

AI로 영어 공부하기

# Translation Analysis

# 결과

## 웹페이지에 들어갔을 때

- 오류 내용 출력: 번역된 문장에서의 오류 (예: 의미, 구문, 단어 선택 오류 등).
- 영어 문장에서 사용된 문법 규칙: 문장에서 사용된 문법 용어와 그 설명.
- 의미와 스타일 개선을 위한 제안: 더 자연스럽게나 명확한 표현 제안.
- 수정된 한국어 번역: 수정된 번역이 있다면 제공.

- 
- + Google Generative AI
  - + Django
  - + JSON 및 HTTP 요청 처리

### AI NLP Translation Analysis

생성된 영어 문장:

The quick brown fox jumps over the lazy dog.

한국어 번역 입력:

한국어 번역을 입력하세요

번역 분석

초기화

# 번역 과정



# 시스템

## 프론트엔드

영어 문장 표시, 번역 입력 폼 제공, 분석 결과 출력.

## 백엔드

클라이언트 요청 처리, 영어 문장 생성 및 분석 로직,  
Google Generative AI와 통신, 분석 결과 파싱.

## GOOGLE GENERATIVE AI

자연어 문장 생성 및 번역 분석.

## 데이터 처리 및 세션 관리

영어 문장 및 번역 데이터 세션에 저장

## 결과 포매팅 및 시각화

AI 응답 포매팅, 분석 결과 구조화,  
클라이언트에 JSON 형식으로 전달.

```
# 분석 텍스트를 섹션별로 파싱하는 함수
def parse_and_format_analysis(analysis_text):
    sections = {
        "오류 내용 출력": "",
        "영어 문장에서 사용된 문법 규칙": "",
        "의미와 스타일 개선을 위한 제안": "",
        "수정된 한국어 번역": "",
    }

    analysis_text = re.sub(r"^[^a-zA-Z가-힣0-9\s:()\*\`\`\",,]+", "", analysis_text)
    section_pattern = re.compile(r"^\s*(.+?):\s*$")

    current_section = None
    for line in analysis_text.split("\n"):
        line = line.strip()

        match = section_pattern.match(line)
        if match:
            section_title = match.group(1).strip()
            content = line[match.end():].strip()
            if section_title in sections:
                current_section = section_title
                sections[current_section] = content
            else:
                current_section = None
        elif current_section:
            sections[current_section] += " " + line.strip()

    for section in sections:
        sections[section] = re.sub(r'["', '\'', sections[section]).strip()

    # Formatting the "영어 문장에서 사용된 문법 규칙" section for better readability
    if sections["영어 문장에서 사용된 문법 규칙"]:
        grammar_rules = re.findall(r"([^\s:]+): ([^\s]+\s+)", sections["영어 문장에서 사용된 문법 규칙"])
        formatted_rules = "<ul>"
        for term, description, details in grammar_rules:
            formatted_rules += f"<li><strong>{term.strip()}</strong>: {description.strip()}"
        formatted_rules += "</ul>"
        sections["영어 문장에서 사용된 문법 규칙"] = formatted_rules

    return sections
```

# 문제점 해결

- chat GPT는 유료 API를 가진다.
- Hugging Face로 시도했지만 실패

TO BE

- Google Generative AI api를 사용해 해결

- AI 응답의 일관성 부족 문제가 있었다.
- 프롬프트로 줬던 형식과 다른 형식의 응답이 있었다.

TO BE

- 프롬프트를 여러번 조정해 해결

- ipynb파일로 보기 불편함

TO BE

- Django를 이용해 인터페이스 개선

# 주제 바꾸게 된 계기

번역 모델 개선 -> ai가 내가 번역한 것에 대한 피드백

## Before

original	translated	
I can't wait to do this.	빨리 해보고 싶어요.	The Korean translation "빨
ch that with Korean subtitles.	당신은 한글자막으로 보게 됩니다.	The Korean translation "당신은 한
So, you hear words.	그러면, 말이 들립니다.	The translation "그러면, 말이
ootball players do that 'goal'.	축구 선수들은 그것을 '골'이라고 합니다.	The translation "축구 선수들은 그
earing is so hard because s...	심지어 스페인인들도 한국어처럼 들리는 부분이 있어서 듣기가 아주 어렵습니다.	The translation is
...	...	...
as helpful for you to under...	이 영상이 이 드라마 이면에 담긴 한국적 맥락과 사회적 맥락을 이해하는 데 도움이 ...	Error: 429 Resour
re questions or if you want ...	더 궁금한 사항이 있거나 당신의 견해를 공유하고 싶다면 댓글을 달아주십시오.	Error: 429 Resour
o this video, I recommend g...	그리고 또한 이 영상이 마음에 드셨다면 이태원 클라쓰 자세히 살펴보기 영상으로 돌아...	Error: 429 Resour
een a hit on the top charts...	화사의 마리아가 한동안 차트에서 1위를 달리고 있습니다.	Error: 429 Resour
us the fact that participated...	그리고 그녀가 작사, 작곡, 뮤직비디오의 스토리 메이킹과 스타일링까지 직접 참여했다...	Error: 429 Resour

+ ai에게 요청 했을 때 결과가 좋지 않았다.

+ 할당량이 초과 하면 할 수 없다는 문제가 있었다.

## After

The quick brown fox jumped over the lazy dog.

[오류 내용 출력]  
단어 선택 오류 원은 동작을 나타내는 명사이지만, 영어 문장에서는 동작을 나타내는 동사 jumps가 사용되었습니다.

[영어 문장에서 사용된 문법 규칙]  
the: 특정 명사를 가리킴 (정관사) fox and dog: 명사 (사물이나 생물을 나타냄) jumped: 과거형 동사 (과거의 동작이)

[의미와 스타일 개선을 위한 제안]  
뛰 대신 뛰었다를 사용하여 동사로 표현

[수정된 한국어 번역]  
개으른 개 위로 갈색 여우가 뛰었다

+ 시에 동일한 양식으로 요청하면 일정한 데이터를 추출할 수 있었다.

+ 할당량에 대한 부담감이 적다.