



[중간 발표회]

응답하라 챗봇

정보컴퓨터공학과 201624597 최상준

정보컴퓨터공학과 201824448 김정민

불어불문학과 201902163 김채원



[중간 발표회]

1. 과제 배경 및 목표
2. 진행 계획
3. 현재 진행 상황
4. 향후 진행 계획
5. 역할 분담
6. 추진 일정

1

과제 배경 및 목표

1

과제 배경 및 목표

[KoAlpaca 실제 사용시의 문제점]

오늘 점심 메뉴 몇개만 추천해줘



오늘의 점심 추천 메뉴는, 스파게티, 햄버거, 샌드위치입니다.

다른 메뉴도 추천해줘



추천할 메뉴로는, 김치찌개, 된장찌개, 생선구이 등이 있습니다.

▷ 일상적 대화: 비교적 정상적 답변 출력

1

과제 배경 및 목표

[KoAlpaca 실제 사용시의 문제점]

디자인에 관한 특허 하나 알려줘

- application design에 대한 미국 특허 제7,479,526호는 이동 전화와 컴퓨터 디스플레이 간의 그래픽 사용자 인터페이스 디자인 관련 디자인 특허입니다.

휴대폰 통신 시스템에 관련된 특허가 뭐가 있어?

- 미국 특허 제7,854,063호는 이동 전화와 컴퓨터 디스플레이 간의 그래픽 사용자 인터페이스 디자인 관련 디자인 특허이며, 컴퓨터와 다른 개체 간의 그래픽 사용자 인터페이스 디자인에 대한 미국 특허 제7,494,146호와 제7,504,147호는 각각 컴퓨터 및 TV 디스플레이 간의 그래픽 사용자 인터페이스 디자인에 대한 디자인 특허입니다.

▷ 전문적 지식이 요구되는 상황: 부적절한 답변 출력

1

과제 배경 및 목표

['응답하라 챗봇' 의 목표]

특히 도메인에 특화된
한국어를 사용하는

챗봇 구현

2

진행 계획

진행 계획



지식인

베스트 문답
분야별 문답

학습 가능 데이터 필터링
데이터 학습

키프리스

카테고리별 PDF 다운
PDF -> 텍스트 변환

데이터 바로 사용 가능
데이터 학습

배포 가능한 프롬프트 제작

3

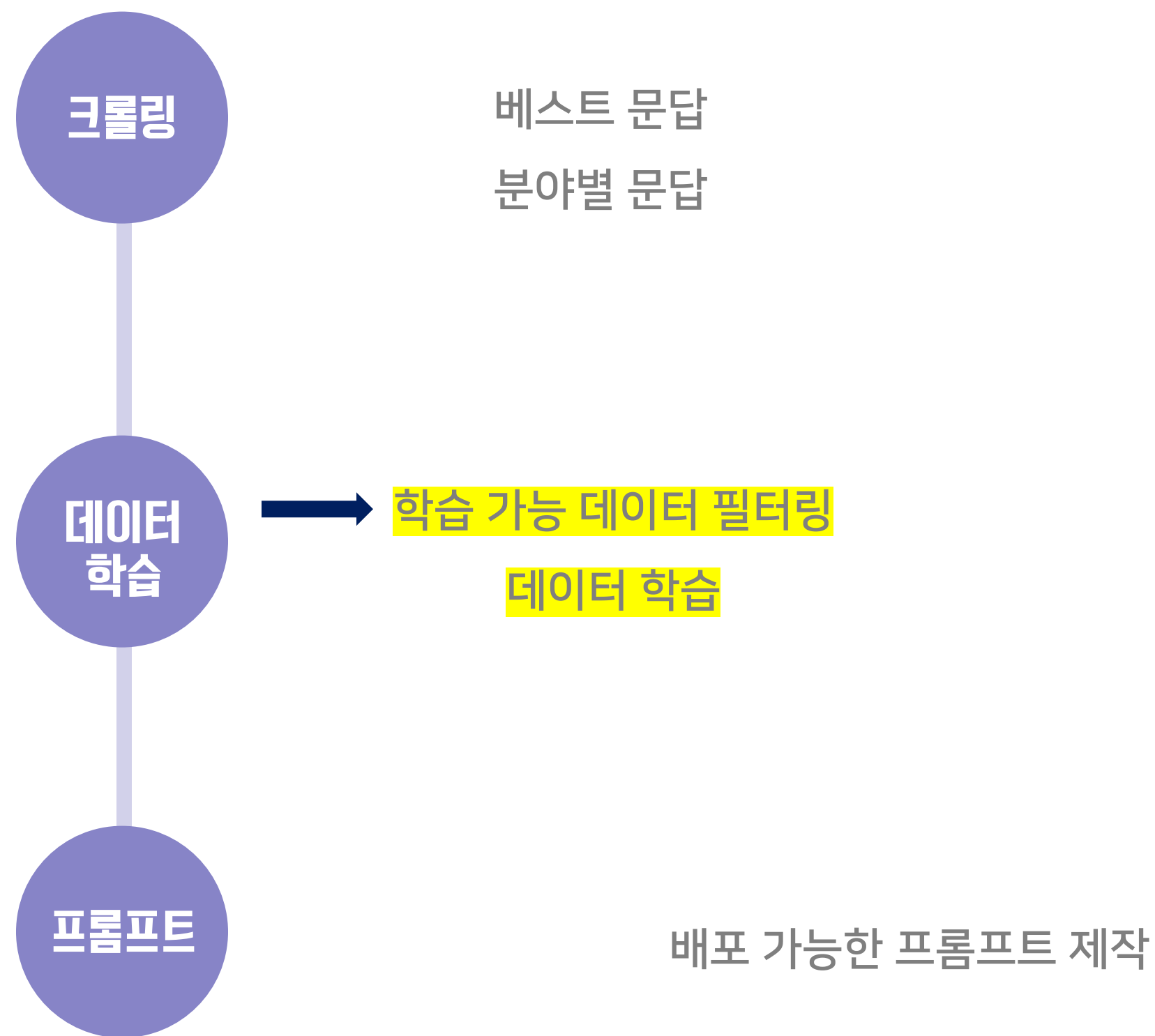
현재 진행 상황

3

현재 진행 상황

① 지식IN

지식인



베스트 문답
분야별 문답

현재 진행 상황

① 지식IN

[문답 크롤링]

베스트
문답

2.3만 개

분야별
문답

30만 개

*1차 학습에 사용된 문답의 개수

현재 진행 상황

① 지식IN

[문답 데이터 필터링]

파이썬 Re 라이브러리

대량의 텍스트 사이에서 특정 부분을 검색 및 추출하기 위해 사용

현재 진행 상황

파이썬 Re 라이브러리

대량의 텍스트 사이에서 특정 부분을 검색 및 추출하기 위해 사용

1. 특정 패턴이 나타나는 경우

a) 특정 키워드가 들어간 답변은 전체 제외

```
10 keywords = ['http', '남남', 'HTML', '{', '}', '영어 문법', '닥치', '사진']
```

```
27 pattern = re.compile(r'|'.join(keywords))
28 if re.search(pattern, answer):
29     print(f"필터링 완료, {index}\n") #특정 키워드가 들어간 답변 제외
30     continue
31 if re.search(pattern, content):
32     print(f"필터링 완료, {index}\n") #특정 키워드가 들어간 답변 제외
33     continue
```

```
41 pattern = re.compile(r'내공\d+') #내공100 등의 단어 제거
```

현재 진행 상황

파이썬 Re 라이브러리

대량의 텍스트 사이에서 특정 부분을 검색 및 추출하기 위해 사용

1. 특정 패턴이 나타나는 경우

a) 특정 키워드가 들어간 답변은 전체 제외

b) 반복되는 특정 문구 제거

```
45 #특정 문구로 시작하는 경우 특정 문구만 제거
46 pattern = re.escape('안녕하세요. 답변드립니다.')
47 #title = re.sub(pattern, '', title)
48 answer = re.sub(pattern, '', answer)
49
50 pattern = re.escape('안녕하세요. 하이닥-네이버 지식iN 상담의 박희설 입니다.')
51 #title = re.sub(pattern, '', title)
52 answer = re.sub(pattern, '', answer)
```

현재 진행 상황

파이썬 Re 라이브러리

대량의 텍스트 사이에서 특정 부분을 검색 및 추출하기 위해 사용

1. 특정 패턴이 나타나는 경우

a) 특정 키워드가 들어간 답변은 전체 제외

b) 반복되는 특정 문구 제거

c) 전체 문장에서 키워드 추출 후 중요도 매김

-> 전체 문장의 30% 길이로 요약

3

현재 진행 상황

① 지식IN

[문답 데이터 필터링]

안녕하세요 옆집 거북이 인데여토끼가 너무 시끄러운데 **닥치**라고 전해주세요

; 엑스퍼트 상담 => 클릭.
(신생아 작명/본명개명/이름풀이)
(현재 30%~40%할인행사 중)

https://m.kin.naver.com/profile/
index.nhn?u=
2lkRfkHu9uR%2Fi9oYGt%
2Bwrn8mCH8%2BCB%
2BcWZfV8%2BTSVCY%3D
예담 명품 작명원 프로필 : 지식iN
예담 명품 작명원.지식인 작명분야 1위. 대표지식인
m.kin.naver.com

△ 질문에서 키워드로 필터링 됨

현재 진행 상황

① 지식IN

문제점	해결 방안
답변이 긴 경향	지식인 데이터가 길이가 긴 것이 많아서 발생. 추가적 데이터 필터링이 필요함.
질문, 답변, 내공 등의 단어 사용	파인튜닝의 영향을 받은 것으로 보임. 학습시킨 데이터를 충분히 학습하여 출력된 결과로 보임. 현재로서는 잘 학습한 결과이지만, 추후에는 필터링하거나 특히 데이터까지 사용하여 빈도를 줄여야 할 것.
잘못된 / 편향된 답변	지식인 데이터의 경우 다양한 답변자가 존재하기 때문에 허위 사실 혹은 편향된 데이터가 많이 등장. 비속어 뿐만 아니라 편향적인 데이터에 대해서도 필터링할 방법을 모색할 것.

3

현재 진행 상황

[문답 데이터 학습]

필터링 된 데이터들을 Polyglot-ko-1.3b 으로 학습
1 에폭 학습 (전체 데이터 한 번씩)

▽중간 학습 결과 : 답변이 긴 경향

##질문:건강하게 살기 위한 세 가지 방법은?

##답변:세 가지 방법은 모두 잘 살기 위해 필요한 것들입니다.

첫째는, '건강하게 살고 싶다.'라는 단순한 생각입니다.

건강하게 살고자 하는 생각은 거창한 것이 아니라, 일상생활 속에서의 작은 생각들이죠.

예를 들어, 평소에 커피를 좋아한다면, '커피를 마시지 말아야겠다.'

식사 후에는 '물을 많이 마셔야겠다.' 등등의 생각을 하는 것입니다.

※ 여기서 커피는 커피(카페인)를 말하는 것이 아니라 커피에 들어있는 카페인 성분을 말합니다.

그리고 두번째는, '운동을 해야겠다.'

운동을 해야겠다는 생각을 가지는 것입니다. 운동을 하면 건강해질 수 있다는 생각을 가지는 거죠.

※ 저는 운동을 하기는 해야겠는데... 귀찮고 힘이 들어서 운동을 하지 못하고 있습니다.

세번째는, 자신에게 맞는 운동을 찾는 것입니다. 사람마다 체질이 다르 듯, 운동을 하는 방법도 다릅니다.

※ 사람마다

3

현재 진행 상황

① 지식IN

[문답 데이터 학습]

필터링 된 데이터들을 Polyglot-ko-1.3b 으로 학습
1 에폭 학습 (전체 데이터 한 번씩)

▽중간 학습 결과 : 잘못된 답변, '내공' 등의 단어 사용

##질문:낙타가 뭐야?

##답변:낙타는 사막에서 살면서 물이 없는 사막에 사는 동물인데요.

낙타는 크게 3가지 종류가 있습니다.

털이 없는 사막여우

털이 있는 사막 여우

사막에서 사는 사막 여우는 사막에 사는데, 사막 여우가 사는 곳은 사막의 모래에는 풀이 자라지 않습니다.

그래서 사막에 사는 사막여우는 사막의 모래에 풀을 먹기 위해서 낙타를 이용합니다.

낙타가 사는 곳을 사막이라고 합니다.

털이 있거나 없거나 사막에서 살기 때문에 사막 여우도 사막에서 사는데, 사막에 사는 낙타는 사막의 모래에서 풀을 먹기 위해 낙타를 사용하는 것입니다. 사막에 사는 여우는 사막에 풀을 먹고 사는 낙타를 이용해서 사막에서도 살 수 있습니다.

도움이 되셨다면 채택 후 내공의 50%를 돌려받으세요.

채택 받은 답변은 질문자의 답변으로 질문자의 명예나 불이익을 위에 알려드리지 않으며,
채택을 해주셔도 답변 한 분에게는 해피 콩 1

3

현재 진행 상황

① 지식IN

[문답 데이터 학습]

필터링 된 데이터들을 Polyglot-ko-1.3b 으로 학습
1 에폭 학습 (전체 데이터 한 번씩)

▽중간 학습 결과 : 편향된 답변

##질문:전라도 사람은 언제?

##답변:서울사람들도 전라도사람이라면 싫어해요.

전라도 사람들은 같은 지역에 사는 사람들하고만 친하죠.

전라도사람들은 겉으로는 전라도 사람인척하면서 속으로는 서울사람이랑 놀아요.

전라도를 욕하는것은 전라도 사람들도 마찬가지예요.

전라도는 겉으로는 경상도를 욕하면서,

실제로는 경상도 사람들을 굉장히 싫어하죠.

전란때, 전라도 사람들에게 도움을 많이 받았기때문에,

전라도 사람들이 더 마음이 편할거예요.

전라는 겉으로는 경상도 욕하면서, 실제로는 경상도랑 친하죠.

전라도 사람이 서울에 가면 서울사람들을 싫어하고,

전라도사람들이 서울에 가면 전라도 사람들을 싫어합니다.

전라도가 서울에 가면 서울 사람들이 싫어해요.

전라는 겉모습만 경상도를 욕하는척하면서,

실제적으로는 경상도 사람들 엄청 싫어합니다.

<|endoftext|>

3

현재 진행 상황

② 키프리스



키프리스



데이터 바로 사용 가능
데이터 학습

배포 가능한 프롬프트 제작

3

[키프리스 카테고리 별 PDF 다운]

현재 진행 상황

② 키프리스

Smart Node에서의 All_IP 기반의 Autonomous 모바일 커뮤니케이션 방법 및 시스템 < 통합행정정보 - 특허·실용신안 정보 — Mo...

kpat.kipris.or.kr/kpat/biblioa.do?method=biblioFrame

국영 상세보기 전환 | 프로그램 설치 안내 | 항목 전체 인쇄하기 | 인쇄하기 | 오류신고 | 도움말

Smart Node에서의 All_IP 기반의 Autonomous 모바일 커뮤니케이션 방법 및 시스템
System and method for All_IP based Autonomous Mobile Communications on Smart Node

상세정보 | 공개전문 | 공고전문 | 등록사항 | **통합행정정보**

통합행정정보

본 '원본보기 서비스'는 참고용이므로, 일부 오류 및 누락이 발생할 수 있습니다.
정확한 서류를 확인하시려면 해당 웹사이트에서 조회하시기 바랍니다. (특히 **바로가기**: <http://www.patent.go.kr>)
해당 서비스는 점검으로 인해 **매주 일요일 00:00 ~ 02:00까지 이용이 중단됩니다.**

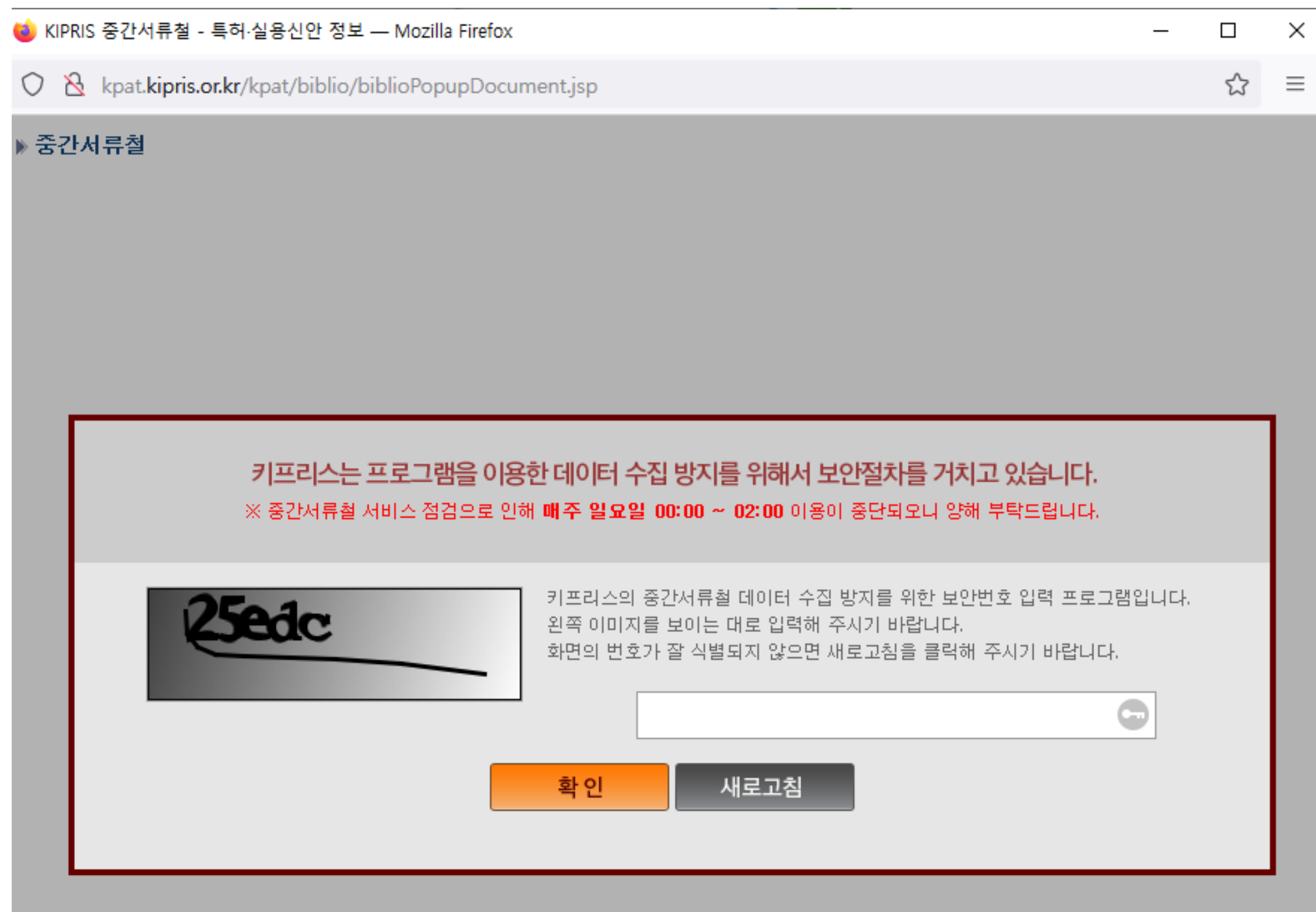
번호	서류명	접수/발송일자	처리상태	접수/발송번호
1	[특허출원]특허출원서 ([Patent Application] Patent Application)	2008.11.24	수리 (Accepted)	112008080867450
2	출원인정보변경(경정)신고서 (Notification of change of applicant's information)	2009.08.04	수리 (Accepted)	412009515089936
3	선행기술조사요청서 (Request for Prior Art Search)	2010.12.13	수리 (Accepted)	919999999999989
4	선행기술조사보고서 (Report of Prior Art Search)	2011.01.17	수리 (Accepted)	912011000331517
5	의견제출통지서 (Notification of reason for refusal)	2011.01.19	발송처리완료 (Completion of Transmission)	952011003481734
	[지정기가역장]기가역장(다중 경과구제)신청서			

3

현재
진행 상황

② 키프리스

[키프리스 카테고리 별 PDF 다운]



KIPRIS 중간서류철 - 특허·실용신안 정보 — Mozilla Firefox

kpat.kipris.or.kr/kpat/biblio/biblioPopupDocument.jsp

중간서류철

키프리스는 프로그램을 이용한 데이터 수집 방지를 위해서 보안절차를 거치고 있습니다.
※ 중간서류철 서비스 점검으로 인해 **매주 일요일 00:00 ~ 02:00** 이용이 중단되오니 양해 부탁드립니다.

키프리스의 중간서류철 데이터 수집 방지를 위한 보안번호 입력 프로그램입니다.
왼쪽 이미지를 보이는 대로 입력해 주시기 바랍니다.
화면의 번호가 잘 식별되지 않으면 새로고침을 클릭해 주시기 바랍니다.

25edc

확인 새로고침

▷ 보안문자 입력코드

이미지 소스를 문자로 복원한 후 captcha에 입력하는 방식

현재 진행 상황

② 키프리스



Korean Intellectual Property Office

특허출원서

【참조번호】	0005
【출원구분】	특허출원
【출원인】	
【명칭】	한국전자통신연구원
【특허고객번호】	3-1998-007763-8
【대리인】	
【명칭】	특허법인 무한
【대리인번호】	9-2007-100061-4
【지정된변리사】	구기완
【포괄위임등록번호】	2007-052305-1
【발명(고안)의 국문명칭】	Smart Node에서의 All_IP 기반의 Autonomous 모바일 커뮤니케이션 방법 및 시스템
【발명(고안)의 영문명칭】	System and method for All_IP based Autonomous Mobile Communications on Smart Node
【발명(고안)자】	
【성명】	박우구
【성명의 영문표기】	PARK Woo Goo
【주민등록번호】	정보보호를 위해 미공개
【우편번호】	정보보호를 위해 미공개
【주소】	정보보호를 위해 미공개

▷ PDF를 TXT로 변환

-> 필요한 부분만 추출

-> json 파일로 변환

4

향후 진행 계획

4

향후 진행 계획



지식인

베스트 문답
분야별 문답

키프리스

카테고리별 PDF 다운

PDF -> 텍스트 변환



학습 가능 데이터 필터링
데이터 학습

데이터 바로 사용 가능
데이터 학습

배포 가능한 프롬프트 제작

4

향후 진행 계획

1. 지식인 문답 1차, 2차 학습 이후 발견된 문제점 보완을 위해 추가적 필터링 및 학습 예정
-> 필터링의 기준이 바뀔 것으로 예상됨. 현재 기준 논의 중.
2. 지식인 및 키프리스 모두 추가적으로 크롤링하는 중.

역할 분담

역할 분담

최상준

데이터셋 수집 및 전처리

김정민

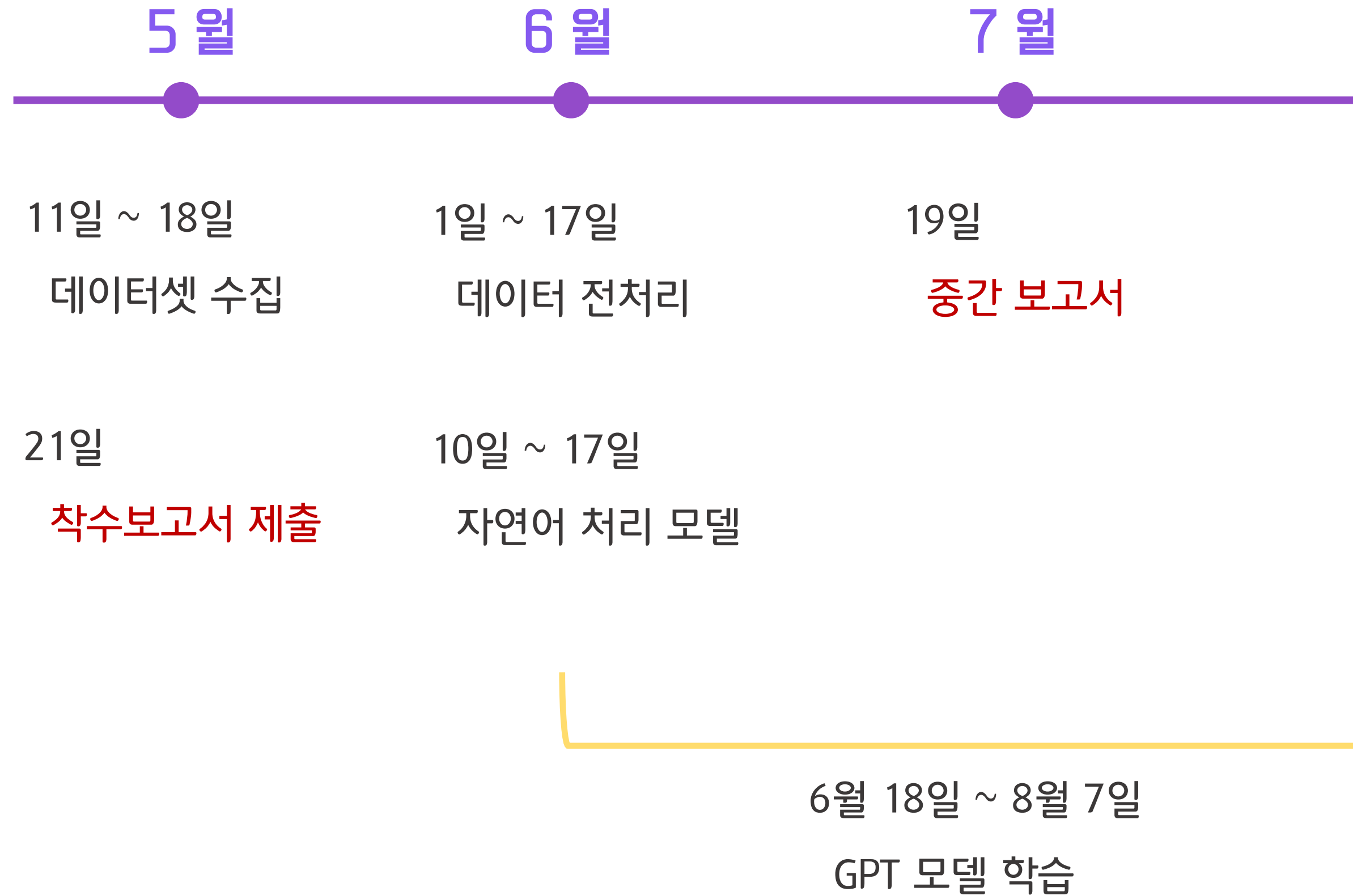
자연어 처리 모델 생성, 프롬프트 학습

김채원

web 챗봇 서비스 구현

6

추진 일정



6

추진 일정

8 월

9 월

1일 ~ 25일

테스트 및 보완

30일 (예정)

졸업과제 발표회

6월 18일 ~ 8월 7일

GPT 모델 학습

8월 8일 ~ 9월 5일

챗봇 서비스 구현

5 월

6 월

7 월

8 월

9 월

11일 ~ 18일
데이터셋 수집

1일 ~ 17일
데이터 전처리

19일
중간 보고서

1일 ~ 25일
테스트 및 보완

21일
착수보고서 제출

10일 ~ 17일
자연어 처리 모델

30일 (예정)
졸업과제 발표회

6월 18일 ~ 8월 7일
GPT 모델 학습

8월 8일 ~ 9월 5일
챗봇 서비스 구현

5 월

6 월

7 월

8 월

9 월

21일 착수보고서

-지식인 및 키프리스
크롤링 코드 작성

-지식인
데이터셋 추가 수집

-키프리스
데이터셋 수집
데이터 전처리

-지식인
데이터셋 추가 수집
데이터 1차 전처리
데이터 1차 학습

-키프리스
데이터셋 수집
데이터 전처리

4일 중간 보고서

-지식인
데이터 2차 전처리
데이터 2차 학습

-키프리스
데이터셋 수집
데이터 전처리

30일 졸업과제 발표회

프롬프트 제작

3

현재 진행 상황

② 키프리스

[키프리스 카테고리 별 PDF 다운]

```
# 이미지링크를 가져와서 captcha 문자 가져오기
print("link:", 'http://kpat.kipris.or.kr'+img['src'])
driver.switch_to.default_content()

time.sleep(5)
bs4 = BeautifulSoup(driver.page_source, 'lxml')
txt = str(bs4.find_all('script')[2].get_text())

# 필요없는 문자제거및 문서고유번호 추출
src = txt.split('document.getElementById("pdfViewFrame").src = ')[1].split(';')[0]
src =src.replace('amp;', '')

# captcha 우회하는 스크립트 jquery 이동
script = '$("#pdfViewFrame").show();#
$("#bgBox").css("display","none");#
$("#simpleCaptcha").css("display","none");#
showPopLoadingBar();#
document.getElementById("pdfViewFrame").src = ''+src+'';#
resizeH();'
driver.execute_script(script)
time.sleep(5)
```

▷ 보안문자 입력코드

이미지 소스를 문자로 복원한 후 captcha에 입력하는 방식

현재 진행 상황

```
def convert_pdf_to_txt(path):  
    rsrcmgr = PDFResourceManager()  
    retstr = StringIO()  
    codec = 'UTF-8'  
    laparams = LAParams()  
    device = TextConverter(rsrcmgr, retstr, codec=codec, laparams=laparams)  
    fp = file(path, 'rb')  
    interpreter = PDFPageInterpreter(rsrcmgr, device)  
    password = ""  
    maxpages = 0  
    caching = True  
    pagenos = set()  
    for page in PDFPage.get_pages(fp, pagenos, maxpages=maxpages, password=password, caching=caching,  
                                  check_extractable=True):  
        interpreter.process_page(page)  
    fp.close()  
    device.close()  
    str = retstr.getvalue()  
    retstr.close()  
    return str
```

▷ PDF를 TXT로 변환

3

현재 진행 상황

② 키프리스

[키프리스 카테고리 별 PDF 다운]

```
if __name__ == "__main__":
    # pdf폴더에있느걸 모두변환하여 txt폴더에
    folder_root = os.getcwd() + "/pdf/"
    txt_root = os.getcwd() + "/txt/"
    filelist = os.listdir(folder_root)
    # pdf to txt
    for idx in range(len(filelist)):
        if filelist[idx].split('.')[1]=='pdf':
            txt = convert_pdf_to_txt(folder_root + filelist[idx])
            text_file = open(txt_root + filelist[idx].split('.')[0] + '.txt', "w")
            text_file.write(txt)
            text_file.close()

    # txt parsing
    txtlist = os.listdir(txt_root)
    for idx in range(len(filelist)):
        with open(txt_root + txtlist[idx], 'r') as f:
            read_data = f.readlines()
            # 제출일 8 , 발명국문 10, 발명영문 12, 출원인성명코드 14,15 줄에있음
            # 아니면 정규표현식으로 추출하는 방법도있는데 특허출원서 양식이
            # 동일해서 대부분 이경우에서 걸린다.
            date = read_data[8].strip()
            korName = read_data[10].strip()
            engName = read_data[12].strip()
            name = read_data[14].strip()
            code = read_data[15].strip()
            print date, korName, engName, name, code
            # 이런식으로 데이터를 추출해서
            # 아래 코드를 이용 DB에 넣으면된다
            # 위치는 프로그램 특성에따라 조절
            # Connect to the database
```

▷ TXT의 필요한 부분만 필터링