

자연어처리 2조



프로젝트 발표: Offensive language Identification in Korean

고수현 • 이규빈 • 이규진 • 최인석



PPT에 일부 혐오 표현과 욕설이 포함되어 있습니다.

Contents

1. Introduction

- Social Media Era
- Offensiveness Attack
- Offensiveness Defense

2. Tasks

- Task: Korean OLI
- 관련 논문 1: KOAS
- 관련 논문 2: Kodoli

3. Architectures

- BERT
- ELECTRA
- Multi-task Learning

4. Results

- Modeling Strategies
- Hyperparameters
- Best Accuracy
- Classifier Modeling
- Freezing Strategies
- Task Combination

Introduction

- Social Media Era
- Offensiveness Attack
- Offensiveness Defense

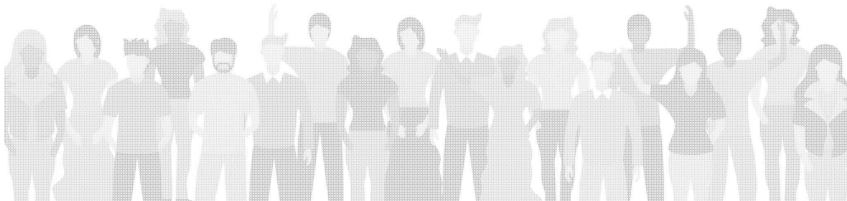
Social Media Era



온라인 커뮤니티 및 소셜 미디어 이용자 수는 매년 증가

Social Media Era

더불어, 참여자의 **욕설** 및 **혐오 표현** 게시 증가



Offensiveness Attack

"왜 이재명 인성 논란·범죄 혐의에도 열광했나"...오세훈
얘기 들어보니

입력 2024.04.29. 오전 10:52 수정 2024.04.29. 오전 10:53 기사원문

박상길 기자 TALK

101

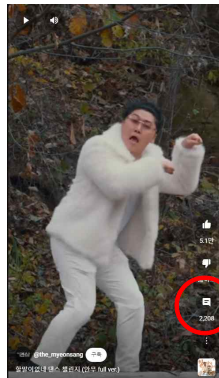
205



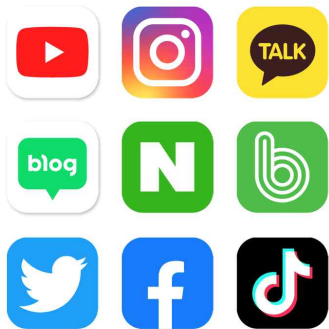
가



'설레면서 댓글창을 열었다'



Offensiveness Attack



남녀노소 접근성이 뛰어난 소셜 미디어



무분별하게 노출되는 부적절한 표현들



플랫폼 자체에서 자동으로 필터링 필요

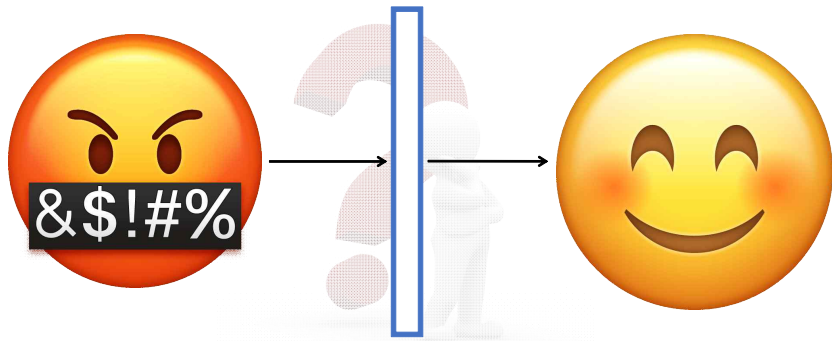
Offensiveness Defense

한국어의 언어적 특성상 필터링이 쉽지 않음

개x끼 그냥 나가 뒤져 봐야

기다려라 찾아간다 ㅋㅋ 평생 누워있게 해줄게

Offensiveness Defense



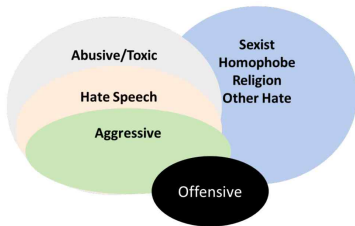
Tasks

- Task: Korean OLI
- Paper 1: KOAS
- Paper 2: Kodoli

Task: Offensive language Identification in Korean

Offensive language identification

- 문장 내에 공격적인 단어가 있는지 문장이 공격적이고 모욕적인 의미를 담고 있는지 등을 감지하고 분류해내는 과제
- 먼저 문장을 이해하고 문장이 누군가에게 모욕적인지에 대해 추론할 수 있어야 한다.
- 사회 문화적 맥락을 파악해서 특정 집단에게 불편한 의미를 담고 있을 수 있음을 인지해야 한다.



Related Works

Paper 1 :

KOAS: Korean Text Offensiveness Analysis System

**San-Hee Park^{1*} Kang-Min Kim^{3*} Seonhee Cho^{1*} Jun-Hyung Park¹
Hyuntae Park² Hyuna Kim¹ Seongwon Chung¹ SangKeun Lee^{1,2}**

¹ Department of Computer Science and Engineering ² Department of Artificial Intelligence
Korea University, Seoul, Republic of Korea

³ Department of Data Science, The Catholic University of Korea, Bucheon, Republic of Korea
carpediem20@korea.ac.kr kangmin89@catholic.ac.kr
{ehcho8564, irish07, pht0639}@korea.ac.kr
{kiipo0623, syc1013, yalphy}@korea.ac.kr

Paper 2 :

“Why do I feel offended?”

Korean Dataset for Offensive Language Identification

**San-Hee Park^{1*} Kang-Min Kim^{2*} O-Joun Lee² Youjin Kang¹
Jaewon Lee³ Su-Min Lee² SangKeun Lee¹**

¹ Korea University, Seoul, Republic of Korea

² The Catholic University of Korea, Bucheon, Republic of Korea

³ Seoul National University, Seoul, Republic of Korea

carpediem20@korea.ac.kr, {kangmin89, ojlee}@catholic.ac.kr yjkang10@korea.ac.kr
enotchi@snu.ac.kr, sumini0516@catholic.ac.kr, yalphy@korea.ac.kr

Paper 1: KOAS : Korean Text Offensiveness Analysis System

2 classification task(multi-task learning) :

abusive language detection

Positive/Neutral/Negative

Sentiment analysis

Abusive/Non-abusive

두 task를 통해 offensiveness 점수를 얻는다

(handmade) $O = \sigma(\alpha \times (y^{neg} - \max(0, y^{pos})) + \beta \times y^{ab})$

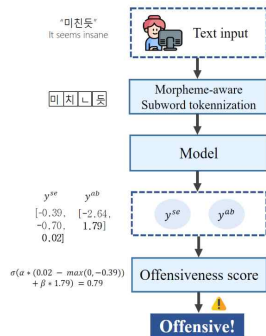


Figure 2: This is a flowchart of KOAS, from taking a Korean text input to eliciting the offensiveness score. y^{se} and y^{ab} denote output vectors of the sentiment analysis and abusiveness detection respectively.

Paper 2: "Why do I feel offended?" Korean Dataset for Offensive Language Identification

- Main :
 1. KODOLI(Korean Dataset for Offensive Language Identification) 데이터셋
 2. 멀티-태스크 러닝 프레임워크
- (Multi)Task
 - Main task : classify Offensiveness
 - 3 labels : OFFEN / LIKELY / NOT
 - Auxiliary task 1 : abusive language detection
 - 2 labels : ABS / NON
 - Auxiliary task 2 : sentiment analysis
 - 3 labels : POS / NEG / NEU
 - Hypothesis : jointly learning offensive language, abusive language and sentiment information improves the performance of offensive language identification

Main Task : classify Offensiveness

- 3 labels : OFFEN / LIKELY / NOT
 - Comment에 offensive language를 포함하고 있는지 여부를 레이블
 - Offensive : non-acceptable language를 포함하고 있는 코멘트
 - Likely Offensive : could be likely offensive(분명하지 않은/의도를 숨기고 있는 것으로 판단되는 코멘트들)
 - Not offensive : direct or indirect offense를 포함하고 있지 않는 코멘트
- Examples
 - "모던타임즈는 봐야한다 왜냐하면 찰리채플린 이니까..." - Not
 - "밥 줬나안나오네 하" - Offensive
 - "이제 한국에서 철수해라. 지겹다" - Likely

Auxiliary Task 1 : abusive language detection

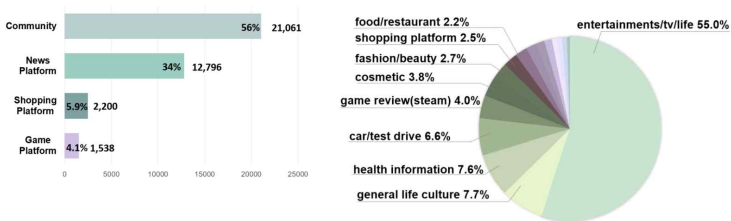
- 2 labels : Abuse / Non-abuse
 - Comment에 외재적으로 신성모독, 혐오 표현 등을 포함하고 있는지 여부를 레이블
 - Abuse : 신성모독, 혐오 표현 등을 포함하고 있는 코멘트
 - Non- Abuse : 신성모독, 혐오 표현 등을 포함하고 있지 않는 코멘트
- Examples
 - "남편이 좋아하네요. 오늘 처음사용해서 아직 잘 모르지만 일단 남편이 좋아하니 만족입니다. 많이파세요." - Non-abuse
 - "좇같은게 차단해도 댓글알람뜸" - Abuse

Auxiliary Task 2 : Sentiment Analysis

- 3 labels : Positive / Negative / Neutral
 - Comment의 긍/부정 여부를 레이블
 - Positive : 긍정적 의미를 갖는 코멘트
 - Negative : 특정 부류를 비판하고 공격하는 코멘트
 - Neutral : 사실이나 정보를 전달하는 코멘트(가치 판단 x)
- Examples
 - "배송도빠르고품질도좋아요강추요" - Positive
 - "도와달라고개씨뺏년들아 사람우습게보이냐" - Negative
 - "ㅎㅎ 난 질것같아서 케이블 스포츠채널에서 배구 봤는데.. 지상파에서 안하면 케이블 스포츠 채널로 돌려보세요" - Neutral

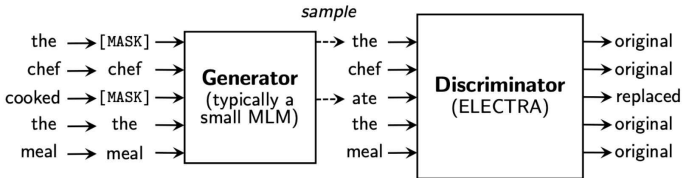
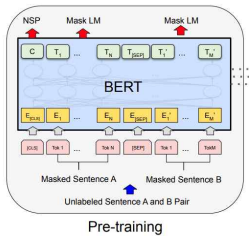
Dataset

- KODOLI (총 38,525 코멘트 with Undergrad/grad student annotation)
 - Oct 2020 ~ Dec 2020 사이의 DC-inside,
 - July 2021 ~ Sep 2021 사이의 Naver News platform(Top-ranked articles)



Model

- use PLM(Pretrained Language Model) + task specific classifier
 - Finetuning BiLSTM, CNN, KoBERT, KoELECTRA



Modeling

- Tokenizer

: morpheme-level pre-tokenization (Mecab-ko), WordPiece tokenizer

- 멀티 태스크 프레임워크:

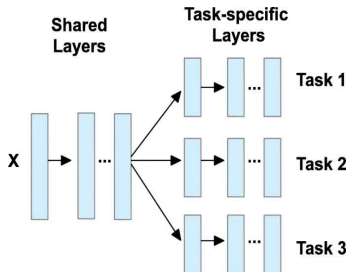
Share model weights between related tasks(parameter sharing)

→ 한 task를 학습하는 것이 다른 task를 학습하는 것에 도움을 줄 수 있다.

→ shared part(encoder layer) and task-specific parts(specific layer)

- (+) 전이 학습 프레임워크 :

인터넷 데이터로 사전학습된 PLM 모델을 파인 튜닝해서 과제를 진행



Process

- 순서

1. Embedding layer를 거쳐서 embedding matrix를 얻음
2. 다음으로 emd mat를 encoder에 입력 $\rightarrow h_1, h_2, h_3 \dots$ 을 얻는다.
3. 그 중 [CLS] token의 hidden feature를 h 로 얻는다.
4. 다음으로 task-specific layer에 입력 output logit z 를 얻는다(using softmax).
5. Calculate cross-entropy loss

$$L_{CE}(U) = \lambda_o L_{OLI}(U) + \lambda_a L_{ALD}(U) + \lambda_s L_{SA}(U)$$

Findings

- 멀티 태스크 학습이 Offensiveness 학습에 도움이 되었다.
- main task Offensiveness 예측값 중 Likely label에 대한 정확도가 가장 낮았다.
- KoELECTRA 가 가장 좋은 퍼포먼스를 보였다.

Architectures

- BERT
- ELECTRA
- Multi-task Learning

BERT: Intro

BERT란?

Transformer 인코더의 구조를 가져와 문장을 양방향으로 이해할 수 있게 만든 모델

Transformer란?

인코더-디코더 구조로 이루어져 있다.

인코더로 입력 시퀀스를 이해하고

디코더로 출력 시퀀스를 만들어간다.

인코더 구조만을 활용한 BERT는 이해를 잘 하는 모델이다.

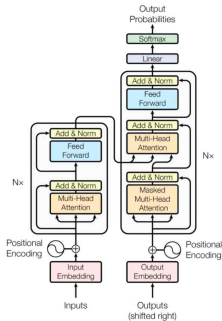


Figure 1: The Transformer - model architecture.

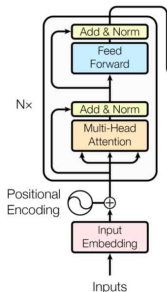
BERT: Pre-training task

이전 모델들과 BERT가 다른점

이전 모델들은 모델을 단방향으로 학습시켰다.

[SOS] l eat an apple [EOS]

하지만 트랜스포머의 self-attention은 전역적인 정보를 잘 이해하는 layer다.



BERT: Pre-training task

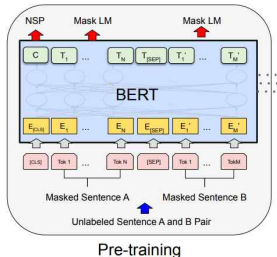
BERT는 MLM & NSP로 양방향 이해가 가능하게 했다.

MLM: 시퀀스에서 몇몇 토큰을 오염시키고 그것이 원래 무엇이었는지 예측하기

[SOS] | [MASK] an apple [EOS]

NSP: 시퀀스 2가 시퀀스 1 다음에 나오는 문장인지 아닌지 예측하기

[CLS] {Seq1} [SEP] {Seq2} Binary classification



ELECTRA: Intro

ELECTRA란?

Transformer 인코더의 구조를 가져와 문장을 양방향으로 이해할 수 있게 만든 모델

BERT와 다른 점?

BERT는 시퀀스의 일부만 학습에 활용하고

Pre-train에 사용하는 [MASK] 토큰은 실제로 사용되지 않는 토큰이라는 한계가 있다.

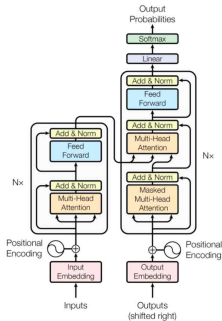


Figure 1: The Transformer - model architecture.

ELECTRA: Pre-training task

Replaced Token Detection

Generator와 Discriminator를 함께 학습시킨다.

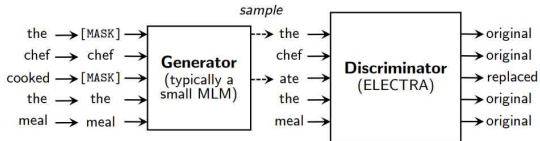
Generator

: 기존 시퀀스에서 [MASK]로 교체된
토큰이 원래 무엇이었는지 예측한다.

Discriminator

: 시퀀스 내의 모든 토큰에 대해 그것이
기존 시퀀스와 같은지 같지 않은지
예측한다.

최종적으로는 Discriminator만 사용한다.



Multi-task Learning

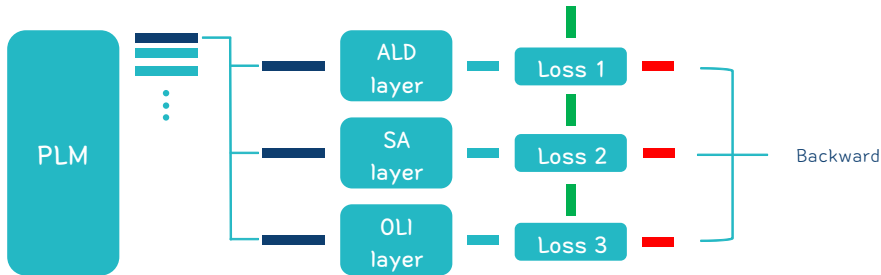
기존 논문의 방법론

Shared Part와 Task-Specific Part로 나뉨

ALD: Abusive Language Detection

SA: Sentiment Analysis

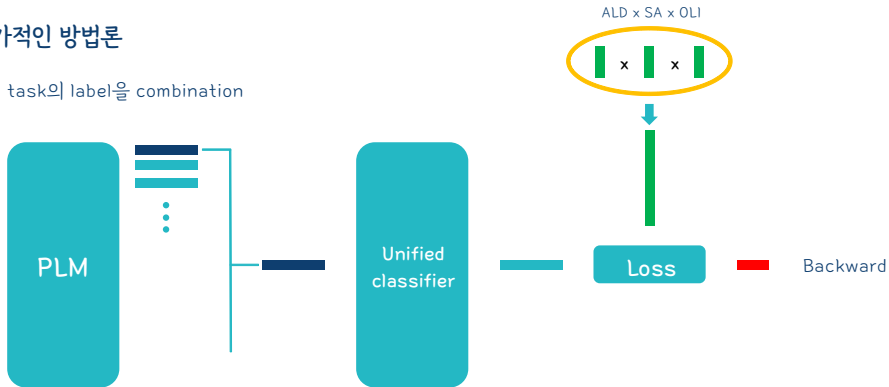
OLI: Offensive Language Detection (main)



Multi-task Learning

추가적인 방법론

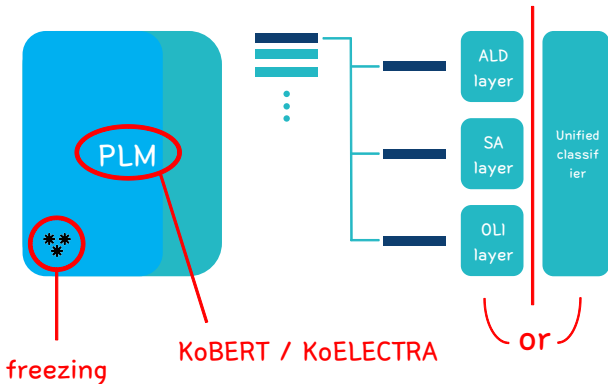
여러 task의 label을 combination



Results

- Modeling Strategies
- Hyperparameters
- Best Accuracy
- Classifier Modeling
- Freezing Strategies
- Task Combination

Modeling Strategies



Hyperparameters

전처리

- 시퀀스의 최대 길이
Ex) 64, 32 ...
- 배치 크기
Ex) 64, 128 ...
- Train / Test 비율
Ex) 80/20, 90/10 ...
- Tokenizer
Ex) Basic, Mecab ...

모델링

- Classifier Hidden Dimension
Ex) no, 512, 256 ...
- Classifier Activation F
Ex) ReLU, Sigmoid, Tanh ...
- Classifier Initialization
Ex) normal, xavier, he ...
- Classifier Dropout
Ex) 0, 0.1, 0.5 ...

학습

- Optimizer
Ex) AdamW ...
- Scheduler
Ex) Cosine ...
- Gradient Clipping
Ex) 1, 5 ...
- Learning Rate
Ex) $1e-3$...
- Epochs
Ex) 5, 10 ...

KoBERT- best

ALD	SA	OLI
91.47%	75.91%	81.38%

주요 세팅

KoBERT 12개 layer 전체 학습

Unified Classifier 사용 (label combination)

KoELECTRA – best

ALD	SA	OLI
88.86%	74.24%	80.49%

주요 세팅

KoELECTRA 12개 layer 중 마지막 11, 12만 학습

Unified Classifier 사용 (label combination)

KoBERT – Classifier Modeling

Unified Classifier

ALD	SA	OLI
91.47%	75.91%	81.38%

Specific Classifier for each Task

ALD	SA	OLI
90.62%	79.68%	78.12%

KoELECTRA – Classifier Modeling

Unified Classifier

ALD	SA	OLI
88.86%	74.24%	80.49%

Specific Classifier for each Task

ALD	SA	OLI
65.##%	70.##%	78.##%

Why Unified better than Specific?

클래스 상호 작용과 데이터 특성의 통합

부정적인 감정이 표현될 때 욕설과 공격성 수반될 확률이 높다.

Label Combination 이 모델로 하여금 클래스를 상호 고려할 수 있도록 학습했을 가능성이 존재한다.

3가지 task 는 독립적이지 않고 어떠한 상관관계가 분명 존재하기 때문.

데이터 불균형

세 클래스를 독립적으로 평가하므로 데이터 불균형이 학습과정에서 극대화됐을 가능성이 존재한다.

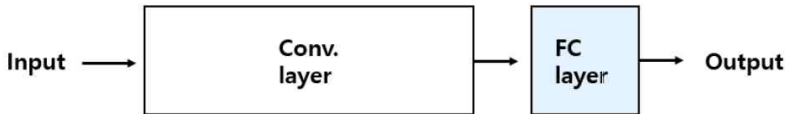
	Abuse
NON	25,941
ABS	12,584


	Sentiment
NEG	13,940
NEU	13,647
POS	10,938

	Offensiveness
NOT	25,241
OFFEN	7,676
LIKELY	5,608

Freezing Strategies

1. 사전 학습 모델의 데이터셋과 유사하고 작은 데이터셋




 : Train (LR = original LR / 10)

 : Frozen (LR = 0)

Freezing Strategies

2. 사전 학습 모델의 데이터셋과 유사하고 큰 데이터셋




 : Train (LR = original LR / 10)

 : Frozen (LR = 0)

Freezing Strategies

3. 사전 학습 모델의 데이터셋과 다르고 큰 데이터셋




 : Train (LR = original LR / 10)

 : Frozen (LR = 0)

Freezing Strategies

4. 사전 학습 모델의 데이터셋과 다른 작은 데이터셋



 : Train (LR = original LR / 10)

 : Frozen (LR = 0)

Freezing Strategies

초기 가설

같은 한국어 데이터이면서 충분히 많은 데이터셋 (38525 rows) 이므로 2번 전략을 취하는 것이 성능 가장 잘 나올 것이다.

즉, 구체적인 feature 를 학습하는 상위 레이어 일부 및 classifier 만 학습시켜도 충분할 것이다.

KoBERT – Freezing (Unified)

BERT no freezing

ALD	SA	OLI
91.47%	75.91%	81.38%

BERT 12th, 11th, 10th layer train

ALD	SA	OLI
88.99%	72.24%	78.61%

BERT 12th, 11th layer train

ALD	SA	OLI
87.95%	70.58%	77.28%

BERT 12th layer train

ALD	SA	OLI
84.83%	67.53%	74.71%

BERT no train

ALD	SA	OLI
67.12%	40.59%	64.39%

KoBERT – Freezing (Specific)

BERT no freezing

ALD	SA	OLI
64.06%	35.93%	81.38%

BERT 12th, 11th, 10th layer train

ALD	SA	OLI
89.06%	65.62%	70.31%

BERT 12th, 11th layer train

ALD	SA	OLI
90.62%	79.68%	78.12%

BERT 12th layer train

ALD	SA	OLI
81.25%	70.31%	70.31%

BERT no train

ALD	SA	OLI
87.50%	60.93%	81.25%

KoELECTRA – Freezing (Unified)

ELECTRA no freezing

ALD	SA	OLI
65.##%	35.##%	65.##%

ELECTRA 10th, 11th layer train

ALD	SA	OLI
88.86%	74.24%	80.49%

ELECTRA 11th layer train

ALD	SA	OLI
88.58%	73.58%	79.09%

ELECTRA no train

ALD	SA	OLI
81.##%	57.##%	71.##%

KoELECTRA – Freezing (Specific)

ELECTRA no freezing

ALD	SA	OLI
65.##%	35.##%	65.##%

ELECTRA 10th, 11th layer train

ALD	SA	OLI
65.##%	70.##%	78.##%

ELECTRA 11th layer train

ALD	SA	OLI
65.##%	71.##%	78.##%

ELECTRA no train

ALD	SA	OLI
65.##%	65.##%	74.##%

Freezing Strategies

결과

KoBERT with Unified Classifier

모델 전체를 재학습하는 3번 전략이 가장 성능 높았다.

→ 사전 학습 모델의 데이터셋과 다르고 큰 데이터셋

KoBERT with Specific Classifier for each Task

상위 layer 두 개와 classifier를 학습시키는 2번 전략이 가장 성능 높았다.

→ 사전 학습 모델의 데이터셋과 유사하고 큰 데이터셋

Freezing Strategies

결과

KoELECTRA with Unified Classifier

상위 layer 두 개와 classifier를 학습시키는 2번 전략이 가장 성능 높았음

→ 사전 학습 모델의 데이터셋과 유사하고 큰 데이터셋

KoELECTRA with Specific Classifier for each Task

상위 layer 한 개와 classifier를 학습시키는 2번 전략이 가장 성능 높았음

→ 사전 학습 모델의 데이터셋과 유사하고 큰 데이터셋

Freezing Strategies

KoBERT(Unified)와 KoELECTRA의 동결전략 차이

KoBERT, KoELECTRA 모두 한국어 위키 데이터로 사전학습을 수행한 모델이다.

KoELECTRA의 경우 '모두의 말뭉치'를 써서 추가 학습을 수행했다.

실제 사람들의 수필, 블로그 글 등이 담겨있다.

KODOLI 데이터가 KoBERT보다 KoELECTRA의 사전학습 데이터에 더 유사하다고 모델이 판단했을 가능성이 존재한다.

Freezing Strategies

실제 '모두의 말뭉치' 데이터

근데 그런 기대와 행복한 상상은 시작부터 깨져버렸다.

나에게는 아무도 없는거 같았다 아이들과 같이 있어도 혼자 있는것만 같은,함께 이야기...

시간이 지나면 나아지겠지했던것도 그냥 내 상상에 불과했다.

수많은 사람,친구들을 만났지만 달라지는건 없었다

그렇게 내가 변하고보니 이게 과연 나를 위한거였을까하는 생각이 가장 먼저 떠올랐다.

그땐 그냥 좋아하는 배우따라 보러갔었다, 가끔은 스토리나 넘버를 듣고 좋아서 보러간...

밥먹기조차 너무너무 귀찮다.

근데 이렇게 밥을 걸러도 왜 살은 안빠질까...

Freezing Strategies

KoBERT with Unified Classifier와 with Specific Classifier for each Task의 동결전략 차이

앞서 Label Combination이 모델로 하여금 클래스를 상호 고려할 수 있도록 학습했을 가능성에 대해 언급.

이와 같은 작업은 feature 간의 복잡한 상호작용을 모델링해야할 필요가 있다.

보다 깊은 모델의 tuning이 필요하다. → 모델 전체 재학습

반면 Specific Classifier for each Task의 경우 abuse, sentiment, offensiveness 에 대해 독립적으로 고려한다.
상대적으로 낮은 복잡성을 가진다.

상대적으로 사전 학습 구조를 유지하며 학습 용이하다. → 상위 layer + classifier 재학습

KoBERT– Task Combination

ALD + SA + OLI

ALD	SA	OLI
91.47%	75.91%	81.38%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.67	92.71	91.17	41.89	36.08	38.77	75.01	74.41	74.71	68.85	67.73	68.21

주요 세팅

KoBERT 12개 layer 전부 학습

Unified Classifier 사용 (label combination)

ALD+SA+OLI 예선 classifier activation을 GELU, init을 he 사용.

KoBERT– Task Combination

ALD + SA + OLI

ALD	SA	OLI
91.47%	75.91%	81.38%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.67	92.71	91.17	41.89	36.08	38.77	75.01	74.41	74.71	68.85	67.73	68.21

SA + OLI

ALD	SA	OLI
Non	76.91%	80.95%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
90.56	92.00	91.27	40.77	36.24	38.37	74.03	76.12	75.06	68.45	68.12	68.23

KoBERT– Task Combination

ALD + SA + OLI

ALD	SA	OLI
91.47%	75.91%	81.38%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.67	92.71	91.17	41.89	36.08	38.77	75.01	74.41	74.71	68.85	67.73	68.21

ALD + OLI

ALD	SA	OLI
91.33%	Non	80.98%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
90.20	92.21	91.19	40.82	37.51	39.09	75.80	74.57	75.18	68.94	68.09	68.48

KoBERT– Task Combination

ALD + SA + OLI

ALD	SA	OLI
91.47%	75.91%	81.38%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.67	92.71	91.17	41.89	36.08	38.77	75.01	74.41	74.71	68.85	67.73	68.21

OLI

ALD	SA	OLI
Non	Non	80.23%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
90.36	91.35	90.85	39.27	42.11	40.64	77.28	70.19	73.56	68.97	67.88	68.35

KoELECTRA – Task Combination

ALD + SA + OLI

ALD	SA	OLI
88.86%	74.24%	80.49%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.45	91.40	90.41	37.92	28.38	32.46	70.62	75.80	73.12	66.00	65.19	65.33

주요 세팅

KoELECTRA 12개 layer 중 마지막 11, 12만 학습

Unified Classifier 사용 (label combination)

ALD+SA+OLI 예선 classifier activation을 ReLU, init을 kaiming normal 사용.

나머지에서 동일한 세팅 했을 때 likely 레이블 예측을 안 해버리는 문제가 있어서 나머지에선 Tanh + xavier normal 사용
Tanh + xavier normal로 ALD+SA+OLI한 결과도 첨부하겠습니다.

KoELECTRA – Task Combination

ALD + SA + OLI

ALD	SA	OLI
88.86%	74.24%	80.49%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.45	91.40	90.41	37.92	28.38	32.46	70.62	75.80	73.12	66.00	65.19	65.33

SA + OLI

ALD	SA	OLI
Non	72.92%	79.55%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
87.15	92.23	89.62	31.94	15.79	21.13	68.84	75.93	72.22	62.64	61.31	61.00

KoELECTRA – Task Combination

ALD + SA + OLI

ALD	SA	OLI
88.86%	74.24%	80.49%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.45	91.40	90.41	37.92	28.38	32.46	70.62	75.80	73.12	66.00	65.19	65.33

ALD + OLI

ALD	SA	OLI
88.19%	Non	80.21%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
86.38	93.80	89.94	37.13	17.16	23.47	70.90	73.36	72.11	64.80	61.44	61.84

KoELECTRA – Task Combination

ALD + SA + OLI

ALD	SA	OLI
88.86%	74.24%	80.49%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
89.45	91.40	90.41	37.92	28.38	32.46	70.62	75.80	73.12	66.00	65.19	65.33

OLI

ALD	SA	OLI
Non	Non	77.19%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
82.51	93.55	87.68	23.91	5.03	8.32	65.04	66.80	65.90	57.15	55.13	53.97

KoELECTRA – Task Combination

ALD + SA + OLI

(tanh + xavier init)

ALD	SA	OLI
88.28	72.84	80.02

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
87.38	92.15	89.70	36.51	21.05	26.71	70.60	75.42	72.98	64.83	62.87	63.13

SA + OLI

ALD	SA	OLI
Non	72.92%	79.55%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
87.15	92.23	89.62	31.94	15.79	21.13	68.84	75.93	72.22	62.64	61.31	61.00

KoELECTRA – Task Combination

ALD + SA + OLI
(tanh + xavier init)

ALD	SA	OLI
88.28	72.84	80.02

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
87.38	92.15	89.70	36.51	21.05	26.71	70.60	75.42	72.98	64.83	62.87	63.13

ALD + OLI

ALD	SA	OLI
88.19%	Non	80.21%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
86.38	93.80	89.94	37.13	17.16	23.47	70.90	73.36	72.11	64.80	61.44	61.84

KoELECTRA – Task Combination

ALD + SA + OLI
(tanh + xavier init)

ALD	SA	OLI
88.28	72.84	80.02

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
87.38	92.15	89.70	36.51	21.05	26.71	70.60	75.42	72.98	64.83	62.87	63.13

OLI

ALD	SA	OLI
Non	Non	77.19%

OLI 결과 분석

Pr - not	R - not	F1 - not	Pr - likely	R - likely	F1 - likely	Pr - off	R - off	F1 - off	Pr - avg	R - avg	F1 - avg
82.51	93.55	87.68	23.91	5.03	8.32	65.04	66.80	65.90	57.15	55.13	53.97



Challenges



Challenges

- Resource에 한계가 있어 Hyperparameter Tuning 에 어려움이 있었다.
- PLM이 큰 모델이라 학습 중 gradient flow가 원활하게 이루어지지 않아서 freezing하지 않았을 때 성능이 덜 나온 부분이 존재했던 것으로 추측된다.
- 가설 검증을 위한 충분한 시간이 부족해 정확도가 높고 낮게 나오는 확실한 이유에 대해 알기는 어려웠다.

Challenges

- Resource에 한계가 있어 Hyperparameter Tuning 에 어려움이 있었다.
 - PLM이 큰 모델이라 학습 중 gradient flow가 원활하게 이루어지지 않아서 freezing하지 않았을 때 성능이 덜 나온 부분이 존재했던 것으로 추측된다.
 - 가설 검증을 위한 충분한 시간이 부족해 정확도가 높고 낮게 나오는 확실한 이유에 대해 알기는 어려웠다.
- Optuna 와 같은 라이브러리로 주어진 resource 내에서 해결하려는 노력.
- 충분한 시간을 가지고 재시도하는 방안.



Service & Social Impact



Service & Social Impact

- 고파스, 에브리타임 댓글 숨기기 기능
- 저연령대를 위한 댓글 필터링
- LLM 모델의 출력에 대한 Model Alignment 확보

Service & Social Impact

- 고파스, 에브리타임 댓글 숨기기 기능
 - 저연령대를 위한 댓글 필터링
 - LLM 모델의 출력에 대한 Model Alignment 확보
-
- 무분별하게 욕설 및 혐오 표현에 노출될 위험성 감소
 - 소통과 즐거움의 창구여야 할 인터넷 속에서 스트레스 받을 확률 감소
 - 언어 습득기에 있는 저연령층의 올바른 언어 사용능력 함양에 도움.
 - AI 윤리 준수

Reference

Paper

- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- Clark, - K., Luong, M. T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555.
- Alkomah, F., & Ma, X. (2022). A literature review of textual hate speech detection methods and datasets. Information, 13(6), 273.
- Park, S. H., Kim, K. M., Lee, O. J., Kang, Y., Lee, J., Lee, S. M., & Lee, S. (2023, May). "Why do I feel offended?" -Korean Dataset for Offensive Language Identification. In Findings of the Association for Computational Linguistics: EACL 2023 (pp. 1142-1153).
- Park, S. H., Kim, K. M., Cho, S., Park, J. H., Park, H., Kim, H., ... & Lee, S. (2021, November). KOAS: Korean text offensiveness analysis system. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (pp. 72-78).

Blog

- pedro marcelino. (n.d.). Transfer Learning from Pre-Trained Models.

GitHub

- KODOLI : <https://github.com/cardy20/KODOLI>
- KoELECTRA : <https://github.com/monologg/KoELECTRA>

자연어처리 2조 발표 마치겠습니다.

프로젝트 발표

고수현 • 이규빈 • 이규진 • 최인석