



cakd3

한국어 도서 말뭉치를 활용한 인공지능 서비스  
**문해력 향상 프로그램**

2조 | 송유빈, 이슬, 오수문, 정하림, 조경림, 최혜정



# CONTENTS

01 서비스 기획 목적

03 데이터 수집 및 전처리

05 서비스 제작

02 서비스 시나리오

04 사용 기술

06 서비스 기대효과



# 01

---

## 서비스 기획 목적

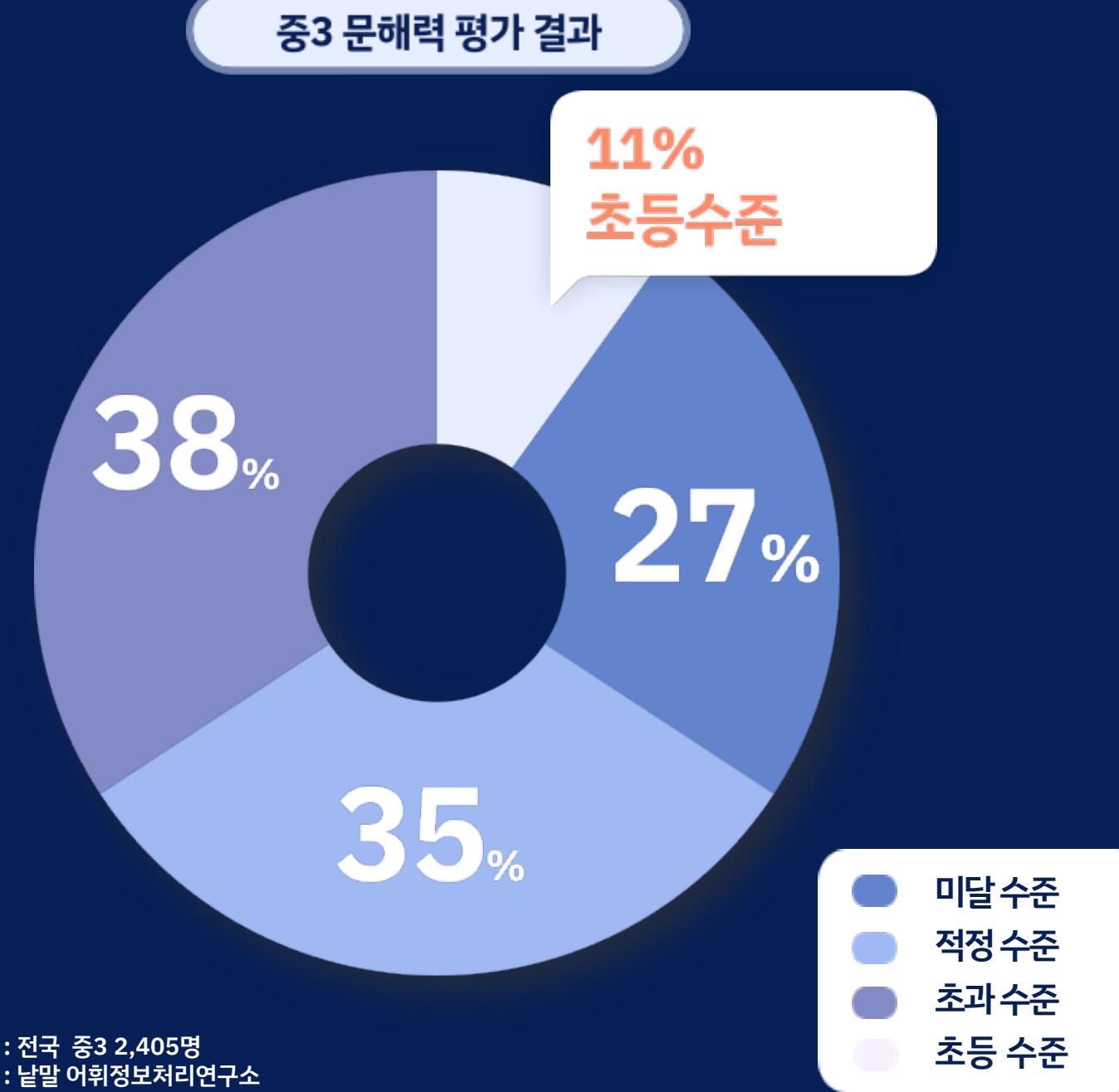
Part 1

# 서비스 기획 목적

## "실질적 문맹, 난독 증가 추세"

전체 학생 27%, 교과서도 제대로 이해하지 못하는 수준

= 문해력이 낮으면 교육, 사회적응 어려워



01

## 서비스 기획 목적

국제 성인역량조사(PIAAC)에 따르면,  
높은 수준의 문해력(상위 11.8%)를 갖춘 사람은  
문명을 갖춘 정도인 사람(최하위 3.3%)보다  
평균 시급이 60% 이상 높음.

= 문해력 낮은 사람은 실업자가 될 확률이 2배 이상 높아짐.

(출처: 중앙일보)



01

# 서비스 기획 목적

문해력 향상을 위한 서비스 아이디어

## : 문서 요약

가장 많이 사용되는 글쓰기, 독해 학습법 중 하나

01

## 서비스 기획 목적

“

**Target**

중·고등학생

”

“

**Purpose**

독해 및 학습  
능력상승

”

02

---

서비스  
시나리오

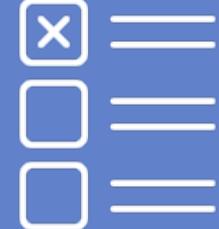
02

# 서비스 시나리오

서비스 제작 시나리오



시나리오 설계



데이터 구축



Fine tuning



웹 서비스 구현

- 프로그램 전체적인 시나리오 구성

- 웅진 한국어 도서 말뭉치 데이터 확인
- AIHub 도서자료 요약 데이터 확인
- 전처리된 DB 생성

- 데이터를 이용한 문서 요약 학습

- CSS / HTML
- Figma
- Flask

02

# 서비스 시나리오

웹서비스 시나리오 설계



읽기 지문

한국어 도서 말뭉치에서  
특정 길이의 독해지문 제공

주어진 지문을 읽은 사용자가 생각  
하는 답안(핵심 문장)을 입력

(사용자) 핵심 문장 입력



(ET5) 요약문 생성: 답안

ETRI의 ET5 한국어 언어 이해 생성 모델을  
이용하여 주어진 지문의 요약문 생성  
-> 레퍼런스(정답)으로 사용

ET5가 생성한 레퍼런스(정답) 문장과  
사용자가 입력한 핵심 문장을 대조하여  
정확도(일치하는 정도) 출력

정확도 출력



02

# 서비스 시나리오

시나리오 예시

## SYSTEM

"이러한 서술체계는 '항일무장투쟁 시기'의 문학에서도 일관되고 있다. '혁명연극'이라는 장르 명칭 대신 '혁명적 극문학'을 장 제목으로 사용하고 있는 것은 국내 문학을 서술할 때 사용하는 장르 명칭과의 통일을 기하여 보다 객관화된 장르 명칭을 부여한 것이라 하겠다. 이 시기에도 '제4절 조선인민혁명군의 필승불패의 위력과 일제의 패망상을 보여준 불후의 고전적 명작 <경축대회>, 풍자극의 새로운 발전', '제5절 자주적 인간의 탄생과 민족해방, 계급해방의 위대한 진리를 밝힌 불후의 고전적 명작 <피바다>와 <한 자위단원의 운명>'과 같이 김일성의 '창작'에 긴 수식어구가 붙여지고 있다. 또 『(구)조선문학사3』에서는 <피바다>와 <한 자위단원의 운명>이 별도의 절로 서술되던 것을 한 절로 묶어 기술하였고, 대신 <경축대회>를 별도의 절로 처리하고 있다. '풍자극의 새로운 발견'이라는 부제목이 말하듯 극양식의 측면에 강조점이 두어지면서 <경축대회>의 문학사적 위상이 높아진 것이 아닌가 한다."

## USER

02

# 서비스 시나리오

시나리오 예시

## SYSTEM

"이러한 서술체계는 '항일무장투쟁 시기'의 문학에서도 일관되고 있다. '혁명연극'이라는 장르 명칭 대신 '혁명적 극문학'을 장 제목으로 사용하고 있는 것은 국내 문학을 서술할 때 사용하는 장르 명칭과의 통일을 기하여 보다 객관화된 장르 명칭을 부여한 것이라 하겠다. 이 시기에도 '제4절 조선인민혁명군의 필승불패의 위력과 일제의 패망상을 보여준 불후의 고전적 명작 <경축대회>, 풍자극의 새로운 발전', '제5절 자주적 인간의 탄생과 민족해방, 계급해방의 위대한 진리를 밝힌 불후의 고전적 명작 <피바다>와 <한 자위단원의 운명>'과 같이 김일성의 '창작'에 긴 수식어구가 붙여지고 있다. 또 『(구)조선문학사3』에서는 <피바다>와 <한 자위단원의 운명>이 별도의 절로 서술되던 것을 한 절로 묶어 기술하였고, 대신 <경축대회>를 별도의 절로 처리하고 있다. '풍자극의 새로운 발견'이라는 부제목이 말하듯 극양식의 측면에 강조점이 두어지면서 <경축대회>의 문학사적 위상이 높아진 것이 아닌가 한다."

## USER

### (요약문 작성)

'항일무장투쟁 시기'의 문학에서도 일관되고 있다. '혁명연극'이라는 장르 명칭 대신 '혁명적 극문학'을 장 제목으로 사용하고 있는 것은 국내 문학을 서술할 때 사용하는 장르 명칭과의 통일을 기하여 보다 객관화된 장르 명칭을 부여한 것이라 하겠다. 극 양식의 측면에 강조점이 두어지면서 <경축대회>의 문학사적 위상이 높아진 것이 아닌가 한다.

02

# 서비스 시나리오

시나리오 예시

## SYSTEM

### <평가 기준>

0.8 이상 Perfect

0.6 이상 0.8 미만 Great

0.4 이상 0.6 미만 good

0.4 미만 Try again

"이러한 서술체계는 ‘항일무장투쟁 시기’의 문학에서도 일관되고 있다. ‘혁명연극’이라는 장르 명칭 대신 ‘혁명적 극문학’을 장 제목으로 사용하고 있는 것은 국내 문학을 서술할 때 사용하는 장르 명칭과의 통일을 위하여 보다 객관화된 장르 명칭을 부여한 것이라 하겠다. 이 시기에도 ‘제4절 조선인민혁명군의 필승불패의 위력과 일제의 패망상을 보여준 불후의 고전적 명작 <경축대회>, 풍자극의 새로운 발전’, ‘제5절 자주적 인간의 탄생과 민족해방, 계급해방의 위대한 진리를 밝힌 불후의 고전적 명작 <피바다>와 <한 자위단원의 운명>’과 같이 김일성의 ‘창작’에 긴 수식어구가 붙여지고 있다. 또 『(구)조선문학사3』에서는 <피바다>와 <한 자위단원의 운명>이 별도의 절로 서술되던 것을 한 절로 묶어 기술하였고, 대신 <경축대회>를 별도의 절로 처리하고 있다. ‘풍자극의 새로운 발견’이라는 부제목이 말하듯 극양식의 측면에 강조점이 두어지면서 <경축대회>의 문학사적 위상이 높아진 것이 아닌가 한다."



'항일무장투쟁 시기'의 문학에서도 일관되고 있다. ‘혁명연극’이라는 장르 명칭 대신 ‘혁명적 극문학’을 장 제목으로 사용하고 있는 것은 국내 문학을 서술할 때 사용하는 장르 명칭과의 통일을 위하여 보다 객관화된 장르 명칭을 부여한 것이라 하겠다. 극 양식의 측면에 강조점이 두어지면서 <경축대회>의 문학사적 위상이 높아진 것이 아닌가 한다.

# 03

---

## 데이터 수집 및 전처리

03

# 데이터 수집 및 전처리

데이터 수집

## 인공지능 서비스 공모전(웅진 북센)

한국어 도서 기반의 대규모 말뭉치를 통해  
다양한 한국어 기반 인공지능 서비스가 개발될 것을 기대

- 대규모의 한국어 빅데이터를 활용한 인공지능 서비스
- **한글, 한국어 활용 및 학습 관련 소프트웨어**

2021 한국어 도서 말뭉치를 활용한  
**인공지능 서비스 공모전**



**■ 공모주제**

- 대규모의 한국어 빅데이터 말뭉치를 활용한 인공지능 서비스
- 한글, 한국어 활용 및 학습 관련 소프트웨어

**■ 참가자격**

- 전국민 누구나 (개인, 기업 모두 응모 가능)

**■ 접수방법**

- email 접수 : proposal@wboxen.com
- 접수기간 : 2021. 11. 24 ~ 2021. 12. 17

**■ 제출자료**

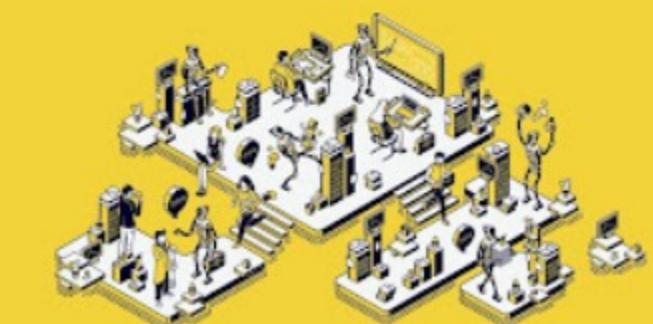
- 응모지원서, 서비스 기획안 (2페이지 이내, 자유양식)
- 프로토타입 : AI모델과 기본 기능이 구현된 실행 가능한 품주소 또는 업. 인공지능 모델관련 소스코드
- 홈페이지 참고 : <http://ebook.boxen.com>

**■ 상금**

- 최우수상 500만원
- 우수상 200만원
- 장려상 100만원

**■ 심사 / 결과발표**

- 심사 : 2021. 12. 17 ~ 2021. 12. 22
- 구현 가능성, 주제 적합성, AI 모델 및 데이터 활용성, 창의성
- 결과발표 : 2021. 12. 22 (공모전 안내 페이지 공지 및 개별 연락)



본 공모전은 과학기술정보통신부가 주관하고 한국지능정보사회진흥원이 지원하는 전문적인 학술증진사업 중 대규모 도서 한국어 말뭉치 대비디전에서 출보, 대비디자인을 위해 시행됩니다.

주최:  과학기술정보통신부 NIA 한국지능정보사회진흥원 주관:  웅진북센

03

# 데이터 수집 및 전처리

데이터 수집

## 2 AI hub > 개방 데이터 > 음성/자연어 > 문서요약 텍스트

도서를 기반으로 한 원문의 핵심 내용, 의미 전달을 적절히 포함하는 요약문을 자동으로 생성하는  
AI기술 개발을 위한 도서 요약 텍스트 데이터

(텍스트 형식의 문단(각 300-1000자) 20만 건과 각 문단별 요약문 20만 건)



03

# 데이터 수집 및 전처리

데이터셋

## ▶ AI hub 데이터 (.json)

```

"passage_id": "123456_0001",
"metadata":
{
    "doc_id": "123456",
    "doc_type": "도서",
    "doc_name": "북미정상회담: 창조적 혁혁성이 될 것인가?",
    "author": "정성윤",
    "publisher": "통일연구원",
    "published_year": "2018",
    "kdc_label": "사회과학",
    "kdc_code": "300"
},
"chapter": null,
"passage": "최근 정세 변동의 의미와 평가 북핵 문제는 지난 25년 동안 다양한 철학과 접근법 그리고 전략의 동원에도 불구하고 않았던 난제 중 난제이다. 그 결과 북핵 문제는 한반도와 동북아의 모든 이슈를 살피고 가두어 버리는 불핵화로 되어 버렸다. 그러나 북핵 문제가 새로운 국면에 진입하고 있다. 결정적 계기는 할후 개최될 남북 정상회담과 북미 정상회담이 될 것이다. 특히 북핵 위기 25년 만에 처음으로 북미 정상이 직접 답판을 하게 됨으로써, 북핵 문제의 획기적 전환에 대한 기대가 높아지고 있다. 두 차례 정상회담이 합의된 가장 큰 배경은 북한의 태도 변화, 미국의 대화 흥을, 우리 정부의 강력한 남북관계 진전 의지와 외교학이다. 이 중 김정은 스스로가 비핵화 의지를 밝힌 것이 정세 변화의 핵심이자 축발 동인이다. 이러한 북핵 정세 변화가 추동하고 있는 구조적 변화 양상, 이를 가능하게 만든 북한의 전략전환 이유, 북미 정상회담 전후의 정세 방향, 그리고 한국의 정책적 고려사항을 제시한다.",
"summary": "북핵 문제는 지난 25년 동안 다양한 노력에도 해결되지 않은 난제로 한반도와 동북아를 모두 아우르는 주요 이슈가 되었다. 그러나 북핵 문제가 남북 정상회담과 북미 정상회담을 계기로 새로운 양상에 접속한다. 이는 북한, 미국, 우리 정부의 태도 변화 및 대화 의지가 바탕이 되었고, 이 중 특히 김정은의 비핵화 의지 표명이 가장 결정적이다. 이러한 구조적 변화와 숨은 의도, 예상 정세 및 한국의 정책 고려사항을 제시한다."

```

## ▶ 공모전 데이터 (.json)

```

{"id": "BOOK_CORPUS_300.2",
"info": {"author": {"birth_year": 1942, "jobs": ["교수"], "write_age": 61},
"class": 0,
"kdc": "309",
"published_year": 2003},
"sentences": [{"char-count": 27,
'id': "BOOK_CORPUS_300.2.1",
'noise-ratio': 0.0,
'original-text': '또 다른 측면에서 북한의 관행도 시정되어야 한다.',
'text': '또 다른 측면에서 북한의 관행도 시정되어야 한다.',
'word-count': 7},
{'char-count': 47,
'id': "BOOK_CORPUS_300.2.2",
'noise-ratio': 0.0,
'original-text': '북한은 경험을 기본적으로 성공한 남한기업이 북한동포를 지원하는 것으로 인식하고 있다.',
'text': '북한은 경험을 기본적으로 성공한 남한기업이 북한동포를 지원하는 것으로 인식하고 있다.',
'word-count': 10},
{'char-count': 55,
'id': "BOOK_CORPUS_300.2.3",
'noise-ratio': 0.0,
'original-text': '그리고 북한은 가능한 한 당국을 배제시키려 하고 시장경제에 적합한 비즈니스 마인드도 결여되어 있다.',
'text': '그리고 북한은 가능한 한 당국을 배제시키려 하고 시장경제에 적합한 비즈니스 마인드도 결여되어 있다.',
'word-count': 13}]
}

```

03

# 데이터 수집 및 전처리

데이터 전처리과정

파일 형식 변환

**(json → csv)**: 모델 활용을 위한  
데이터 파일 형식 변환

AI 말뭉치 변환

: 문장별로 구별되어있던  
데이터를 문단별로 묶음

컬럼 생성

**Level 1) 400~600자**  
**Level 2) 600자 이상**: 난이도 구별을 위한  
음절별 카운트 컬럼 생성

데이터 병합

**AI 말뭉치**  
**+ 웅진 공모전 말뭉치**: 카테고리별로 병합  
(AI 말뭉치 데이터는  
대부분 사회과학 카테고리에  
들어갔으며, 합쳐지지못한  
약 3개의 카테고리 데이터는  
삭제)

파일 형식 변환

**(csv → db)**: 모델의 빠른 구동속도를 위한  
데이터 파일 형식 변환

03

# 데이터 수집 및 전처리

데이터 결합

= AI hub 말뭉치 + 공모전 데이터 말뭉치

분야	text & summary
기술과학	23919
예술	13891
사회과학	470000
융합	6753

# 04

---

## 사용 기술

Transformer → T5 → ET5 → 코드

04

# 사용 기술

Transformer

## Transformer의 전체 모델 구조

2017년 구글이 발표한 논문인  
 “Attention is all you need”에서 나온 모델로  
 기존의 seq2seq의 구조인 encoder-decoder를 따르면서도  
 Attention만으로 구현한 모델이며  
 RNN을 사용하지 않음에도 RNN보다 우수한 성능을 보인다.

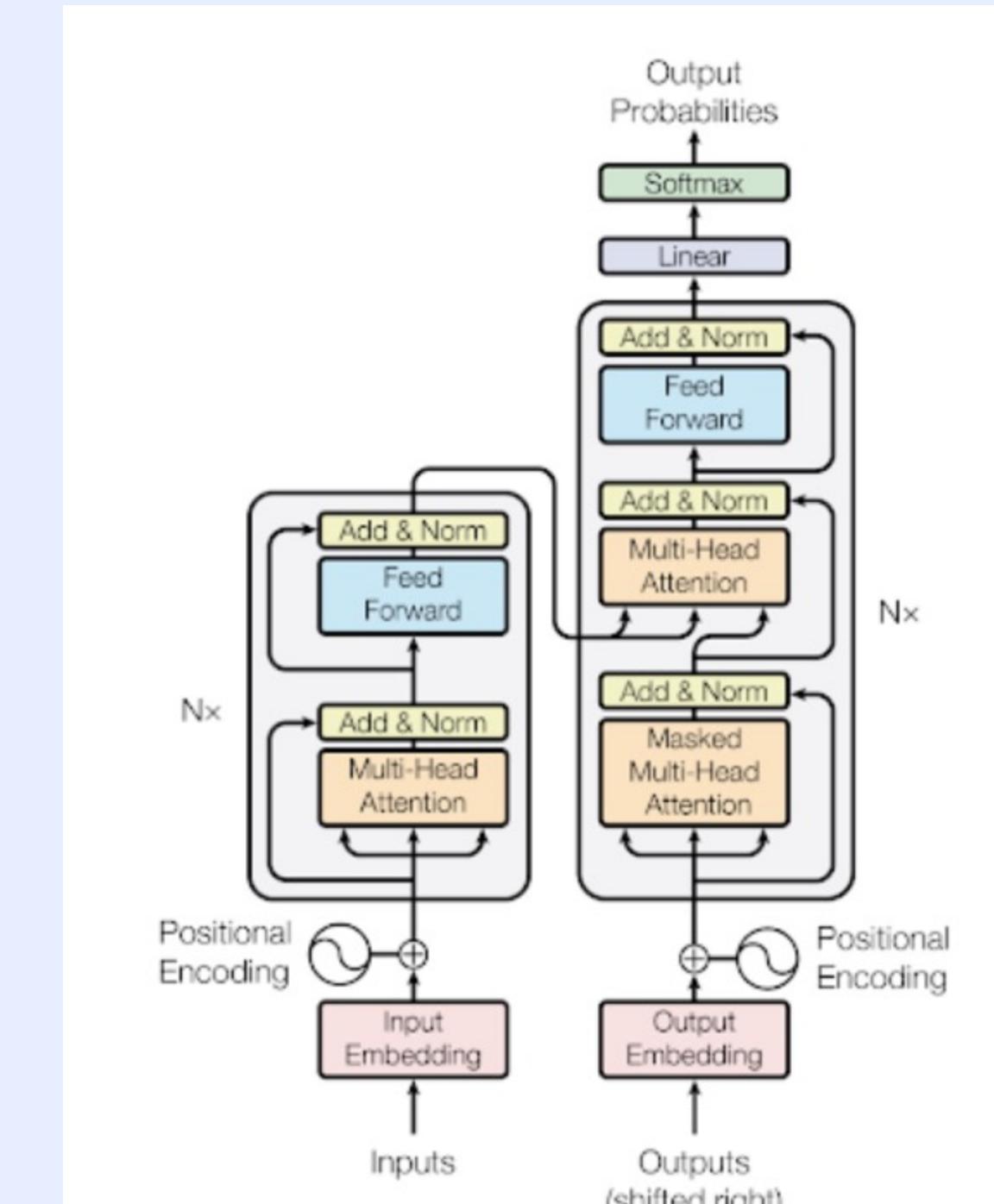


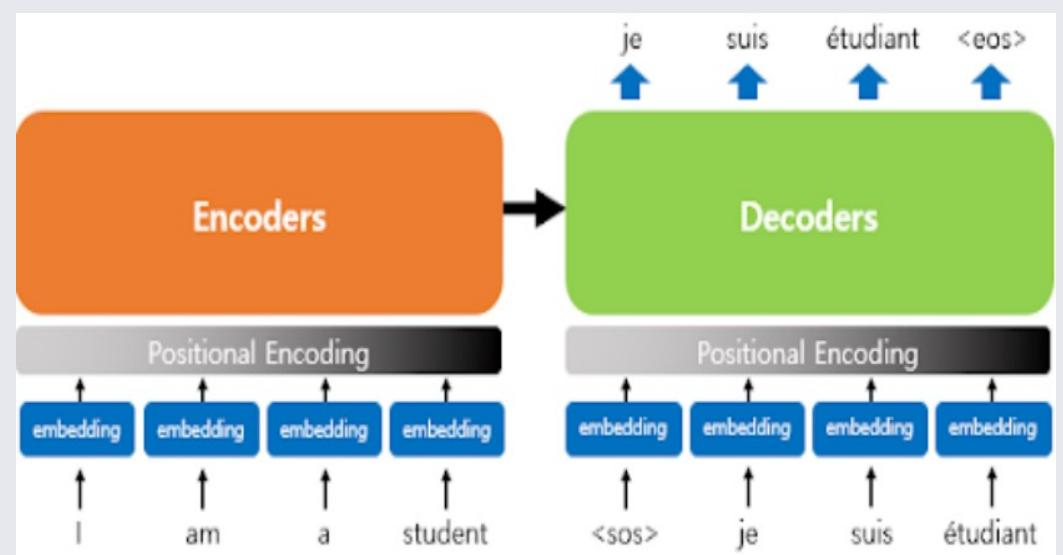
Figure 1: The Transformer - model architecture.

04

# 사용 기술

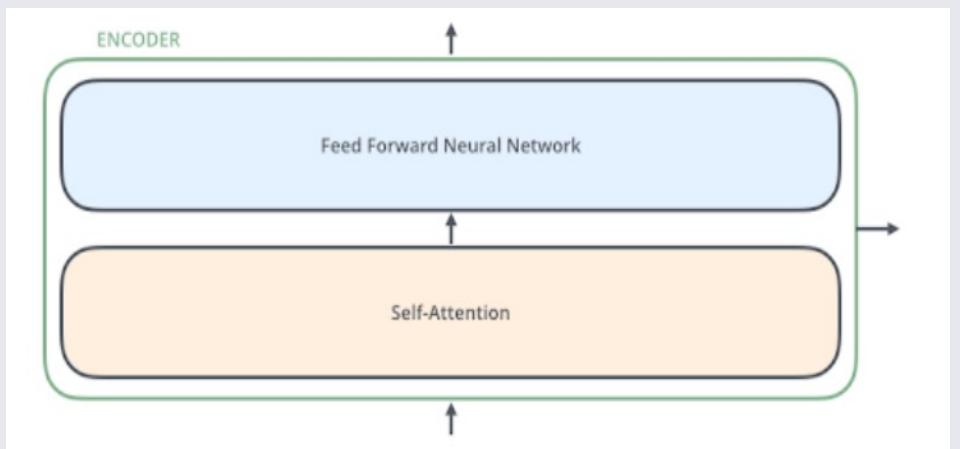
Transformer의 특징

## 특징1 : positional encoding



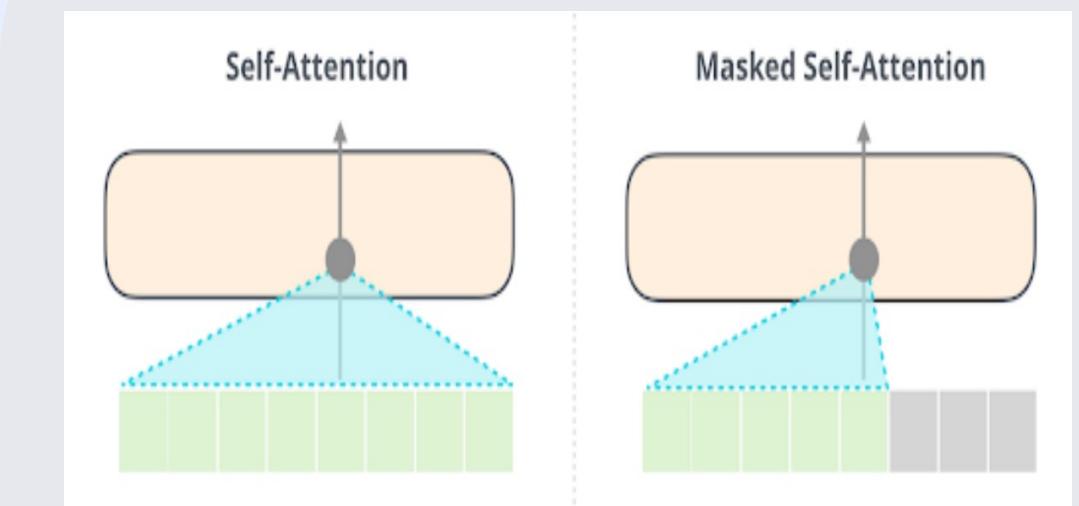
단어 입력을 순차적으로 받는 방식이  
아닌 각 단어의 임베딩 벡터에 위치 정보  
들을 더하여 모델의 입력으로 사용

## 특징2 : self-Attention



Encoder에 들어온 입력은  
self-Attention layer를 지나며  
이 때 한 문장 속에서 특정 토큰의 전체적인 의미를  
가진 값을 도출한다.  
예) 'I study at school'에서 I의 전체적인 의미를  
갖는 값을 도출

## 특징 3 : Masked self-attention



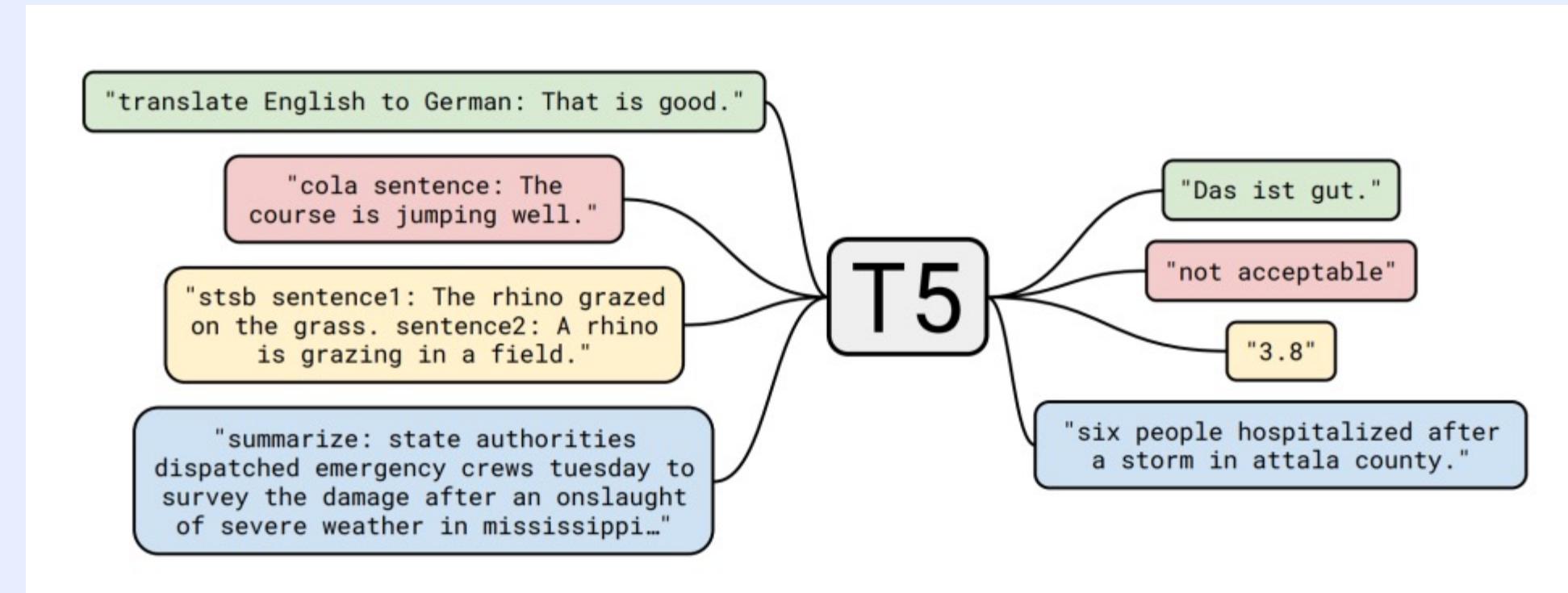
타깃 단어 뒤에 위치한 단어는 self-attention  
에 영향을 주지 않도록 마스킹 처리해 가린다.  
예) 'I' 'study' 'at' 'school'이란 토큰에서 'I'  
'study'란 단어 이후에 올 말을 예측할 때 뒤의 단어  
를 Masking

04

# 사용 기술 : T5

unified Text-to-Text Transformer, T5 특징

Text to text란? text 형태의 문제를 text로 출력하는 것

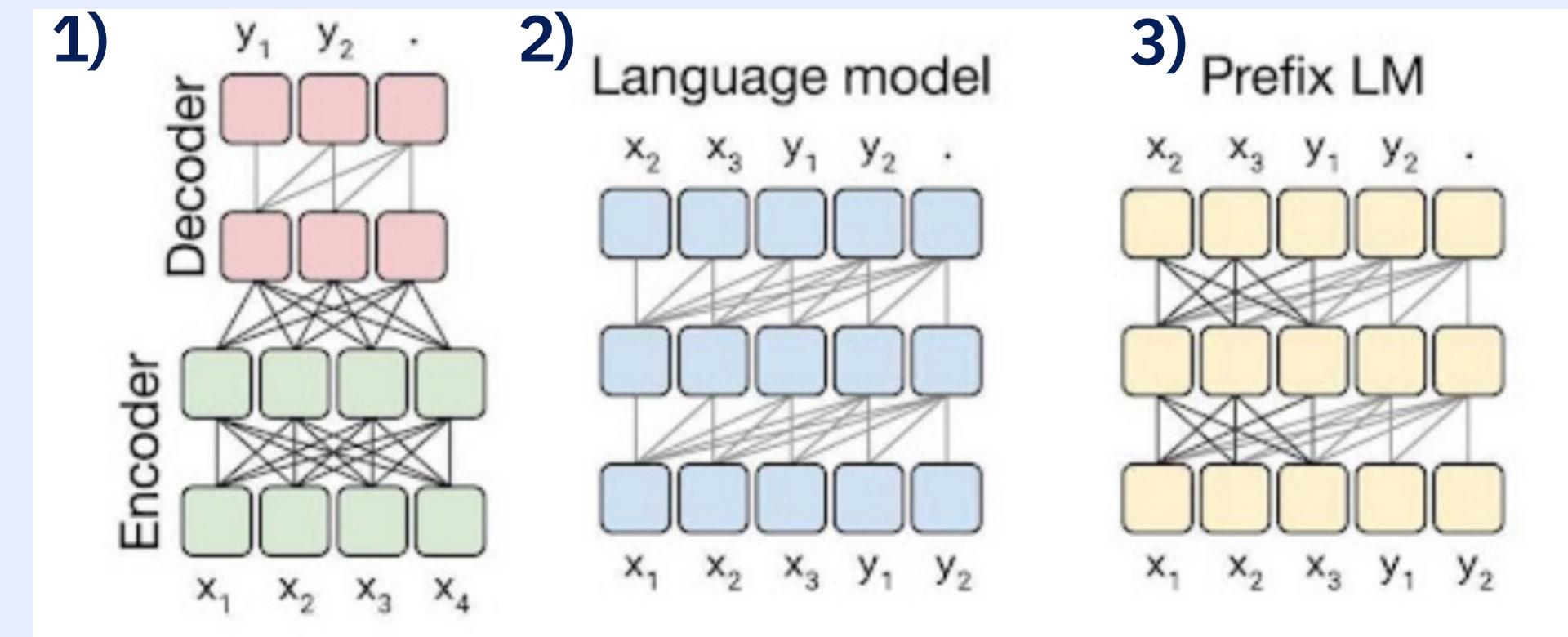


- 같은 모델, 손실 함수, 하이퍼파라미터 등을 여러가지 NLP task에 사용 가능
- C4(Colossal Clean Crawled Corpus)라는 대규모 사전 훈련 데이터셋 사용
- Pretraining objectives : noising된 input을 denoising하며 단어를 예측
- Pushing the limits : 110억 개의 파라미터를 갖는 모델을 훈련하여 SOTA(State of the Art) 달성

# 사용 기술 : T5

T5 구조

cakd3



**1) Encoder – Decoder 모델**

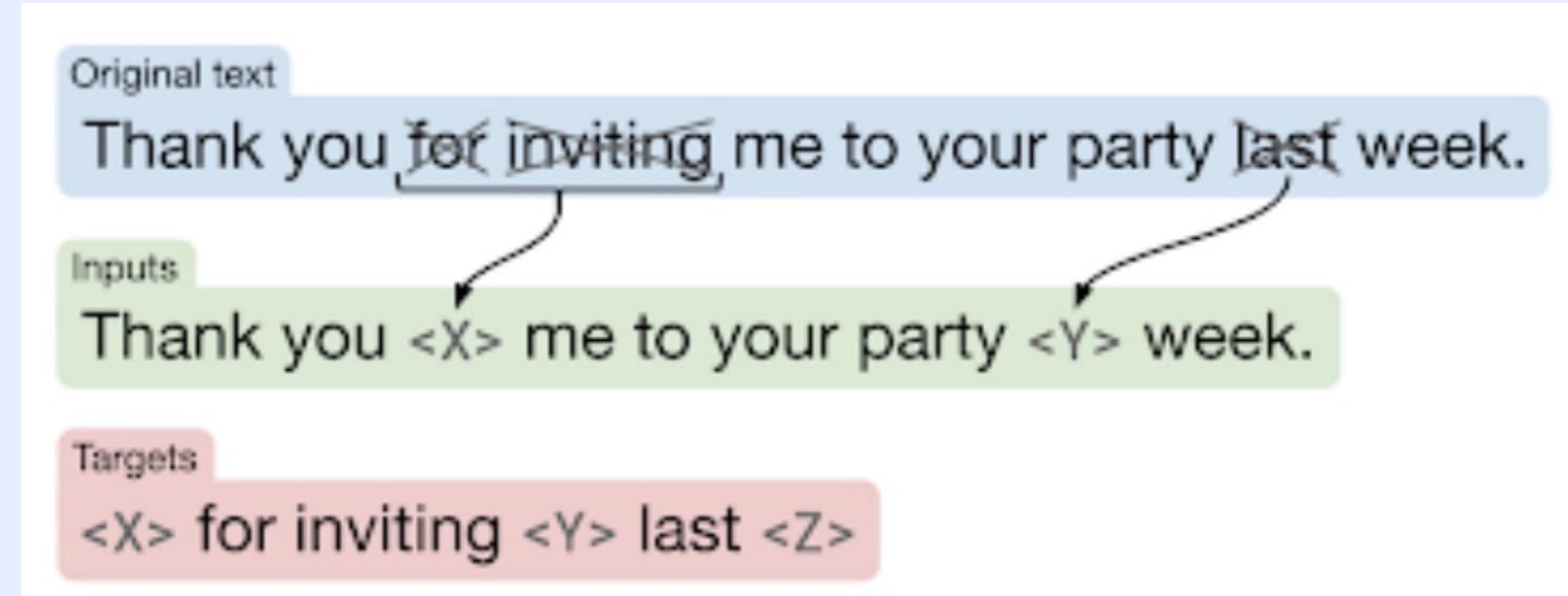
**2) Language model** : 자기회귀적으로 output sequence를 만든다.  
즉, 앞에 주어진 단어들을 통해 뒤에 올 단어를 예측한다.

**3) Prefix Language model** : source data(source text, target)은 bidirectional attention  
generated text는 unidirectional attention

04

# 사용 기술 : T5

T5 Flow

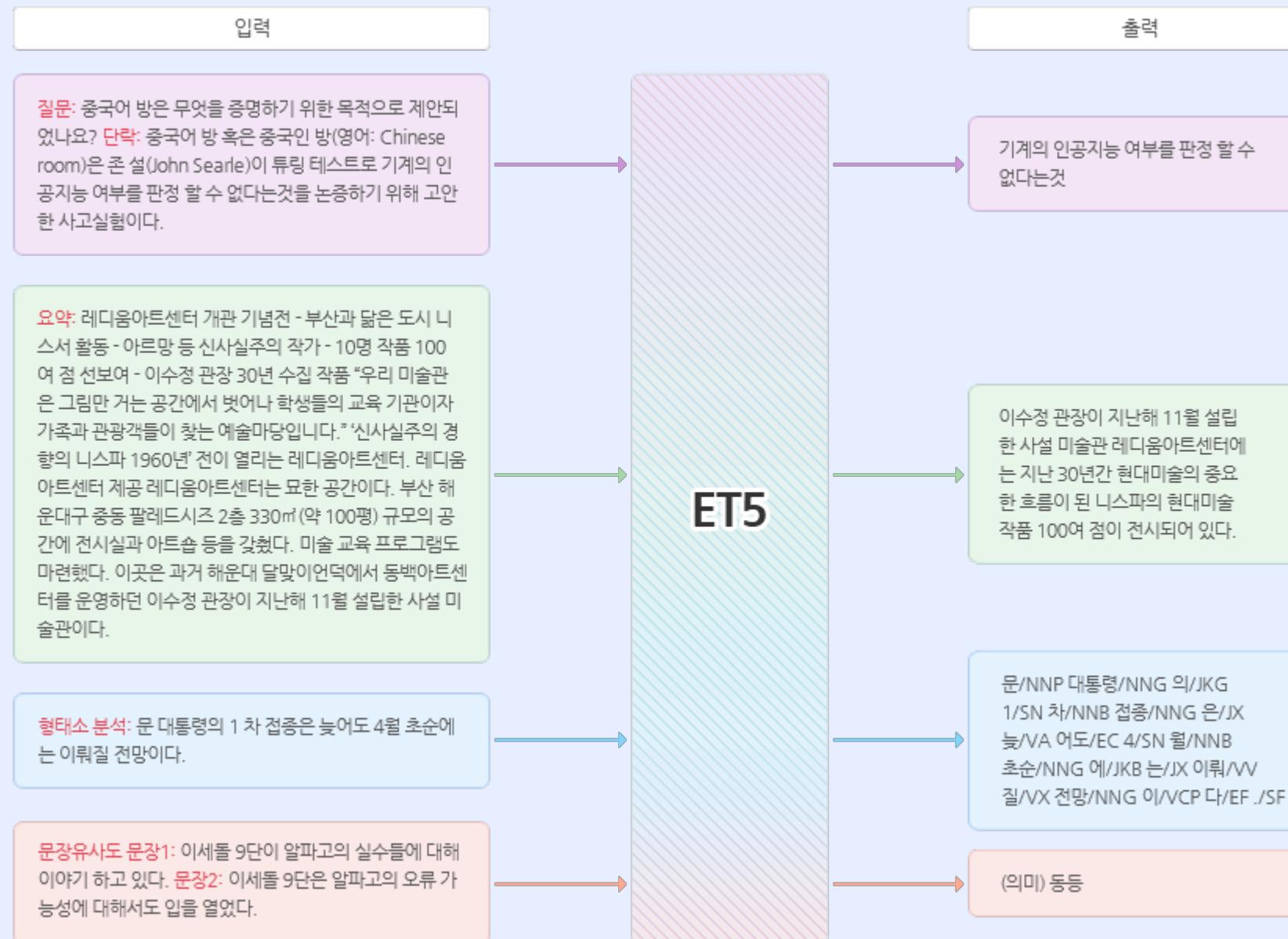


- BERT는 하나의 token에 masking을 하지만 T5 연속된 token을 하나의 mask로 바꿈 BART와 비슷
- Input에서 mask 되지 않은 부분을 target에서 맞춰야 함 MASS와 비슷
- Output level에서 FFNN + Softmax를 통해 시퀀스 생성

04

# 사용 기술 : ET5

## 한국어 이해생성 언어모델



- 대용량 원시 텍스트로부터 빠른 단어열 맞추기(T5 학습 유형)와 다음 단어 맞추기(GPT 학습 유형)를 동시에 사전학습(pre-train)하여 언어이해와 언어생성 능력을 향상시킨 모델
- 대표적인 한국어 처리 태스크 5종(기계독해, 요약, 단락 순위화, 형태소 분석, 문장유사도 추론) 대상 평가 결과 비슷한 한국어 모델 중 최고 수준의 성능을 보임.
- [질의응답] 기계 독해(Machine Reading Comprehension)
- [언어생성] 문서요약(Abstractive Summarization)
- [정보검색] 단락 순위화(Passage Ranking)
- [언어분석] 형태소 분석(Part Of Speech Tagging)
- [문장의미 분석] 문장 유사도 추론(Natural Language Inference)

04

# 사용 기술 : 미세조정

코드

```
1 import pandas as pd  
2 dataset_file = '/Training/training_df'  
3 df = pd.read_csv(dataset_file)  
4 df = df[['text', 'summary']]
```

1. 데이터셋을 DataFrame 형태로 불러오기

2. 설치해야하는 라이브러리 : sentencepiece, transformers, torch

사용한 라이브러리: sentencepiece, transformers, torch, numpy, pandas, os

```
# Setting up the device for GPU usage  
from torch import cuda  
device = 'cuda' if cuda.is_available() else 'cpu'
```

3. device 설정

04

# 사용 기술 : 미세조정

코드

4. 학습용 데이터인 Source Text(원본 내용)과 target(요약문)을 간단히 정제한 후 토큰화 한다.

```
class YourDataSetClass(Dataset):
    def __init__(self, dataframe, tokenizer, source_len, target_len, source_text, target_text):
        self.tokenizer = tokenizer
        self.data = dataframe
        self.source_len = source_len
        self.summ_len = target_len
        self.target_text = self.data[target_text]
        self.source_text = self.data[source_text]
```

```
def __getitem__(self, index):
    source_text = str(self.source_text[index])
    target_text = str(self.target_text[index])

    # 데이터 정제
    source_text = " ".join(source_text.split())
    target_text = " ".join(target_text.split())

    source = self.tokenizer.batch_encode_plus(
        [source_text],
        max_length=self.source_len,
        pad_to_max_length=True,
        truncation=True,
        padding="max_length",
        return_tensors="pt",
    )
    target = self.tokenizer.batch_encode_plus(
        [target_text],
        max_length=self.summ_len,
        pad_to_max_length=True,
        truncation=True,
        padding="max_length",
        return_tensors="pt",
    )
    source_ids = source["input_ids"].squeeze()
    source_mask = source["attention_mask"].squeeze()
    target_ids = target["input_ids"].squeeze()
    target_mask = target["attention_mask"].squeeze()
```

04

# 사용 기술 : 미세조정

코드

```
def train(epoch, tokenizer, model, device, loader, optimizer):

    model.train()
    for _, data in enumerate(loader, 0):
        y = data["target_ids"].to(device, dtype=torch.long)
        y_ids = y[:, :-1].contiguous()
        lm_labels = y[:, 1:].clone().detach()
        lm_labels[y[:, 1:] == tokenizer.pad_token_id] = -100
        ids = data["source_ids"].to(device, dtype=torch.long)
        mask = data["source_mask"].to(device, dtype=torch.long)

        outputs = model(
            input_ids=ids,
            attention_mask=mask,
            decoder_input_ids=y_ids,
            labels=lm_labels,
        )
        loss = outputs[0]

        optimizer.zero_grad()
        loss.backward()
        optimizer.step()
```

```
def validate(epoch, tokenizer, model, device, loader):

    model_path = ''
    model = torch.load(model_path + 'model.pt')
    model.eval()
    predictions = []
    actuals = []
    with torch.no_grad():
        for _, data in enumerate(loader, 0):
            y = data['target_ids'].to(device, dtype = torch.long)
            ids = data['source_ids'].to(device, dtype = torch.long)
            mask = data['source_mask'].to(device, dtype = torch.long)

            generated_ids = model.generate(
                input_ids = ids,
                attention_mask = mask,
                max_length=150,
                num_beams=2,
                repetition_penalty=2.5,
                length_penalty=1.0,
                early_stopping=True
            )
            preds = [tokenizer.decode(g, skip_special_tokens=True, clean_up_tokenization_spaces=True) for g in generated_ids]
            target = [tokenizer.decode(t, skip_special_tokens=True, clean_up_tokenization_spaces=True)for t in y]
            predictions.extend(preds)
            actuals.extend(target)
    return predictions, actuals
```

## 6. validation

**preds** : 학습한 model을 통해 생성된 요약문

**target** : 기존의 요약문을 noiseing → denoising하여 만든 요약문

## 5. Train

→ bert\_score와 같은 비교 평가를 위해 사용된다.

04

# 사용 기술 : 미세조정

코드

## 7. def T5Trainer():

Train과 validation을 위한 Datasets을 나누고 DataLoader를 만든다.

```
def T5Trainer(
    dataframe, source_text, target_text, model_params, output_dir="./outputs/"
):
    model_path = '축적해서 학습한 model.pt의 경로'
    token_path = './ETRI_ET5/'
    torch.manual_seed(model_params["SEED"]) # pytorch random seed
    np.random.seed(model_params["SEED"]) # numpy random seed
    torch.backends.cudnn.deterministic = True

    tokenizer = T5Tokenizer.from_pretrained(token_path)

    model = torch.load(model_path + 'model.pt')
    model = model.to(device)

    # raw dataset을 importing
    dataframe = dataframe[[source_text, target_text]]
    display_df(dataframe.head(2))

    # Dataset 및 Dataloader 만들고 train, val split
    train_size = 0.8
    train_dataset = dataframe.sample(frac=train_size, random_state=model_params["SEED"])
    val_dataset = dataframe.drop(train_dataset.index).reset_index(drop=True)
    train_dataset = train_dataset.reset_index(drop=True)

    # testing과 validation을 위한 dataloader 생성
    training_loader = DataLoader(training_set, **train_params)
    val_loader = DataLoader(val_set, **val_params)

    optimizer = torch.optim.Adam(
        params=model.parameters(), lr=model_params["LEARNING_RATE"]
    )
```

```
# Train, Test datasets 만들기
training_set = YourDataSetClass(
    train_dataset,
    tokenizer,
    model_params["MAX_SOURCE_TEXT_LENGTH"],
    model_params["MAX_TARGET_TEXT_LENGTH"],
    source_text,
    target_text,
)
val_set = YourDataSetClass(
    val_dataset,
    tokenizer,
    model_params["MAX_SOURCE_TEXT_LENGTH"],
    model_params["MAX_TARGET_TEXT_LENGTH"],
    source_text,
    target_text,
)

train_params = {
    "batch_size": model_params["TRAIN_BATCH_SIZE"],
    "shuffle": True,
    "num_workers": 0,
}

val_params = {
    "batch_size": model_params["VALID_BATCH_SIZE"],
    "shuffle": False,
    "num_workers": 0,
}
```

04

# 사용 기술 : 미세조정

코드

8. 학습시킨 모델을 저장하여 validation을 진행하고  
모델이 학습을 통해 만든 요약문 모음을 csv 파일로 저장한다.

```
#Train
for epoch in range(model_params["TRAIN_EPOCHS"]):
    train(epoch, tokenizer, model, device, training_loader, optimizer)

#학습한 model 및 tokenizer 저장
path = os.path.join(output_dir, "model_files")
model.save_pretrained(path)
tokenizer.save_pretrained(path)

# Validation
for epoch in range(model_params["VAL_EPOCHS"]):
    predictions, actuals = validate(epoch, tokenizer, model, device, val_loader)
    final_df = pd.DataFrame({"Generated Text": predictions, "Actual Text": actuals})
    final_df.to_csv(os.path.join(output_dir, "predictions.csv"))
```

```
df["text"] = "summarize: " + df["text"]

T5Trainer(
    dataframe=df,
    source_text="text",
    target_text="summary",
    model_params=model_params,
    output_dir="results"
)
```

04

# 사용 기술 : 미세조정

모델 파라미터 및 학습 방식

## 모델 학습 파라미터

Training Batch Size	2
Valid Batch Size	2
Train Epochs	3
Valid Epochs	1
Learning Rate	1e-4
Max Source Text Length	800
Max Target Text Length	200
Seed	42

## 모델 생성 요약문 파라미터

num_beams	2
no_repeat_ngram_size	3
min_length	30
max_length	250
repetition_penalty	2.5
length_penalty	1.0
early_stopping	True

## 학습 방식

기술과학 6000개



사회과학 6000개



예술 6000개



기타 3000개



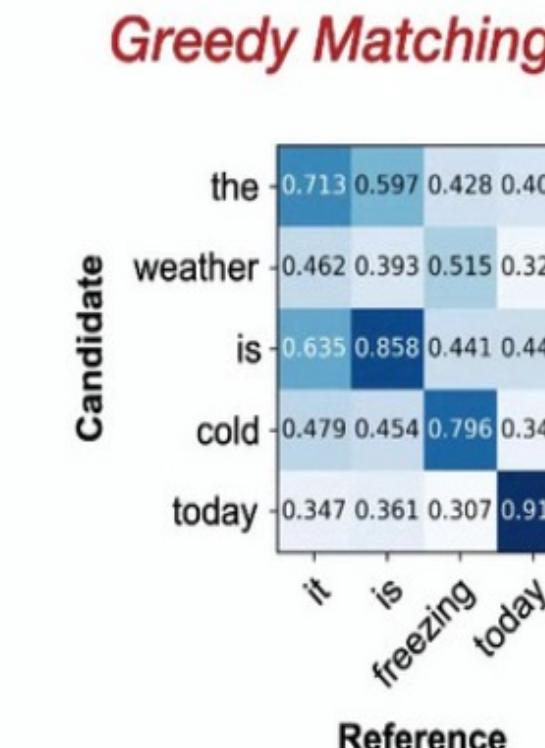
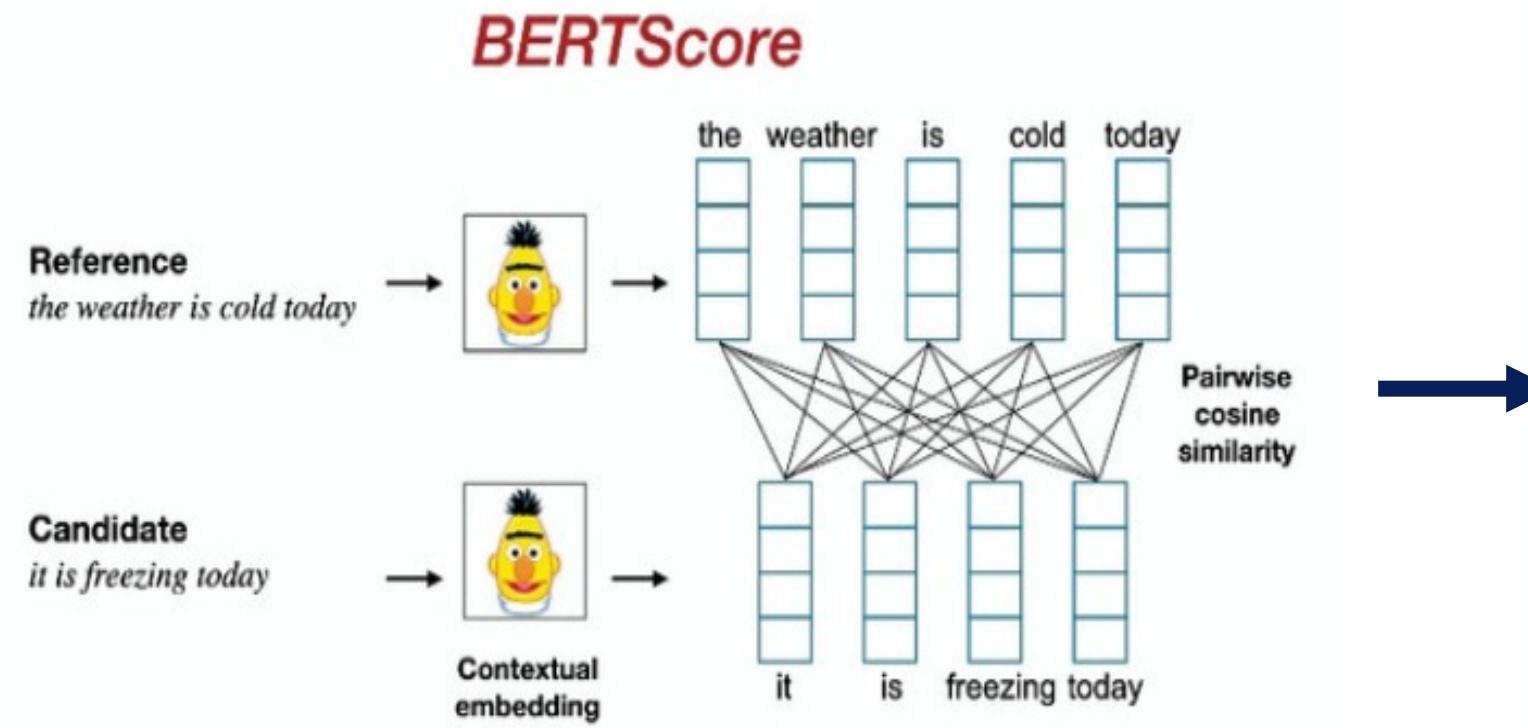
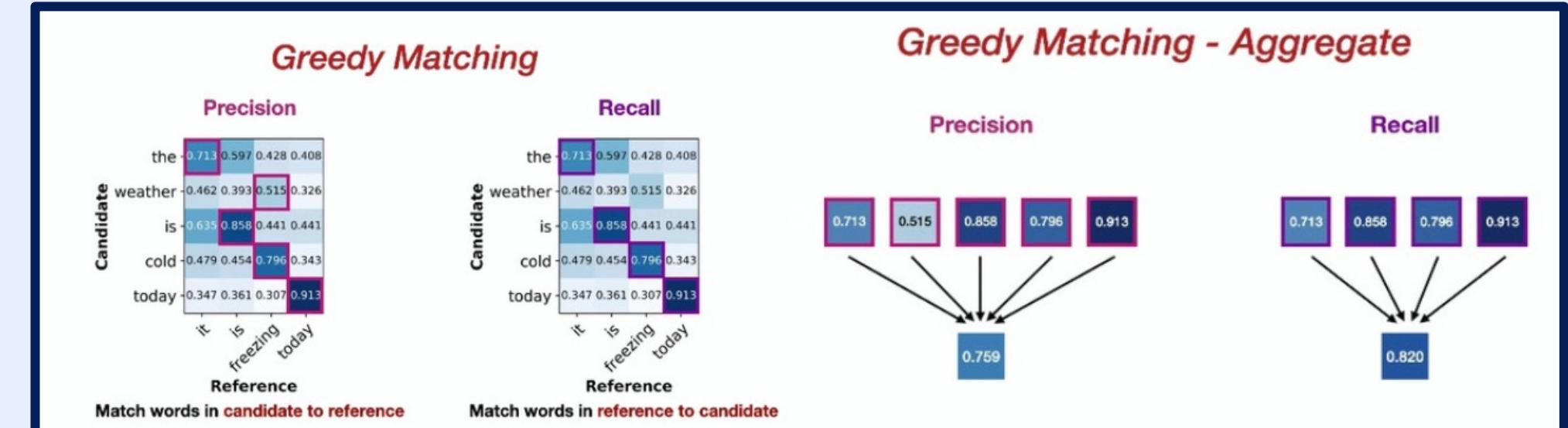
총 21000개 데이터 축적 학습

04

# 사용 기술 : BERTScore

## 모델 평가

기존 문장 요약 평가 방법으로 많이 쓰이는 ROUGE는 exact match로 평가해 문맥을 평가하지 못한다. 반면에 BERTScore는 문장간의 문맥을 평가할 수 있다.

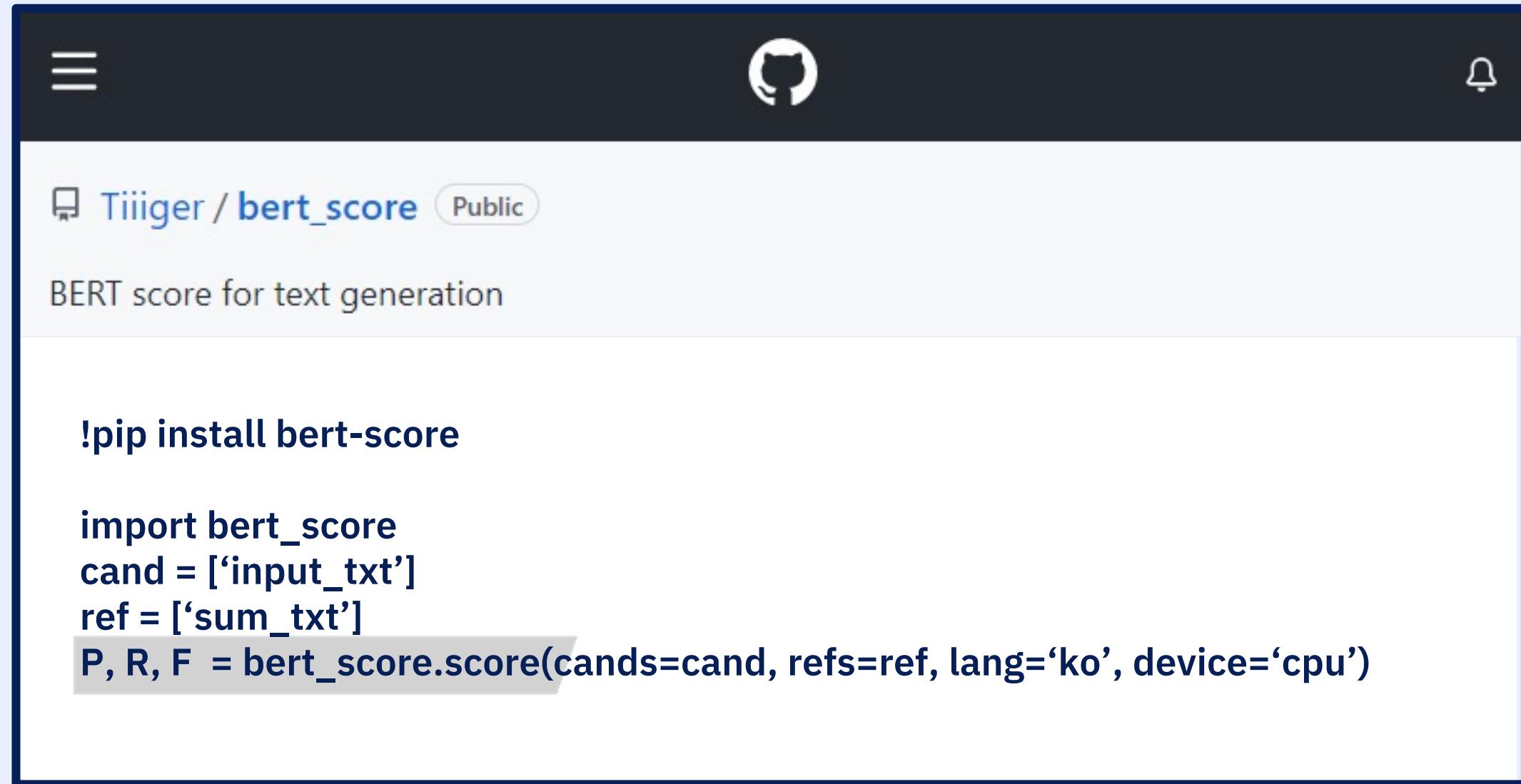


**Precision**  
**Recall**  
**F1 score**( $F1 = 2 \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$ )

04

# 사용 기술 : BERTScore

평가 결과



The screenshot shows a GitHub repository page for 'Tiiiger/bert\_score'. The repository is public and describes itself as 'BERT score for text generation'. The README contains the following code examples:

```
!pip install bert-score

import bert_score
cand = ['input_txt']
ref = ['sum_txt']
P, R, F = bert_score.score(cands=cand, refs=ref, lang='ko', device='cpu')
```

Bert-Score  
최종 0.8391

Fine tuning  
T5Trainer로 진행  
기술과학 6000/사회과학 6000/ 예술 6000/ 기타 3000  
데이터 이용

# 05

---

## 서비스 제작

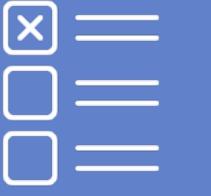
05

# 서비스 제작



## 시나리오 설계

- 프로그램 전체적인 시나리오 구성



## 데이터 구축

- 전처리된 DB 생성



## Fine tuning

- 데이터를 이용한 문서 요약 학습



## 웹 서비스 구현

- CSS / HTML
- Figma
- Flask

05

# 서비스 제작

웹 구현

주요 이용 툴과 언어



## 웹 구현(HTML/CSS)

HTML: 문서와 문서를 연결시키는 마크업 언어, CSS: HTML로 만들어진 문서를 꾸며주는 역할  
의미있는 구조화된 문서 작성 후, 시각적인 효과로 웹페이지 구현



## Figma

웹 브라우저 기반의 디자인 툴, 직관적인 UI/UX로 간단하게 프로토타입 제작 가능  
라이브러리 개념이 있어 체계적인 작업 가능(동시작업)



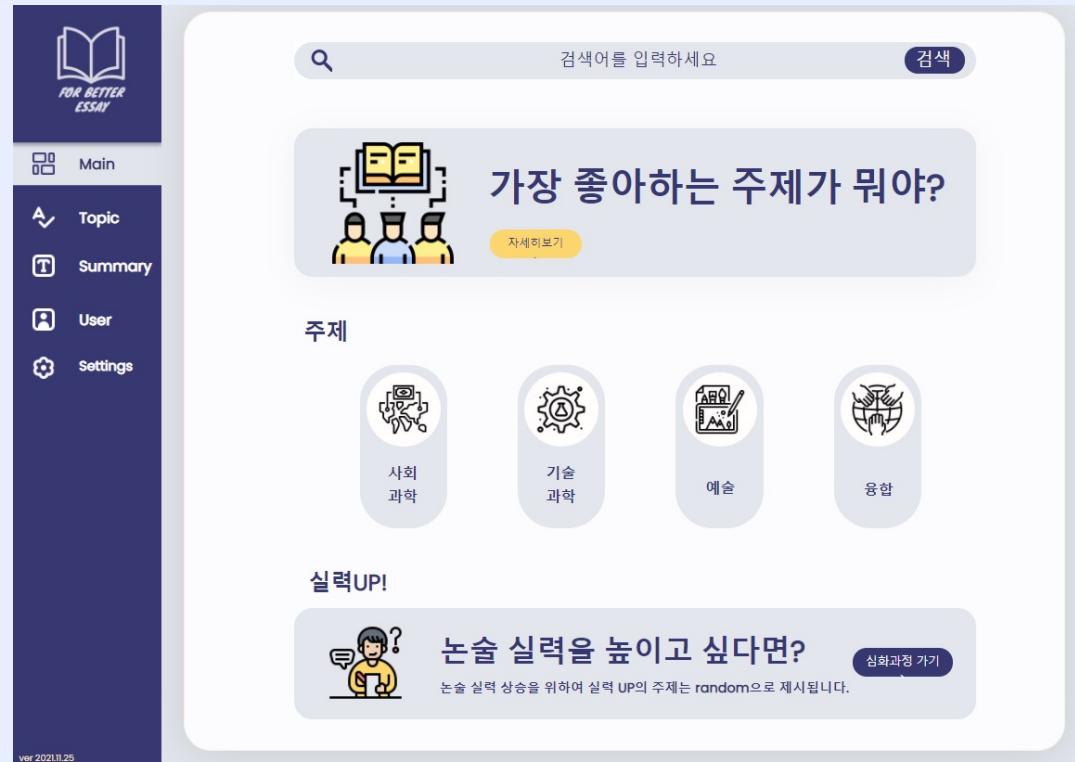
## Flask

웹 어플리케이션 개발을 위한 프레임워크, Django는 무겁고 기능이 많아서 가벼운 Flask 사용  
ngrok을 통해 로컬과 외부 연결

# 05 서비스 제작

## 웹 구현 (대시보드)

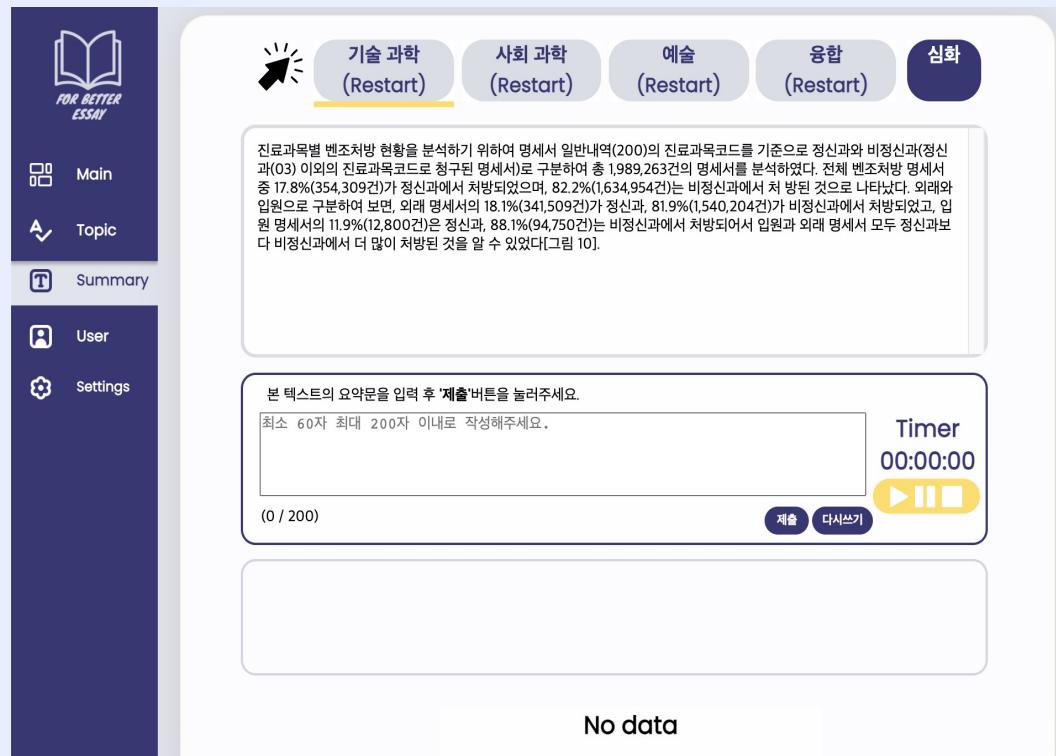
✓ 첫페이지  
- Main



- Topic



- Summary



- User



- Settings

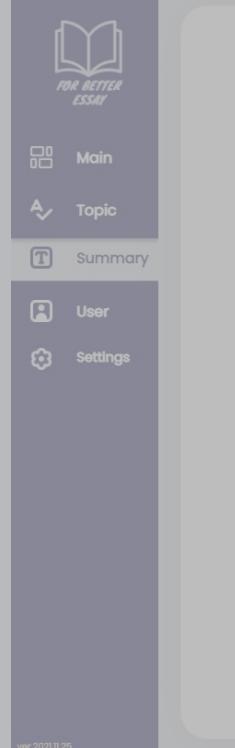


05

## 서비스

웹 구현 (대시보드)

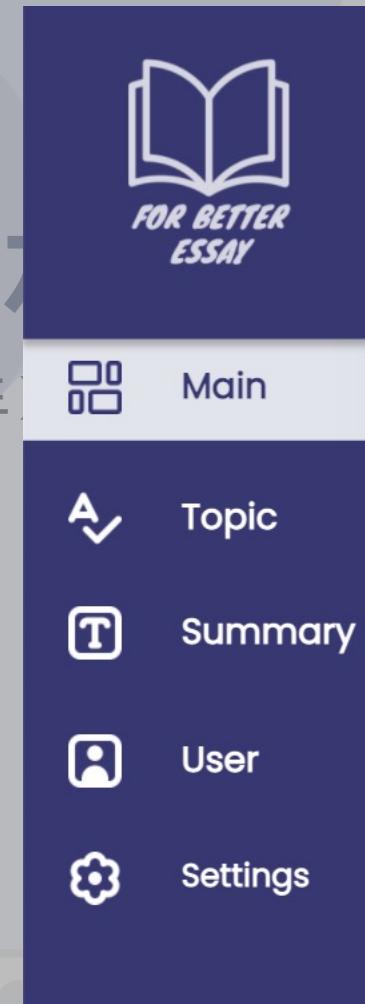
- Summary



4가지 주제 및  
심화 글쓰기  
선택 가능

✓ 첫페이지  
- Main

- Topic



주제



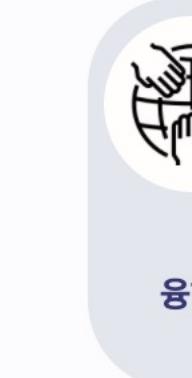
사회  
과학



기술  
과학



예술



융합

실력UP!



논술 실력을 높이고 싶다면?

논술 실력 상승을 위하여 실력 UP의 주제는 random으로 제시됩니다.

심화과정 가기 >

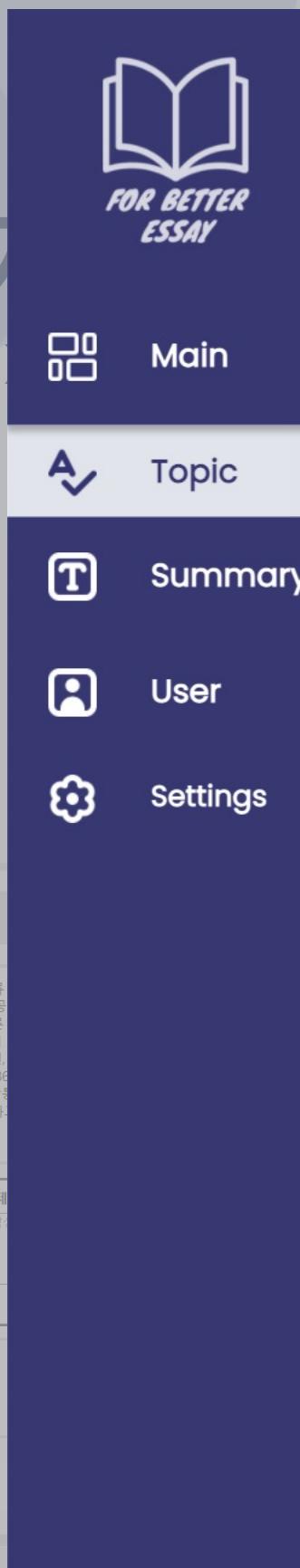




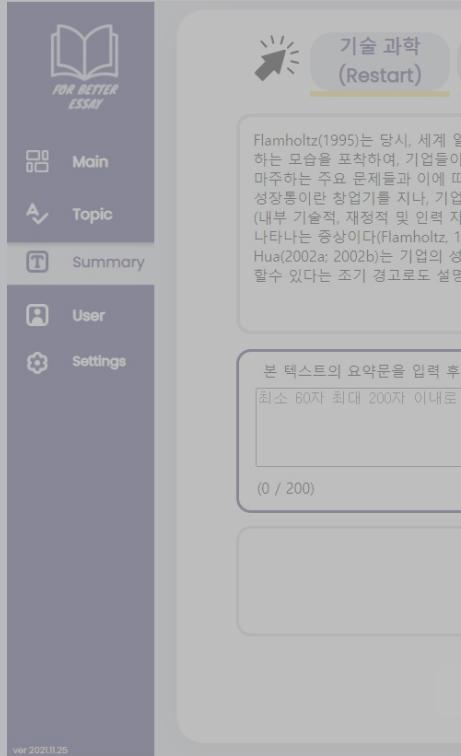
05

## 서비스

웹 구현 (대시보드)



## - Summary



✓ 첫페이지  
- Main

- Topic

검색어를 입력하세요

검색

실력UP!

논술 실력 상승을 위하여 실력 UP의 주제는 random 으로 제시됩니다.

주제가 랜덤으로 나오는 이유는 뭔가요?

- 다양한 주제의 요약을 진행함으로써 여러가지 주제의 글의 문맥과 핵심을 쉽게 파악 할 수 있습니다.
- 다양한 주제의 글을 요약해봄으로써 한 분야에 편향되지 않은 문해력을 향상시킬 수 있습니다.

랜덤 주제  
심화  
글쓰기

글쓰기 시작

05

# 서비스

웹 구현 (대시보드)

## - Summary

FOR BETTER ESSAY

Main

Topic

Summary

User

Settings

Flamholtz(1995)는 당시, 세계 일류하는 모습을 포착하여, 기업들이 공모주하는 주요 문제들과 이에 따른 성장통이란 창업기를 지나, 기업이 (내부 기술적, 재정적 및 인력 자원) 나타나는 증상이다(Flamholtz, 1986). Hua(2002a, 2002b)는 기업의 성장통 할 수 있다는 조기 경고로도 설명하고 있다.

본 텍스트의 요약문을 입력 후 '제출'버튼을 눌러주세요.

최소 60자 최대 200자 이내로 작성해주세요.

(0 / 200)

ver 2021.11.25

✓ 첫페이지 - Main - Topic

기술 과학 (Restart)

사회 과학 (Restart)

예술 (Restart)

융합 (Restart)

심화

서정시인은 그래서 자아가 안식할 수 있는 공간을 상상력을 통해 환각으로 만들어낸다. 서정시가 그리는 낙원이 바로 그것이다. 이러한 낙원은 시인마다 다른 모습으로 나타나기도 하지만, 대부분의 경우 '고향'이나 '자연'으로 그려진다. 고향은 시인이 경험한 가장 아름다운 시절의 상징이며, 자연은 인간 존재의 본질적 근원으로서의 의미를 지니고 있기 때문이다. 다시 말해 고향은 어른이 된 자아가 경험하는 현실의 고통과 아픔을 보상해 줄 수 있는 원체험으로 작용하며, 자연은 현대의 도시 공간에서 부대끼는 삶을 초극할 수 있는 안식이 공간이 되는 것이다. 주근육 시인의 시세계를 지탱하는 중요한 축 또한 여기에 있다. 그의 시는 첫 시집 '산노을 등에 지고'에서부터 최근의 시집 '갈대 속의 비비새'에 이르기까지 이러한 이상향으로서의 '고향'과 '자연'이 매우 중요한 요소로 형상화되어 있다. 초기 시에서는 힘겹고 고통스러운 삶의 현실을 뛰어넘을 수 있는 공간으로서의 '자연'이 주로 형상화되어 있는데, 시인은 그러한 자연과 동화되고자 하는 노력을 쉬지 않는다. 이러한 자연에의 동화 노력은 그러나 현실의 힘겨운 삶의 무게에 눌려 끊임없이 방해받는데, 이러한 방해는 시인으로 하여금 '고향'에 대한 추구로 발전해 가게 만든다. 현대적인 삶의 방식 때문에 근원으로서의 자연에 도달할 수 없을 때 시인은 어린 시절의 '고향'으로의 회귀를 감행하는 것이다. 그러나 그러한 회귀가 자아에게 완전한 안식을 허용하는 절대적 공간에 이르도록 만들

본 텍스트의 요약문을 입력 후 '제출'버튼을 눌러주세요.

최소 60자 최대 200자 이내로 작성해주세요.

(0 / 200)

제출

다시쓰기

Timer  
00:00:00

No data

입력하세요 검색

자주 출제됐던 주제

가장 흔히 출제되는 핵심 출제어

모의논술고사 자료집

및 보안

언어 및 지역

정책 쿠키 정책

## 주제 선택 탭, 클릭시 주제별 랜덤 문단 출력

기술 과학  
(Restart)

진료과목별 벤조처방 현황을 분석하기 위하여 명세서 일반내역(200)의 진료과목코드를 기준으로 정신과와 비정신과(정신과(03) 이외의 진료과목코드로 청구된 명세서)로 구분하여 총 1,989,263건의 명세서를 분석하였다. 전체 벤조처방 명세서 중 17.8%(354,309건)가 정신과에서 처방되었으며, 82.2%(1,634,954건)는 비정신과에서 처방된 것으로 나타났다. 외래 입원으로 구분하여 보면, 외래 명세서의 18.1%(341,509건)가 정신과, 81.9%(1,540,204건)가 비정신과에서 처방되었고, 원 명세서의 11.9%(12,800건)은 정신과, 88.1%(94,750건)는 비정신과에서 처방되어서 입원과 외래 명세서 모두 정신과에 비정신과에서 더 많이 처방된 것을 알 수 있었다[그림 10].

# 주제별 출력된 랜덤 문단

본 텍스트의 요약문을 입력 후 '제출'버튼을 눌러주세요

최소 60자 최대 200자 이내로 작성해주세요

## 사용자 요약문 작성란

(0 / 200)

Timer  
00:00:0

제출 다시쓰기

기계 요약문 출판

# 제출 후 평가 스코어 출력

No date

- <평가 기준>
- 0.8 이상 Perfect
- 0.6 이상 0.8 미만 Great
- 0.4 이상 0.6 미만 good
- 0.4 미만 Try again

# 서비스

웹 구현 (대시보드)

메뉴탭

-  Main
-  Topic
-  Summary
-  User
-  Settings

## - Summary

The screenshot shows the 'Restart' section of the 'Technology and Science' app. At the top right, there is a yellow button labeled '기술 과학 (Restart)'. Below it is a large icon of a computer monitor with a brain-like circuit board on the screen. The main content area contains a text box with the following text:

Flamholtz(1995)는 당시, 세계 열린 기업으로서의 이미지를 확립하는 모습을 포착하여, 기업들이 주로 중시하는 주요 문제들과 이에 따른 성장통이란 창업기를 지나, 기업 내부 기술적, 재정적 및 인력 자원에 나타나는 증상이다(Flamholtz, 1995; Hua(2002a; 2002b)는 기업의 성장을 할 수 있다는 조기 경고로도 설명

Below the text box is a smaller input field with the placeholder text '분 텍스트의 요약문을 입력 후 최소 60자 최대 200자 이내로' and '(0 / 200)' below it. On the left side of the screen, there is a vertical sidebar with icons and labels: 'Main' (document icon), 'Topic' (arrow icon), 'Summary' (document icon), 'User' (person icon), and 'Settings' (gear icon). The bottom left corner of the sidebar has the text 'ver 2021.11.25'.

# 타이머 (시작, 중지, 리셋)

05

# 서비스 제작

웹 구현 (시연 영상)

The screenshot shows a web browser window with multiple tabs open. The active tab displays a web application interface.

**Left Sidebar:**

- Main
- Topic
- Summary
- User
- Settings

**Top Bar:**

- 내 드라이브 - Google Drive
- 1208\_flask\_ngrok구현.ipynb
- Document
- Your Authtoken - ngrok
- YouTube
- 데모영상 - Google Drive

**Content Area:**

기계 되며 약제의 승인과정에 있어서 인허가의 의사결정에 대한 시비가 를 수 있다. 따라서 이런 의사결정을 하는 조직이나 위원회는 보다 객관적으로 치료법의 득과 실을 판단하는 방법이 필요하며 이에 대하여는 많은 연구들이 이루어져 왔으나 아직 어떤 방법이 최선의 방법인지 국제적으로 합의된 부분은 부족하다. 하지만 득과 실에 대한 trade off는 의사결정이나 권고사항 결정과정에서 피할 수 없는 부분이며 다양한 이해당사자들이 존재하는 보건의료분야의 특성상 투명하고 객관화 시키는 노력을 기울이지 않을 수 없다. 이상적인 방법은 CHMP에서 제시한 바와 같이 가장 중요한 이득과 가장 심각한 위험을 쉽게 확인하게 해주는 것, 개별 이득과 위험에 대한 명백한 가중치를 부여하는 것, 근거의 강도와 확인된 불확실성에 대해 제시하는 것이며 이러한 원칙하에 각 치료법이 갖는 부차적인 이득과 위해를 종합적으로 반영하고 종합하여 결론을 내릴 수 있다면 좋겠지만 많은 변수를 고려할수록 방법론이 복잡해지고 실제로 사용하기에 어려울 수도 있다.

본 텍스트의 요약문을 입력 후 '제출'버튼을 눌러주세요.

최소 60자 최대 200자 이내로 작성해주세요.

(200 / 200)

Timer  
00:00:00

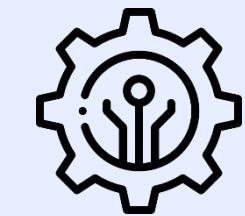
제출      다시쓰기

**Bottom Note:**

1. 글자 수 제한 기능 시연 영상.

# 서비스 제작

개선계획



## 기술

- 모델의 정확도 향상
- 고객서비스 속도 개선



## 서비스

- 주제 카테고리 다양화·세분화
- 나이도 세분화(타겟의 다양성)
- 웹과 앱 구현 및 연동 = 접근성 향상
- 학습 기록을 볼 수 있는 페이지 추가

# 06

---

## 기대효과



## 1. 실질적 문해력 상승

실질적인 문해력 상승으로  
언어 뿐만 아니라 다양한 분야의 학습 능력 향상  
기대

## 2. 자기주도적 학습 가능

AI를 이용한 학습 프로그램으로 따로 선생님을  
만나는 시간을 낼 필요 없이 원하는 때에 원하는  
만큼 자기주도적인 학습이 가능

## 3. 논술 및 글쓰기 실력 향상

핵심 문장을 찾는 연습을 하며 독해 능력 뿐만 아  
니라 문장을 구성하는 능력도 향상 될 것을 기대

## 4. 정보 신뢰성 판단력 향상

문해력이 높아지면서 주어진 정보에 대한 신뢰도  
를 판단 할 수 있는 능력이  
향상 될 것으로 기대

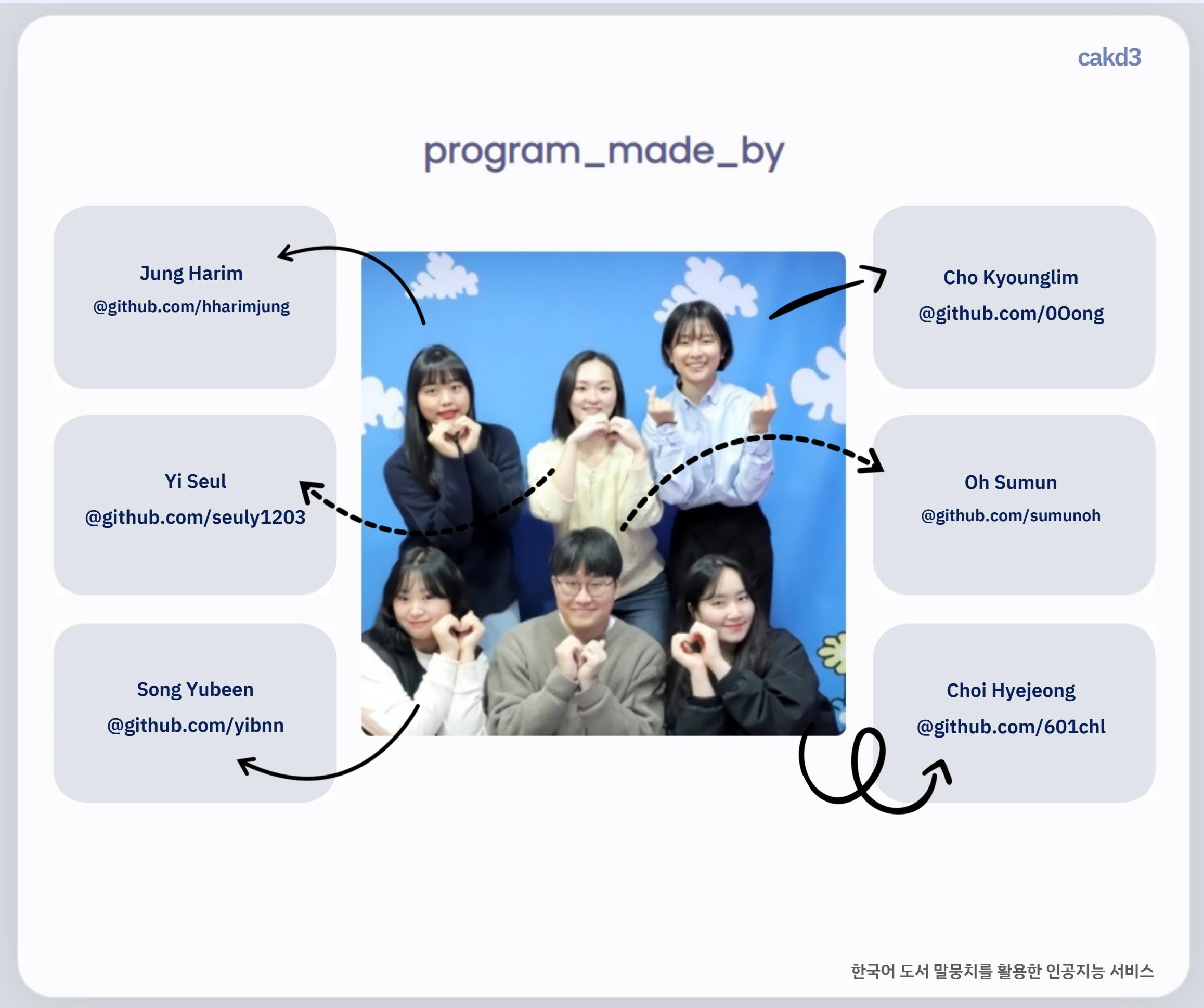
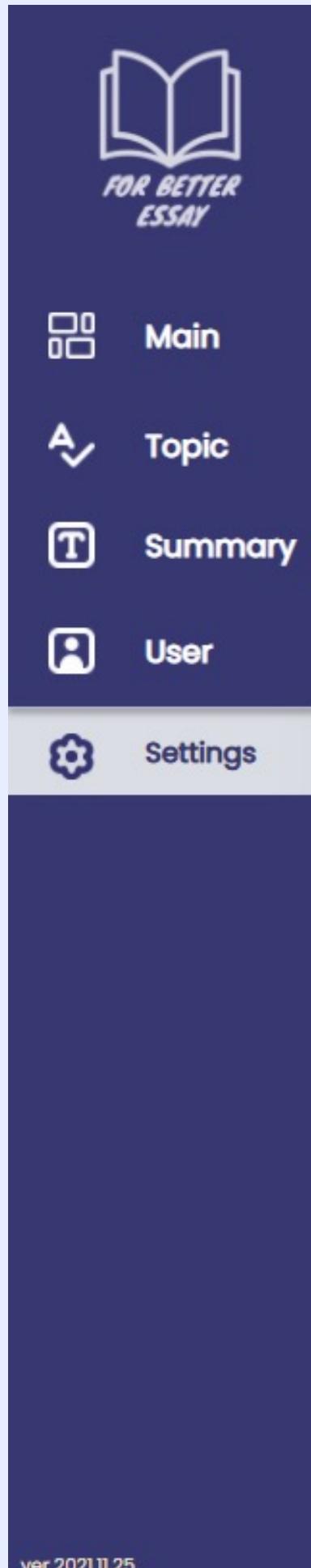
# Our Team

팀원 소개

## Contact us

Questions? Concerns?  
Feedback? Let us know, we're  
here to help.

48





cakd3

2조 | 송유빈, 이슬, 오수문, 정하림, 조경림, 최혜정

# Thank you