



# Final Project\_NLP\_팀 리포트(10조)

## ○ 프로젝트 개요

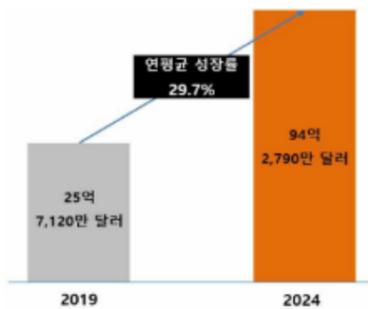
### • 프로젝트 소개

- WERODA(위로다)
  - 사용자에게 심리상담을 해주는 Multi-Turn 기반의 챗봇
  - KoGPT2 및 Elastic Search를 이용해 구현

### • 주제 선정 배경

- 현대 사회에서 꾸준히 증가하고 있는 개인 심리 상담 수요
- 코로나19 통합 심리 지원단에 따르면 20년 2월에 13,265건에서 21년 8월에 135,407건으로 상담 횟수가 920% 증가
- 이는 코로나로 인해 현재 다양한 고객 서비스 지원에 활용되고 있는 챗봇 서비스에 대한 수요가 앞으로도 커진다는 점을 시사

### • 시장성



→ 2019년 기준으로 챗봇 시장은 연평균 성장률 29.7%로, 2024년에는 약 11조원에 이를 것으로 전망 (Marketsandmarkets, chatbot market, 2019)

### • 기존 서비스의 한계

- 기존 심리 상담 챗봇 현황 사례
  - 과학기술 정보통신부 & 한국연구재단 : ‘마이멘탈 포켓’
  - 서울시 : ‘누구나 챗봇’

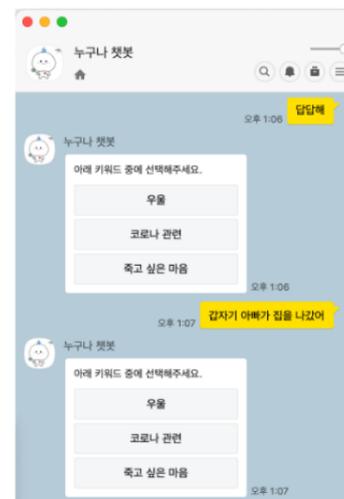
"우울증 치료? 복잡 터져"...디지털치료제 시범서비스 직접 써보니

댓글 0 | 공감 0

과거정통부, 마이멘탈포켓 1월부터 시범 서비스  
인공지능 챗봇 '상담' 코너, 영동한 답변 속출  
다른 서비스는 아직 준비 중!



[아시아경제 김병수 기자] "우울증 치료하려 상담했다가 복잡 터지게 생겼다".



- 현재 심리상담원과 직접 대화할 수도 있는 챗봇은 주로 유료 서비스
- 무료 심리 상담 챗봇 서비스는 대부분 단순한 트리형 구조로 사용자의 만족도가 낮음

- **프로젝트 목표**

- 챗봇이 기존 서비스보다 향상된 성능으로 사용자의 심리 상담을 할 수 있도록 구현하는 것을 목표로 함
  - 단순 응답이 아닌 일상 대화형 채팅
  - 사용자의 발화 의도를 정확히 파악한 문맥에 맞는 응답 서비스

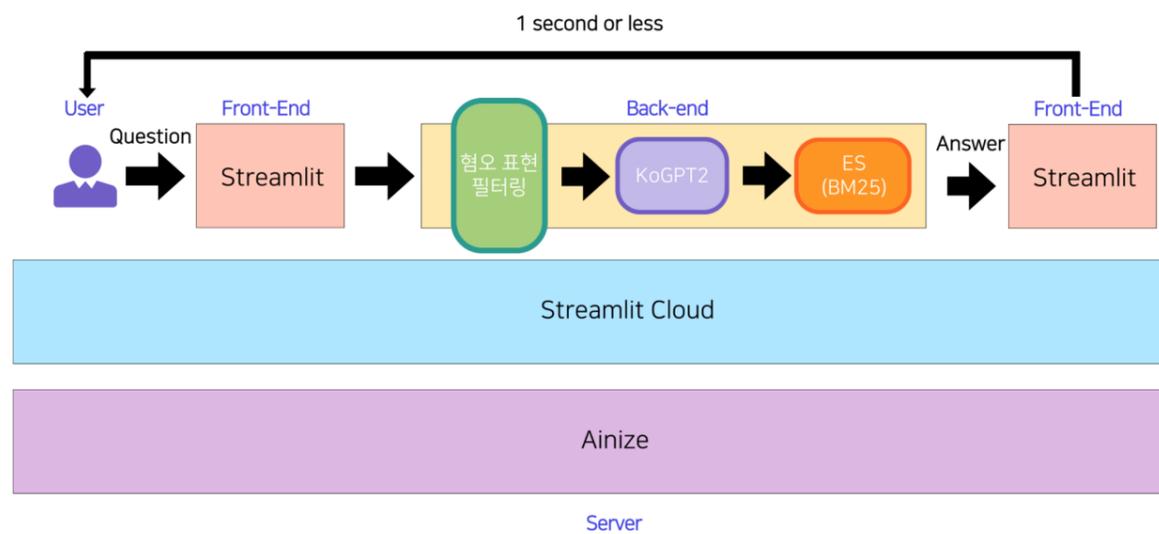
- **기대 효과**

- 딥러닝 기술로 심리 상담원이 없는 챗봇을 구현함으로써 인건비 절감
- 사용자의 성향, 취미 등의 정보를 이해해 맞춤형 상품 추천, 이미지 광고 등 다양한 서비스로 발전 가능

- **활용 장비 및 재료(개발 환경, 협업 tool 등)**

- 서버환경 : Ubuntu 18.04.5 LTS , GPUv100
- 개발툴 : vscode, Streamlit, Ainize, jupyter notebook
- 협업툴 : Git, Github Project Slack, Zoom

- **프로젝트 Workflow**



- 사용자가 Question을 입력
- Korean UnSmile Dataset으로 학습한 혐오 표현 필터링 모델을 통해 Question의 혐오 표현 점수 확인 → 0.1점 미만이면, 혐오 표현으로 간주해 사용자에게 '순화된 표현을 사용하세요' 경고 및 Question 재입력 요구
- Model을 이용해 Question에 대한 적절한 Answer 생성 및 출력
- Streamlit을 이용해 위 과정을 Chatting 형태의 Web 구현

- **프로젝트 팀 구성 및 역할**

- 김남현(T3021) : Hate Speech Filtering, Data Collection, ChatBot Evaluation
- 민원식(T3079) : PM, Validation Server, ChatBot Evaluation
- 전태양(T3194) : EDA, Data Collection & Preprocessing, ChatBot Evaluation
- 정기원(T3195) : Data Collection, Text Style Transfer, Elasticsearch, ChatBot Evaluation
- 주정호(T3211) : Model Research, UI, Server Deploy, ChatBot Evaluation
- 최지민(T3223) : Data Collection, DialogBERT, ChatBot Evaluation

## ○ 프로젝트 수행 절차 및 방법

### ● 프로젝트 개발 Process

개발 과정을 아래와 같이 크게 5가지 파트로 분류함.

- EDA : 웰니스 대화 스크립트 데이터셋의 분포 확인
- Data Collect & Processing : 데이터 수집 후 모델 학습 및 Retrieval을 위한 전처리
- Modeling : 여러 모델을 구현하고 성능 향상을 위해 Parameter Tunning 및 다양한 기능 추가
- Model Test & Monitor : Monitoring Tool을 이용하여 모델을 서버 환경에서 테스트
- 협업 Tool 관리 및 기타(문서 정리) : Notion, Git, 구글 스프레드 시트 이용

### ● 프로젝트 수행 및 완료 과정(Work Breakdown Structure)

2022년 5월							< 오늘 >
일	월	화	수	목	금	토	
8	9	10	11	12	13	14 프로젝트 주제 선정 회의	
15 프로젝트 주제 선정 회의	16	17	18	19	20 dialogBERT를 이용한 멀티턴 챗봇 구현	21	
챗봇 평가 메트릭 정하기							
백엔드 구현							
koGPT2를 이용한 멀티턴 챗봇 및 UI 구현							
koGPT2를 이용한 싱글턴 챗봇 및 UI 구현							
RoBERTa를 이용한 챗봇 및 UI 구현							
데이터 수집작업(명언, 음악)							
데이터 EDA							
단일턴 챗봇 베이스라인 모델 성능 최적화 +							
22	23	24	25	26	27	28	
챗봇 평가 메트릭 정하기							
챗봇 SSA 성능평가 Tool 제작							
데이터 수집작업(명언, 음악)							
서버 구축 및 서비스 연결 작업							
dialogBERT를 이용한 멀티턴 챗봇 구현							
데이터 EDA							
문장 유사도 측정 모델 구현							
단일턴 챗봇 베이스라인 모델 성능 최적화 +							
모델 서비스 배포 작업							
29	30	31	6월 1일	2	3	4	
문장 유사도 측정 모델 ...							
챗봇 최종 성능평가							
모델 서비스 배포 작업							
1차 발표자료 작성(내부)							
5	6	7	8	9	10	11	
1차 발표자료 작성(내부)							
2차 발표자료 작성							
멘토 피드백 ... 오전 11:00							
멘토 피드백 ... 오전 11:00							
최종 결과 제출 오후 5:00							

## ○ 프로젝트 수행 결과

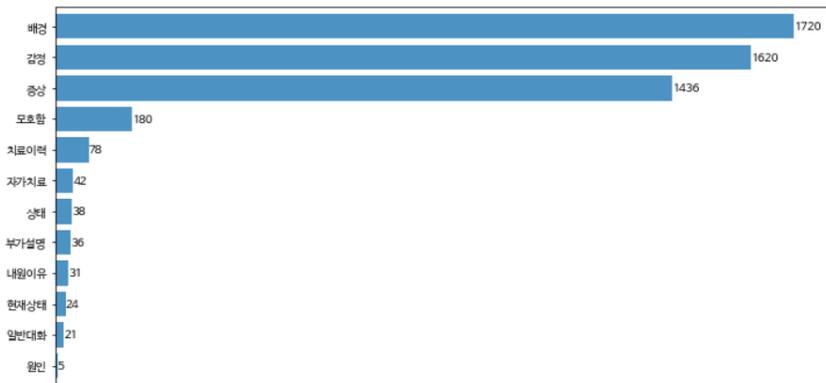
### 1. 사용된 데이터셋

- [AI-Hub] 웰니스 대화 스크립트 데이터셋
  - 정신 건강 상담 주제의 359개 대화 의도에 대한 5,232개의 사용자 발화 및 1,023개의 챗봇 발화
  - KoGPT2 모델 학습, Retrieval 데이터셋 구축에 사용
- [songys/Chatbot\_data] Chit-Chat 데이터셋
  - 다음 카페 "사랑보다 아름다운 실연([http://cafe116.daum.net/\\_c21\\_/home?grpId=1bld](http://cafe116.daum.net/_c21_/home?grpId=1bld))"에서 자주 나오는 이야기들을 참고하여 제작
  - 챗봇 학습용 문답 페어 11,876개
  - 일상 0, 이별(부정) 1, 사랑(긍정) 2로 라벨링
  - Retrieval 데이터셋 구축에 사용

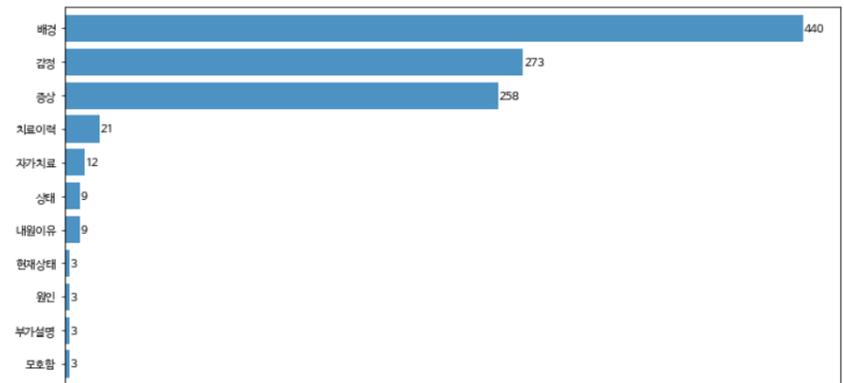
- [AI-Hub] 감성 대화 말뭉치
  - 60가지 다양한 감정의 코퍼스 27만 문장
  - 우울증 관련 및 대화 응답 시나리오 포함
  - 감성 대화 엔진 또는 챗봇을 개발하려는 목적에 맞추어 제작
  - Retrieval 데이터셋 구축에 사용

## 2. EDA

- 초기 데이터셋은 감정/감정조절이상/화, 감정/감정조절이상과 같은 방식으로 카테고리를 분류, 총 359개의 카테고리로 한눈에 살펴보기가 어려움
- 따라서 '/'를 기준으로 첫 카테고리를 대주제로 취급하여 여러 분포를 살펴봄
- 먼저 데이터 분포를 살펴보고 데이터 불균형 문제를 확인함



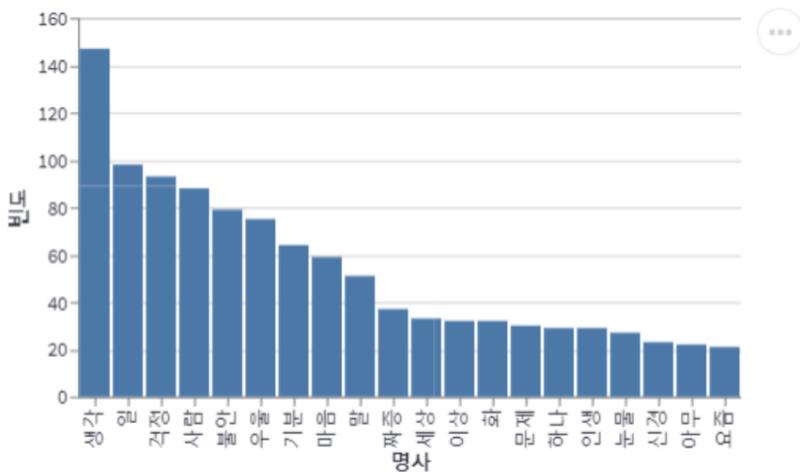
질문 데이터셋 분포



답변 데이터셋 분포

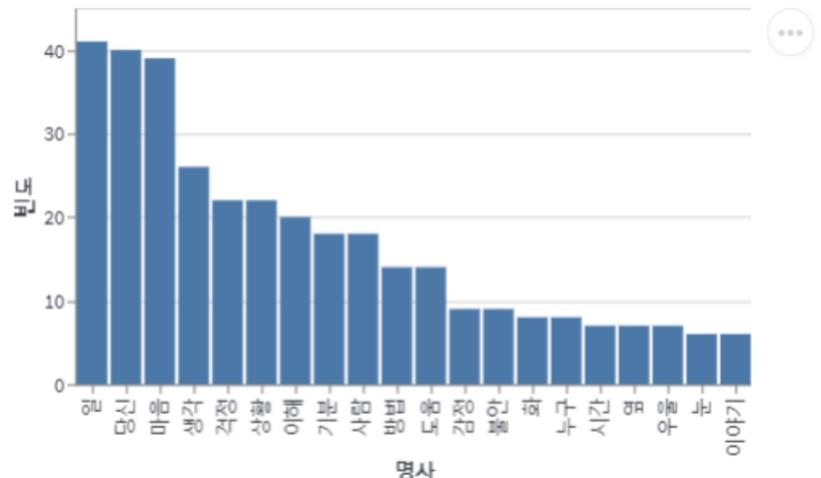
- 카테고리별 자주 등장하는 단어 빈도 확인

### "감정" 키워드 top 20



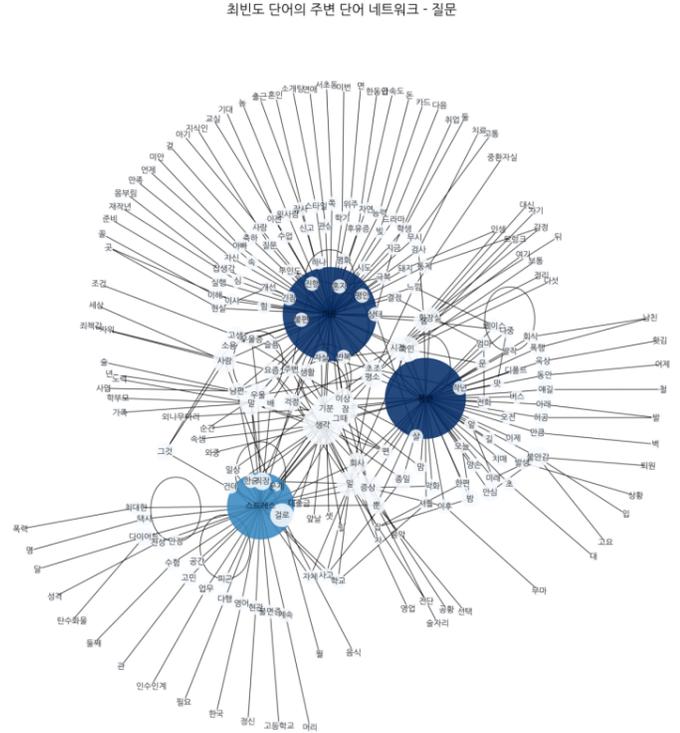
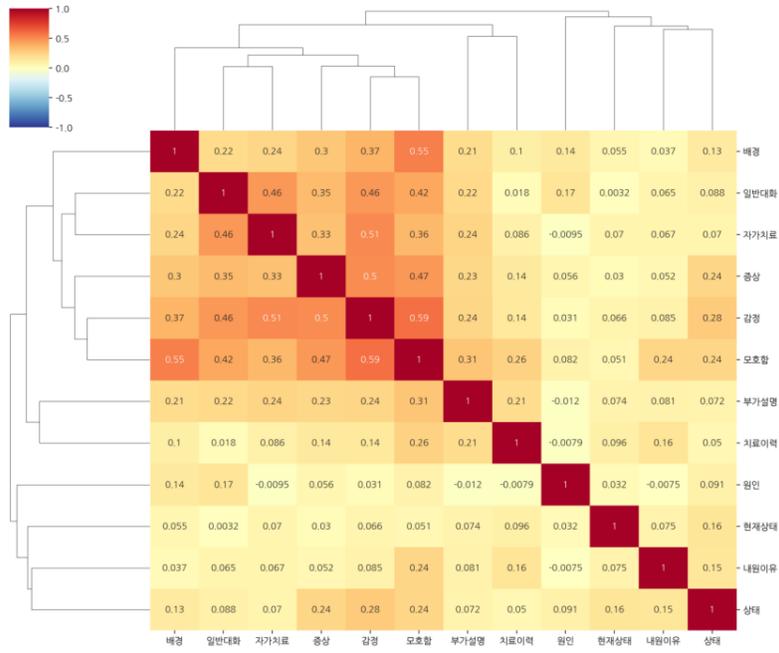
질문 데이터셋의 '감정' 카테고리

### "감정" 키워드 top 20



답변 데이터셋의 '감정' 카테고리

- 카테고리를 줄인다면 발화자의 감정을 분류할 때 더 쉽게 분류할 수 있을거라 생각했고, 카테고리별로 유사한 카테고리가 있는지 파악하기 위한 EDA 진행
- count 기반 상위 500 단어에 대한 카테고리 상관관계 분석
- 스트레스,불안, 마음 키워드(예시)로 네트워크 분석

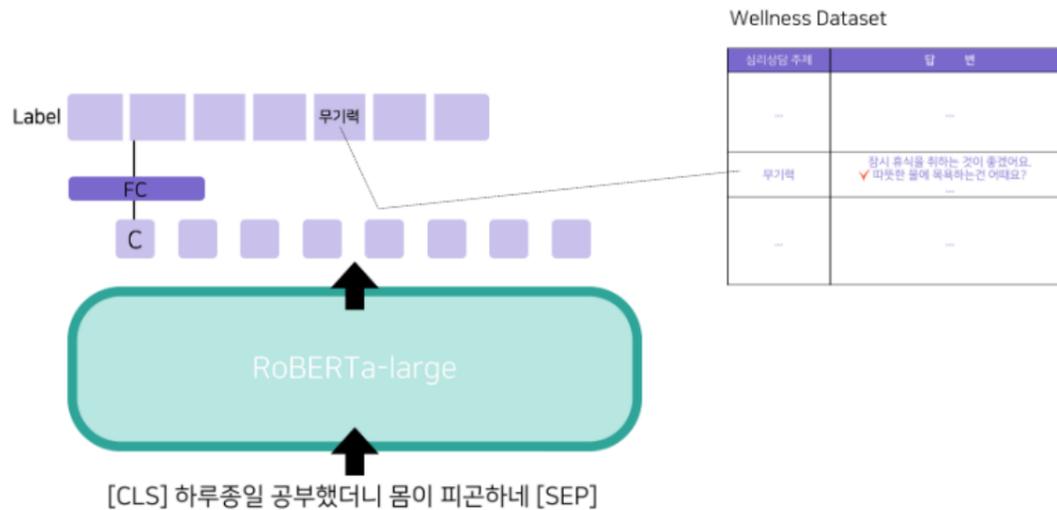


### 3. 데이터 처리

- 전처리
  - 챗봇 답변 데이터 중복 제거
  - Elasticsearch Nori 형태소 분석기 사용
  - 불용어는 사용하지 않을 때 챗봇 성능이 더 높게 나옴 (SSA)
- 후처리
  - 발화와 답변에서 키워드를 추출하여 챗봇의 성능을 향상시키려 하였으나 실패
- Augmentation
  - Text Style Transfer
    - Conditional BERT
    - Stable Style Transfer
    - 성능이 낮게 나와 실사용에는 무리가 있음

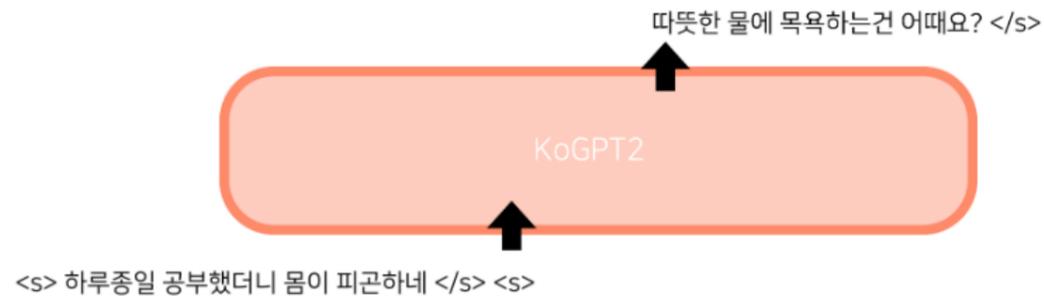
### 3. 모델

- RoBERTa



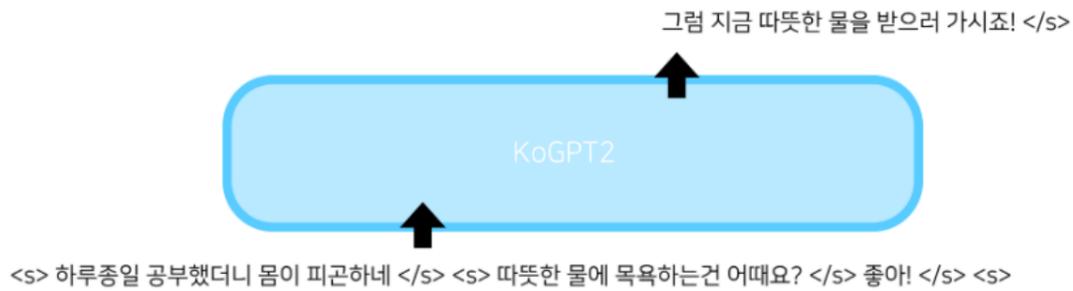
- Question을 심리상담 주제로 분류하여 해당 주제의 답변을 무작위로 선택
- 일상 대화도 심리상담 주제로 분류 → 일상 대화 불가능

- KoGPT2 (Single-Turn)



- Question으로 Answer을 생성 → 분류 모델에 비해 훨씬 자연스러운 대화 가능
- 오직 사용자의 Question으로 Answer을 생성하기 때문에, 이전 대화를 활용하지 못함 → 새로운 맥락의 대화 불가능
- 생성 모델이기 때문에 생성된 답변의 완성도가 떨어짐

- KoGPT2 (Multi-Turn)



- 최대 4턴의 이전 대화와 Question을 결합해 Answer을 생성 → 이전 대화를 활용한 연속된 대화 가능
- 이전 대화와 상관없는 Question도 이전 대화와 함께 결합되어 Answer을 생성 → 맥락에 맞지 않는 답변 생성
- 생성 모델이기 때문에 생성된 문장의 완성도가 떨어짐

- KoGPT2 (Multi DownStream Task)



- 다양한 DownStream Task를 수행할 수 있는 GPT2의 장점을 이용 → Question으로 심리상담 주제 분류와 Answer 생성을 동시에!
- 심리상담 데이터는 Wellness DataSet, 일상 대화 데이터는 Chit-Chat DataSet을 이용
  - Wellness DataSet에서 챗봇의 답변에 대한 응답 데이터는 <긍정답변>과 <부정답변>으로 전환 → Multi-Turn 대화에서 사용

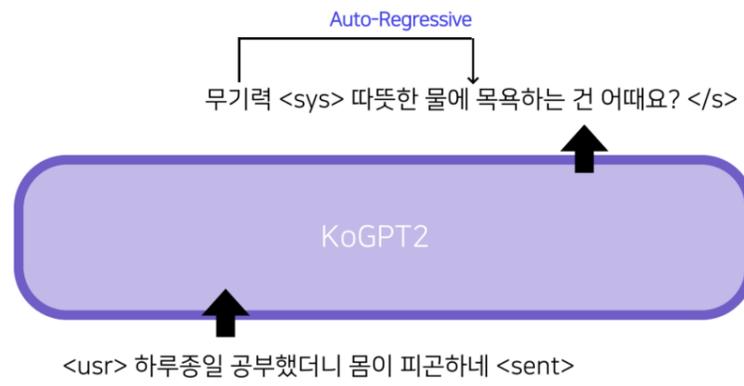
user	system	sentiment
너무 불안한거 있지.	불안감을 느끼고 계시는군요. 00님이 불안감으로 많이 힘드신 건 아닌지 걱정이 돼요.	불안
완전 힘들고 짜증나요.	평소 즐기던 일이 더 이상 즐겁지 않거나 흥미가 사라졌나요?	무기력
다 때려치고 싶어.	마음이 잠깐 지쳤나 봐요. 아무 의미 없는 일은 없을 거예요.	의욕 상실
어 맞아	그러셨군요. 많이 당황스러우셨을 것 같아요. 이런 상황이 오래 지속될까 걱정돼요.	긍정답변
그건 아닌데	그렇군요. 불안을 줄이는 방법으로 꾸준한 운동이나 명상, 규칙적인 수면이 도움이 될 수 있어요.	부정답변

[KETI] Wellness Dataset

user	system	sentiment
12시 명!	하루가 또 가네요.	일상
안녕	안녕하세요!	일상
3박4일 놀러가고 싶다.	여행은 언제나 좋죠!	일상
반가워!	저도 반갑습니다!	일상
넌 누구야?	저는 '핫식스' 팀에서 개발한 위로봇입니다.	일상

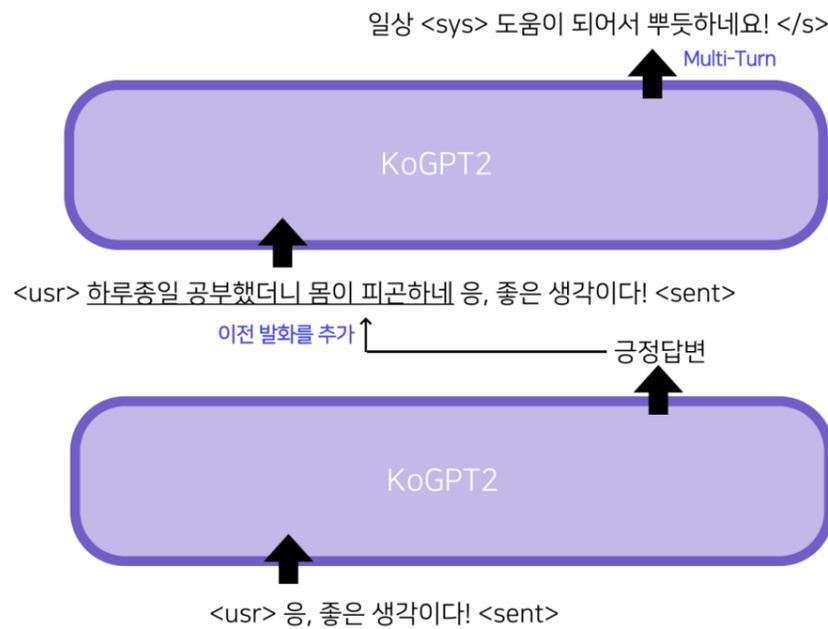
[songys/Chatbot\_data] Chit-Chat Dataset

o Auto-Regressive



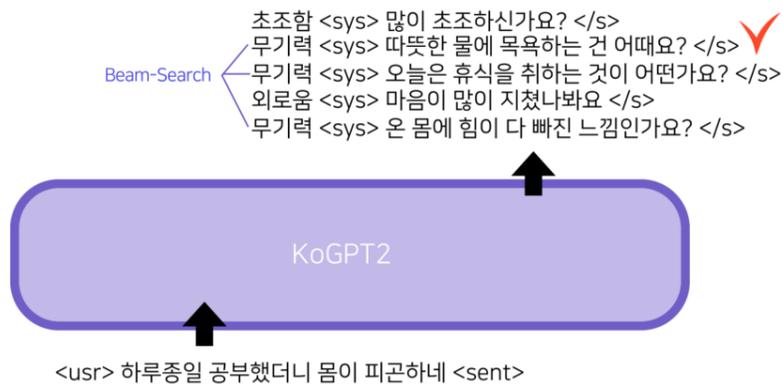
- GPT2는 이전 토큰들을 이용해 다음 토큰을 예측 → Question 뿐만 아니라, 예측한 심리상담 주제를 이용해 Answer을 생성 → 문맥에 맞는 Answer 생성

o Multi-Turn



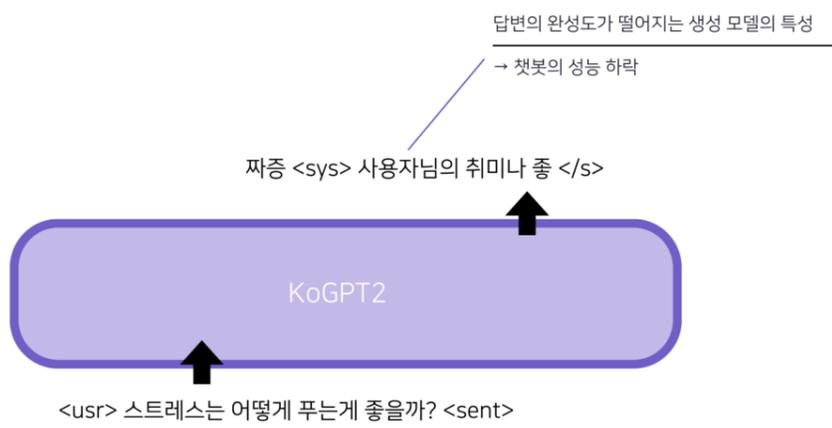
- Question을 긍정 답변 및 부정 답변으로 예측했을 때, 사용자의 이전 Question을 추가해 다시 Task 진행 → 연속한 대화 수행 가능

o Beam Search

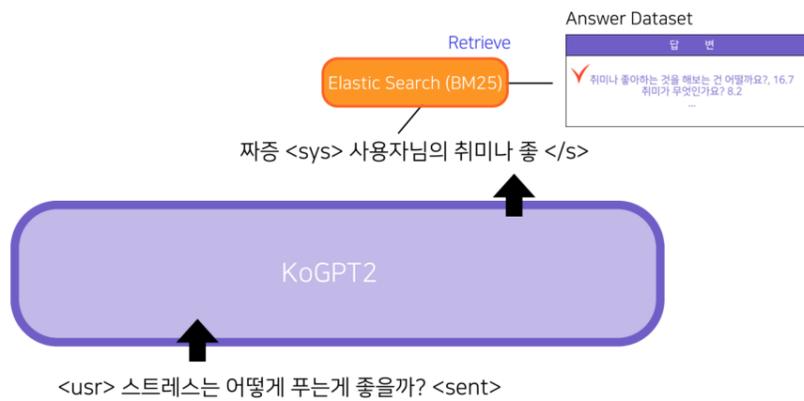


- Task를 여러 번 수행하여 최다 예측된 심리상담 주제의 Answer를 최종 답변으로 추출 → Answer의 정확도 향상

○ Retrieve



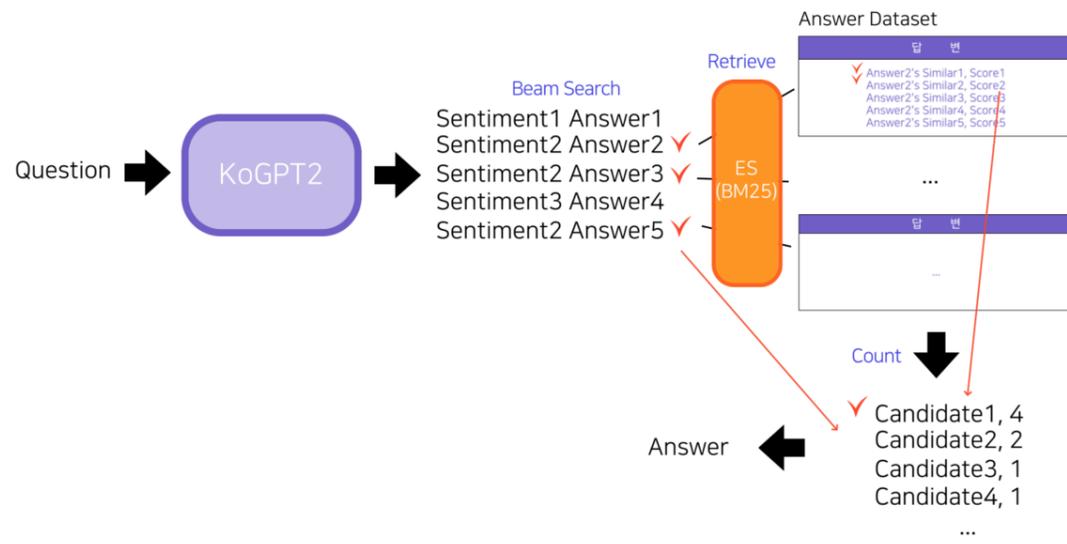
- GPT 계열의 생성 모델의 특성 상 Answer의 완성도가 떨어져 챗봇의 성능 하락을 야기



→ Answer Dataset을 구축하여 Elastic Search의 BM25 Retrieval을 이용해 생성된 Answer과 가장 유사한 답변을 Answer Dataset에서 추출하여 답변으로 채택

○ Conclusion

- Workflow



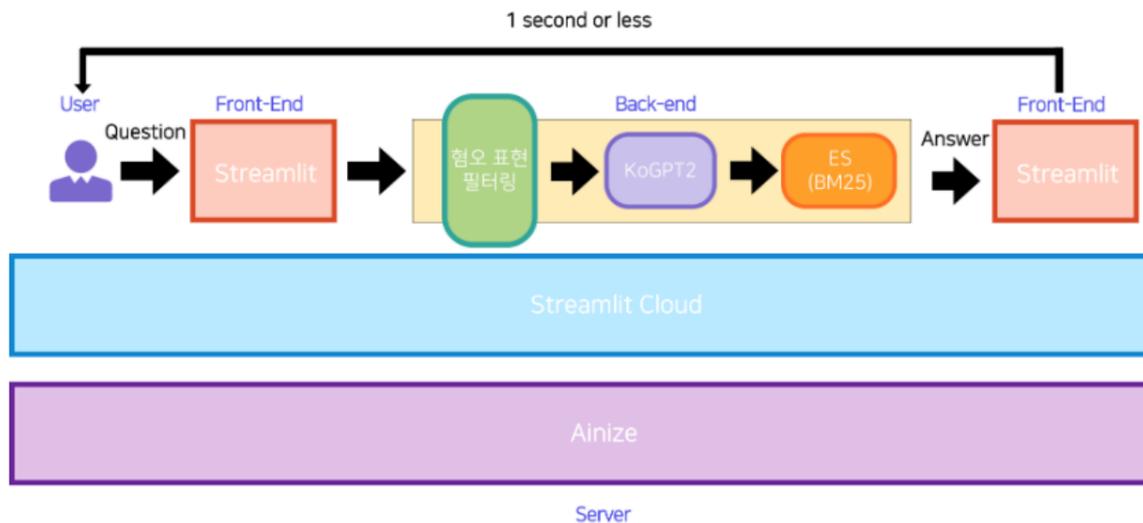
- 위의 기능들을 모두 결합하고 발전시켜 다음과 같은 과정으로 최종 답변으로 선택
  - KoGPT2: Question을 이용해 심리상담 주제와 Answer를 5번 생성
  - Beam Search: 최다 예측된 심리상담 주제의 답변을 채택 → 답변 후보에 추가
  - Retrieve: 채택된 답변과 유사한 답변을 Answer Dataset에서 5개 검색 → 유사도 점수가 10점 이상인 유사 답변을 답변 후보에 추가
  - Count: 답변 후보에서 최다 빈도의 답변을 최종 답변으로 선택

■ Timeline



5. Product Serving

- Service Architecture



## 6. Result

- 모델 성능평가 지표 (SSA)
  - 2020년 구글이 챗봇 Meena를 발표하면서 도입한 대화 만족도 평가 지표
  - Sensibleness : 답변의 맥락과 논리성 평가
  - Specificity : 답변의 구체성과 사람과의 유사성 평가
  - e.g. 너 중국 음식 좋아해?
    - 오늘 메뉴가 뭔가요? (Sensibleness :0, Specificity: 0)
    - 응, 좋아해. (Sensibleness :1, Specificity: 0)
    - 응, 난 그 중에 '짜장면이' 제일 좋아. (Sensibleness :1, Specificity: 1)
- 평가자 간 일치도 평가
  - 구글 Meena 발표 논문의 작업자 간 일치도 평가 방법을 참조
  - Krippendorff's alpha : 0~1 사이의 값(0 불일치, 1 일치)을 가짐
  - Agreement는 SSA 계산을 위해 사용.

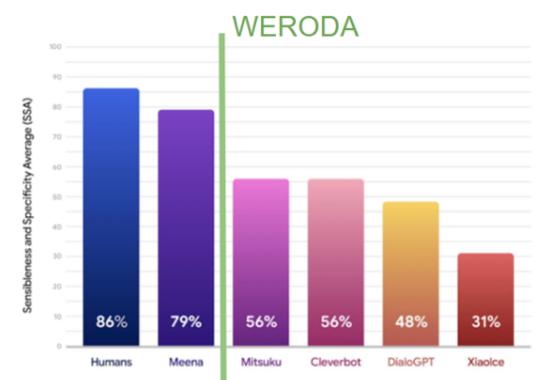
< Our Team Worker Agreement >			< Meena's Crowd Worker Agreement >		
Metric	Agreement(%)	Krippendorff's alpha	Metric	Agreement (%)	Krippendorff's alpha
Sensibleness	66.6	0.74	Sensibleness	76 ± 3	0.42 ± 0.03
Specificity	59.4	0.67	Specificity	66 ± 2	0.30 ± 0.05

⇒ 팀의 작업자 간 일치도 평가 점수가 구글 Meena 발표 논문보다 높음 → 팀의 평가 일치도 ↑

- 모델 별 성능 측정 및 비교
  - 모델 성능 측정

Model	Sensibleness	Specificity	SSA	비 고
RoBERTa	.	.	.	평가 불가
KoGPT2 (Single-Turn)	42.13	38.49	40.31	.
KoGPT2 (Multi-Turn)	39.39	34.57	36.98	.
KoGPT2 (Mutl Downstream-Task)	66.60	59.26	62.93	SOTA

- 타 모델과 비교



## ○ 자체 평가 의견

### • 결과 및 고찰

- 최근에 런칭한 루다와 같은 상용 챗봇은 다양한 주제에 대해 대화를 나눌 수 있어 성능 평가 비교의 대상으로 삼기 어려운 점이 있으나, 심리 상담 주제로 한정한다면 정부의 챗봇에 비해 사용자의 만족도가 향상
- 다만 자연스러운 대화를 위해 더 많은 DataSet을 확보하는 부분과 각 사용자에게 적합한 챗봇의 다양한 페르소나를 반영하는 것이 당면 과제
- 모델학습 구조의 및 Retriever 불용어 설정 변경을 통해 좋은 챗봇을 만들 수 있었음
- Product Serving 수업에서 배운 내용을 활용하여 배포까지 완성하였음

- **후속 개발 및 연구 목표**

- 일상 대화 및 심리 상담과 관련된 더 많은 학습 DataSet 확보
- 사용자와 자연스러운 대화를 위해 챗봇에 페르소나 도입
- Retrieval 후보를 의미에 기반하여 선정하도록 하여 답변의 정확도 향상(현재 BM25 문서유사도에 의해 후보 선정)
- 챗봇 성능 평가 시 정확성을 높이기 위한 평가 가이드라인 구체화
- 사용자 취향, 취미 등을 바탕으로 맞춤형 서비스 제공에 관한 기능 구현 및 해당 서비스로 발전

- **잘한 점과 아쉬운 점**(팀 별 공통 의견 중심으로 작성하며, 2~3장 분량을 고려하여 개인적인 의견은 개인 회고 부분에서 작성할 수 있도록 합니다.)

- **잘한 점들**

- 다양한 모델을 시도했고, 최종 모델이 이전 모델의 단점을 모두 해결 및 우수한 성능을 보여준 것
- Transformer에 Elastic Search를 결합시켜 좀 더 다양한 답변을 생성한 것

- **시도 했으나 잘 되지 않았던 것들**

- 연속 대화를 위해 DialogBERT를 시도했으나, 학습이 부족하여 대화 맥락을 파악하지 못하는 모습을 보임.
- Text Style Transfer을 이용해 챗봇 답변의 말투를 변환시키는 것

- **아쉬웠던 점들**

- 명언 추천, 노래 추천과 같은 부가 서비스를 구현하지 못했다.
- Retrieval 시 BM25 외 DPR 등 다른 시도를 못해 본것이 아쉬움.
- 다양한 한글 데이터가 부족한 현실이 아쉬웠다.

- **프로젝트를 통해 배운 점 또는 시사점**

- 사용자의 발화를 완전히 이해하고 적절한 답변을 생성하는 챗봇 구현이 상당히 어렵다.

개인회고록(민원식).

개인회고록(김남현).

개인회고록(전태양).

개인회고록(최지민).

개인회고록(주정호).

개인회고록(정기원).

# 개인회고록(민원식)

○ 이번 프로젝트에서 나의 목표는 무엇이었는가?

- Product Serving 강의에서 학습한 내용을 되도록 많이 적용해 보고자 하였음.

○ 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

- 우선 모든 Product Serving 강의를 수강 하였고 관련 스킬을 익히기 위해 실습까지 진행함..
- AI stage에서 제공한 서버를 모델 서빙 서버로 만들기 위해 Bontomi로 패키징하는 작업을 함.
- 패키징 작업 시 Poetry를 통한 패키지 관리로 환경설정 이슈를 최소화 하고자 함.
- 모델 성능 평가를 위해 수집된 log에서 챗봇 대화를 추출하고 이를 엑셀파일로 불러오는 작업을 함.
- 모델 서빙 서버와 프론트엔드(Streamlit)를 분리하여 챗봇 성능 평가자들이 모델을 불러오지 않고도 로컬에서 프론트엔드에 접속만으로 작업할 수 있도록 함.

○ 나는 어떤 방식으로 모델을 개선했는가?

- 모델이 답변을 생성한 후 후보 답변을 retrieval 시 후보 답변 별로 점수하는 과정에 대해 아이디어 제공함.

○ 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- Bontomi이라는 툴을 통해 모델서빙과 Docker를 통한 배포작업에 대한 이해도가 향상되었음.
- Poetry를 사용해 Elasticserch 및 기타 사용된 패키지를 관리하고자 하였으나 패키지 버전별 의존성 문제로 관리에 어려움이 있었음. 이를 통해 각 패키지간 의존성 문제에 대해 고민해 볼 수 있었음.

○ 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- Bontomi, FastAPI, Poetry 등 처음 학습한 툴을 사용해 프로젝트를 진행해 보고자 함.
- 강의에서 다른 각종 툴들의 기초적인 내용을 벗어나 스스로 찾고 해결할 수 있는 응용법을 터득함.

○ 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- 시간의 한계로 GCP와 Github Action을 이용한 서비스 환경구축을 시도해 보지 못함.
- 프론트 엔드 구현 시 Streamlit을 제외한 다른 툴을 사용해보지 못함.

○ 한계/교훈을 바탕으로 다음 프로젝트에서 스스로 새롭게 시도해볼 것은 무엇일까?

- MLflow, Airflow 등 사용해 보지 못한 툴들을 이용해 서비스 환경을 보다 효율적으로 구축.
- 챗봇에 페르소나를 적용하여 보다 사람과 같은 대화를 할 수 있도록 할것.

# 개인회고록(김남현)

- 이번 프로젝트에서 나의 목표는 무엇이었는가?
  - product serving 단계까지 마무리하는 것.
  - 다양한 서비스 제공을 위한 풍부한 데이터 수집
- 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?
  - AI-hub, 깃허브 등 다양한 곳에서 챗봇 기능을 위한 한글 데이터를 탐색했다.
  - product serving 관련 제공된 강의를 수강했다.
- 나는 어떤 방식으로 모델을 개선했는가?
  - 혐오 표현 필터링 기능을 위한 Korean UnSmile Dataset의 pretrained model 추가
- 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?
  - 혐오표현 필터 기능 모델을 통해 챗봇 사용자가 입력하는 부적절한 표현을 어느 정도 거를 수 있었다. 처음에는 부적절한 표현 식별 기준점을 0.7로 설정했으나 이후 논의를 거쳐 0.1로 설정했다. 명백한 혐오표현이 아니라면 최대한 규제를 피하자는 게 논의 끝에 내린 결론이었다.
- 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?
  - 어렵게 생각하지 말고 쉬운 방법을 택했다. 바쁜 일정을 감안해서 빠른 판단을 내렸다.  
명언/어록 데이터는 한글 데이터가 없는 수준이었기에 하나씩 직접 수집해서 최소 1000개는 수집하고자 했으나 우선순위 과제가 아닌 데다가 시간이 많이 필요했기에 수집 작업을 중간에 그만두고 다른 일을 시작했다. 혐오표현 필터 모델 같은 경우는 그냥 pretrained 모델을 가져와서 썼다.
- 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?
  - 양질의 한글 데이터가 부족한 현실을 다시 깨달았다. 영어 데이터는 다양한 소재로 정말 쉽게 찾을 수 있는 반면에, 한글 데이터는 기관에서 수집해서 가공한 정형 데이터를 제외하면 구할 수 있는 게 거의 없었다.
  - 원래 내담자의 상황을 고려한 글귀나 음악 따위를 함께 추천하는 기능을 생각했으나 구현하지 못했다.
  - 혐오표현 필터링 기능을 추가할 때, 처음에는 'Korean UnSmile Dataset'으로 모델을 직접 학습시켜서 가져오려 했다. 하지만 기능을 빨리 추가해야 한다는 생각에 마음이 급했고, 제공되는 pretrained 모델의 성능도 괜찮다고 판단해서 직접 모델을 학습시키지 않았다.  
  
부적절한 표현을 대체로 잘 거를 수 있었지만, 간혹 지나치게 검열되거나 또는 부적절한 표현을 식별하지 못하는 경우도 있었다. 이는 다양한 데이터로 혐오표현 필터 모델을 더 학습시켜야 한다는 필요성을 제기하는데, 개선해야 하는 부분이다.
  - 바쁜 일정 속에서 제공된 강의와 실습을 제대로 챙기지 못했다. 최종 프로젝트가 끝나고 다시 강의를 듣고 실습을 진행해야한다.
- 한계/교훈을 바탕으로 다음 프로젝트에서 스스로 새롭게 시도해볼 것은 무엇일까?
  - 혐오표현 필터 모델 직접 학습시키기.
  - product serving 직접 해보기.
  - 더 많은 기능(명언/어록, 음악 추천)을 추가하고, 관련 데이터 수집하기.

# 개인회고록(전태양)

○ 이번 프로젝트에서 나의 목표는 무엇이었는가?

- EDA를 활용하여 데이터를 탐색, 모델의 성능을 올리기 위해 데이터를 최대한 수집하고자 했다.
- 발화 및 답변에서 감정이나 키워드를 추출하여 노래 추천과 같이 챗봇의 부가 서비스를 추가하고자 했다.

○ 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

- 구글의 open - domain 챗봇과 Facebook의 챗봇 레시피 논문을 읽었다.
- eda 대시보드를 제작하기 위해 Product serving 강의를 수강했다.
- Clustering과 관련하여 자료들을 찾아봤다.
- 감정이나 키워드를 기반으로 노래를 추천하기 위해 노래 데이터를 Selenium을 활용하여 크롤링 했다.

○ 나는 어떤 방식으로 모델을 개선했는가?

- Pandas를 활용하여 데이터를 전처리하고 Seaborn, Matplotlib을 활용하여 시각화를 진행했다. 이후 Streamlit을 활용하여 대시보드를 만들었고, 이를 팀원들에게 공유했다.
- 더 자연스러운 대화가 이뤄지도록 만들기 위해 대화 데이터를 더 수집했고, 수집한 데이터 셋을 싱글턴과 멀티턴을 위해 전처리하고 병합했다.

○ 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- 심리 상담 데이터 셋은 배경과 감정을 기반이 담긴 대화의 빈도가 압도적으로 많았고, 일반 대화와 모호한 대화들의 답변이 많지 않았다. 이는 발화자가 조금이라도 모호한 질문을 챗봇에게 던진다면 챗봇이 적절하게 대응하기 어려울 거라 생각했다.

○ 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- 데이터를 더 수집했고, 심리 상담 데이터에 취합하여 모델 학습 시 일반 대화에도 답변이 가능하게 했다.
- 심리 상담 데이터 셋의 질문 카테고리는 359개로 감정 분류를 할 때 너무 많다고 생각했다. 그리하여 K-means와 DBScan을 사용하여 카테고리를 줄이려는 시도를 했으나, 질문 문장의 길이가 짧고, 비슷한 단어들이 많이 섞여있어 클러스터링 결과가 좋지 않았다.

○ 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- 모델 Evaluation을 진행하느라 마지막 주를 전부 소비했고, 수집한 노래 데이터를 활용하여 노래 추천 서비스를 구현하지 못했다. 이는 시간 분배에 대한 개인적인 아쉬움으로 남았다.

○ 한계/교훈을 바탕으로 다음 프로젝트에서 스스로 새롭게 시도해볼 것은 무엇일까?

- 이번 프로젝트를 통해 유사도 기반 추천 시스템에 관심이 생겼고, 관련된 Task에 대한 학습을 진행한 뒤에 감정 및 키워드에 따른 노래 추천 시스템을 개발해 보려 한다.

# 개인회고록(최지민)

- 이번 프로젝트에서 나의 목표는 무엇이었는가?
  - 심리상담 목적에 맞는 대화가 가능한 챗봇을 만드는 것
  - 챗봇의 답변이 더욱 완성도 있게 만드는 것
- 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?
  - 이어지는 대화를 위해 DialogBERT 탐색 및 적용
  - Sensibleness and Specificity(SSA)로 챗봇 성능 평가
- 나는 어떤 방식으로 모델을 개선했는가?
  - 생성 모델이 상황에 맞는 답변을 생성하지만, 문장의 완성도가 떨어진다는 것을 확인  
→ 답변 데이터셋을 구축하여 생성된 답변과 가장 유사한 것을 찾아서 반환하기를 제안
- 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?
  - Generation+ElasticSearch로 적절하면서 완성도 있는 답변을 내놓을 수 있게 됨
  - 생성 모델의 한계점을 파악하고, 보완하는 방법이 필요함을 깨달음
- 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?
  - 일정을 고려하여 기존 라이브러리와 Github 자료, 사전 학습 모델을 적극 활용하려고 노력함
- 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?
  - DialogBERT가 학습이 부족하였는지, 그 이전 대화 맥락을 파악하지 못하는 모습을 보여 최종 모델에 사용하지 못함
  - 더 나은 웹 구성을 위해 기존 채팅처럼 최신 대화가 가장 아래에 나오는 화면 구성을 시도했으나, streamlit 기능의 한계로 적용하지 못함
- 한계/교훈을 바탕으로 다음 프로젝트에서 스스로 새롭게 시도해볼 것은 무엇일까?
  - 더 나은 Frontend를 위한 공부
  - 모델 경량화를 위한 공부

# 개인회고록(주정호)

- 이번 프로젝트에서 나의 목표는 무엇이었는가?
  - 멀티턴 대화 / 답변 생성 / 일상 대화가 가능한 챗봇 구현
  - 사용자가 쉽게 서비스를 이용할 수 있도록 UI 구현
  - 모든 사용자가 어디서든 사용할 수 있도록 서버 배포
- 나는 내 학습 목표를 달성하기 위해 무엇을 어떻게 했는가?
  - KoGPT2을 Wellness Dataset과 Chit-Chat Dataset으로 Fine-Tuning
  - Streamlit을 이용해 UI 구현
  - Streamlit Cloud 및 Ainize를 이용해 서버 배포
- 나는 어떤 방식으로 모델을 개선했는가?
  - KoGPT2 모델이 심리 상담 주제를 예측하고, 이를 이용해 답변을 생성하게 함 (Auto-Regressive)
  - 심리 상담 주제를 긍정 답변 및 부정 답변으로 예측시, 이전 발화를 추가해 다시 답변을 생성 (Multi-Turn)
  - 위 과정을 5번 하여, 최다 선택된 심리 상담 주제의 답변을 답변 후보로 결정 (Beam Search)
  - Elastic Search를 이용해 각각의 답변 후보와 유사한 답변들을 검색해 답변 후보 추가 (Retrieve)
  - 답변 후보 중 최다 빈도의 답변을 최종 답변으로 결정 (Count)
- 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?
  - 시중에 나와있는 다른 모델과 비교했을 때, 보다 나은 성능의 심리 상담 챗봇 개발
  - 사용자의 발화를 완전히 이해하고, 적절한 답변을 생성하는 챗봇을 개발하는 것이 상당히 어려운 것을 깨달았다.
- 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?
  - KoGPT2를 Multi Downstream-Task가 가능하도록 설계하였다. 이를 통해 답변을 좀 더 우수하게 생성해주는 효과가 있었다.
- 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?
  - 3턴 이상의 사용자의 발화는 고려하지 못하는 한계점이 있었다.
- 한계/교훈을 바탕으로 다음 프로젝트에서 스스로 새롭게 시도해볼 것은 무엇일까?
  - DialogBERT 모델을 이용하여 챗봇을 구현하는 것을 새롭게 시도해볼 것이다.

# 개인회고록(정기원)

- 이번 프로젝트에서 나의 목표는 무엇이었는가?
  - Elasticsearch 활용해보기
  - Text Style Transfer 시도해보기
  - 발화 및 답변에서 키워드 추출해서 사용자 맞춤 콘텐츠 추천해주기
  - Product Serving 강의내용 시도해보기
- 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?
  - Elasticsearch 공식 문서 탐독
  - Text Style Transfer 논문 탐독
  - Sentence BERT 등 키워드 추출 관련 논문 읽기
  - FastAPI 등 API 관련 실습
- 나는 어떤 방식으로 모델을 개선했는가?
  - Elasticsearch 불용어 관련 실험
  - Text Style Transfer 한국어 데이터로 직접 실험
  - 전체 챗봇 답변 데이터 크롤링
- 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?
  - 불용어 필터를 사용하지 않도록 해서 챗봇의 성능이 개선되었음
- 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?
  - Elasticsearch 사용법에 대해 자세히 알게 되었다
  - 텍스트 스타일 변화를 시도하며 BERT 구조에 대해 더 자세히 알게 되었다
- 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?
  - 스타일 변화 결과가 썩 좋지 않았다
  - FastAPI 시도 해보았으나 어려웠다
- 한계/교훈을 바탕으로 다음 프로젝트에서 스스로 새롭게 시도해볼 것은 무엇일까?
  - 텍스트 스타일 변화 최신 논문 한국어 데이터로 시도