

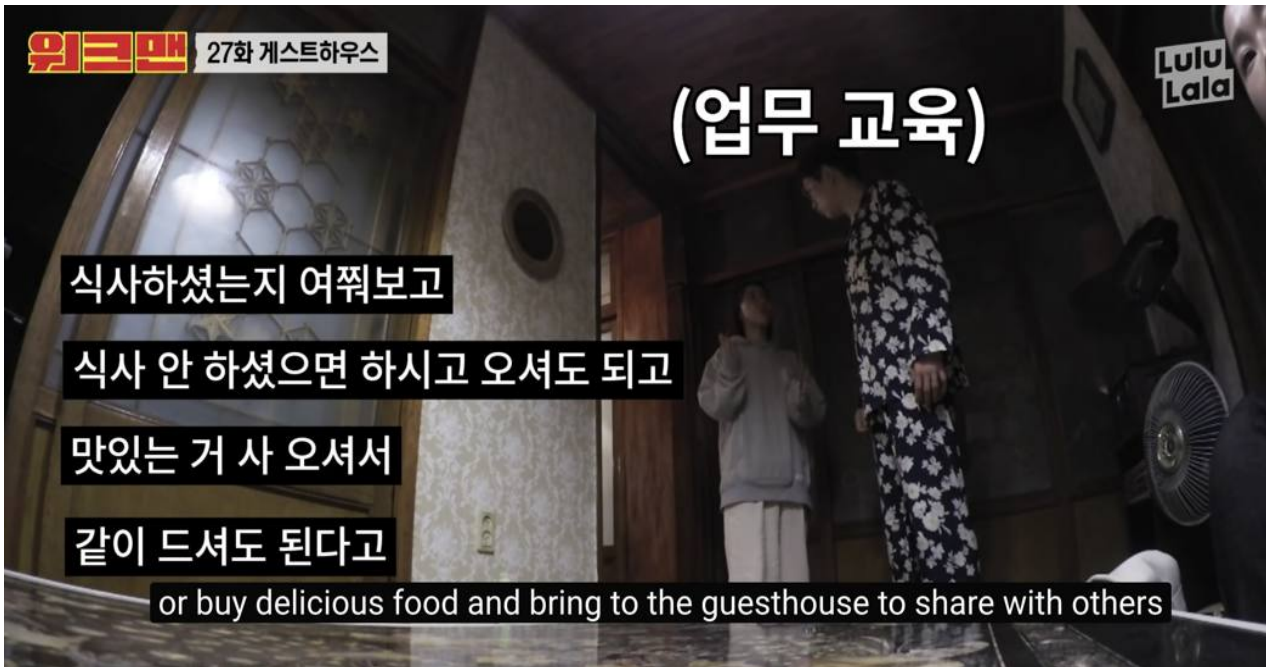
Dynamic Subtitle Auto-generator

OCCR 조상연

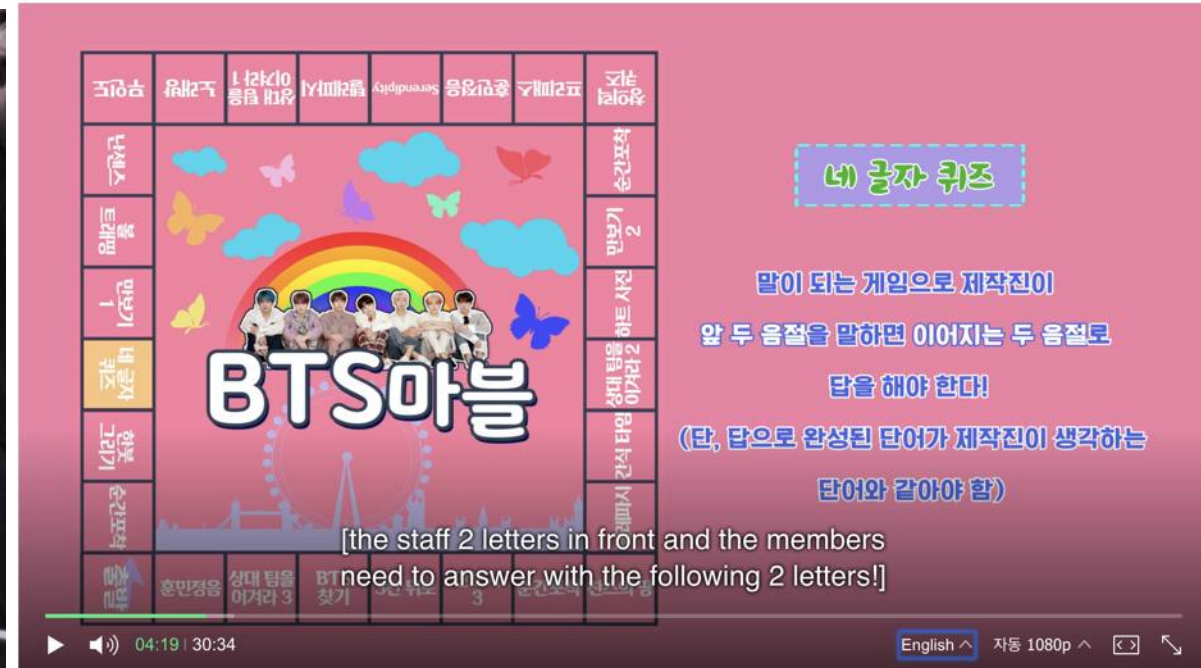


1. 서비스 배경

영상 콘텐츠의 시대 자막의 한계는 존재한다.



Youtube 워크맨 중 화면엔 자막이 많지만 영어 자막은 극히 일부



Vlive Run BTS! 중 게임 설명에 대한 자막이 매우 빠르게 지나감

✓ 한정된 시간 안에 하단에 시선을 고정해야하는 불편함 → 콘텐츠에 대한 이해 및 재미 반감

1. 서비스 배경

WebVTT, TTML 통한 해결 모색



한국어 캡션 On



한국어 캡션 Off

```
<layout xmlns:tts="http://www.w3.org/ns/ttml#styling">
  <region xml:id="subtitleArea"
    style="s1"
    tts:extent="560px 62px"
    tts:padding="5px 3px"
    tts:backgroundColor="black"
    tts:displayAlign="after"
  />
</layout>
```

```
<body region="subtitleArea">
  <div>
    <p xml:id="subtitle1" begin="0.76s" end="3.45s">
      It seems a paradox, does it not,
    </p>
    <p xml:id="subtitle2" begin="5.0s" end="10.0s">
      that the image formed on<br/>
      the Retina should be inverted?
    </p>
  </div>
</body>
```

TTML 스타일링, 레이아웃 샘플

✓ 결국 직접 조정해야하는 불편함과 성가심 → 자동화된 자막 제작 및 편집툴이 필요

Auto Generator + Editor

Auto Generator Module : Python, Flask

Youtube Link



Parsing Multi-Frame



- ❑ Clova의 OCR 기술을 이용하여 영상 속 위치 및 텍스트 인식
- ❑ 각 단어를 이어진 자막으로 변환, 위치 중복된 문장 병합
- ❑ Positioning 적용된 WebVTT 자막 파일 자동 생성

Clova OCR API



결과 병합 및
자막 제작

Web based Dynamic Subtitle Editor : React, Javascript

Edit Subtitle (style, Text) / Delete



- ❑ 원하지 않는 부분까지 인식하거나 틀린 인식의 경우 수정, 삭제 가능
- ❑ webVTT 및 자동 번역된 자막으로 Export

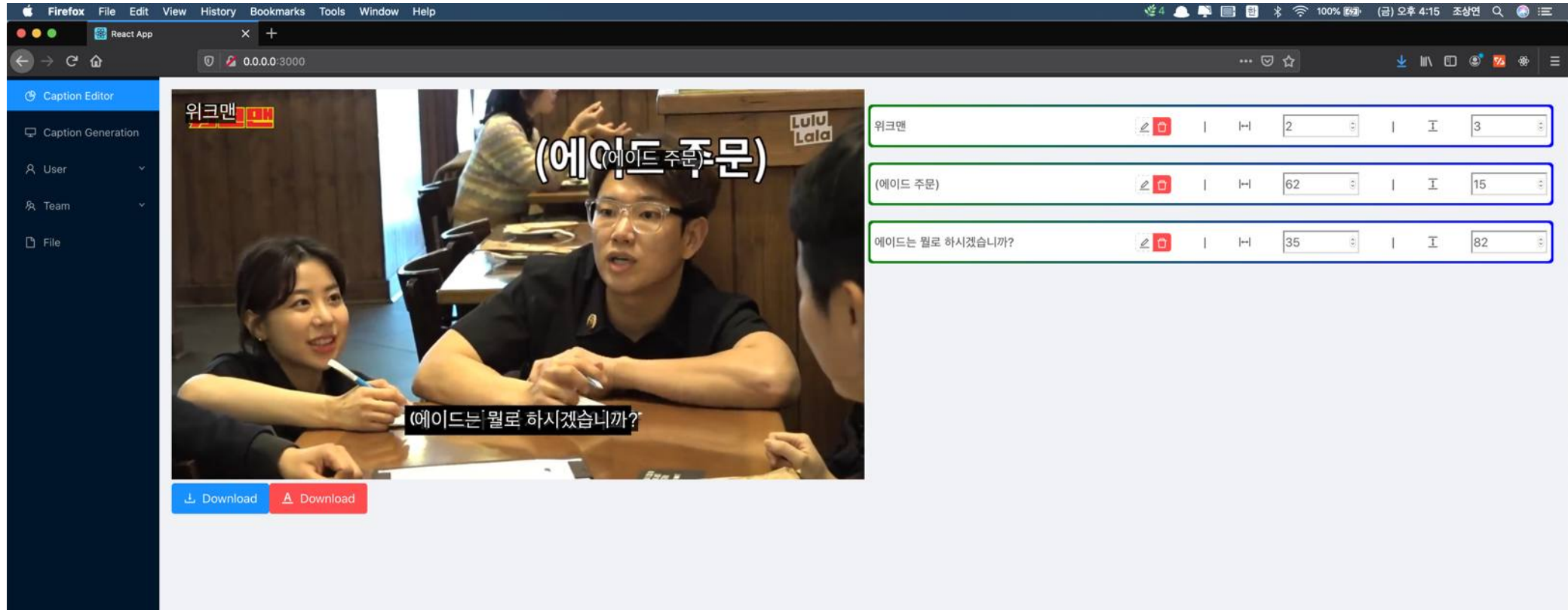
Export to
WebVTT

Papago
Translate



3. 시연

DEMO



https://www.youtube.com/watch?v=zKR_4aC83iA

Development

AI Burning Day 준비를 위한 프로젝트 전체 설계

우선 핵심은 Clova OCR API를 이용하여 자막을 제작하는 일이다.
여기서 필요한 핵심 Function을 정리하고 이에 대한 기본 구현을 직접 해보고 테스트 케이스를 만들어 본다.

1. Functional Req.

1.1 영상 캡처

- ✓ (Optional) Youtube 링크 입력시 0.3초마다 스크린샷 생성
- ✓ 영상 업로드시 초당 1~4장의 Frame 스크린샷 생성
- ✓ 생성된 스크린샷을 OCR API로 전송

1.2 자막 생성

- ✓ API Response Data Preprocessing) 위치 정보 기반하여 같은 문장 묶음
- ✓ 통일된 데이터 형식 제작 필요
- ✓ 가공된 데이터를 바탕으로 webVTT 생성 + id 부여

1.3 자막 에디터 만들기

- ✓ 1. 리액트로 기본 틀 잡기
- ✓ 2. actviecue를 이용한 실시간 자막 송출
- ✓ 3. 자막 수정하기 (Text, position)
- ✓ 4. 수정된 자막 정보 id값에 담아 api로 전송
- ✓ 5. 최종본으로 다운로드 하기 버튼
- ✓ 6. 일본어로 번역하기 버튼

자막 에디터 화면 구성

- 유튜브 링크 입력
- 자막 에디터
- 최종 다운로드 버튼 / 번역 다운로드 버튼

1.4 전용 API 서버 제작

- ✓ 5. api에서 vtt 파일 서빙 - 1hour
- ✓ 5. api에선 해당 id값의 caption 수정
- ✓ 6. 수정된 자막 정보에 맞게 webVTT 재생성 -> 200 Return
- ✓ 7. webVTT 정보 중 Text만 읽은 뒤 Papago 번역 서비스 실행
- ✓ 8. 새로운 번역본 webVTT생성

□ 동영상 캡처

- Pytube3 를 이용한 영상 다운로드
- OpenCV를 이용하여 영상 내 화면 프레임별 캡처

□ 자막 생성

- OCR API를 통해 얻은 정보 바탕으로 각 단어들 한 문장으로 조합
- 프레임 마다 끊어진 자막들을 다음 프레임 자막과 연결시켜 하나의 Caption으로 병합
- 생성된 Caption으로 webVTT 규격에 맞게 Position 적용하여 생성

□ 자막 에디터

- React 를 기반으로 하여 컴포넌트 2개 구성 (VideoContainer, CaptionView)
- useRef를 이용하여 Video 내 Caption DOM 접근 및 업데이트
- useReducer를 이용하여 상태 관리 로직 분리

□ 전용 API 서버

- Flask API 를 이용하여 영상 및 자막 파일 Serving
- MIME 타입 문제 해결

1. 영상 콘텐츠의 글로벌 마케팅 차별화

- 영상 콘텐츠 이해도 및 재미 증가, **시각적 만족도 극대화**

2. 기존 자막 업체보다도 저렴한 가격 & 빠른 속도 & 다양한 효과

- OCR API 1회 3원 기준 분당 500원대 << **기존 대비 1/4 수준** << 자X이크 기준 자막 제작 원가 분당 2000원 측정
- 기존 단조로운 텍스트 대비 훨씬 다양한 효과 및 스타일링 지원

3. 사용자 수정 자막 OCR 학습 데이터로 이용하여 모델 고도화 선순환

- 사용자가 삭제하거나 수정한 부분에 대한 데이터를 수집, OCR Model 학습 데이터로 재이용
- 다양한 번역 데이터 역시 번역 모델 학습데이터로 사용 가능 -> 모델 선순환 및 **추후 자동 번역 시스템까지 가능**