

이청수와 아이들 중간 발표

신개념 토론 싱크탱크 방구석 대한민국



✓ Schedule

- 매주 일요일 3시 정기 미팅 진행
 - 이번 주 진행 사항, 다음 주 진행 예정 사항 공유
 - 합의가 필요한 부분과 애로사항 공유, 해결방안 모색
 - 맡지 않은 domain의 이해를 위해 각각의 role마다 기본적인 내용 설명

○ 진행 사항

- 앱: 채팅방 UI 및 프로토타입 톤론 flow 완성, 실험을 통한 톤론 flow 세부 조정 예정
- 서버: multi-label 비방성 표현 마스킹 모델 서빙 준비중
- 인공지능 모델(대화요약): 1차 모델 완성 및 휴리스틱 평가 단계 진행중
- 인공지능 모델(비방성 표현 마스킹): Multi-Label Classification 완성 및 evaluation 진행중

✓ Role of Each Member

김석: 백엔드 서버 개발

박진우: 대화요약 모델 개발

이청수: 프론트엔드 개발

장채윤: 비방성표현 마스킹 모델 개발

Our Notion

< Process 공유 및 회의록 작성을 위해 Notion을 이용하였습니다.

A Team ⌂ 100% 🔋

캡스톤설계프로젝트_SWE... Task List Quick Find All Updates Settings & Members

WORKSPACE

- 방구석 대한민국의 필요성
- debate frame Q/A
- Task List**
- Meeting Notes
- Docs

PRIVATE

- + Add a page
- Templates
- Import
- Trash

Board View Table + Add view Filter Sort ... New

✓ Task List

To Do 4

- Text Summarization Repo
 - 1
- 와이어프레임 Wireframe이란 화면 설계서입니다. Figma를 이용해 앱 UI를 디자인하고, 개발할때 참고하려고 합니다.

Doing 4

- 비방표현 데이터셋
 - 1. kobart로 가능한 다양한 시뮬레이션을 하는 종 >> 모델 성능이 아쉬울 경우 Kobertsum이나 다른 요약 모델 사용해볼 예정
 - 2. inference 시간은 그렇게 길지 않아 보임 gpu 사용시 약 50초 (beam search: 50)
 - 3. beam search는 50 정도가 가장 성능이 좋음(경험적 성능)
- 대화요약 모델
 - 토큰 사전 공지 내용
 - 맞춤법에 유의하여 토큰할 것
 - 글자수 제한을 지킬 것
 - 욕설 및 비방표현은 토큰에 불리하게 작용될 수 있으니 유의할 것
 - 기호나 특수문자 사용을 주의할 것(?)

Done 4

- 깃허브 만들기
 - 1. 가용 가능한 데이터셋에 대한 정보와 그에 대한 논의
 - 방구석 대한민국 오픈카톡방 데이터셋(박진우가 전담해서 출해오기)
 - <https://d2.naver.com/helloworld/7753> 273 읽어보기
 - 데이터 저처리.
- 데이터셋 찾아보기

No Status 2

토론 플랫폼 개발

- 1. 공정하고 원활한 토론을 위한 방장봇
 - 비방성 표현 마스킹
 - 대화 주제 요약
- 2. 상용화와 연대감, 게임틱한 요소를 위한 국가 개념

주제 구체화

- fine-tuning 데이터셋
- [korean-hate-speech](#) 9,381개 labeled dataset

train_loss
— multi-label naive — multi-label case

방구석 대한민국의 목적

- ✓ 토론 형식 설명
- ✓ 방구석 대한민국 설명
- ✓ 기대 효과



고전식 토론 이란?

A Team ⌂ ⌃ ⌄ 100% 🔋

특정한 논제에 찬성과 반대로 대립하는 두 편이 각자 주장하고자 하는 바로 상대방과 청중을 설득하고자 하는 말하기이다.



✓ 방구석 대한민국의 목표

토론의 장점을 살리고 AI 기술을 활용하여 단점을 보완 함으로서 합리적인 의사결정 과정을 도출하는 모범적인 토론 모델을 만드는 것을 목표로 한다.

✓ 방구석 대한민국만의 장점

1. 인공지능 사회자 도입

- 토론 전반적인 진행을 주도하고 발언권을 제한하며 전체 대화에 대한 요약을 해줌으로써 기존의 사회자의 역할을 수행함.

2. 익명성 플랫폼

- 익명성을 도입하는 것은 쉽지만 그로 인해 파생되는 단점인 비방성 표현을 남발 하는 행위를 비방성 표현 마스킹 모델로 통제함.

3. 토론 방식을 간소화

- 기존 복잡한 토론 방식들을 간소화하고 토론 진행방식 및 구조를 직관적으로 만들어 초심자도 재미있게 즐길 수 있게 만듦

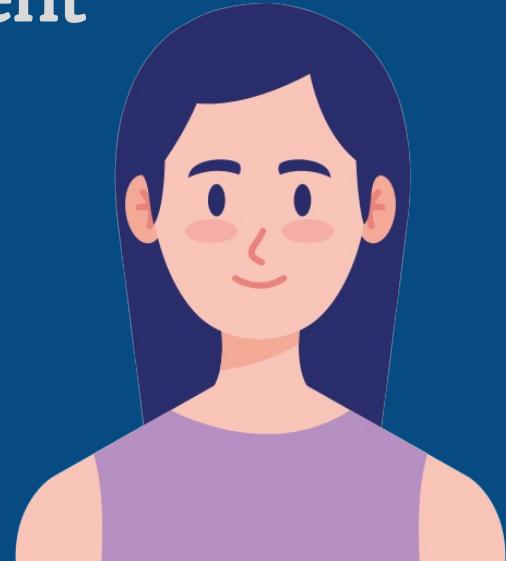
✓ 방구석 대한민국의 기대 효과

프로젝트 '방구석 대한민국'을 통해 AI 기술과 기존 토론 방식의 장점을 융합하여 젠더 갈등, 세대 갈등, 지역 갈등 등 사회에 만연해 있는 극단성을 극복하고, 생산적이고 유의미한 토론을 통해 현안들을 혁명하게 해결해 나갈 수 있을 것이라 생각한다.



기술 설명

- ✓ AI Model Development
- ✓ Front-End Development
- ✓ Back-End Development



✓ Text Summarization

- Dataset: [Dacon] 한국어 문서 생성요약 AI 경진대회 (40k)
[AI HUB] 문서요약 텍스트 (350k)
[AI HUB] 일상대화 요약 텍스트 (120k)
- Model description: [model name] Kobart (SKT에서 만든 한국어 BART 모델)
[Pretrained model] 'digit82/kobart-summarization' 사용
- Evaluation metrics: test dataset으로 디베이팅 데이에서 크롤링한 토론 샘플 2500개 사용
heuristic method을 활용하여 랜덤하게 샘플 선정하여 모델별로 점수화
(뒤에서 자세한 설명)
- Hyperparameters : beam, epoch, combination of datasets

✓ Text Summarization

Challenges:

- 데이터셋의 종류와 양이 많음
 - >> 데이터셋 조합을 간소화하고 고성능 gpu확보함
- 좋은 성능의 metric이 존재하지 않음 (Rogue score는 metric 성능이 좋지 않음)
 - >> 사람이 1~10점으로 점수화 하여 비교

Limitation: 모든 토론 샘플을 평가하는데 드는 휴먼 코스트가 매우 높음

>> 랜덤으로 100개 정도를 랜덤하게 추출하여 평가 할 예정

✓ Text Summarization

- metric example

model_version	3	4	5	6	7
노키즈존	5	4	4	7	7
인공지능	3	4	5	9	8
9시등교	7	7	5	8	5
컴퓨터계산	9	7	7	10	8
사드배치	8	9	6	7	7
sum	32	31	27	41	35

사람의 평가

✓ Text Summarization Example

origin text: 저는 찬성입니다. 저는 개인적으로 수학을 좋아하고 즐기는데 수학은 계산보단 어떤 문제를 풀 방법을 생각하는 것에 더 가깝다고 생각합니다. 실제로 유명한 수학자들도 계산을 못합니다. 저는 계산은 기계에게 맡기고 창의적인 수학적 방법풀이에 더 집중했으면 좋겠습니다. 오일러라는 계산을 잘하는 굉장히 유명한 수학자가 있습니다. 저도 굉장히 존경하는 분인데, 그 분은 살아생전 수많은 논문을 내셨습니다. 그러나 오일러가 살던 시대에는 계산기가 없었기 때문에 오일러는 항상 계산이라는 단순 육체 노동을 해야 했습니다. 그는 암산에도 능했지만 그 천재적인 능력을 단순 육체 노동인 계산을 하시느라 시간을 빼앗겼습니다. 그 분은 소수들의 규칙을 찾는데 열중하셨는데 그 시대에는 계산기가 없어서 소수를 직접 찾아야 했습니다. 저는 오일러가 이 시대에 살았더라면 아름다운 소수 공식을 만들 수 있었지 않았을까 생각합니다. 이 이야기를 하면서 제가 하고 싶은 얘긴 이와 같이 계산은 시간을 잡아먹는 단순 작업입니다. 여러분은 컴퓨터와 계산 배틀을 해서 이길 자신이 있으십니까? 계산은 컴퓨터한테 시키고 단축된 시간에 더 연구해서 모르던 사실을 알아내면 좋겠습니다.

'summary6'

'저는 개인적으로 수학을 좋아하고 즐기는데 수학은 계산보단 어떤 문제를 풀 방법을 생각하는 것에 더 '
'가깝다고 생각하며 굉장히 유명한 수학자들도 계산을 못하기 때문에 계산은 기계에게 맡기고 창의적인 '
'수학적 방법풀이에 더 집중했으면 좋겠습니다.'

✓ Offensive Comment Masking

- Dataset:

[Naver] 2021.01 ~ 2021.11 Top10 news comment 크롤링 (20K)

[욕설감지데이터셋] <https://github.com/2runo/Curse-detection-data> (5K)

[Korean-hate-speech] <https://github.com/kocohub/korean-hate-speech> (15k)

= Total : 약 40K Binary Classification Dataset

[Korean-unsmiled-dataset] https://github.com/smilegate-ai/korean_unsmile_dataset (18K)

= Total : 약 18K Multi-Label Classification Dataset

Model description: [model name] KLUE-Roberta Model

[Pretrained model] 'klue/roberat-base' 사용

- Evaluation metrics: F1 score, accuracy
- Hyperparameters : epoch, number of label, learning rate, type of model(2 type)

✓ Offensive Comment Masking

Challenges:

- 모델의 성능이 어플리케이션에 임베디드 하기에 좋지가 않다 (F1 70점 정도)
 >> Fine-Tune 기법 변경 (뒷장에서 설명)
- 같은 단어라도 다른 맥락을 가질 경우 비방성의 여부에 차이가 날 수 있다
- 단순히 비방표현을 분류하는 것이 아니라 비방의 정도나 카테고리도 분류할 수 있으면 방장봇이 사용자에게 경고와 패널티를 주는데에 용이하게 사용 가능하다
 >> Multi-Label로 된 데이터셋을 활용

Limitation: Multi-Label의 경우 데이터의 개수가 대용량 언어코퍼스로 사전 학습된 모델을 완전히 Fine-Tuned 시키기에 한계가 있음

>> Binary한 모델과 Multi-Label 모델을 양상별하여 최선의 성능을 도출할 것

✓ Offensive Comment Masking

How to Fine-Tune BERT for Text Classification? (<https://arxiv.org/pdf/1905.05583.pdf>)

last layer of BERT gives the best performance.
Therefore, we use this setting for the following experiments.

Layer	Test error rates(%)
Layer-0	11.07
Layer-1	9.81
Layer-2	9.29
Layer-3	8.66
Layer-4	7.83
Layer-5	6.83
Layer-6	6.83
Layer-7	6.41
Layer-8	6.04
Layer-9	5.70
Layer-10	5.46
Layer-11	5.42
First 4 Layers + concat	8.69
First 4 Layers + mean	9.09
First 4 Layers + max	8.76
Last 4 Layers + concat	5.43
Last 4 Layers + mean	5.44
Last 4 Layers + max	5.42
All 12 Layers + concat	5.44

Table 3: Fine-tuning BERT with different layers on
IMDb dataset.

• 논문에서 제안하는 방식

- 일반적으로 뉴럴넷의 다른 layer들은 다른 정보를 capture 한다.
- 하위의 레이어는 좀 더 일반적인 정보를 담고 있다.
- 실제로 분류 모델에서 Bert의 12개 layer중 마지막 4개의 layer만을 사용하여 작업한 경우 성능이 가장 좋게 나온 사례가 있다.
- layer에 대한 parameter를 나누어 learning rate decay시키는 방법이 있다.

✓ Offensive Comment Masking

How to Fine-Tune BERT for Text Classification? (<https://arxiv.org/pdf/1905.05583.pdf>)

So What?

```
pooled_output = torch.cat(tuple([outputs.hidden_states[i] for i in [-4, -3, -2, -1]]), dim=-2)
pooled_output = torch.max(pooled_output, dim=-2).values
pooled_output = pooled_output[:, :]
pooled_output = self.dropout(pooled_output)
logits = self.classifier(pooled_output.view(-1, 768))
```

- In Forward Code,
- 마지막 네 개의 hidden states의 logits값들의 column 방향으로 max인 값을 최종 아웃풋을 위한 hidden state로 사용하였습니다.
- 모델 학습 초반 learning rate 를 2e-5로 매우 낮게 낮추었습니다.

위 두가지 방법은 기존에 사전 학습된 정보를 소실하지 않으면서 파인튜닝을 하는데에 도움이 됩니다.

✓ Offensive Comment Masking

- Evaluation



- Validation Loss가 더 느리게 상승
- epoch 1에서 epoch 3 돌린 기존 모델과 거의 유사한 성능
- Training 진행중입니다.

✓ Offensive Comment Masking Example

- Test Dataset에 있는 Sample을 이용해 확인한 결과입니다.

Type the sentence : 한남동에는 한남들이 많이 사나
해당 댓글은 72%의 확률로 남성에 해당하는 비방성 댓글입니다.

Type the sentence : 한남동에 카페가고 싶다
해당 댓글은 지역에 해당하는 표현이며, clean한 댓글입니다.

Type the sentence : 딱 전라도 마인드임 통수찌는 위선자들
해당 댓글은 82%의 확률로 지역에 해당하는 비방성 댓글입니다.

Type the sentence : 게이 새끼들 다 정상으로 돌아오게 치료받자
해당 댓글은 94%의 확률로 성소수자에 해당하는 비방성 댓글입니다.

Type the sentence : 개독교인들 꺼져라
해당 댓글은 94%의 확률로 종교에 해당하는 비방성 댓글입니다.

Type the sentence : 문재앙
해당 댓글은 75%의 확률로 개인지칭에 해당하는 비방성 댓글입니다.

✓ Offensive Comment Masking Example

Type the sentence : 니 얼굴 박원순 닮았다
해당 댓글은 50% 의 확률로 개인지칭에 해당하는 비방성 댓글입니다.

Type the sentence : 우리나라도 미개한 조선족들 넘친다
해당 댓글은 92% 의 확률로 인종/국적에 해당하는 비방성 댓글입니다.

Type the sentence : 아이돌 빠순이 된장녀 맘충아
해당 댓글은 91% 의 확률로 여성/가족에 해당하는 비방성 댓글입니다.

Type the sentence : 한국페미 = 고유정 수준
해당 댓글은 45% 의 확률로 여성/가족에 해당하는 비방성 댓글입니다.

✓ 채팅 기능 구현

Challenges: 앱의 핵심 기능인 실시간 채팅 기능 구현하기

별도 채팅 서버 구현없이 채팅 기능을 빠르게 구현하기 위해 [Firebase Realtime Database](#) 사용

- 실시간 클라우드 데이터베이스
- 값 이벤트를 수신 대기하여 데이터를 읽을 수 있는 기능을 제공

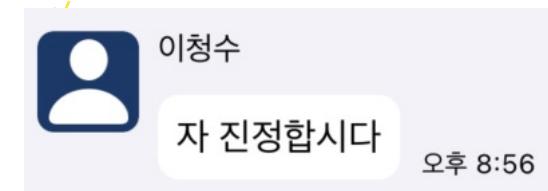
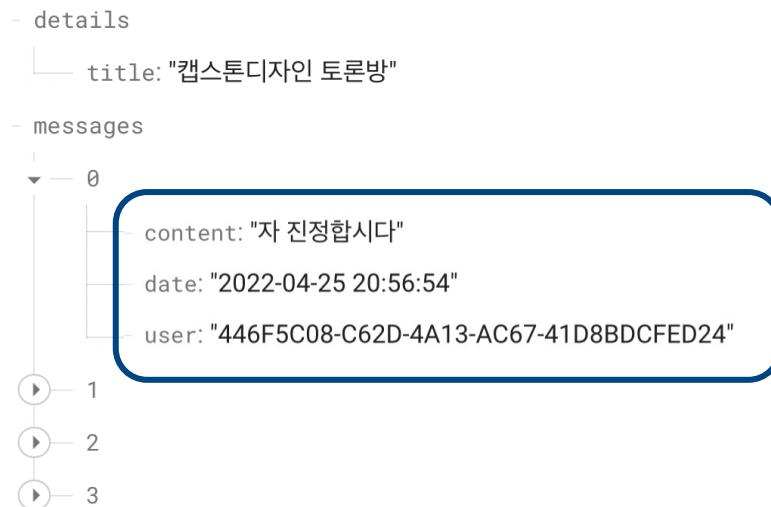
```
refHandle = reference.observe(DataEventType.value, with: { snapshot in
    // Do something with snapshot
})
```



✓ 채팅 기능 구현

왼쪽 아래 사진과 같이 트리형태로 구조를 설계: 사용자가 입력한 채팅을 messages라는 Node에 추가

- messages Node에서 채팅 추가 이벤트가 감지되면 전달된 값을 사용자 화면(View)에 반영

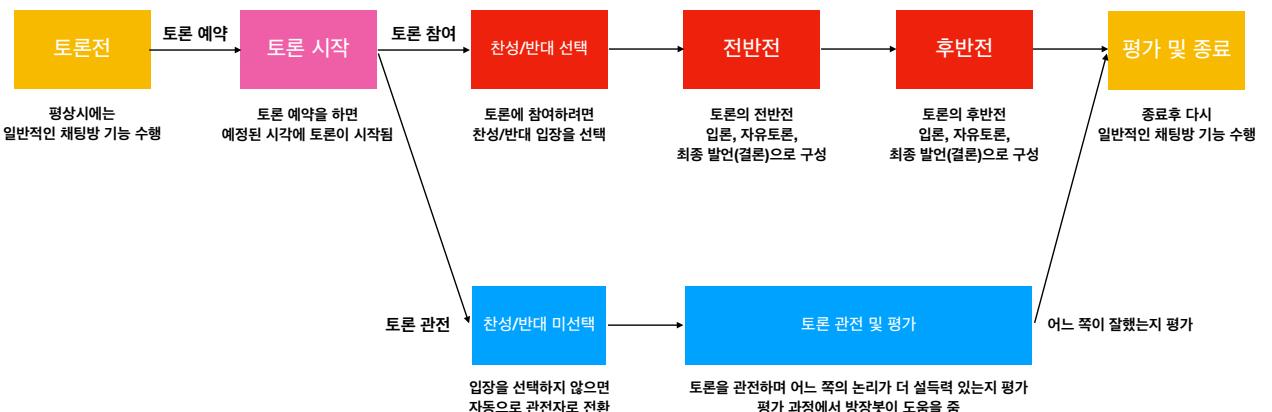


✓ 토론 기능 구현

Challenges: 앱의 두 번째 핵심 기능인 토론 기능을 구현하기

실시간으로 토론이 어떤 상태에 있는지를 알 필요가 있었음

예: 토론이 예약된 상태, 토론이 시작되어 진행을 선택중인 상태 등



✓ 토론 기능 구현

Firebase 실시간 데이터베이스에 토론의 상태 값을 나타내는 flag 값을 저장

해당 값으로 토론이 어떤 상태에 있는지를 파악하고 사용자 화면(View)에 반영

예) 찬성측 발언 시간에는 반대측에서 채팅을 입력할 수 없도록 View의 속성을 변경

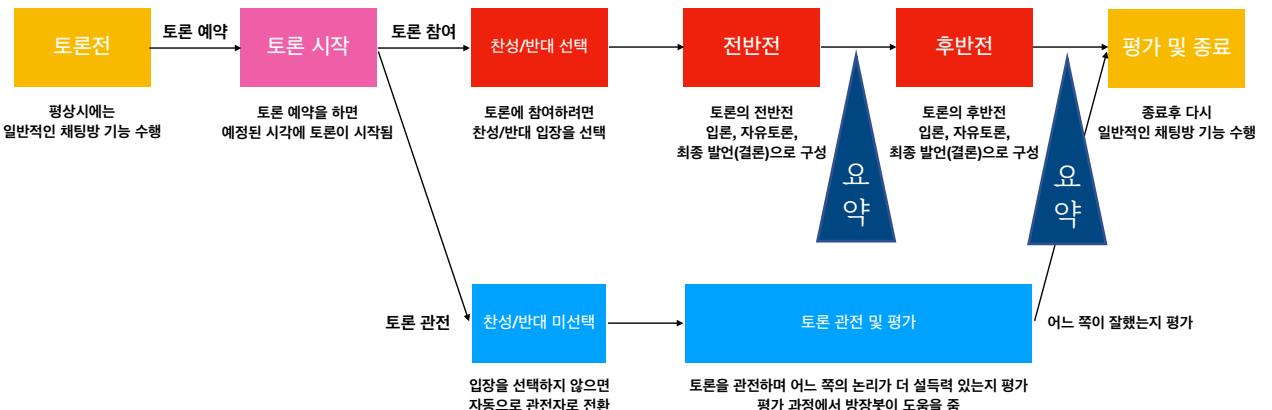
- phase
 └ value: 0

flag	내용	비고
0	평상시	시작 대기중인 토론이 없음. 채팅방 기능 수행
1	시작 대기	토론 시작 시간이 되었으나 찬성/반대/판정단이 최소 1명씩 모이지 않아 시작이 보류됨
2	입론(1)	첫번째 입론 시간. 해당 진영만 발언 가능
3	입론(2)	두번째 입론 시간. 해당 진영만 발언 가능
4	자유토론	자유 토론시간. 찬성/반대측만 발언 가능
5	결론(1)	첫번째 결론 시간. 해당 진영만 발언 가능
6	결론(2)	두번째 결론 시간. 해당 진영만 발언 가능
7	투표	투표 시간. 아무도 발언할 수 없음
8	결과 집계 및 발표	투표 결과 집계 및 발표 시간. 결과 발표 후 0번으로 돌아감

✓ 토론 기능 구현

Limitation: 현재 계획했던 전체 토론 흐름을 구현하지는 못했음

- 전반전/후반전이 아니라 한번만 진행됨
- 요약 기능도 아직 제공하지 못하고 있음



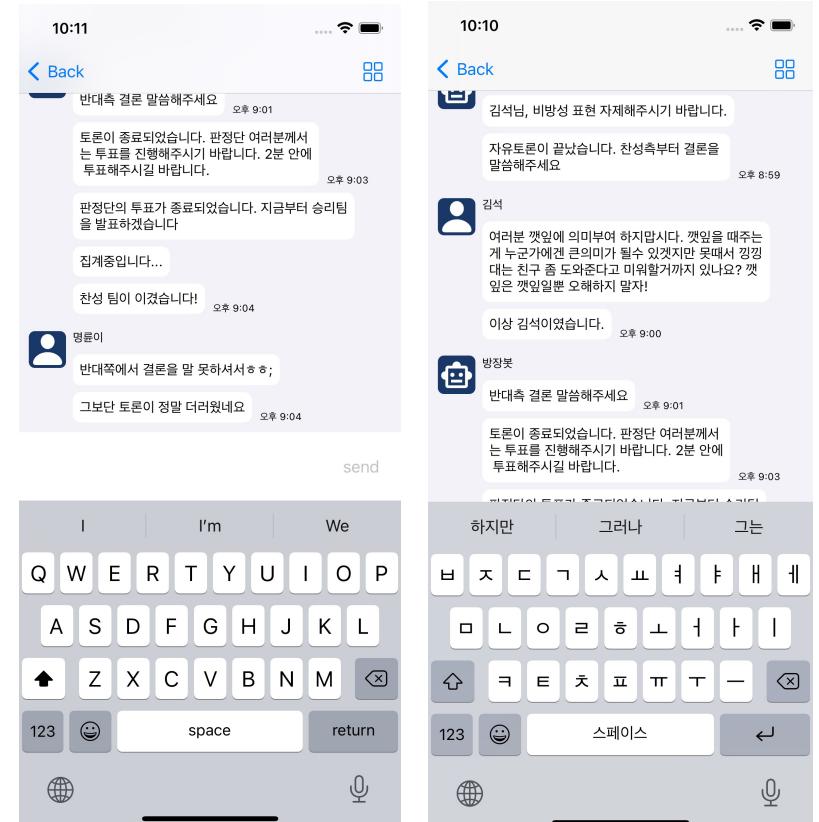
Front-End Development

A Team ⚡ 100% 📰

✓ 사용성 개선

사용자에게 직접 보여지는 화면을 만든다는 Frontend 특성상 사용성을 높이기 위한 시도

- 글자를 입력할 때 올라오는 키보드를 고려한 View 처리
- Darkmode와 같은 접근성(accessibility) 고려



✓ 사용성 개선

사용자에게 직접 보여지는 화면을 만든다는 Frontend 특성상 사용성을 높이기 위한 시도

Limitations: 여전히 제공하지 못하는 접근성 기능이 많음

- 시각 장애인 및 저시력자를 위한 VoiceOver 기능: 화면에 있는 텍스트를 읽어주는 기능
- 사용자가 글자 크기를 정의할 수 있는 Dynamic Type 기능



Handler 작성

Initialize, preprocess, inference, postprocess로 구성

1. Initialize

mar file 압축에 필요한 파일들을 load하고 연결

2. preprocess, inference, postprocess

Json request로 받은 data를 파싱해서 모델에 추론, 결과를 list에 저장한 후 json으로 변환

문제점:

- module에 새로운 class를 생성할 수 없고, return value와 parameter를 특정하는 등의 규정 존재
- 핸들러 파일을 잘못 작성한 경우 서버가 다운됨
- 어떤 부분에 에러가 있는지 디버깅하기 어려움

Back-End Development

A Team ⌂ ⌂ ⌂ 100% 🔋

모델 서빙 도구 선택

1. 초기 시도: Flask api 사용

문제점

- 추론 시 GPU를 사용하기 위한 I/O 처리, GPU 자원에 대한 처리 어렵다
- 여러 개의 input이 동시에 들어갈 때 exception을 핸들링하기 어렵다

2. 해결: Torchserve 사용

장점

- GPU 사용과 동시에 들어가는 input에 대한 자바 서버가 구현되어 있다
- Java server와 inference model을 연결하는 handler만 작성하면 서빙 가능

Back-End Development

A Team 100%

모델을 서빙할 서버 컴퓨터 선택

1. 초기 시도: AWS EC2 사용하여 추론하기로 결정

문제점:

성능 나쁜 CPU 1개로 BERT 기반 무거운 모델 inference를 해서 response time이 너무 길다

2. 해결 방안

Distillbert를 사용해서 크기가 작은 모델을 서빙

GPU를 사용하여 추론이 가능한 환경에서 서빙

3. 결정

Distillbert의 성능이 기존 roBERTa모델보다 떨어짐

-> 모델은 Roberta base로 유지하고, GPU를 사용하여 추론이 가능한 환경에서 서빙

4. 환경

Windows OS

Linux OS 환경을 위해 Docker 설치

GPU 사용을 위한 nvidia-docker, cuda 설치

DDNS설정, 포트포워딩 설정

Back-End Development

A Team 100%

서버 response time

GTX 1050Ti

1. klue-Roberta-base classification

평균 0.087초

분명 둘 중 한 상대는 사랑이 아닌 의지, 편함, 익숙함과 같은 감정으로 함께하는 것이며, 다른 한 상대로만으로는 총 족되지 못하고 있다는 것입니다.	85	79	105	90	60
만약 등록세를 징수할 경우, 분명히 이 과정에서 등록세를 내지 않고 유기를 하는 사람들도 생길 것이고 이 또한 문제입니다.	70	62	56	58	54
만약 등록세를 징수할 경우, 분명히 이 과정에서 등록세를 내지 않고 유기를 하는 사람들도 생길 것이고 이 또한 문제입니다.	53	55	58	80	59
만저십새끼 표정에 항상 스웨이 넘치노	51	54	72	77	67
확실히 40대 새끼들이 명청한 새끼들이 많음. 공부에 손 놓은 새끼들이 많아서 그런지 40대들은 확실히 좀 지능이 떨어지는 새끼들 많은거 같다	63	74	95	78	74
차라리 등록세 보다는 반려동물을 키우게 되면 예전에 티비에서 봤던 것인데, 반려동물의 몸에 반려동물과 주인에 대한 정보가 있는 칩 같은 것을 이식해서 주인이 유기할 경우 칩을 통해 주인을 찾아 처벌을 하는 등, 이런식으로 해나가는 것이 더 효율적일 것이라고 생각합니다	87	96	89	97	91

Back-End Development

A Team 100%

서버 response time

GTX 1050Ti

2. Kobart
summarization
평균 4.45초

부모, 가족, 친구와의 관계 속 생기는 감정과 연인과의 감정은 다릅니다. 연인은 부 모, 가족, 친구와 다르게 서로의 인생을 약속하며 함께 재산, 자식을 꾸려나가는 것이 가능한 암묵적 존재입니다. 서로의 성생활을 유일하게 공유할 수 있는 존재이며, 친구, 부모, 선생 다양한 역할을 기대할 수 있는 존재이기도 합니다. 연인의 존재는 신뢰, 공유, 아낌을 베이스로 단단하게 이뤄집니다. 두 사람을 연인의 기준으로 동시에 사랑한다는 것은 존재할 수 없습니다. 분명 둘 중 한 상대는 사랑이 아닌 의지, 편함, 익숙함과 같은 감정으로 함께하는 것이며, 다른 한 상대로만으로는 충족되지 못하고 있다는 것입니다. 두 사람을 동시에 사랑하는 것은 사랑의 의미가 정확히 무엇인지 모르는 것이거나, 사랑의 무게를 제대로 짊어질 자신이 없어 늘어놓는 변명일 뿐입니다.	5580	4505	5025	4471	4774
아 그게 동물 등록제라고 불리는 거군요. 그렇다 해도 전 여전히 반려동물 등록세에 대해서는 부정적인 입장이네요. 차라리 현재 동물 등록제를 더 신경 써서 부족한 점들을 보완하고, 이게 이미 시행이 되고 있는 건 알고 있었는데 그래도 여전히 유기되는 동물들이 많은 걸로 알고 있습니다. 개를 예로 들면, 한때 유행을 타서 많이 키웠던 웰시코기가 한동안 유기를 많이 당했고, 그 후에는 시바견의 유기가 좀 많았던 것으로 알고 있습니다. 만약 동물 등록제가 제대로 시행이 안돼서 주인을 찾을 수 없기 때문에 유기를 한 것이라면, 부족한 점을 보완해서 제대로 시행해야 하고, 제대로 시행이 됐든 안됐든 떠나서 유기에 대한 처벌을 강화해야 한다고 생각합니다. “동물을 물건과 구별되는 별도의 존재로 인정한다”는 조항에 맞게, 생명을 유기한 것에 대해 강하게 처벌해서, 끝까지 책임질 수 없다면 아예 키우지 못하게 해야 된다고 생각합니다. 전 비단 동물 문제만이 아니라, 여러 문제들에서 우리나라의 처벌이 너무 약하지 않나? 하는 생각들이 듭니다. 타인에게 피해나 소음 문제도 제도적 방침을 마련함과 동시에 처벌도 수위가 약한 것이 아닌, 강하게 했으면 합니다. 그래야지 사람들이 제대로 책임을 질테니까요.	3810	3819	3785	3831	4831
저는 반려동물 등록세를 반대합니다. 확실히 반려동물 가구 수에 비해 제도적 준비가 되어있지 않다고 저도 생각하지만, 그렇다고 해서 반려동물 등록세가 문제를 해결할 거라고 생각되지 않기 때문입니다. 만약 등록세를 징수할 경우, 분명히 이 과정에서 등록세를 내지 않고 유기를 하는 사람들도 생길 것이고 이 또한 문제입니다. 제도적으로 필요한 것은, 단순히 반려동물을 키우기 위해 세를 내는 것이 아니라, 반려동물을 키울 때 제대로 책임을 지게 하기 위한 제도가 필요합니다. 반려동물 등록세를 징수한다고 해서 유기가 사라지진 않을 겁니다. 말 그대로 반려동물 등록세는, 처음 반려동물을 키우기 시작할 때만 내지, 그 후에는 내야하는 비용이 없기 때문입니다. 분명 또 키우다가 유기하는 사람들이 나올테고, 더 나아가 소음이나, 타인을 공격하는 문제에 대한 근본적인 해결책은 반려동물 등록세가 아닙니다. 차라리 소음이나 공격에 대한 책임을 해당 반려동물의 주인에게 묻는 제도 등을 도입해 사람들이 길거리나 집에서 이러한 문제들을 더 신경 쓰고 책임지게 만드는 것이 효과적이라고 봅니다. 사람들이 반려동물에 대해 책임의식이 떨어지는 이유는, 달리 처벌을 받지 않기 때문입니다. 차라리 등록세 보다는 반려동물을 키우게 되면 예전에 티비에서 봤던 것인데, 반려동물의 몸에 반려동물과 주인에 대한 정보가 있는 칩 같은 것을 이식해서 주인이 유기할 경우 칩을 통해 주인을 찾아 처벌을 하는 등, 이런식으로 해나가는 것이 더 효율적일 것이라고 생각합니다	4273	4269	4284	4694	4802

데모 시연

- ✓ 토론 프로세스 설명
- ✓ 데모 시연



토론중 발생하는 비방성 표현 경고 및 마스킹

