

< 캡스톤디자인 최종보고서 >

Its'me : 스마트워치에서의 사용자 말투 기반 답장 제안

팀원(참여학생): 박건우(산업경영공학), 김유환(소프트웨어융합학),
정성호(소프트웨어융합학)

요약

스마트 워치는 휴대폰을 지니고 있지 않더라도 전화, 메세지 등의 연락에 즉시 확인할 수 있다는 장점을 가지고 있다. 하지만 타이핑을 하기에는 인터페이스가 매우 작기 때문에 답장을 하기는 어렵다는 단점이 필수불가결하다. 이를 해결하기 위한 방안으로 주로 답장 제안 기능을 활용하고 있으나, 현재 제안되는 답장 후보들의 경우 무뚝뚝한 경우가 많아 무심코 보낼 경우 상대방과의 관계 악화를 초래할 수 있다. 본 연구에서는 사용자의 상대방마다의 카톡 대화 내역을 통해 사용자 말투를 학습하여 사용자 말투로 변환된 답장 후보를 제안하는 시스템을 개발했다.

1.1. 연구배경

답장 제안(reply suggestion)이란 스마트폰, 스마트워치, 웹메일등의 메시지 관련 서비스에서 이메일, 메세지, 전화 등의 연락이 도착했을 때 사용자가 쉽고, 빠르게 상대방에게 답장할 수 있도록 답장 후보를 제안하는 기능이다. 특히 스마트워치의 경우 작은 인터페이스로 인해 사용자가 텍스트를 직접 입력하지 못하기 때문에 답장 제안 기능의 활용도 및 의존도가 매우 높다.

하지만 현재 각종 스마트워치에서 제안되고 있는 답장 후보들은 발신인에 관계없이 ["응", "다시 연락할게", "나중에 전화드려도 될까요?"]등의 다소 무뚝뚝한 말투로 작성되어 있다. 이러한 상황속에서 사용자는 상대방과의 관계 악화를 우려하여 답장을 아예 하지 않고 미뤄 상대방을 기다리게 만들거나, 무뚝뚝한 답장을 전송함으로써 성의없다는 이미지를 가지게 될

수 있다.

1.2. 연구 목표

본 연구는 Seq2Seq 알고리즘을 활용하여 사용자의 말투를 학습해, 사용자가 카카오톡 메시지를 수신했을 때 스마트워치 화면에 알림을 전함과 동시에 ["곧 전화드리겠습니다", "나중에 전화하겠습니다", "문자 메시지를 보내주세요", "알바중입니다. 나중에 전화하겠습니다", "회의중입니다. 나중에 전화하겠습니다", "운전중입니다. 나중에 전화하겠습니다", "밥먹고 있습니다. 나중에 전화하겠습니다.", "무슨일 있으신가요?"]의 연락을 길게 이어가지 못하는 상황에서 가능한 10개의 답장 리스트를 사용자의 말투로 변환하여 제안하고, 선택된 답장을 카카오톡으로 전송하는 것을 목표로한다.

1.3. 기대 효과

본 연구는 스마트워치에서 사용자의 말투로 변환된 답장을 제안함으로써 자신의 답장을 직접 입력하지 못해 생기는 답답함을 감소시키고, 상대방에 따라 맞춰진 말투로 관계 악화 우려 없이 스마트워치로 사용자의 상황을 즉각적으로 표현할 수 있도록 한다. 이를 통해 작은 인터페이스로 인해 필수불가결하게 발생했던 스마트워치의 고질적 한계였던 답장 입력 제한을 완화시킬 수 있다.

2. 배경 지식 및 관련 연구

2.1.1. Seq2Seq 모델

Seq2Seq 모델은 한 시퀀스를 다른 시퀀스로 변환하는 작업을 수행하는 딥러닝 모델로, 주로

자연어 처리(NLP)분야에서 활용되는 모델이다. 인코더 모듈과 디코더 모듈로 이루어져있으며, 인코더는 일반적으로 RNN, LSTM등의 순환 신경망 구조를 사용하여 입력 시퀀스를 고정 길이의 벡터로 변환하는 역할을 수행하고, 디코더는 반대로 고정 길이로 변환된 벡터를 기반으로 출력 시퀀스를 반환한다. 주로 입력된 언어를 다른 국가의 언어로 변환하는 번역기에 활용되며, 그외에도 내용 요약, STT(Speech To Text)에 활용되고 있다.

2.1.2. 답장 제안시 고려 사항

Yue Weng et al. [5]은 Uber 앱 내에서 차량 운전자가 탑승예정객의 메시지에 한번의 클릭으로 빠르게 답장할 수 있도록, 탑승예정객의 메시지에서부터 의도를 추출하고, 해당 메시지에 대해 운전자가 선택할 가능성이 높은 상위 4개의 답장을 제안하는 원클릭 채팅 시스템을 구현했다. 한편, Ronald E Roberson et al. [3]은 이메일 답장 제안 시스템에서 사회적 관계(상사, 동료)에 있는 상대방에게는 답장의 공손함이 중요하지만, 형제 자매 등의 가까운 지인에게는 속어를 쓰는 등 상대방과의 관계에 따라 제안하는 답장의 형식 및 말투가 달라져야 함을 밝혔고, 자동으로 제안되는 답장이 무감정하기 때문에 불쾌감 혹은 오해가 생길 수 있다는 내용 또한 밝혀졌다[1, 3]. 본 연구에서는 사용자가 스마트워치를 통해 답장해야하는 상황을 회의중, 수업중, 운전중 등의 “길게 연락을 이어가지 못하는 상황”으로 가정하고, 각 상황에 맞춰 미리 정의한 답장들을 사용자의 말투로 변환하여 제안한다.

2.1.3. 스마트워치에서의 텍스트 입력

스마트 워치는 손목에 착용하는 대표적인 웨어러블 기기로, 메시지, 전화, 캘린더, 건강 지표 등을 손쉽게 확인할 수 있다. 하지만 손목에 착용하는 기기인 만큼 인터페이스의 크기가 작기 때문에 화면 안의 키보드를 통해 텍스트를 입력하는 것은 매우 제한적이라는 단점이 존재한다. 이에 대해 Stephen Oney et al. [4]는 스마트워치에서의 텍스트 입력을 위해, 두 번 탭하여 키보드 영역을 확대한 후, 확대된 영역안의 문자를 선택하는 방식의 키보드 레이아웃을 개발했고, Komninos et al. [2]는 스마트 워치의 화면에 7개 구역을 지정하고 각 구역마다 알파벳을 분배하여, 입력하려는 단어의 철자가 들어있는 구역을 순서대로 클릭했을 때 가장 확률이 높은

단어로 변환해주는 키보드 입력 방식을 개발했다. 본 연구에서는 사용자가 자신의 답장을 직접 입력하지 못해 생기는 답답함을 감소시키기 위해, 상대방의 메시지에 대한 자동 답장을 사용자의 말투로 변환하여 제안한다.

3. 추진 내용

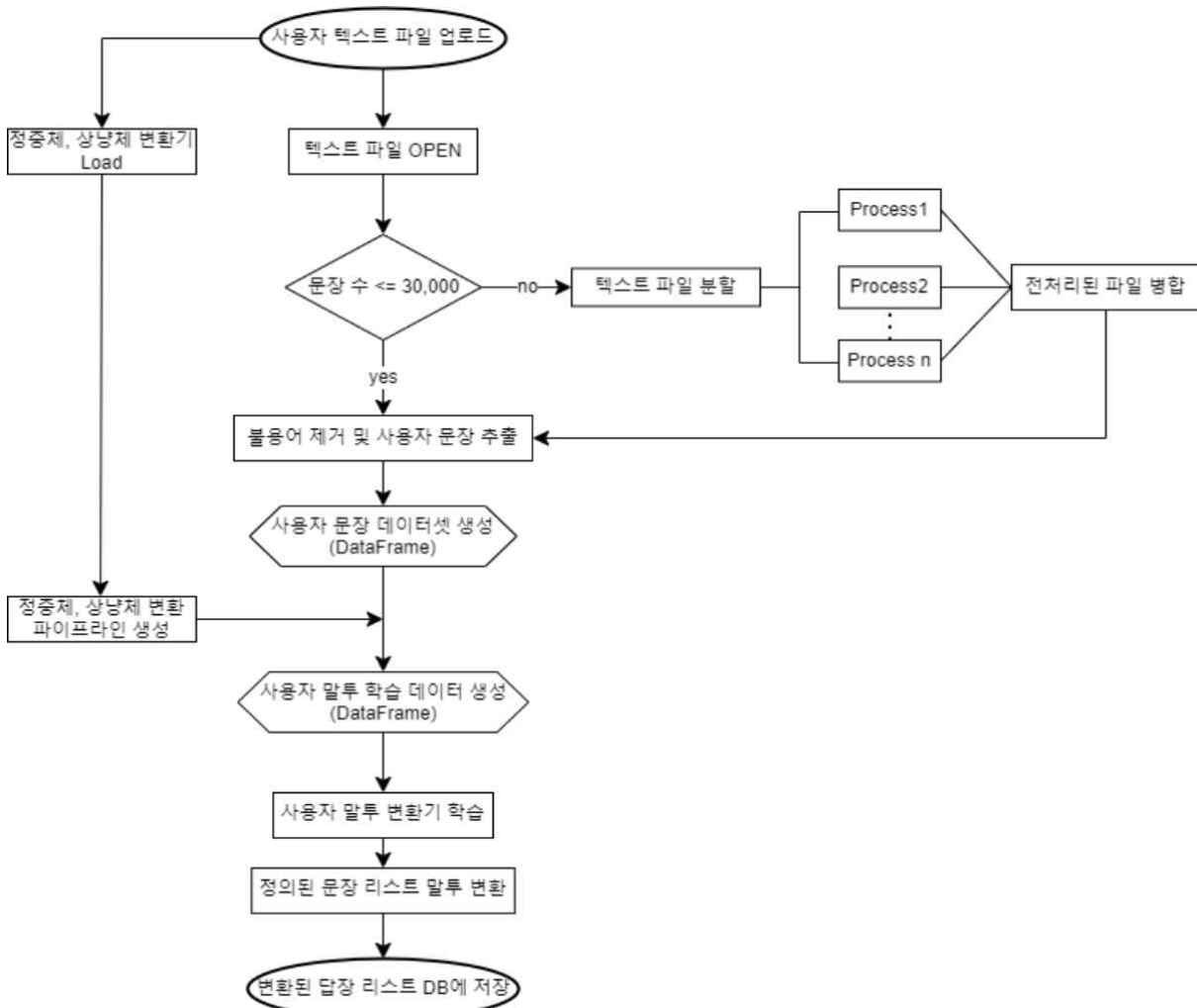
3.1. 팀 구성 및 역할

이름	역할	수행업무
박건우	팀장	<ul style="list-style-type: none"> - 카카오톡 AI 역공학 설계를 통한 대화체에 견고한 상냥체, 정중체 변환 모델 개발 - 사용자 말투 변환 모델 개발 - Django를 활용하여 사용자 INPUT-> 모델 학습 -> DB 저장까지의 파이프라인 설계 - Django를 활용하여 request, response API 개발 - 웹페이지, 모바일 앱의 카카오톡 로그인 기능 개발
정성호	팀원	<ul style="list-style-type: none"> - 모바일 앱에서 사용자 알림을 캐치하는 기능 개발 - 사용자가 스마트워치에서 답장을 선택했을 때 실제로 카카오톡으로 전송하는 기능 개발
김유환	팀원	<ul style="list-style-type: none"> - DB로부터 추출한 답장 리스트를 모바일 앱에서 스마트워치로 전송하고 스마트워치에서 버튼 형식으로 띄우는 기능 개발

3.2. 전체 시스템 구성

3.2.1 웹페이지 <-> Postgresql DATABASE(Amazon RDS)

<플로우 차트>



* 상냥체, 정중체, 사용자 말투 학습 데이터셋 예시

매크로 생성 데이터 셋

사용자(user)	상냥체(formal)	정중체(gentle)
밥 먹는중	밥 먹는 중이에요.	현재 식사를 하고 있습니다.
보고싶어	보고 싶어요.	보고 싶습니다.

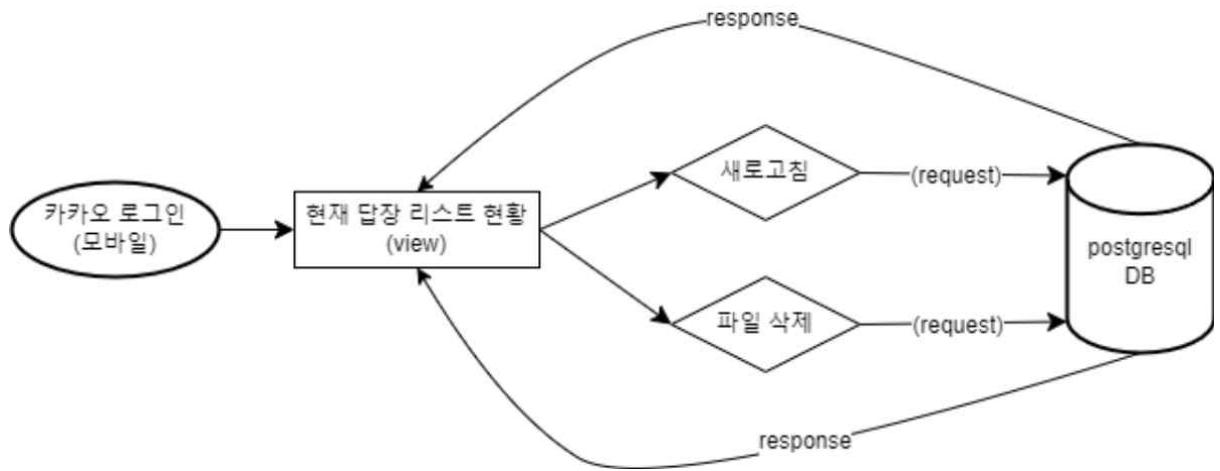
상냥체 학습 데이터셋

상냥체 (formal)	무작위 (random)
밥 먹는 중이에요.	밥 먹는중
보고 싶어요.	보고싶어
밥 먹는 중이에요.	현재 식사를 하고 있습니다.
보고 싶어요.	보고 싶습니다.

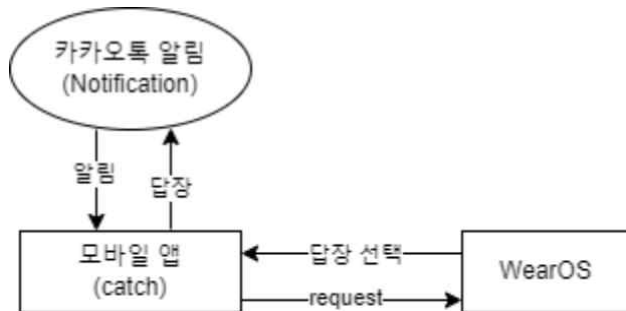
정중체 학습 데이터셋

정중체 (gentle)	무작위 (random)
현재 식사를 하고 있습니다.	밥 먹는중
보고 싶습니다.	보고싶어
현재 식사를 하고 있습니다.	밥 먹는 중이에요.
보고 싶습니다.	보고 싶어요.

3.2.2 Postgresql DATABASE (Amazon RDS) <-> 모바일(스마트폰) 통신

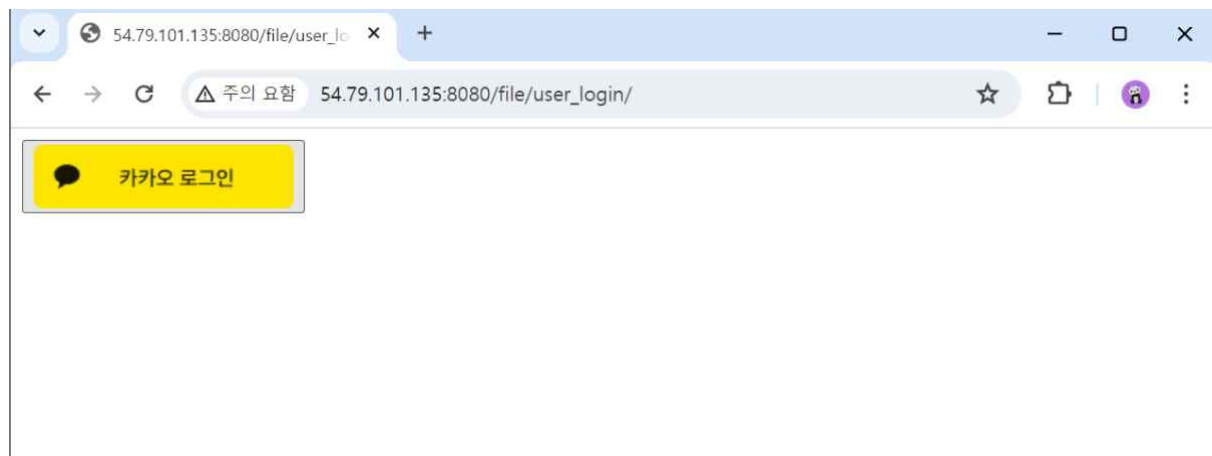


3.2.3 모바일 <-> WearOS 통신

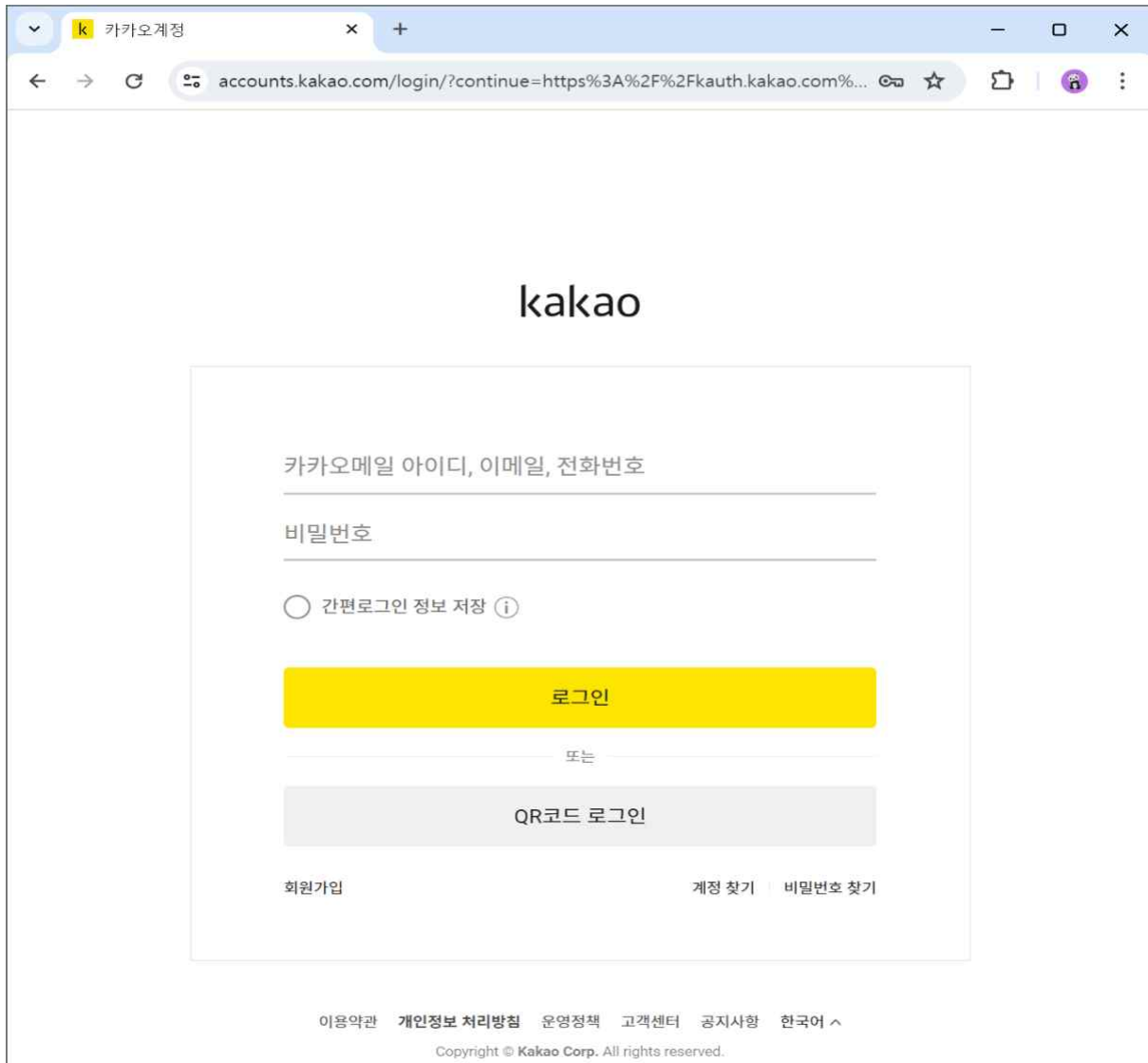


3.3. 사용자 시나리오

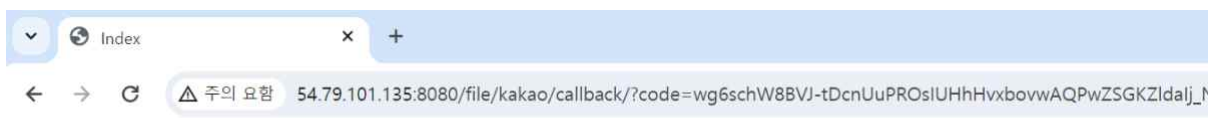
1. 사용자가 웹페이지에 접속한다.



2. 카카오톡 로그인 수행



3. 파일 업로드 웹페이지 이동

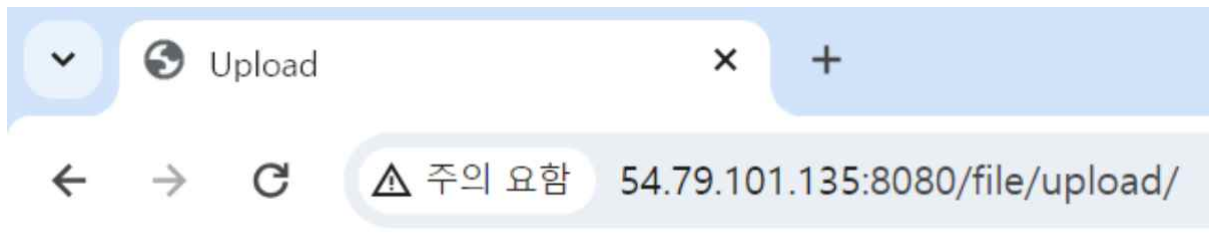


파일 업로드

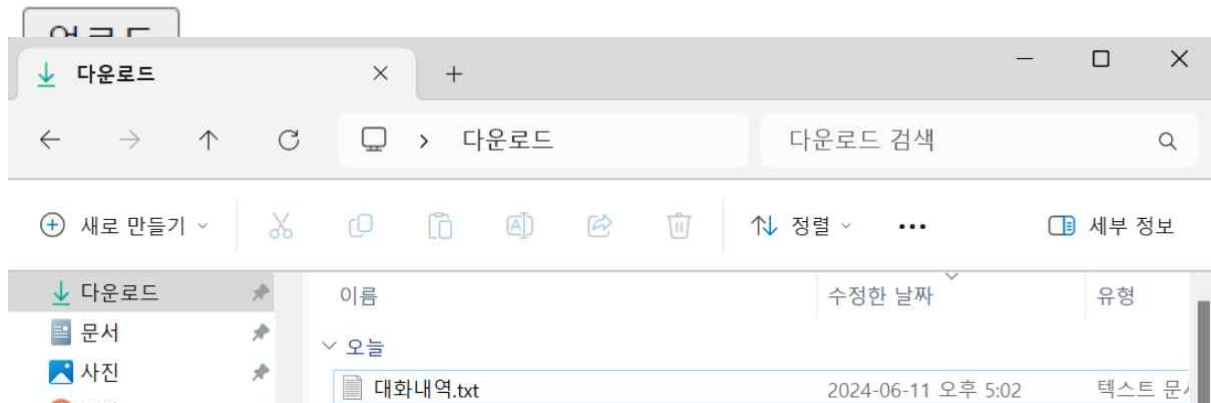
- [카카오톡 채팅방 파일 올리기](#)
- [업로드한 파일 목록](#)



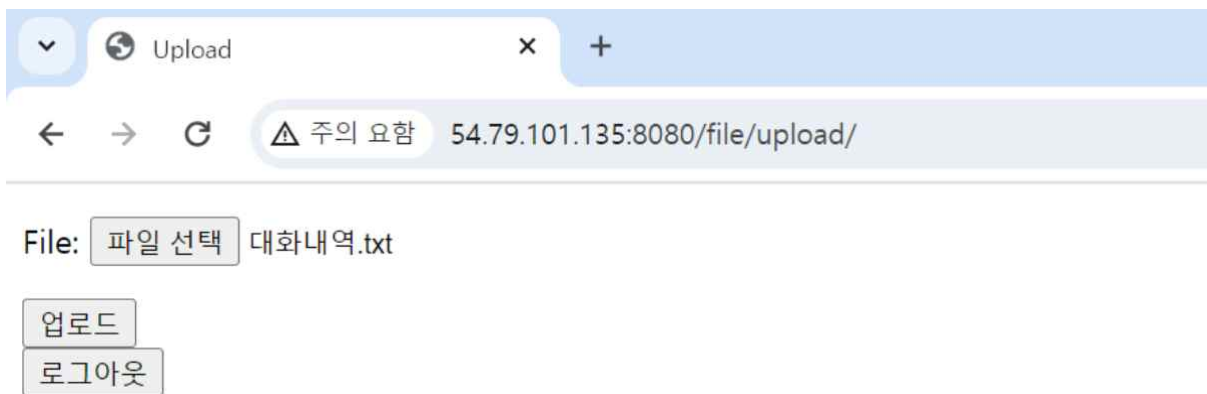
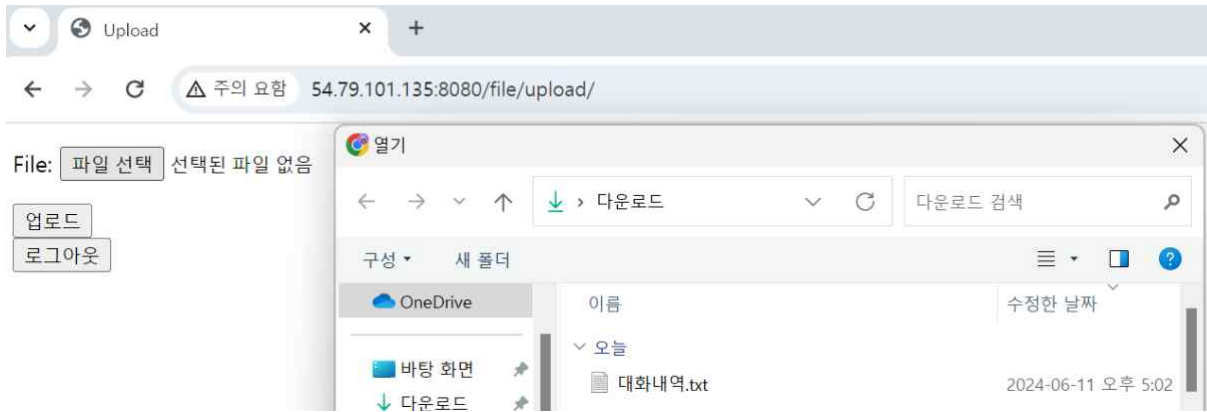
4. 카카오톡 대화 내역 내보내기



File: 선택된 파일 없음

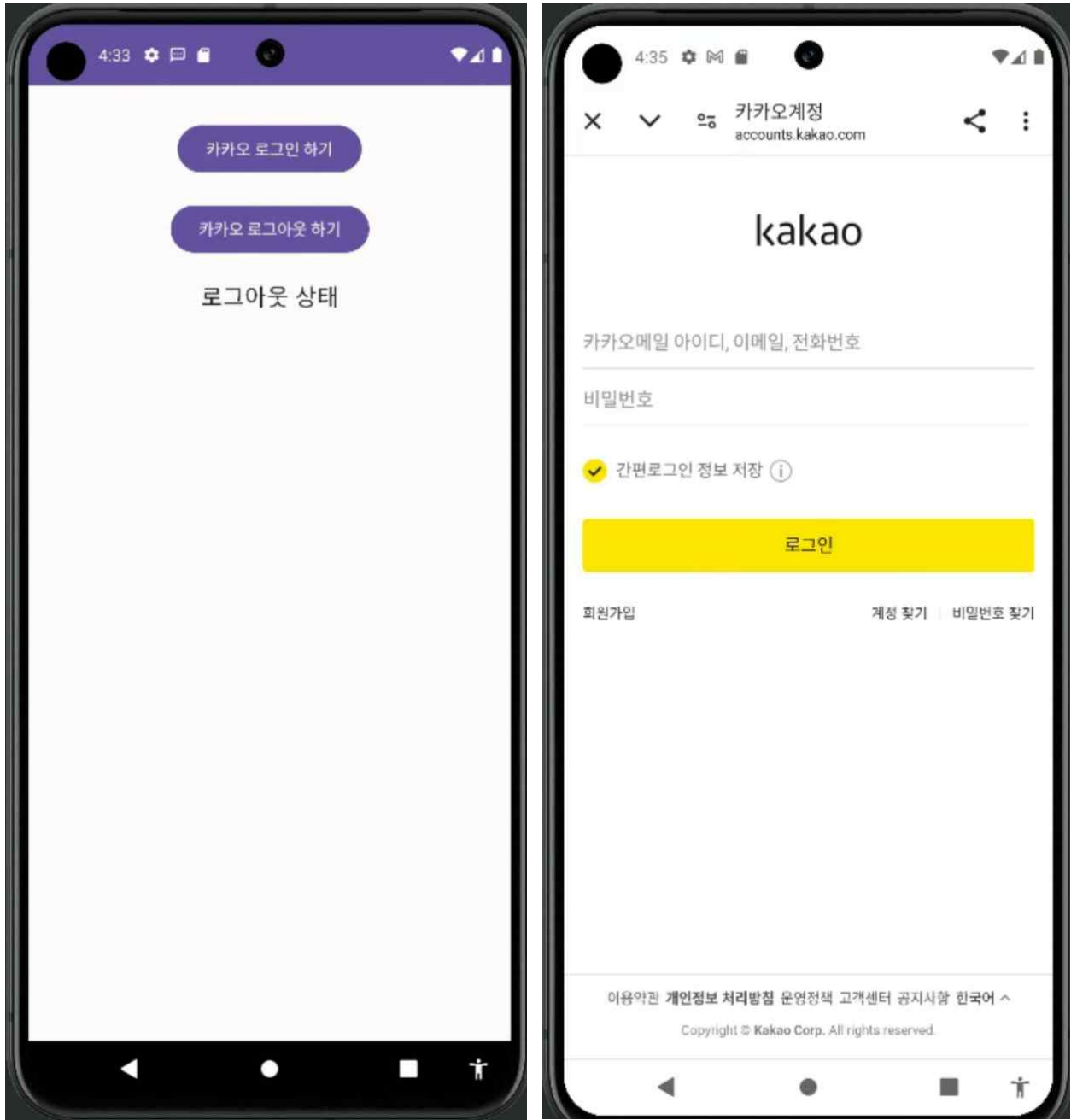


5. 카카오톡 대화내역 txt파일 업로드



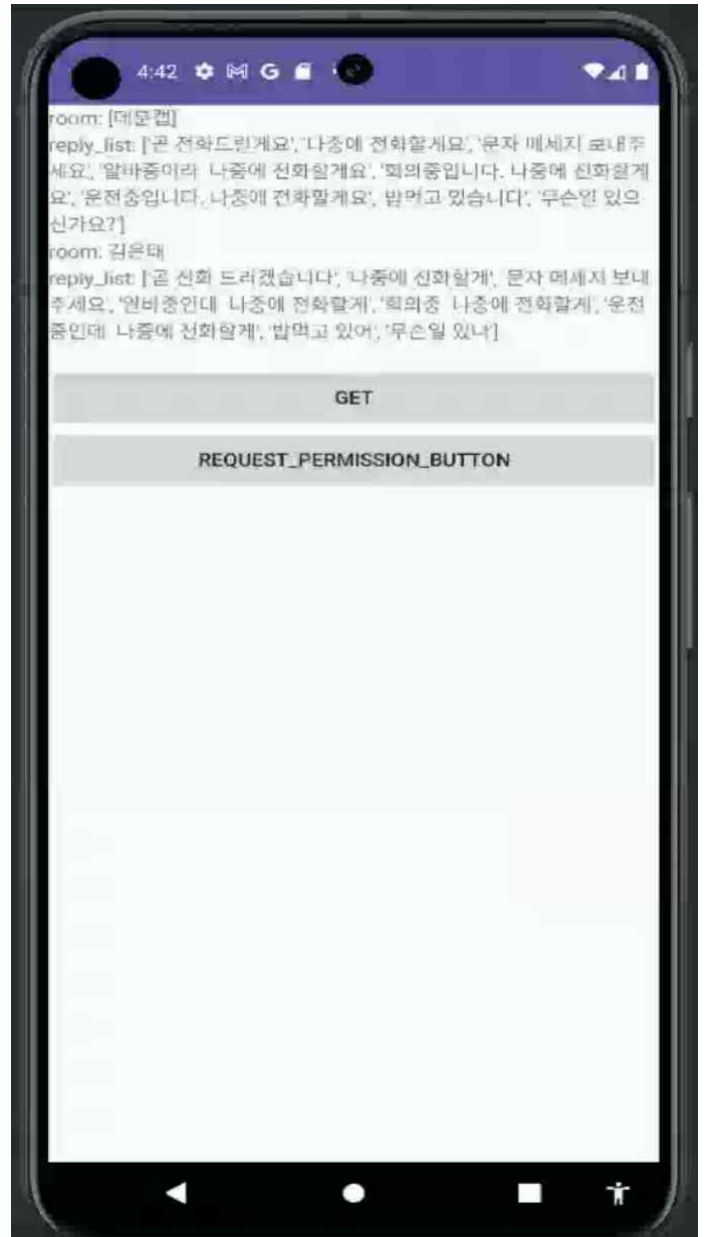
- 업로드 버튼 클릭시 백엔드에서 사용자 말투 학습 시작 (학습과정은 후술함)
- 학습 완료시, RDS (PostgreSQL DB)에 저장

6. 모바일 앱에서 카카오톡 로그인



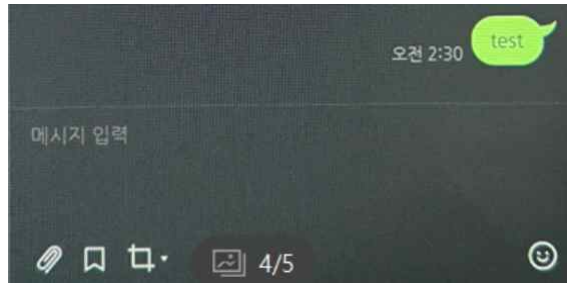
7. 장고에 변환된 답장 리스트 request 전송 (GET 버튼을 누르면 동작함)

->장고에서 rds에서 사용자 id를 기준으로 답장 리스트를 reponse로 반환



* 모바일에 가져옴과 동시에 WEAR OS에도 답장 리스트가 연동되어서 저장됨

8. Wear OS에서 카카오톡 도착 시 알림 + 답장 리스트 표시



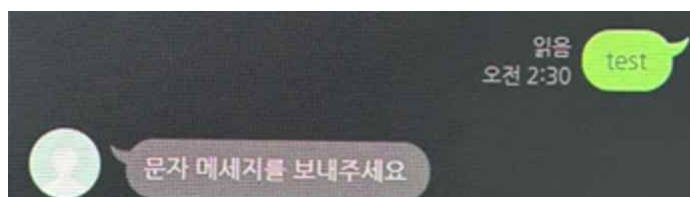
<메시지 전송>



<메시지 도착>



<답장 리스트 표시>



<답장 버튼 클릭시 선택한 메시지 전송>

8.1. 핵심 기능 개발

① 카카오톡 말투 변환기 역공학 설계

Seq2Seq 알고리즘을 활용하여 사용자 말투를 학습하기 위해서는, 사용자가 보낸 문장을 정중체, 상냥체 등 다른 문체로 변환한 문장을 생성하여 사용자 말투를 decoder로 설정하고 상냥체, 정중체로 변환된 문장을 encoder로 맵핑하여 상냥체 혹은 정중체의 문장이 입력되었을 때 사용자 말투로 변환되도록 학습해야한다.

이를 위해 AI Hub의 오픈 데이터를 활용하여 정중체, 상냥체 모델을 개발했으나, 데이터 수 부족으로 사용자의 말투를 상냥체, 정중체로 적절히 변환하기에는 부족한 성능을 보였다.

반면 카카오톡에서 제공하고 있는 말투 변환 서비스의 경우 매우 높은 성능을 보였으므로, Encoder - Decoder 특성을 활용하여 역공학 설계를 통해 카카오톡 말투 변환기와 비슷한 성능의 상냥체, 정중체 변환기를 개발했다.

개발 방법은 다음과 같다.

1. 임의의 대화 문장을 카카오톡 대화창에 입력한다.

밥 먹고 있었엉

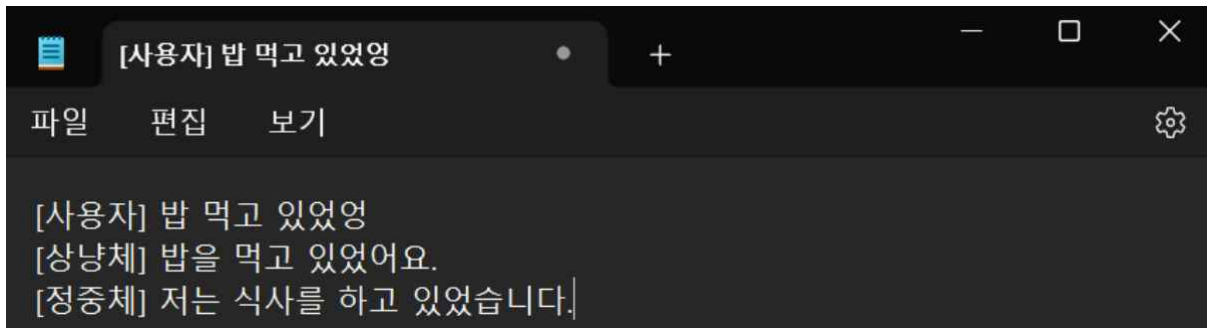


2. AI 버튼을 클릭해 상냥체, 정중체로 변환한 문장을 생성한다.





3. 원본문장, 상냥체로 변환된 문장, 정중체로 변환된 문장을 메모장에 옮겨 적는다.



4. 위의 과정을 자동으로 반복하는 매크로를 개발하여 약 60,000문장에 대해 상냥체, 정중체 학습 데이터를 생성한다. (pyautogui, pywinauto 활용)

formal	random	gentle	random
안녕하세요.	안녕하십니까.	안녕하십니까.	안녕하세요.
반가워요.	반갑습니다.	반갑습니다.	반가워요.
좋은 아침이에요.	좋은 아침입니다.	좋은 아침입니다.	좋은 아침이에요.
간밤에 별일 없으셨나요?	간밤에 별일 없으셨습니까?	간밤에 별일 없으셨습니까?	간밤에 별일 없으셨나요?
안녕하세요?	안녕하십니까?	안녕하십니까?	안녕하세요?
안녕하세요!	안녕하십니까!	안녕하십니까!	안녕하세요!
안녕하세요?	안녕하십니까?	안녕하십니까?	안녕하세요?
오늘도 활기찬 하루 되세요!	오늘도 활기찬 하루 되십시오!	오늘도 활기찬 하루 되십시오!	오늘도 활기찬 하루 되세요!
상쾌한 아침이에요.	상쾌한 아침입니다.	상쾌한 아침입니다.	상쾌한 아침이에요.
잠은 잘 주무셨어요?	잠은 잘 주무셨습니까?	잠은 잘 주무셨습니까?	잠은 잘 주무셨어요?
밤사이 잠은 잘 주무셨어요?	밤사이 잠은 잘 주무셨습니까?	밤사이 잠은 잘 주무셨습니까?	밤사이 잠은 잘 주무셨어요?
출근 하셨네요!	출근 하셨습니까!	출근 하셨습니까!	출근 하셨네요!

* random = 원본 문장 + 상냥체 or 정중체 문장이 병합된 데이터

=> 말투의 종류에 robust하게 변환할 수 있도록 무작위성을 추가로 학습시켰다.

5. 개발 완료한 상냥체 변환기, 정중체 변환기는 Hugging_face 플랫폼에 업로드하여 학습시 호출하여 사용했다.



geonwoopark

9unu

Edit profile

Settings

Models 2

Sort: Recently updated

9unu/gentle_speech_translation

Text2Text Generation • Updated 17 days ago • ↓ 611

9unu/formal_speech_translation

Text2Text Generation • Updated 17 days ago • ↓ 630

② 사용자 말투 변환기 학습

①에서 학습한 상냥체, 정중체 변환기를 활용하여 사용자 말투 변환 데이터셋을 생성하고, 같은 AI 알고리즘을 활용하여 사용자 말투로 변환하는 변환기를 개발했다.

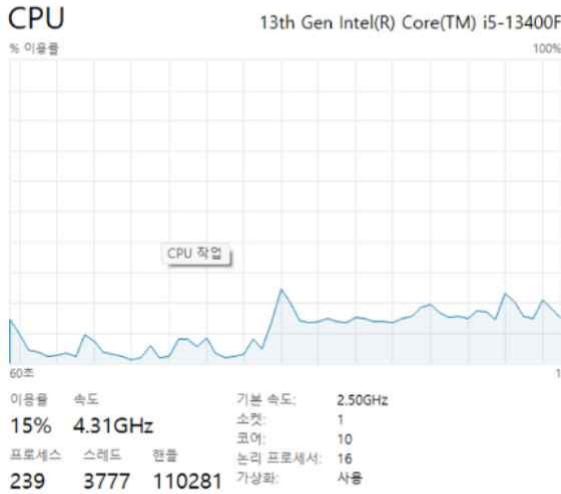
user	random
오 그러면 주말에 보자 나는 토요일 둘다 괜찮아	그렇다면, 주말에 뵙는 것이 어떨까요? 저는 토요일과 일요일 모두 괜찮습니다.
정해지면 알려줘 어디서 볼까??	정해지는 대로 알려주시면 감사하겠습니다. 혹시 어디에서 만나는 것이 좋을지에 대한 의견도 함께 부탁드립니다.
오 나도 좋아 이증집?	저도 좋습니다. 이증집입니까?
넌 저녁에 보자 겸실 집을 좀 알아보겠습 !	내일 저녁에 뵙는 것이 좋을 것 같습니다. 삼겹살 가게를 찾아보도록 하겠습니다.

③ 파이프라인 설계 [텍스트 파일 입력 -> 사용자 말투 학습 데이터 생성 -> 사용자 말투 변환기 학습 -> DB에 변환된 답장 리스트 저장]

장고를 활용하여 사용자가 입력한 카카오톡 대화 내역 txt파일을 업로드하면, 사용자가 보낸 문장을 분류해내고, 상냥체, 정중체 변환기를 활용하여 사용자 말투 변환기 학습 데이터를 생성한다.

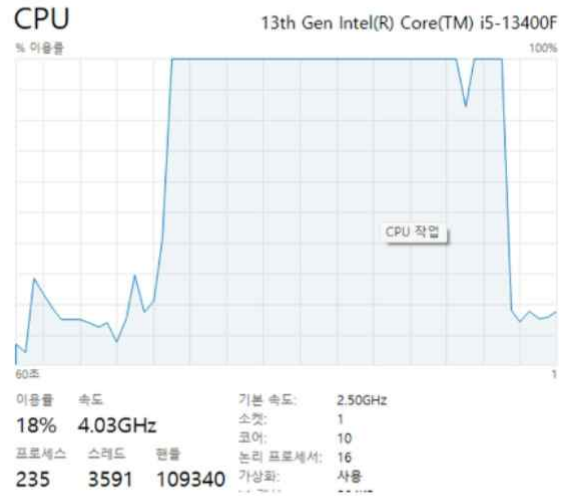
이때, 사용자의 대화내역이 너무 많을 경우 (특히 단체 카톡방) 데이터 전처리의 시간이 너무 오래걸리기 때문에, 파이썬 multiprocessing을 활용하여 텍스트 데이터 전처리 시간을 단축시켰다.

프로세스 분할 전



약 60,000개 문장 → 812.56초 소요

프로세스 분할 후



약 60,000개 문장 → 36.63초 소요

<학습 시간 Worst case 전체 프로세스 동작 Log>

```
<<<카톡 데이터 csv로 변환 중>>>
[Kss]: Oh! You have mecab in your environment. Kss will take this as a backend! :D
<<<변환 완료>>>16.731685400009155 초
총 데이터 수: 1987
학습용 데이터 수 (최대 1000 제한): 1000
<<<학습 데이터 생성 파이프라인 동작 중>>>
<<<학습 데이터 생성 완료>>> 15.0 분
<<<모델 학습 코드 동작 중>>>
<<<모델 학습 및 답장 변환 완료>>> 22.0 분
총 소요 시간: 37.0 분
```

- * 학습 데이터 수 (사용자 전송 문장)가 1000개일 때, 충분히 사용자 말투로 변환된다고 판단하여 학습 시간 효율화를 목적으로 최대 데이터 수를 1000으로 제한함
- * 학습 데이터 수가 1000일 때, 즉 worst case의 경우 총 소요 시간이 약 37분 소요된다.

<PostgreSQL DATABASE에 올라간 형태>

	id [PK] bigint	user_id character varying (300)	user_name character varying (300)	file character varying (100)	room character varying (50)
1	5	3495711455	박건우	박건우/KakaoTalk_20240601_0021_33_110_김은태...	김은태
2	6	3508161050	김유환	김유환/KakaoTalk_20240601_1950_52_438_group.txt	[대분첩]
3	8	3495711455	박건우	박건우/test.txt	[대분첩]

reply_list
character varying (300)

[공 전하 드리겠습니다; '나중에 전화할게; '문자 메시지 보내주세요; '알바중인데 나중에 전화할게; '회의중 나중에 전화할게; '운전중인데 나중에 전화할게; '밥먹고 있어; '무슨말 있나?]
[공 전화드리겠습니다; '나중에 전화 하겠습니다; '문자 메시지를 보내주세요; '알바중입니다. 나중에 전화드릴게요; '회의중입니다; '운전중입니다. 나중에 전화할게요; '밥먹고 있습니다; '무슨말 있으신가요?]
[공 전화드릴게요; '나중에 전화할게요; '문자 메시지 보내주세요; '알바중이라 나중에 전화할게요; '회의중인데 나중에 전화할게요; '운전중입니다. 나중에 전화할게요; '밥먹고 있습니다; '무슨말 있으신가요?]

* 카카오톡 ID에 맞춰 user_id를 선정했고, 같은 채팅방이어도 사용자에게 따라 다른 말투로 변환된 것을 확인할 수 있다.

④ 모바일 (스마트폰) 과 RDS DATABASE 연동

WearOS 스마트워치에서 RDS Database에 저장된 변환된 답장 리스트를 사용자에게 제공하기 위해 모바일 (스마트폰)을 경유하는 시스템을 구축했다.

<스마트폰 <-> RDS DATABASE 통신 순서>

1. 안드로이드 스튜디오 앱내에 카카오톡 로그인 기능 구현
2. 로그인 시 세션에 사용자의 카카오톡 ID를 변수로 저장
3. GET 버튼을 누르면, 사용자의 카카오톡 ID를 GET request header에 포함하여 Django 서버에 요청을 보낸다.
4. 요청을 받은 Django 서버는 전달받은 사용자 ID를 key 값으로 RDS DATABASE에서 사용자의 상대방별 카톡방 이름, 상대방 이름, 변환된 답장리스트를 필터링한다.
5. 필터링된 내용을 json 형식의 serializer객체로 변환하여 response로 반환한다.
6. 스마트폰 앱에 변환된 답장 리스트가 저장되고, 곧바로 WearOS로 답장리스트를 전송한다.

⑤ 모바일 (스마트폰) 과 WEAROS (스마트워치) 연동, 카카오톡 전송

모바일과 WEAROS는 안드로이드 스튜디오 (코틀린)에서 지원하는 Wearable Data Layer API를 활용하여 문자열 형식으로 통신할 수 있다. 해당 함수를 활용하여 스마트폰에 전달된 답장 리스트를 WEAROS로 전달했으며, 스마트워치에서 선택된 답장 또한 같은 함수를 활용하여 스마트폰으로 전달했다.

스마트워치에서 스마트폰으로 전달된 선택된 답장은 StatusBarNotification 클래스를 활용하여 백그라운드 상태에서 즉, 스마트폰 화면이 켜지지 않은 상태에서도 실제 카카오톡 답장을 전송하는 기능을 구현했다.

9. 결과 [사용 시나리오]

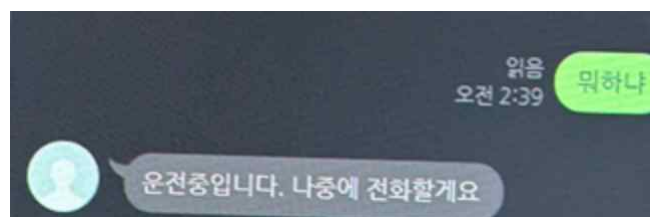
[시나리오 : 운전 중에 친한 친구한테 연락이 온 상황]



<메시지 도착>



<답장 선택>



<선택한 답장 전송>

10. 결론

본 연구는 기존의 스마트워치에서 제공되는 무뚝뚝한 말투의 답장을 사용자의 상대방 별 말투를

반영한 답장으로 변환하여 제시하고, 선택할 시 실제로 카카오톡 답장을 전송하는 기능을 구현했다.

이를 위해 대화체에 견고한 정중체, 상냥체 변환 모델을 카카오톡 말투 변환 AI의 결과를 역변환하는 일종의 역공학 설계를 통해 개발했다. AI Hub에 공유되어 있는 공공 대화 데이터와 실제로 사용자로부터 얻은 대화 데이터를 합쳐 약 60,000개의 상냥체, 정중체 문장 데이터로 학습을 진행했으며, 개발한 두개의 변환기는 허깅페이스에서 각각 약 600회의 다운로드 수를 달성할 정도로 높은 성능을 보였다. 이후 두개의 변환기를 활용하여 사용자가 전송한 문장을 상냥체, 정중체로 변환하여 사용자 말투 학습 데이터 쌍을 생성하고, 정중체, 상냥체 문장이 입력되었을 때 사용자의 말투로 변환되도록 Seq2Seq 모델을 학습시켰다. 문장의 수가 많으면 많을수록 모델의 성능을 향상시킬 수 있으나, 모델 학습 시간을 고려했을 때 1,000개의 문장을 input으로 활용했을 때 학습시간과 출력 결과 성능의 trade off 관계상 최적임을 발견했다.

학습된 모델을 활용하여 사용자의 말투로 변환된 답장 리스트를 실제로 스마트워치에서 사용할 수 있도록 안드로이드 스튜디오 앱을 개발했다. 모바일 앱과 스마트워치 앱을 모두 개발했으며, 모바일 앱에서는 카카오톡 로그인 기능, 변환된 답장 리스트를 RDS DATABASE에서 가져오도록 Django에 요청하는 Request API, 스마트워치에 변환된 답장 리스트를 공유하는 기능, 카카오톡 알림이 도착했을 때 스마트워치에 알림을 전달하는 기능, 백그라운드에서 실제 카카오톡으로 전송하는 기능을 구현했으며, 스마트워치에서는 버튼을 클릭했을 때 선택한 답장을 모바일로 전송하는 기능을 구현했다.

결과적으로, 실제로 카카오톡과 연동하여 사용할 수 있는 스마트워치 앱을 개발하는데에 성공했다. 하지만 아직 실제 사용성 평가는 마치지 못했으므로, 변환된 답장 리스트를 사용하는 A 집단과 변환되지 않은 답장리스트를 사용하는 B집단을 비교하는 A/B 테스트와 실제 사용자 인터뷰를 통한 정성적 평가가 필요하다.

또한 추후 사용자의 상대방 별 말투뿐만 아니라, 운전중, 회의중 등의 사용자 맥락을 추적하여 그에 맞춤형 답장 리스트를 변환하여 제공하거나, 생성형 AI를 활용하여 상대방이 보낸 문장에 대한 적절한 답장을 변환하여 제공하는 등의 후속연구가 필요할 것으로 생각된다.

참고문헌

- [1] Algouzi, Sami, and Ali Abbas Falah Alzubi. "The Study of AI-Mediated Communication and Socio-Cultural Language-Related Variables: Gmail Reply Suggestions.", Applied Artificial Intelligence 37, no. 1, 2023

[2] Komninos, Andreas and Dunlop, Mark. "Text input on a smart watch", IEEE Pervasive Computing, 50-58, 2013

[3] Ronald E Robertson, Alexandra Olteanu, Fernando Diaz, Milad Shokouhi, and Peter Bailey. "I Can't Reply with That": Characterizing Problematic Email

Reply Suggestions,

Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 1–18, 2021

[4] Stephen Oney, Chris Harrison, Amy Ogan, and Jason Wiese. "ZoomBoard: a diminutive qwerty soft keyboard using iterative zooming for ultra-small devices", Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2799–2802, 2013

[5] Yue Weng, Huaixiu Zheng, Franziska Bell, and Gokhan Tur. "OCC: A Smart Reply System for Efficient In-App Communications", Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2596–2603, 2019