2

뭘 바라시는 겁니까

3

하사원
여기 놀러 왔나?

4

허씨 양반길

한국어 텍스트 다중 분류 모델 제작

협박

갈취

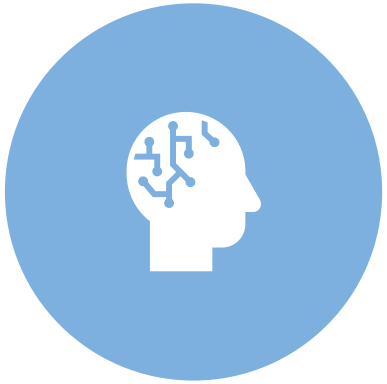
직장 내 괴롭힘

기타 괴롭힘

일반 대화

DKTC

AI



한국어 대화형
자연어 텍스트

텍스트 다중 분류

제한적 데이터

모델 완성

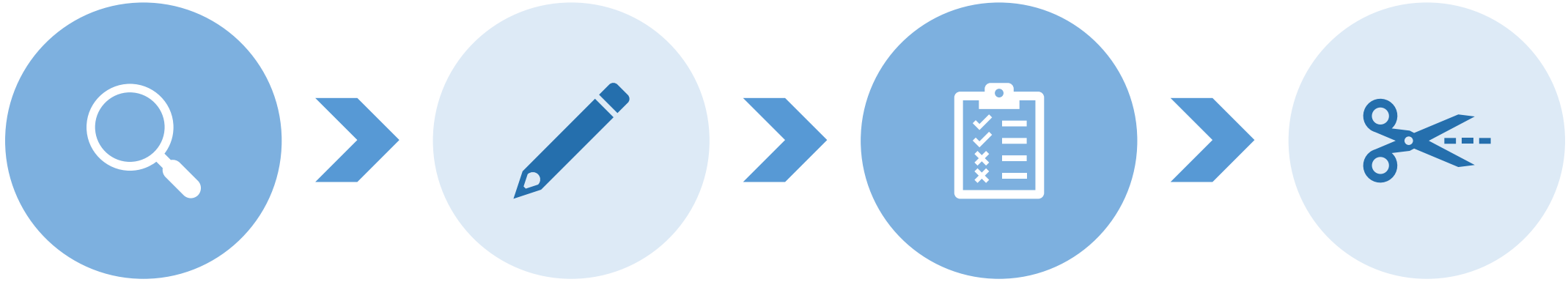
제한된 데이터 내에서 텍스트 분류 성능 높이기

01 양질의 데이터 생성 및 데이터 증강

02 적은 데이터로 비교적 높은 성능을 내는 모델링

02 워크플로우 – 역할 및 진행 과정

03 데이터 생성 – 프로세스



데이터 확인

기존 Train 데이터 샘플의 구성을 확인하여 생성할 데이터 형태를 정의

프롬프트 작성

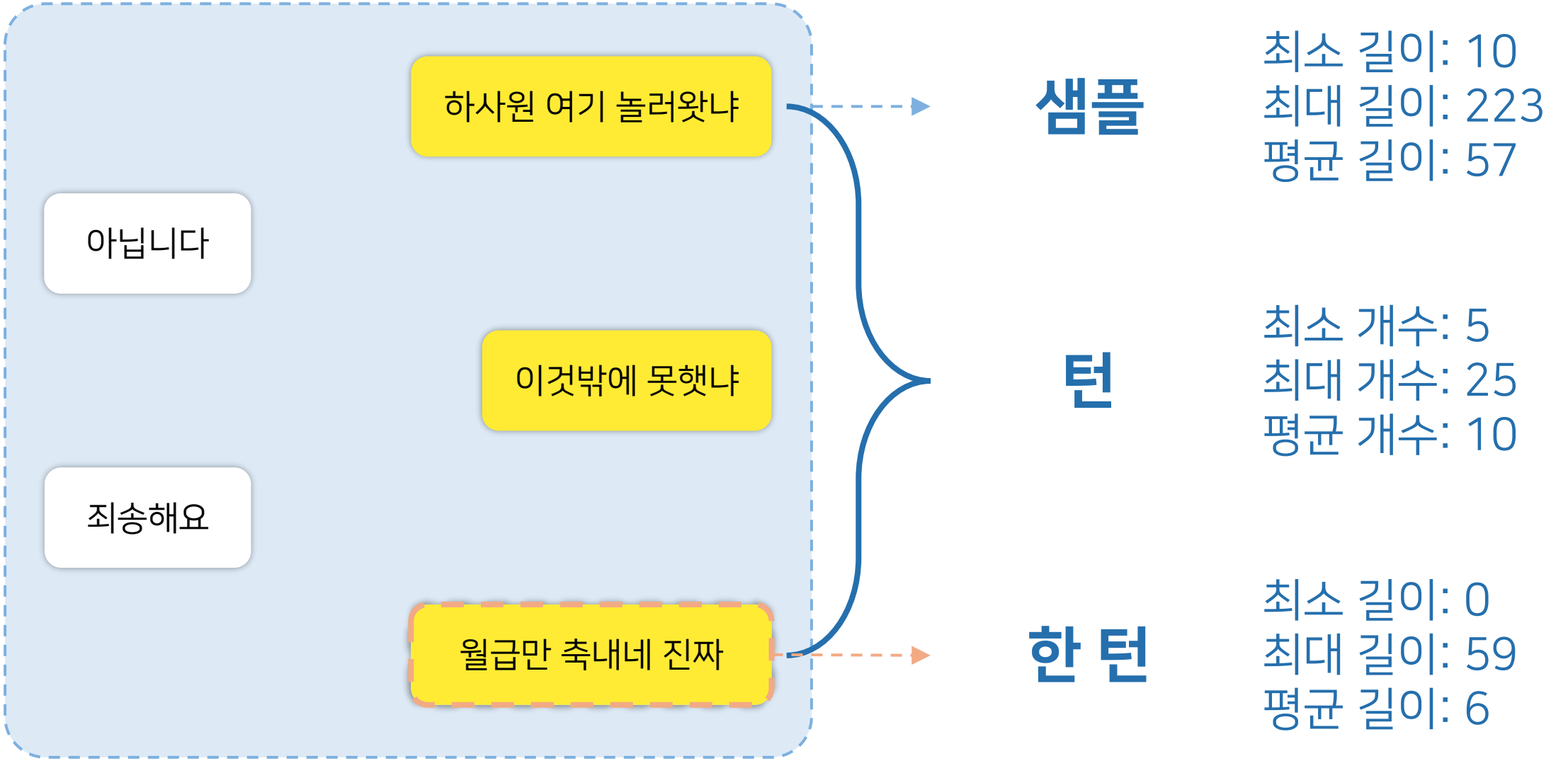
앞서 정의한 내용을 기반으로 대화 생성 규칙을 정하고 이를 프롬프트로 작성

결과물 검수

대화형 인공지능에 데이터 생성 요청 후 결과물의 품질에 따라 데이터셋 추가

데이터 분리

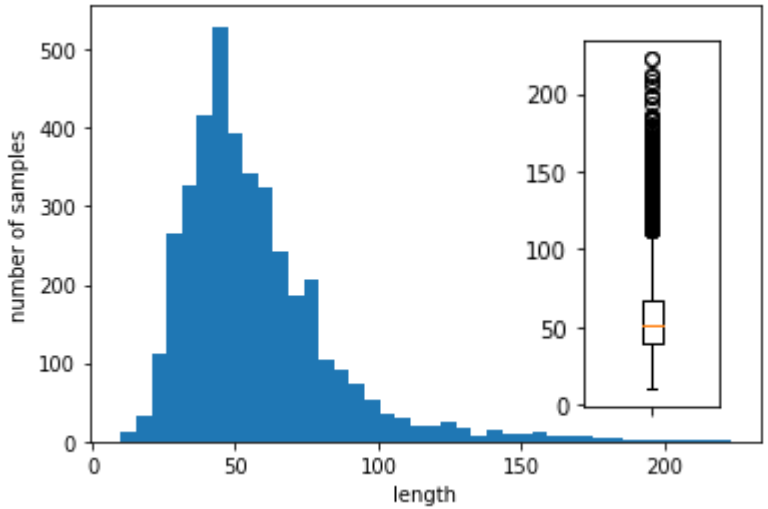
일반 대화 클래스가 포함된 데이터셋을 학습과 검증용으로 분리하여 준비



출처: DKTC train 데이터셋

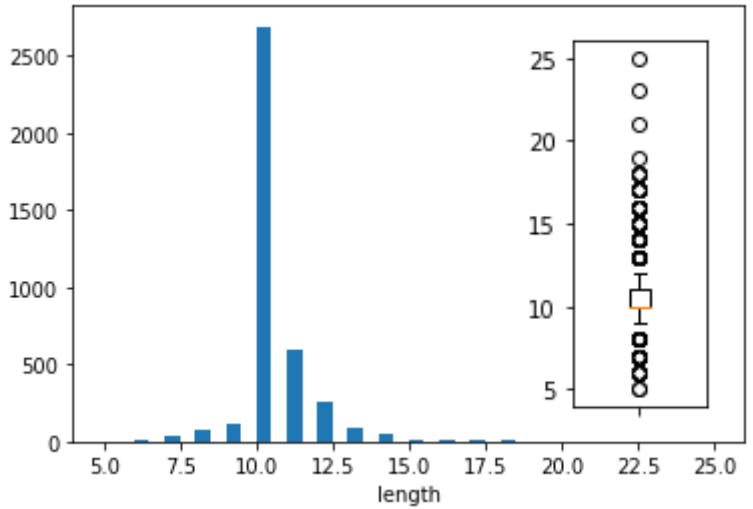
샘플

최소 길이: 10
최대 길이: 223
평균 길이: 57



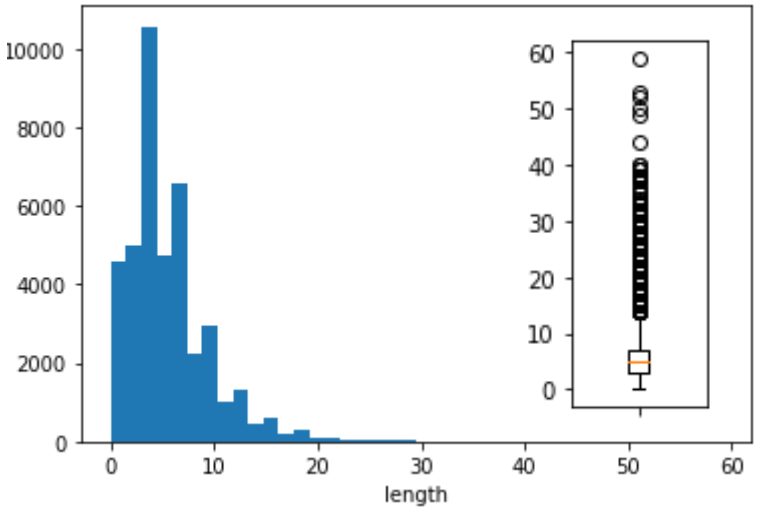
턴

최소 개수: 5
최대 개수: 25
평균 개수: 10

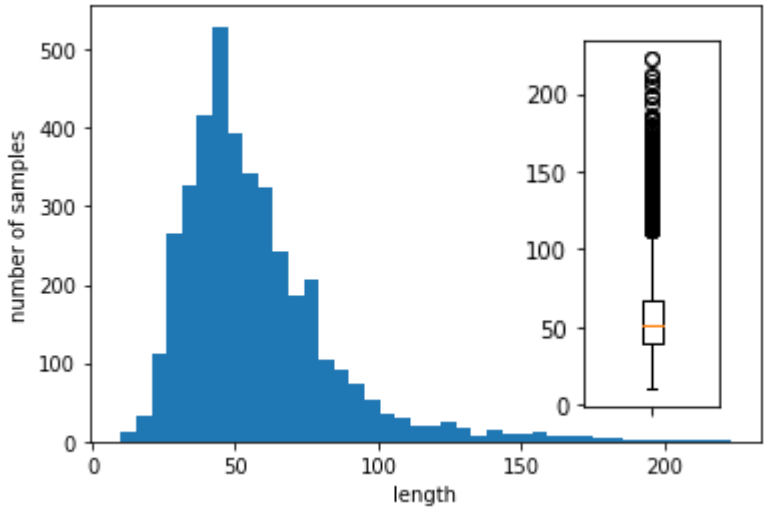


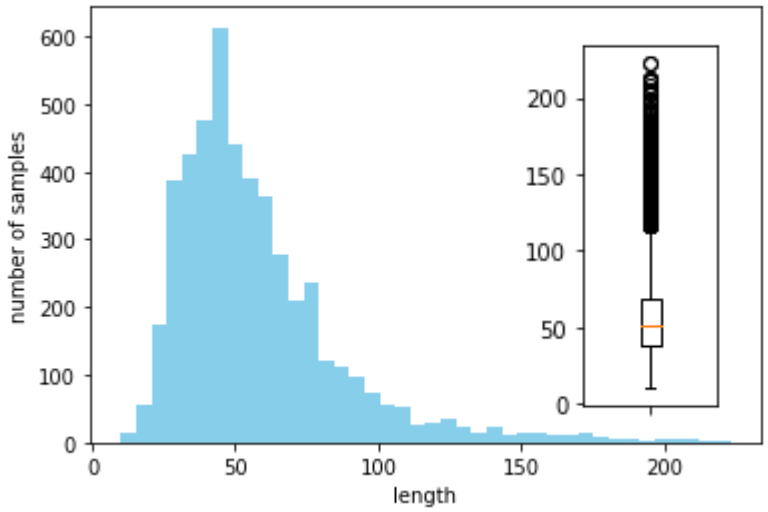
한 턴

최소 길이: 0
최대 길이: 59
평균 길이: 6

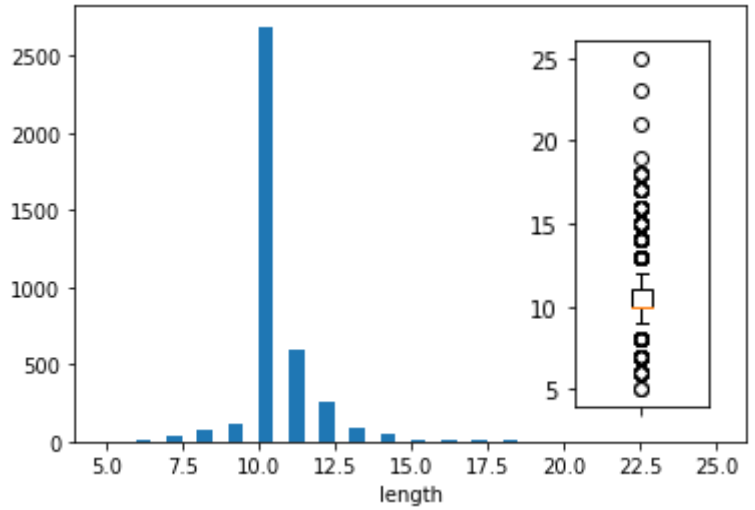


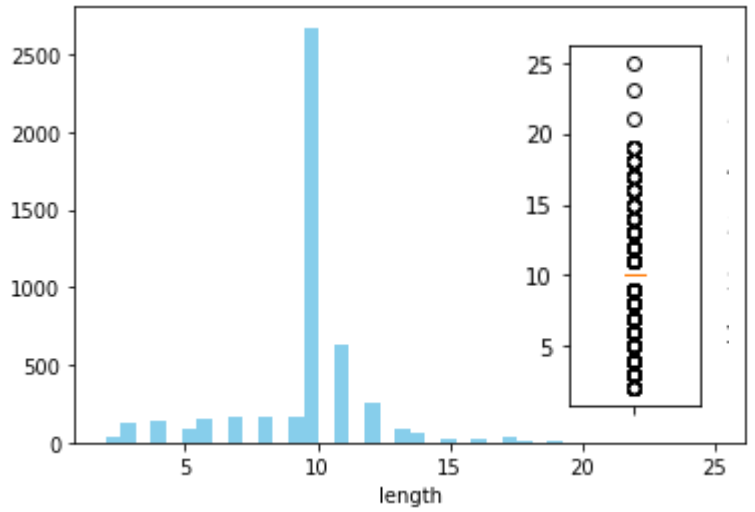
샘플


number of samples
length

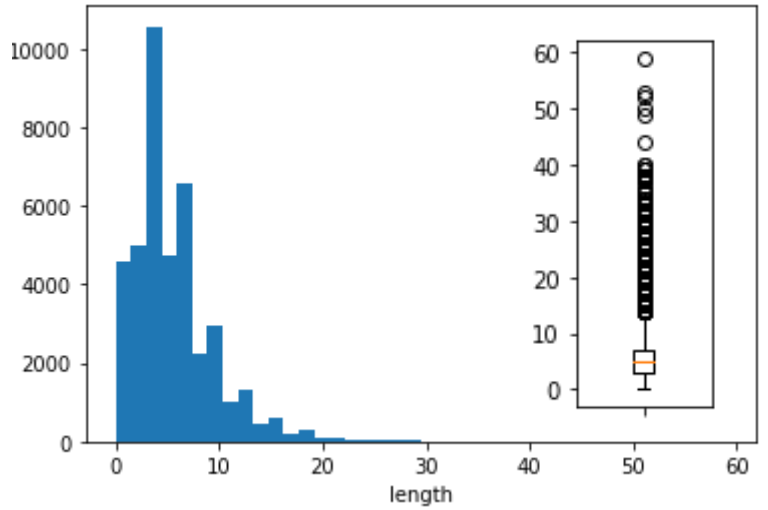

number of samples
length

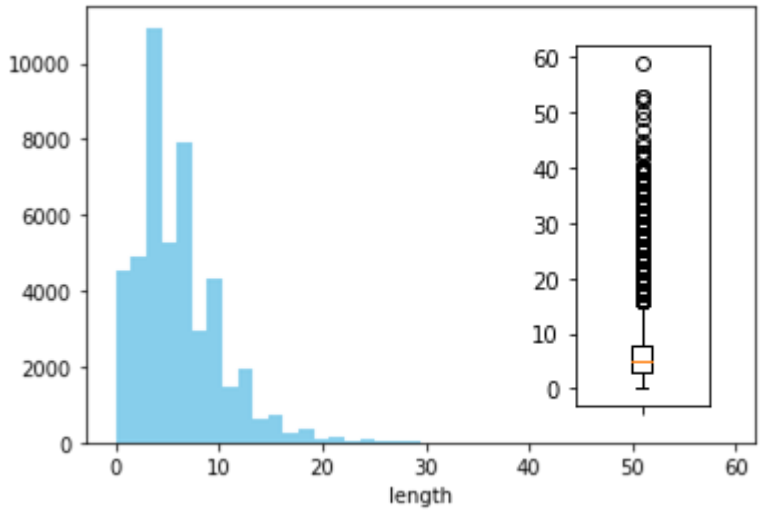
턴


length


length

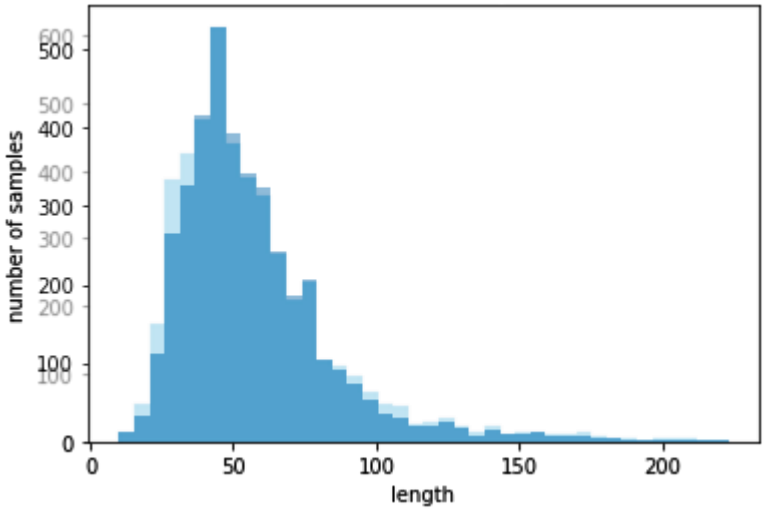
한 턴

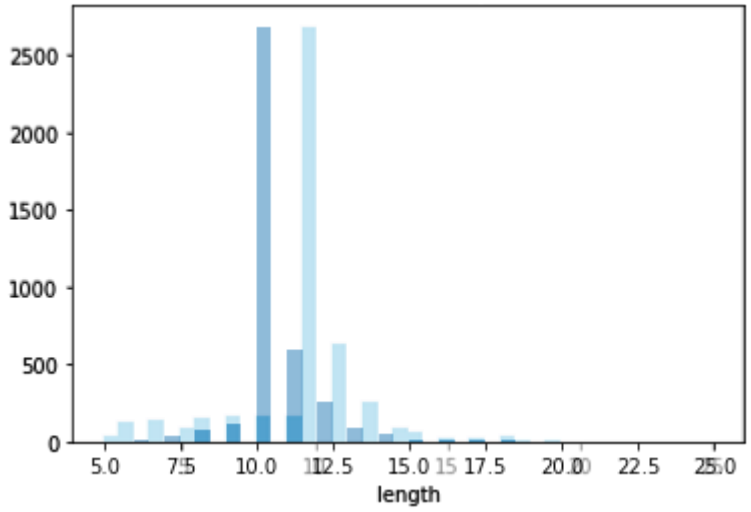

length

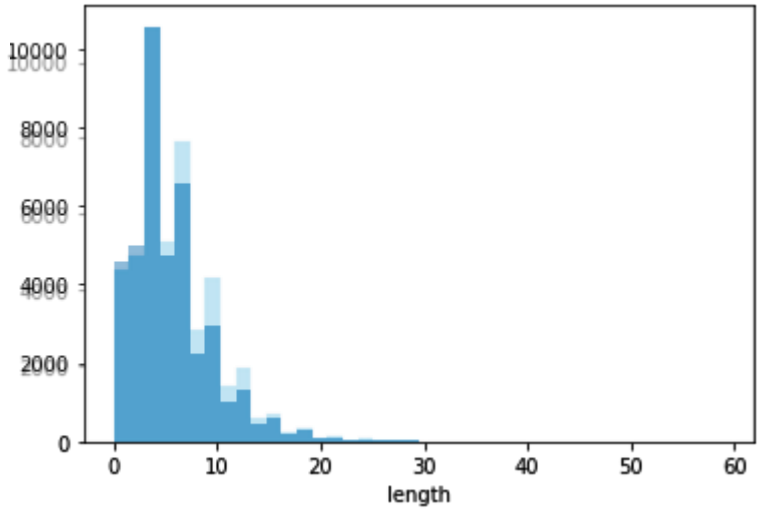

length

기존 생성 데이터 취합 후

샘플 턴 한 턴







두 사람 사이에 일상적으로 할 법한 대화 주제를 다양하게 추천하라.

두 분 사이의 관계와 상황에 따라 다르겠지만, 편안하게 이야기 나눌 수 있는 일상적인 대화 주제들을 다양하게 추천해 드릴게요.

일상생활 및 주변 이야기:

- **오늘 하루는 어떠셨어요?** (가장 기본적인 시작이죠!)
- **오늘 점심/저녁은 뭐 드셨어요?** (음식은 언제나 좋은 이야기 소재가 되죠.)
- **오늘 날씨가 정말 좋네요./춥네요./이상하네요.** (날씨는 가볍게 시작하기 좋아요.)
- **혹시 요즘 재미있게 보는 드라마/영화/책 있으세요?** (취미 공유는 친밀감을 높여줍니다.)
- **최근에 인상 깊었던 뉴스나 이야기거리 있으세요?** (사회 현상에 대한 생각을 나눌 수 있죠.)
- **주변에 새로 생긴 가게나 달라진 점이 있나요?** (함께 공유하는 공간에 대한 이야기.)
- **오늘 회사/학교/집에서 있었던 소소한 이야기.** (일상적인 경험을 공유하며 공감대를 형성.)
- **주말/휴일 계획 있으세요?** (함께 시간을 보낼 계획을 세우거나 각자의 계획을 공유.)
- **최근에 재미있거나 웃겼던 일 있으세요?** (긍정적인 감정을 공유하며 즐거운 분위기를 만들 수 있어요.)
- **혹시 요즘 관심 있는 분야나 배우고 싶은 것이 있으세요?** (서로의 관심사를 알아가는 좋은 기회.)

개인적인 이야기:

- **어렸을 때 재미있었던 추억 이야기.** (과거를 공유하며 서로를 더 깊이 이해할 수 있어요.)

규칙1: 각 **주제 당 10건**의 대화를 생성한다.

규칙2: 각 대화 **샘플**의 **평균 길이**는 **57**이며 **오차범위**는 **20**이내로 한다.

규칙3: 각 대화는 **평균**적으로 **10턴**의 대화로 구성하며 최대 20턴으로 한다.

규칙4: 대화 중 **한 턴의 길이는 60미만**으로 구성한다.

규칙5: 생성한 대화 데이터는 **csv 형식**이며, 대화의 **각 턴은 개행**하여 구분한다.

규칙6: 협박, 갈취, **괴롭힘 관련 내용이 포함되지 않도록 주의**한다.

규칙7: csv파일 형식이므로 대화 내용에는 절대로 ","가 들어가지 않도록 한다.

Gemini

이거 들어봐 와 이 노래 진짜 좋다 그치 요즘 이 것만 들어 진짜 너무 좋다 내가 요즘 듣는 것도 들어봐 음 난 좀 별론데 좋을 줄 알았는데 아쉽네 내 취향은 아닌 듯 배고프다 밥이나 먹으러 가자 그래

위와 같은 형식으로 직장 생활이나 일상 생활에서 나눌 법한 일반 대화 상황 데이터 셋을 10가지 정도 만들어줘. 길이는 50~200자 사이. 티키타카는 5~20 사이.

아이펠 캠퍼스

생성 오류 확인

강원도나 제주도 정말 좋아요. 풍경도 아름답고 맛있는 것도 많고요.
오, 저도 한번 가보고 싶네요. 정보 감사합니다.
네, 꼭 한번 가보세요. 분명 스트레스 해소에 도움이 될 거예요."

대답이 없었습니다.

규칙9를 잊었나?! csv파일 형식이므로 대화 내용에는 절대로 ","가 들어가지 않도록 한다!

정말 죄송합니다! 또다시 규칙을 놓쳤네요. CSV 파일 형식에 맞게 대화 내용에 쉼표(",")가 들어가지 않도록 수정하겠습니다.

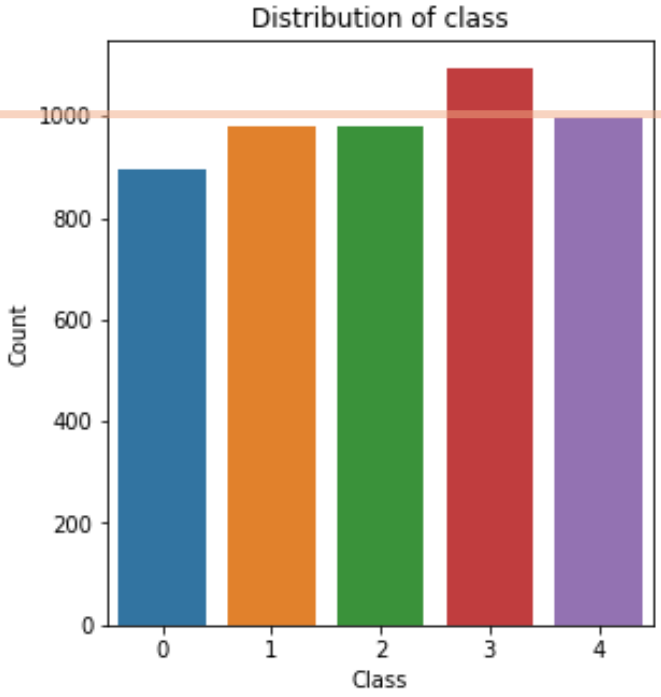
코드 스니펫

```
topic,conversation
좋아하는 운동이나 스포츠,"혹시 평소 즐겨 하는 운동 있으세요?
저는 축구 보는 걸 정말 좋아해요! 직접 하는 건 체력이 안 돼서…
아 저도 축구 좋아해요! 혹시 좋아하는 팀 있으세요?
네 저는 맨체스터 유나이티드 팬이에요
오 라이벌 팀 팬이시네요! 저는 리버풀 응원합니다
하하 ...때마다 긴장감이 넘치겠네요
```

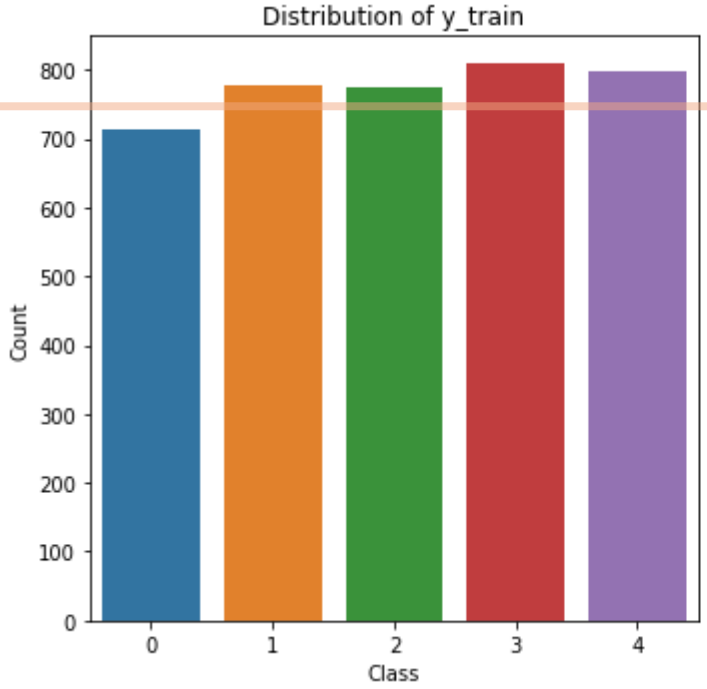
데이터 다양성 확인

	topic	conversation
0	오늘 하루 있었던 일	퇴근길이네요! 오늘 하루 수고 많으셨어요.\r\n네 수고하셨습니다. 특별한 일은 없...
1	오늘 하루 있었던 일	점심 뭐 드셨어요?\r\n비빔밥 먹었어요. 맛있더라고요.\r\n오 저도 비빔밥 좋아...
2	오늘 하루 있었던 일	아침에 지하철이 너무 붐벼서 힘들었어요.\r\n정말요? 저는 버스 탔는데 그것도 만...
3	오늘 하루 있었던 일	날씨가 정말 좋네요. 점심시간에 공원 산책 다녀오셨어요?\r\n네 잠깐 회사 앞 올...
4	오늘 하루 있었던 일	오늘 커피 맛이 평소보다 더 좋았던 것 같아요.\r\n정말요? 저는 오늘따라 커피가...
5	오늘 하루 있었던 일	시간이 정말 안 가는 것 같아요.\r\n저도요. 벌써 이렇게 늦었는데 아직도 할 일...
6	오늘 하루 있었던 일	회의에서 부장님 말씀이 너무 길어서 힘들었어요.\r\n맞아요. 저도 집중하기 힘들더...
7	오늘 하루 있었던 일	새로운 웹사이트 개편 프로젝트 팀이 꾸려졌는데 OO님도 같이하게 됐어요!\r\n정말...
8	오늘 하루 있었던 일	회사 식당에 새로운 메뉴가 나왔는데 드셔보셨어요?\r\n아니요 아직 못 먹어봤어요....
9	오늘 하루 있었던 일	고객사에서 긍정적인 피드백을 받아서 기분이 좋네요.\r\n정말요? 어떤 내용이었는데...
10	최근 관심사	재미있게 보는 드라마 있으세요?\r\n네 비밀의 숲이라는 드라마에 푹 빠져있어요.\...
11	최근 관심사	즐겨 듣는 음악 있으세요?\r\n네 뉴진스의 Hype Boy라는 신곡이 너무 좋아서...
12	최근 관심사	재미있는 책 읽으신 거 있으세요?\r\n네 돌이킬 수 없는 약속이라는 소설을 읽었는...
13	최근 관심사	새로 시작한 취미 있으세요?\r\n네 최근에 프랑스 자수를 시작했어요. 아직 초보지...
14	최근 관심사	관심 있는 뉴스나 이슈 있으세요?\r\n최근에 플라스틱 재활용 문제에 대한 다큐멘터

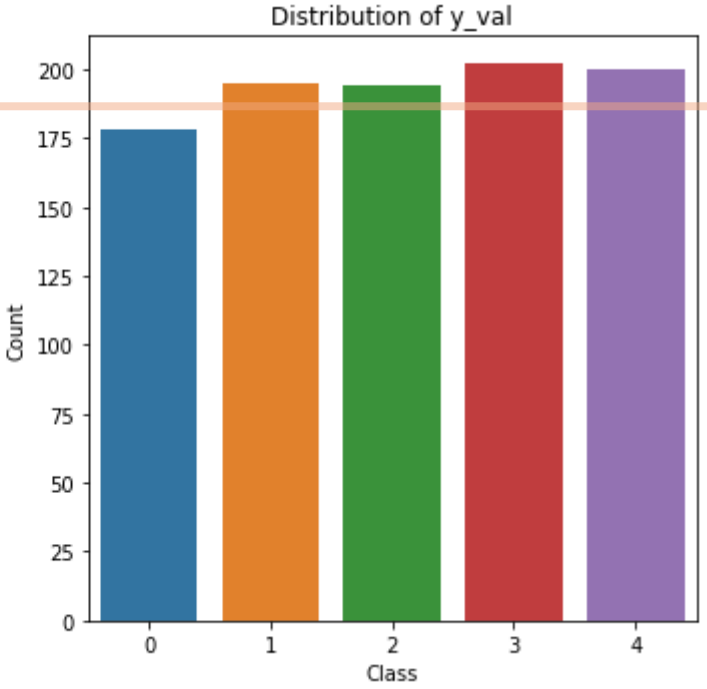
취합 데이터셋



학습용 데이터셋



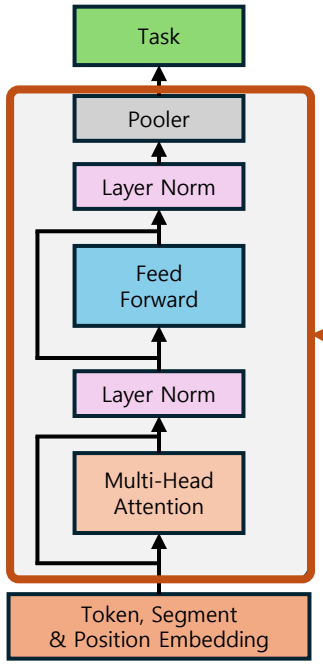
검증용 데이터셋



중복 제거 전

중복 제거 후

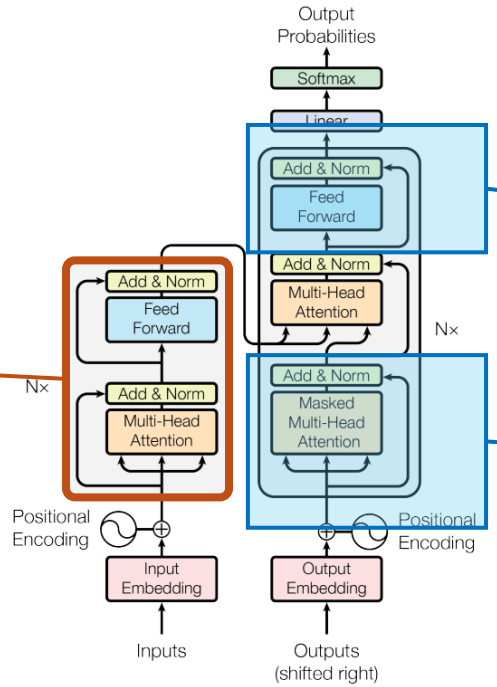
BERT



F1 Score 0.8452

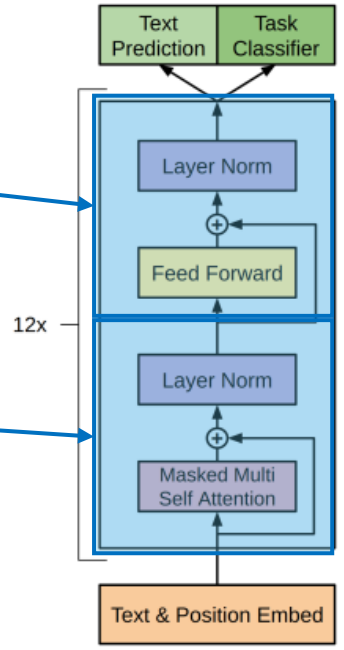
Public Score 0.6543

TRANSFORMERS



F1 Score 0.8206

GPT-1

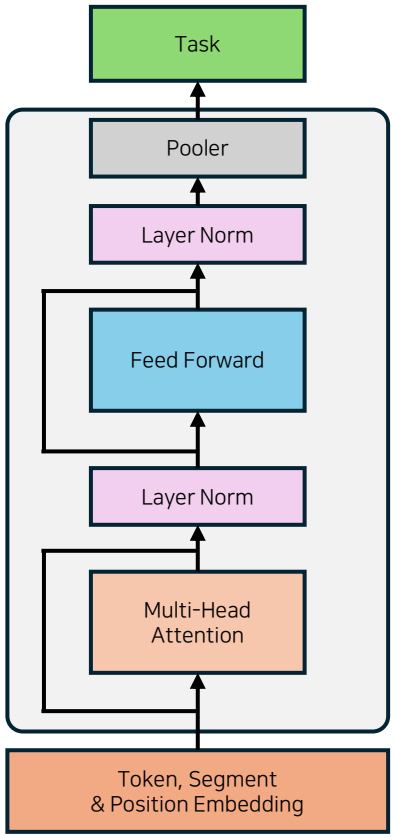


F1 Score 0.8075

Public Score 0.6433

*Validation Set 기준

BERT(Bidirectional Encoder Representations from Transformers)



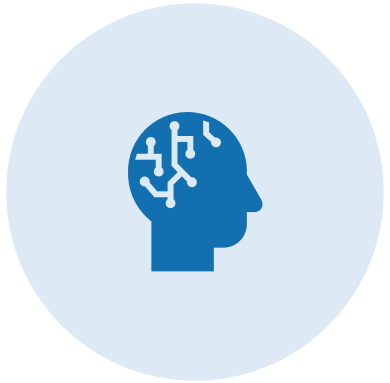
- 기존의 transformer는 한 방향으로만 문맥을 이해
- BERT는 양방향 문맥 이해를 통해 단어의 정확한 의미를 파악
- CLS토큰을 사용하여 정보를 압축, Classifier의 입력으로 사용
- 위와 같은 이유로 텍스트 분류에 효과적인 구조

[참고 자료]
구글 리서치 블로그 "Open Sourcing BERT: State-of-the-Art Pre-training for Natural Language Processing"

아이펠 캠퍼스

실험 설계

성능 향상을 위한
자료를 조사해보고
가설을 설정



모델 테스트

가설에 따라 모델
혹은 데이터 수정을
통해 테스트 진행

결과 분석

실험 결과를 분석
하여 기존 가설을
평가 및 이후 진행
할 실험 설계

가설 1 대용량의 한국어 데이터로 사전 학습된 모델을 활용하면 적은 데이터로도 한국어 분류 작업에 도움이 될 것이다.

가설 2 이에 더해 Dense Layer를 추가하면 학습에 용이하여 분류 성능이 높아질 것이다.

가설 3 개행 문자를 제거하면 분류 성능이 높아질 것이다.

KLUE/BERT-Base
(Korean Language Understanding Evaluation)

Table 1: Task Overview

Name	Type	Format	Eval. Metric	# Class	{|Train|, |Dev|, |Test|}	Source	Style
KLUE-TC (YNAT)	Topic Classification	Single Sentence Classification	Macro F1	7	45k, 9k, 9k	News (Headline)	Formal
KLUE-STS	Semantic Textual Similarity	Sentence Pair Regression	Pearson's r, F1	$[0, 5]$ 2	11k, 0.5k, 1k	News, Review, Query	Colloquial, Formal
KLUE-NLI	Natural Language Inference	Sentence Pair Classification	Accuracy	3	25k, 3k, 3k	News, Wikipedia, Review	Colloquial, Formal
KLUE-NER	Named Entity Recognition	Sequence Tagging	Entity-level Macro F1 Character-level Macro F1	6, 12	21k, 5k, 5k	News, Review	Colloquial, Formal
KLUE-RE	Relation Extraction	Single Sentence Classification (+2 Entity Spans)	Micro F1 (without *no_relation*), AUPRC	30	32k, 8k, 8k	Wikipedia, News	Formal
KLUE-DP	Dependency Parsing	Sequence Tagging (+ POS Tags)	Unlabeled Attachment Score, Labeled Attachment Score	# Words, 38	10k, 2k, 2.5k	News, Review	Colloquial, Formal
KLUE-MRC	Machine Reading Comprehension	Span Prediction	Exact Match, ROUGE-W (LCCS-based F1)	2	12k, 8k, 9k	Wikipedia, News	Formal
KLUE-DST (WoS)	Dialogue State Tracking	Slot-Value Prediction	Joint Goal Accuracy Slot Micro F1	(45)	8k, 1k, 1k	Task Oriented Dialogue	Colloquial

- 위키피디아, 모두의 말뭉치(국립국어원) 등에서 약 62GB의 한국어 데이터 학습
- Base 기준 약 1억 1천만개의 파라미터
- 다양한 TASK에 활용할 수 있도록 공개
  [좌측 사진 참고]

[참고 자료] 논문, "KLUE: Korean Language Understanding Evaluation"

KLUE-BERT Tokenizer

- mecab으로 형태소를 분할하고 wordpiece알고리즘 토크나이저를 사용함

Wordpiece

- 단어를 Sub Word로 분리하는 알고리즘이며 이를 통해 OOV 단어 최소화
- 예시) 단어: '아침밥' → Sub Word: ['아침', '##밥']
- 오탈자가 있더라도 옆에 붙은 Sub Word를 통해 오탈자로 인한 부정적 영향 최소화
- 빈도수가 높은 단어는 분리하지 않고 그대로 사용

[참고 자료]
논문, "KLUE: Korean Language Understanding Evaluation" p.51

결과	KLUE-BERT	KLUE-BERT + Dense	KLUE-BERT + Dense
Public Score	0.70838	0.72262	0.72911
Total parameters - Trainable - Non-Trainable	110,621,189 3,845 110,617,344	111,932,165 1,314,821 110,617,344	111,932,165 1,314,821 110,617,344
Pre-Processing	-	-	개행 문자 제거
Epochs	50	9	15
Remarks	-	Dense1 (1024) , Dense2 (512)	Dense1 (1024) , Dense2 (512)

- [x] 사전 학습 모델을 통해 성능 향상
- [x] Dense Layer 추가로 성능 향상
- [x] 개행 문자 제거로 성능 향상

☑ 사전 학습 모델을 통해 성능 향상

☑ Dense Layer 추가로 성능 향상

가설 1 동결층을 해제하면 학습 데이터셋을 보다 잘 이해하게 되어 분류 성능이 높아질 것이다.

*Validation Set 기준

F1 Score 0.8477 → F1 Score 0.9224

클래스 별 분포	0	1	2	3	4
	98	106	112	165	19

과적합

가설 1 Easy Data Augmentation을 통해 더 많은 데이터를 학습하면 분류 성능이 높아질 것이다.

가설 2 An Easier Data Augmentation을 통해 더 많은 데이터를 학습하면 분류 성능이 높아질 것이다.

아이펠 캠퍼스

EDA
(Easy Data Augmentation)

- 특정 단어를 유의어로 교체
- 임의의 단어를 삽입
- 임의로 두 단어 서로 위치 교체
- 임의의 단어를 삭제

아버지가 객실 아빠 안방 방에 정실 들어가신다.

아버지가 탈의실 방 휴게실 에 안방 탈의실 들어가신다.

AEDA
(An Easier Data Augmentation)

- 문장 내 임의의 지점에 임의의 구두점을 추가

! 어머니가 ! 집 ; 을 ? 나가신다

어머니 ? 가 . 집 , 을 , 나가신다

[참고 자료] github/toriving/KoEDA, 논문 "EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks"

결과	KLUE-BERT + Dense	KLUE-BERT + Dense + EDA	KLUE-BERT + Dense + AEDA
Public Score	0.72911	0.74373	0.73974
Total parameters - Trainable - Non-Trainable	111,932,165 1,314,821 110,617,344	111,932,165 1,314,821 110,617,344	111,932,165 1,314,821 110,617,344
Pre-Processing	개행 문자 제거	개행 문자 제거	개행 문자 제거
Epochs	15	24	28
Remarks	Dense1 (1024) , Dense2 (512)	Dense1 (1024) , Dense2 (512)	Dense1 (1024) , Dense2 (512)

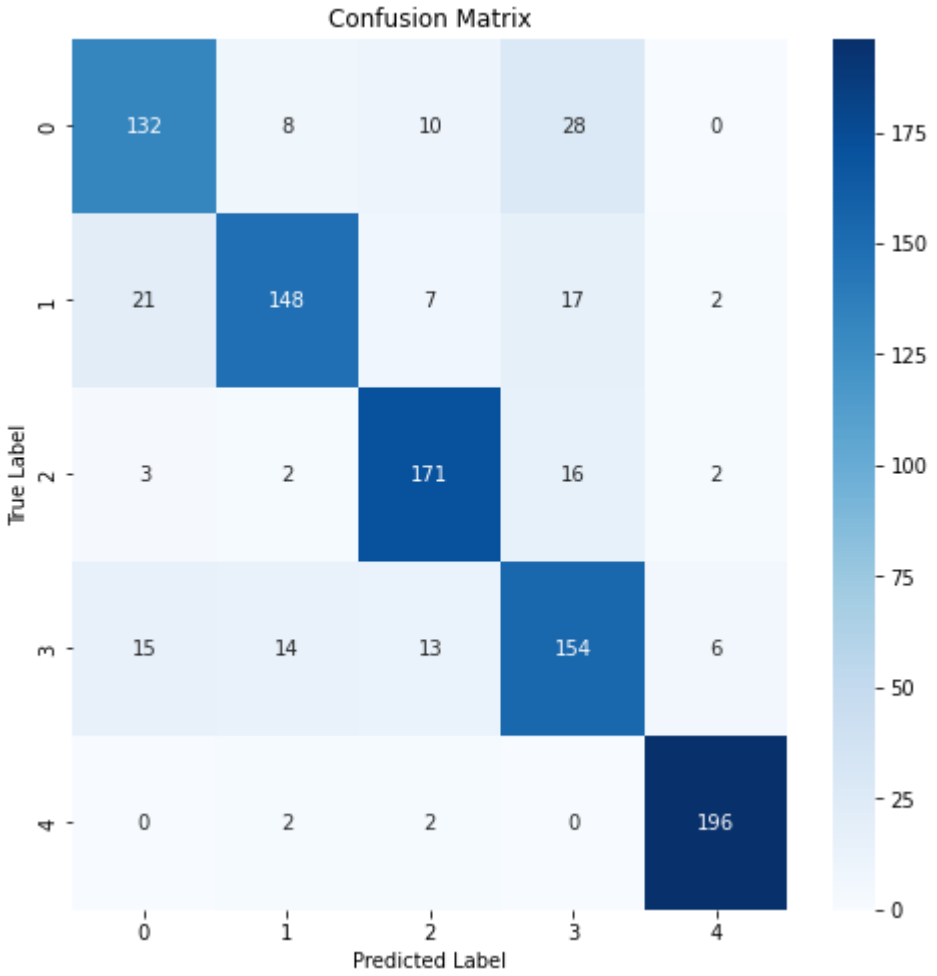
☑ EDA를 통해 성능 향상

☑ AEDA를 통해 성능 향상

실험 01의 결과:

- Validation Set의 F1 Score가 증가하지만 Test Set에 대한 F1 Score는 하락한다.
  - ✓ Model A: Validation F1 Score 0.82 / Test F1 Score 0.72
  - ✓ Model B: Validation F1 Score 0.92 / Test F1 Score 0.70

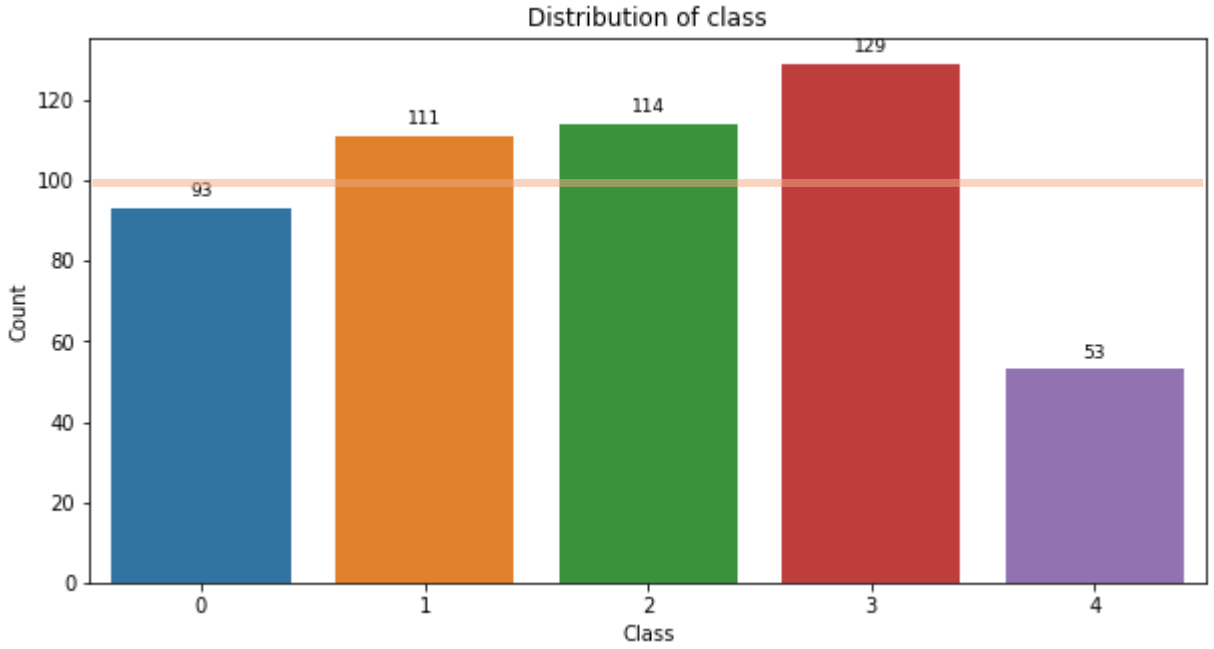
Model A Test F1 Score 0.72262



Validation Result

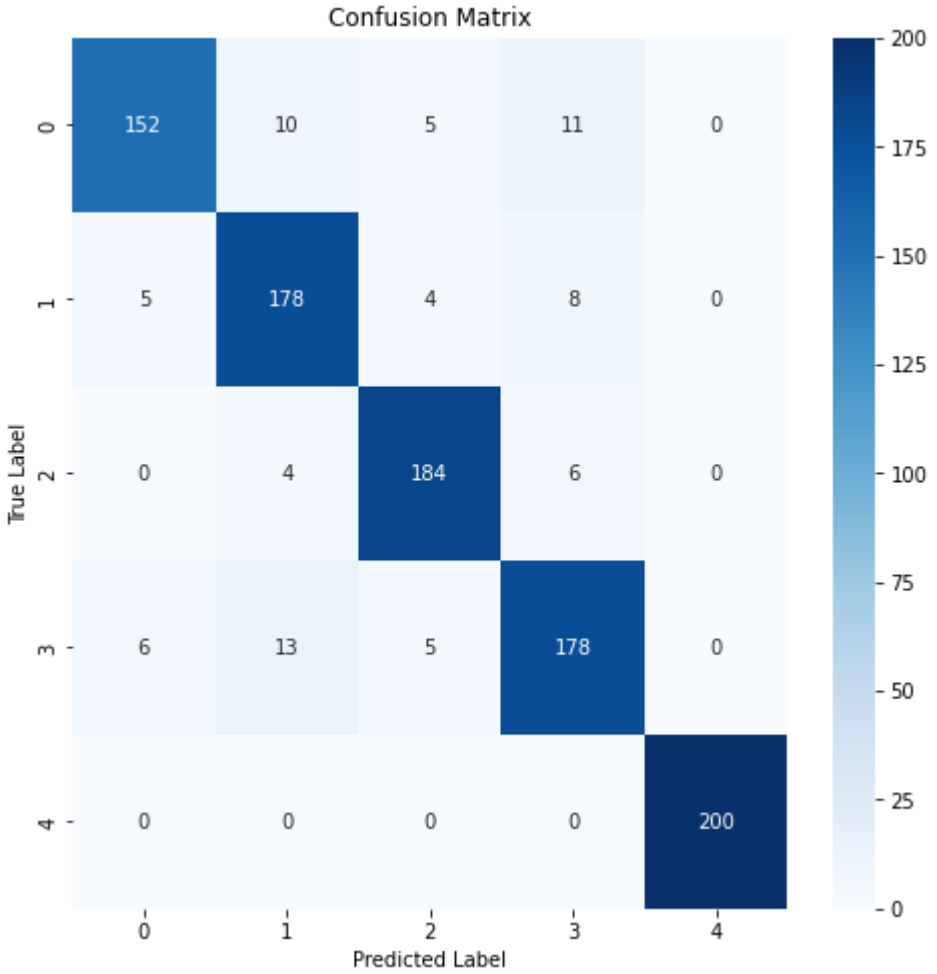
Classification Report: **Validation Result**

	precision	recall	f1-score	support
0	0.77	0.74	0.76	178
1	0.85	0.76	0.80	195
2	0.84	0.88	0.86	194
3	0.72	0.76	0.74	202
4	0.95	0.98	0.97	200
accuracy			0.83	969
macro avg	0.83	0.82	0.82	969
weighted avg	0.83	0.83	0.83	969



Test Result

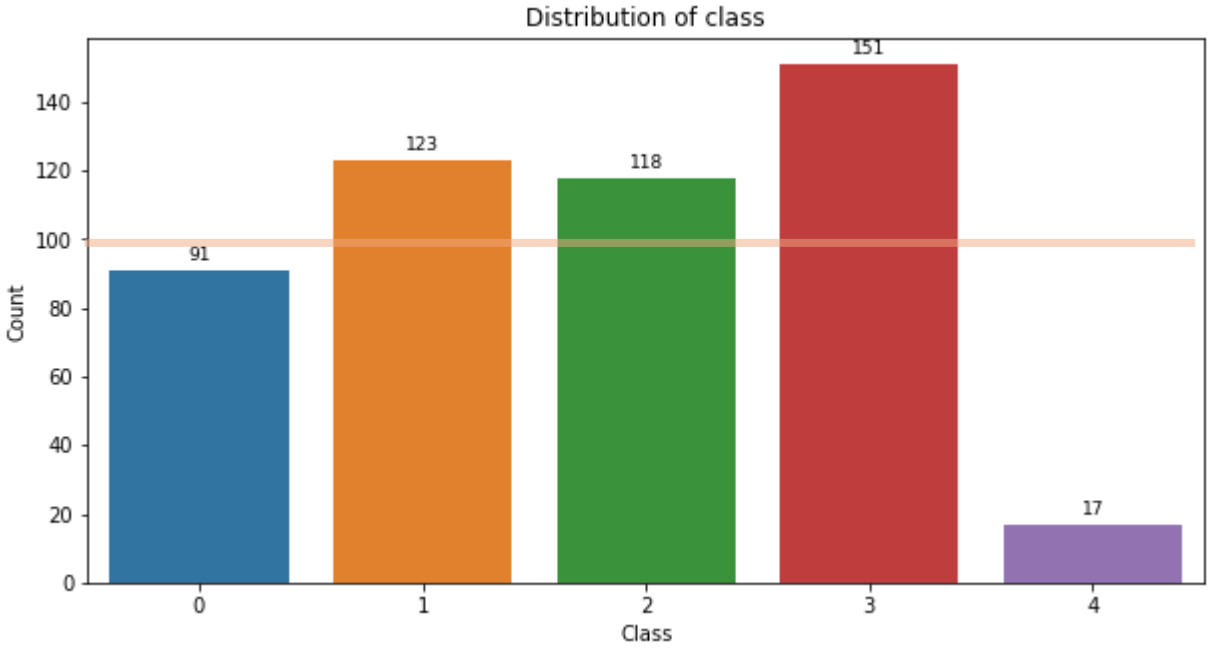
Model B Test F1 Score 0.70765



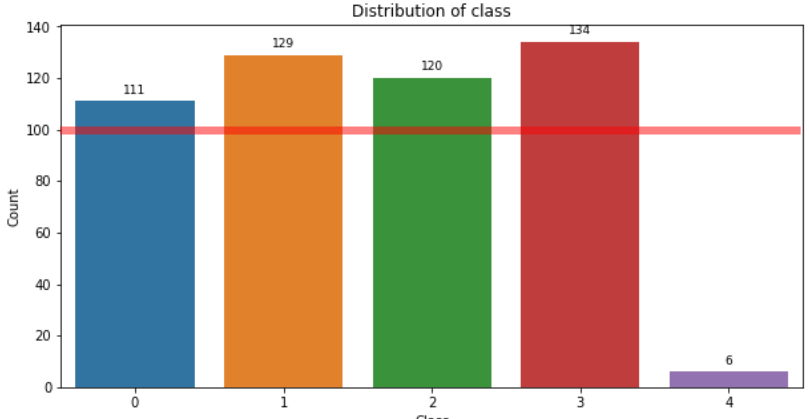
Validation Result

Classification Report: **Validation Result**

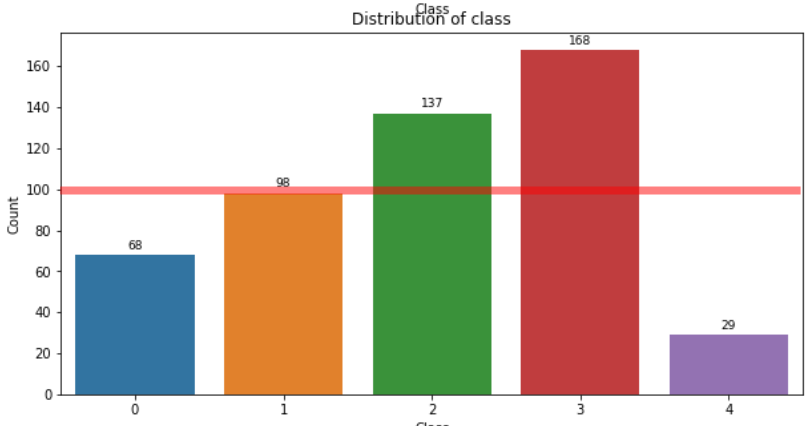
	precision	recall	f1-score	support
0	0.93	0.85	0.89	178
1	0.87	0.91	0.89	195
2	0.93	0.95	0.94	194
3	0.88	0.88	0.88	202
4	1.00	1.00	1.00	200
accuracy			0.92	969
macro avg	0.92	0.92	0.92	969
weighted avg	0.92	0.92	0.92	969



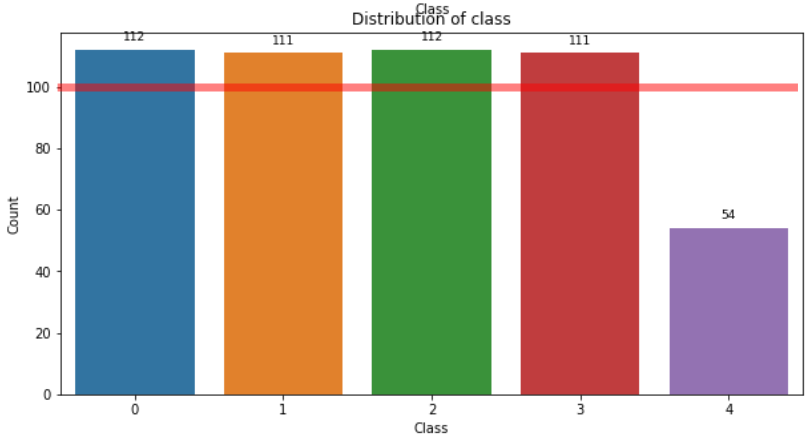
Test Result



Test F1 Score: 0.60260



Test F1 Score: 0.65434



Test F1 Score: 0.74373

실험 01의 결과:

- Validation Set의 F1 Score가 증가하지만 Test Set에 대한 F1 Score는 하락한다.
  - ✓ Model A: Validation F1 Score 0.82 / Test F1 Score 0.72
  - ✓ Model B: Validation F1 Score 0.92 / Test F1 Score 0.70

- **학습 데이터의 일반 대화가 타 클래스와 너무 명확히 구분이 되는 특징이 있다.**
- **하지만 그 특징이 Test의 일반 대화 특징과는 다르다.**

일반 대화임에도 불구하고 특정 클래스에서만 사용될 법한 단어가 등장하면 높은 확률로 오분류

```
[ ]  1 get_predition("김대리. 김부장.")
```

```
1/1 [==============================] - 0s 202ms/step
'직장 내 괴롭힘'
```

```
[ ]  1 get_predition("김대리. 김부장")
```

```
1/1 [==============================] - 0s 249ms/step
'직장 내 괴롭힘'
```

```
[ ]  1 get_predition("김대리. 김대리")
```

```
1/1 [==============================] - 0s 82ms/step
'직장 내 괴롭힘'
```

```
[ ]  1 get_predition("야근. 야근")
```

```
1/1 [==============================] - 0s 196ms/step
'일반 대화'
```

```
[ ]  1 get_predition("야근하고싶어? 싫어")
```

```
1/1 [==============================] - 0s 65ms/step
'직장 내 괴롭힘'
```

```
[ ]  1 get_predition("그러지마.죄송")
```

```
1/1 [==============================] - 0s 31ms/step
'기타 괴롭힘'
```

```
[ ]  1 get_predition("너 진짜 멋지다. 왜 그러세요 부끄럽습니다.")
```

```
1/1 [==============================] - 0s 244ms/step
'일반 대화'
```

```
[ ]  1 get_predition("너 진짜 멋지다. 그러지마세요 부끄럽습니다.")
```

```
1/1 [==============================] - 0s 114ms/step
'기타 괴롭힘'
```

🔴 1. 협박 대화

- **상황**: 신체적 위협, 가족 협박, 생명 또는 신체 훼손을 암시
- **주요 주제**:
  - 제3자(예: 동생, 친구)를 위협 대상으로 삼음
  - 실제 위해 가능성을 암시하거나 암묵적인 협박을 포함
  - **강한 위압감, 공포감 조성**

🗨 **대화 특징**: "죽고 싶냐", "동생을 평생 못 볼 수도 있다", "수틀리면 친구 팔 자른다" 등 **직·간접적 위협**이 포함됨.

🟠 2. 갈취 대화

- **상황**: 상대의 소지품, 금품 등을 협박하거나 강요로 뺏으려는 상황
- **주요 주제**:
  - 금전적 갈취: "그거 엄마 심부름이라 안돼" → "맞기 전에 내놔"
  - 물건 강탈: "그 옷 비싸보이는데 한번 입어보자" → "소문 낼 거야"
  - 상대의 **약점을 이용해 요구사항을 강제함**

🗨 **대화 특징**: 폭력 암시 + 사회적 망신 등 **심리적 압박을 이용한 착취**

🟡 3. 기타 괴롭힘 대화

- **상황**: 직접적인 폭력은 없지만, **사적 정보 침해**나 **언어적 모욕**, **불쾌감 유발** 등의 비정상적 접근
- **주요 주제**:
  - 사적 정보 접근 및 연락 시도 (스토킹에 가까움)
  - 대인 관계에서의 **명예훼손**, **허위 사실 유포**
  - 모욕, 비아냥 섞인 언어 사용

🗨️ **대화 특징**: "카페에서 번호 봤다", "왜 그렇게 예민하세요?", "화해하자 → 옥상에서" 등 **교묘하거나 우회적인 위협 또는 불쾌감 유발**

🔵 4. 직장 내 괴롭힘 대화

- **상황**: 상하 관계 또는 조직 내 위치를 이용한 권력형 괴롭힘
- **주요 주제**:
  - 무리한 업무 지시: "퇴근 10분 전인데도 업무 강요"
  - 인신공격, 가족 모욕: "부모 없니?", "애미애비 없냐?"
  - 감정적 폭언 및 경고: "바로 해고야", "사회생활 처음이야?"

🗨️ **대화 특징**: 업무 외적인 비난을 포함하며, **상대의 인격을 부정**하거나 **퇴사를 압박**

분석 결과

- 각 클래스마다 높은 빈도로 나타나는 특정 단어나 상황이 있다.
- 모델은 해당 단어와 상황 등에 각 클래스 분류 가중치를 높게 평가할 가능성이 크다.
- 이러한 특징이 전혀 나타나지 않은 것만 일반 대화로 분류할 가능성이 높다.

예상 가능한 해결 방안

- 분류기에서 일반 대화 클래스에 가중치를 높게 혹은 임계값을 낮게 설정하여 일정 확률 이상 일반 대화일 가능성이 있다면 무조건 일반 대화로 분류
- 각 클래스를 대표하는 단어와 상황이 들어가지만 위협적이지 않은 일반 대화 데이터를 많이 학습에 포함하여 해당 단어에 대한 편향성을 낮춘다.
- Regularization 기법을 활용하여 특정 단어에 대한 편향성을 낮춘다.

애매한 격차

```
1 get_predition("너 진짜 멋지다. 왜 그러세요 부끄럽습니다.")
```

```
1/1 [==============================] - 0s 244ms/step
'일반 대화'
```

```
[ ] 1 get_prob("너 진짜 멋지다. 왜 그러세요 부끄럽습니다.")
```

```
1/1 [==============================] - 0s 215ms/step
array([[1.4628738e-04, 1.4881685e-04, 3.1008403e-04, 3.9624624e-04,
        9.9899858e-01]], dtype=float32)
```

```
[ ] 1 get_predition("너 진짜 멋지다. 그러지마세요 부끄럽습니다.")
```

```
1/1 [==============================] - 0s 114ms/step
'기타 괴롭힘'
```

```
[ ] 1 get_prob("너 진짜 멋지다. 그러지마세요 부끄럽습니다.")
```

```
1/1 [==============================] - 0s 188ms/step
array([[6.5818347e-02, 5.0498243e-02, 1.2571908e-02, 8.7038392e-01,
        7.2752318e-04]], dtype=float32)
```

극명한 격차

```
1 get_predition("야근하고싶어? 안 할건데? 그래 그럼 잘가")
```

```
1/1 [==============================] - 0s 182ms/step
'직장 내 괴롭힘'
```

```
[ ] 1 get_prob("야근하고싶어? 안 할건데? 그래 그럼 잘가")
```

```
1/1 [==============================] - 0s 202ms/step
array([[0.08457034, 0.01040056, 0.7733973 , 0.12669821, 0.00493358]],
      dtype=float32)
```

```
1 get_predition("하지마라. 안 그럴게요. 죄송합니다.")
```

```
1/1 [==============================] - 0s 216ms/step
'기타 괴롭힘'
```

```
[ ] 1 get_prob("하지마라. 안 그럴게요. 죄송합니다.")
```

```
1/1 [==============================] - 0s 217ms/step
array([[0.18159893, 0.00976149, 0.30147877, 0.5061406 , 0.00102021]],
      dtype=float32)
```

- 각 클래스마다 높은 빈도로 나타나는 특정 단어나 상황이 있다.
- 모델은 해당 단어와 상황 등에 각 클래스 분류 가중치를 높게 평가할 가능성이 크다.
- 이러한 특징이 전혀 나타나지 않은 것만 일반 대화로 분류할 가능성이 높다.
- 실험 02를 통해 데이터가 많으면 분류 성능이 높아지는 효과를 확인했다.

그러므로 주어진 상황에서 분류 성능을 높이는 방법은 다음과 같다.

➢ 각 클래스를 대표하는 단어나 상황이 들어가지만 위협적이지 않은 대화 위주의 일반 대화 데이터로 학습에 포함시켜 특정 단어들에 대한 편향성을 낮춘다.

그 외의 보조 수단으로 아래의 방법이 약간의 도움을 줄 수는 있을 것이다.

- Regularization 기법을 활용하여 특정 단어에 대한 편향성을 낮춘다.

“모델링 개선점을 찾는 것보다
더 많은 데이터를 수집하는 것이
가장 효과적”

- 케창딥 6장 1절 中 -