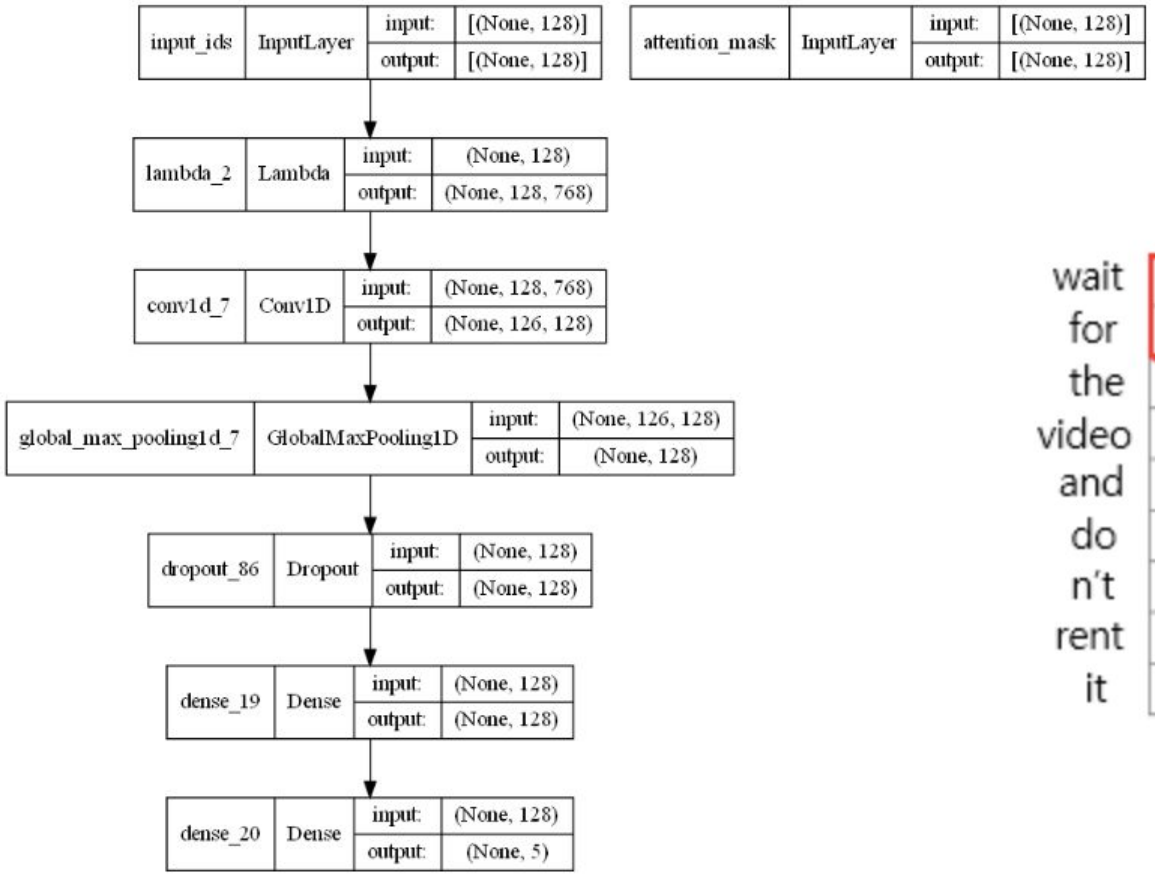


예문: 한국어 단어는 형태소로 구성된다.

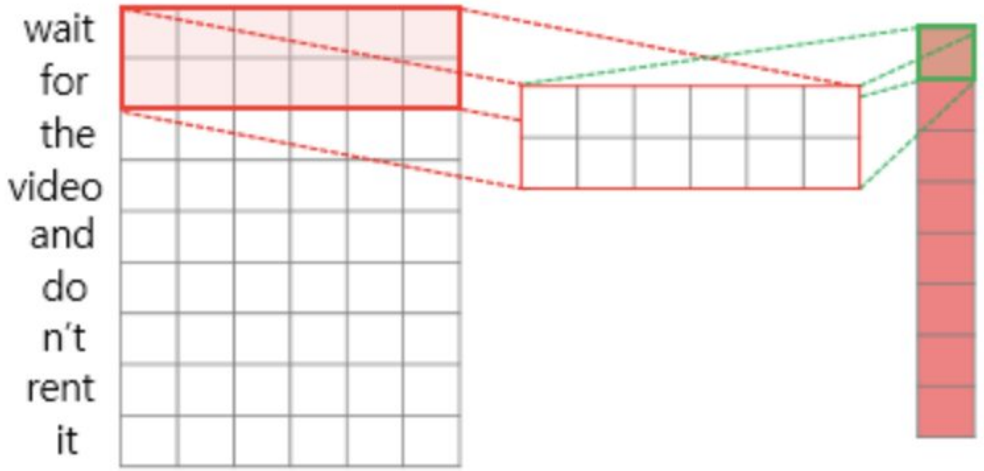
〈ETRI 형태소 기반 언어모델과 구글 언어모델 비교〉

03.3. KoBERT Tuning

pre-trained koBERT + 1D conv:



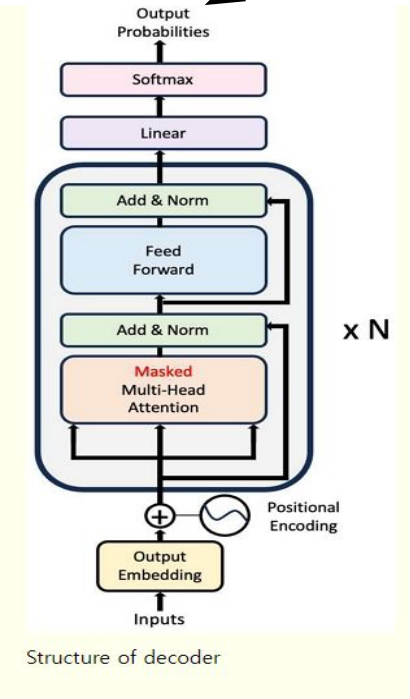
KoBert



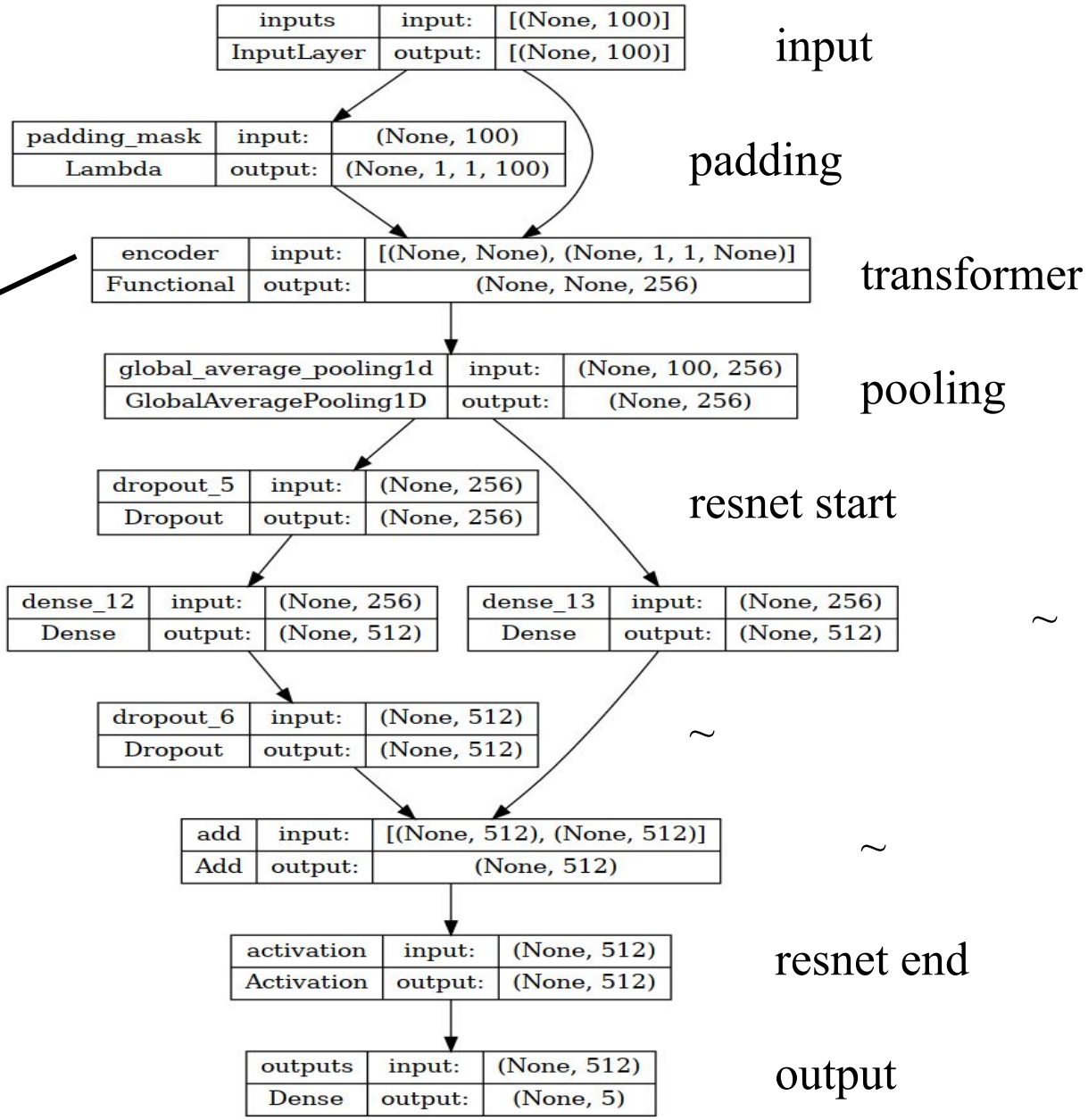
1D conv : 지역적 정보 탐색 가능

03.4. Transformer Tuning

Transformer + Resnet:



Structure of decoder



input

padding

transformer

pooling

resnet start

~

~

~

resnet end

output

04

Results

04. 1. Experiments

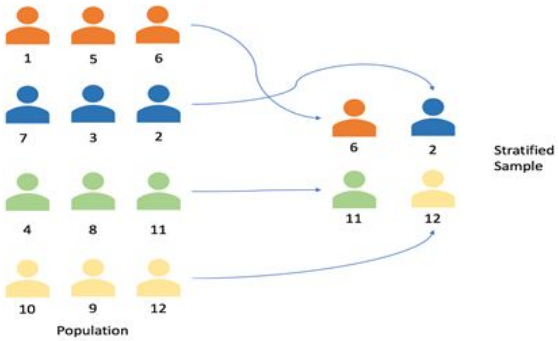
04. 2. Transformer Kobert Result

04. 3. Transformer Kobert Tuning Result

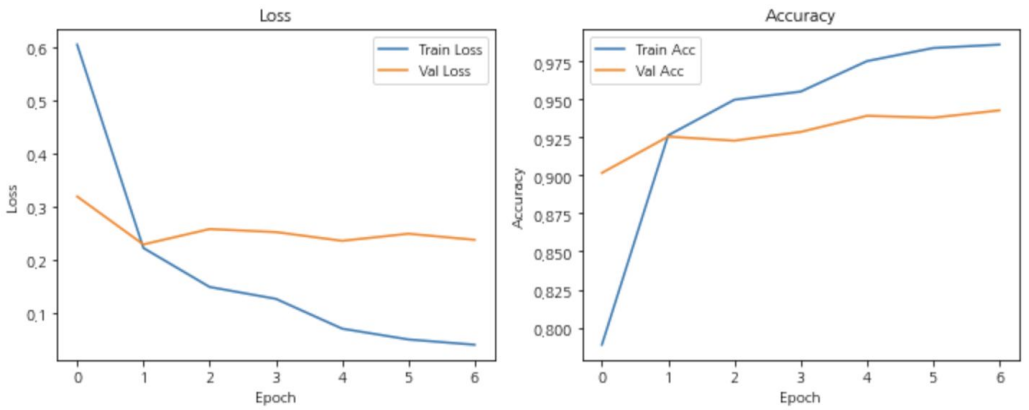
04. 4. Results

04.1. Experiments

실험 내용
1. 테스트 데이터 입력시 전처리(교정) 추가, 형태소 분석기의 성능 향상
2. 모델 학습 시 Stratified Split 이용: 데이터셋 내 클래스 분포를 학습에서 유지하여 모델의 일반화를 도움
3. 모델 Fine Tuning (레이어 동결 이용): BERT내 레이어 12개 동결(lr: 4e-5)로 먼저 훈련, 과적합 후 best_model에서 8개 동결(lr: 2e-5)로 훈련
4. 대화 내 문장 교체: 대화 중 대화의 성격이 드러나는 부분이 후반에 위치한 케이스가 많고, 최대 시퀀스 제한으로 사라지는 문제 → 대화 내에서 문장을 랜덤으로 swap함



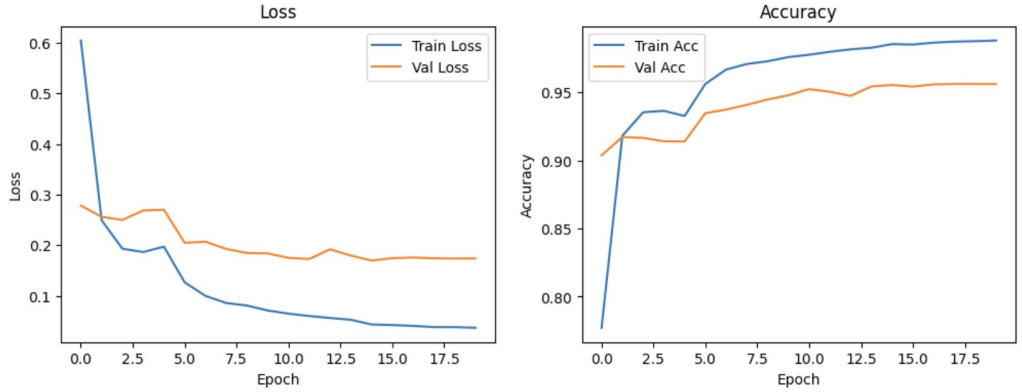
04.2. Transformer Kobert Result



Transformer

	precision	recall	f1-score	support
협박 대화	0.8626	0.9015	0.8816	731.0000
갈취 대화	0.9029	0.8690	0.8856	931.0000
직장 내 괴롭힘 대화	0.9441	0.9036	0.9234	747.0000
기타 괴롭힘 대화	0.8794	0.9134	0.8961	982.0000
일반 대화	0.9724	0.9706	0.9715	2179.0000

f1-score
0.9116

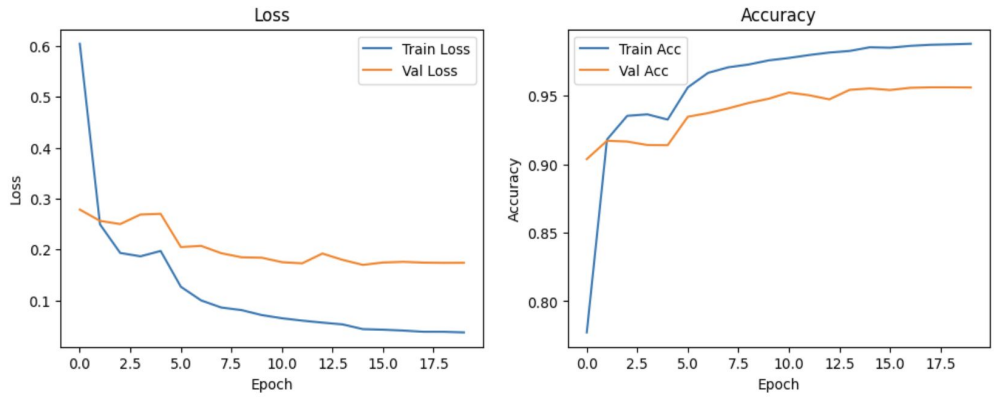


KoBert

	precision	recall	f1-score	support
협박 대화	0.9337	0.9448	0.9392	1267.0000
갈취 대화	0.9486	0.9611	0.9548	1518.0000
직장 내 괴롭힘 대화	0.9561	0.9812	0.9685	1332.0000
기타 괴롭힘 대화	0.9671	0.9242	0.9451	1622.0000
일반 대화	0.9925	0.9932	0.9928	2921.0000

f1-score
0.9601

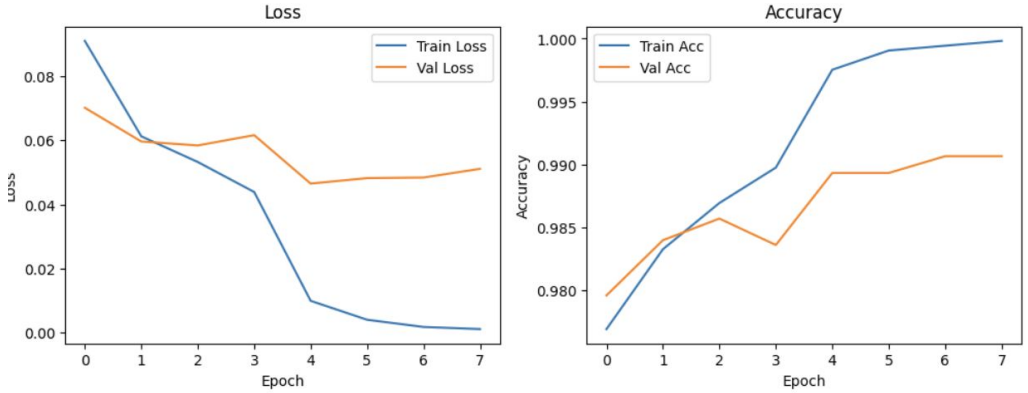
04.3. Transformer Kobert Tuning Result



Transformer + Resnet

	precision	recall	f1-score	support
협박 대화	0.9337	0.9448	0.9392	1267.0000
갈취 대화	0.9486	0.9611	0.9548	1518.0000
직장 내 괴롭힘 대화	0.9561	0.9812	0.9685	1332.0000
기타 괴롭힘 대화	0.9671	0.9242	0.9451	1622.0000
일반 대화	0.9925	0.9932	0.9928	2921.0000

f1-score
0.9601



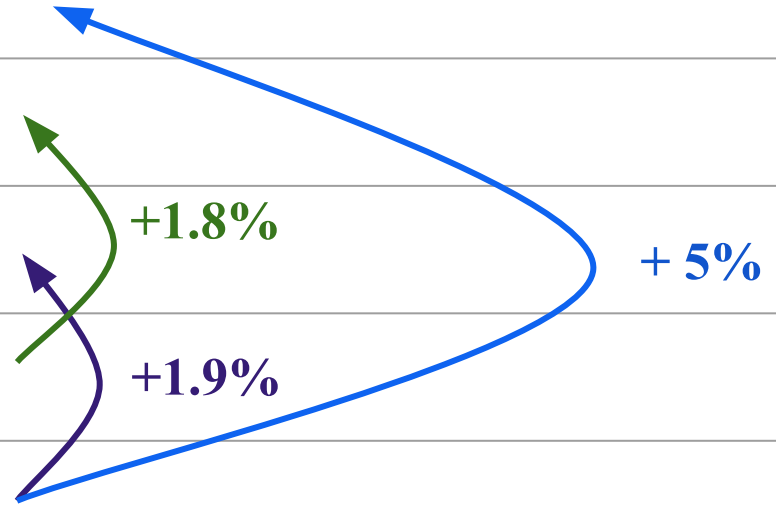
KoBert + Conv1D

	precision	recall	f1-score	support
협박 대화	0.9844	0.9748	0.9796	713.0000
갈취 대화	0.9880	0.9880	0.9880	913.0000
직장 내 괴롭힘 대화	0.9824	0.9891	0.9857	732.0000
기타 괴롭힘 대화	0.9823	0.9853	0.9838	955.0000
일반 대화	0.9979	0.9974	0.9977	1937.0000

f1-score
0.9870

04.4. Results

Models	Public score (F1-score)
koBERT + Resnet	0.8270
Transformer + Resnet	0.8076
koBERT + 1D conv	0.8017
Transformer	0.7927
koBERT	0.7872



05

Conclusion

05. 1. Conclusion

05. 2. Reference

05. Conclusion

1. Transformer + ResNet, koBERT + 1d conv, koBERT + ResNet 이 각각 F1-score 0.808, 0.802, 0.827로 상식 수준과 기준점 이상의 준수한 분류 성능을 보여줌

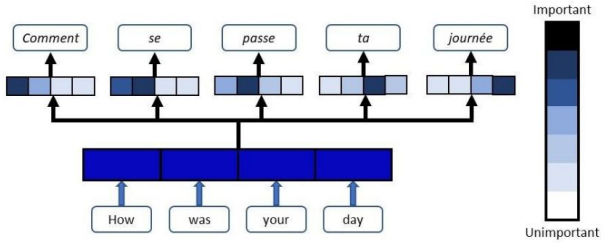
2. 텍스트 기반 분류 모델은 문장 구조 뿐만 아니라 특정 단어(Ex : 갈취 클래스 예측을 위해 "돈"에 가중치를 둠)에도 집중을 요함

3. koBERT + 1d conv은 koBERT가 제공하는 임베딩에, 로컬 패턴을 감지할 수 있는 1d conv을 이용해 성능을 기존 koBERT 대비 1.8% 개선하였음

4. Transformer/koBERT + ResNet은 언어 모델이 높은 레이어로 갈 수록 단어에 대한 집중도가 감소하는 문제와 1d conv 적용 시 계산 복잡성이 증가하는 문제를 개선하기 위해 잔차 연결을 이용해 성능을 기존 Transformer 대비 1.9% / koBERT 대비 5% 개선하였음

Reference

- *GitHub - bab2min/Kiwi: Kiwi(지능형 한국어 형태소 분석기)*
- *데이터부터 한글 텍스트 데이터 증강까지*
- *Combining ResNet and Transformer for Chinese Grammatical Error Diagnosis*
- *검색엔진의 Analyzer, 형태소분석기 ≠ 토크나이저*
- *https://arxiv.org/pdf/1511.06709*
- *https://www.tensorflow.org/tfmodels/nlp/fine_tune_bert*



Thank you for your attention