

영상 자막 자동전환 기반 텍스트 요약 알고리즘

2015104194 이규호

2014104112 신주민

요약

프로젝트는 유튜브(주로 강의 영상)에 업로드 되는 자막을 가져와서 이를 형태소 단위로 나누고, graph centrality를 텍스트에 적용한 textrank 알고리즘을 활용하여 요약하여 사용자에게 제공하는 서비스이다. 서비스는 웹으로 보여준다.

1. 서론

1.1 연구 배경

2020년 초부터 찾아온 코로나 바이러스(COVID-19)는 기존 우리의 생활방식을 완전히 뒤바꿔놓았다. 외부행사들이 취소되고, 개학이 미뤄지고, 재택근무가 실행되는 등 사람들과 접촉이 잦은 실외활동이 줄어들고 비대면 방식의 실내활동이 많이 늘어나게 되었다. 그리고 이러한 시간이 길어짐에 따라 사람들은 온라인 안에서라도 사람들을 만나고자 하였고, 그 결과 인터넷 방송과 유튜브(Youtube) 시장은 더욱 더 커지게 되었다.

하루에만 66년치의 동영상이 업로드 되는 방대한 유튜브 시장에서 모든 영상을 다 시청하기란 불가능에 가까워졌다. 예전에는 블로그나 카페에 글로 올라오던 정보들이 이제는 동영상으로 올라와 내가 원하는 짧은 정보를 얻기 위해서 그 영상을 모두 봐야 할 수도 있는 상황이 되었다. 또한 영상의 길이가 대체적으로 긴 강의 영상들도 유튜브에 업로드 되기 시작하면서 이러한 영상들을 요약해 주는 프로그램의 필요성에 대해 느끼게 되었다.

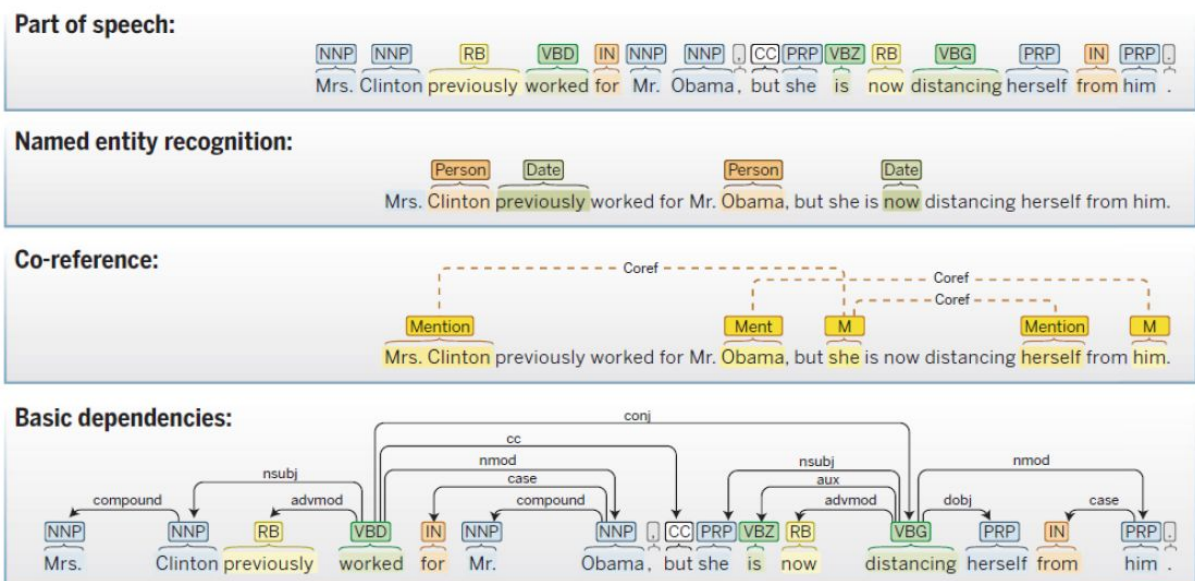
따라서 우리는 많은 사람들이 유튜브를 통해 새로운 지식을 보다 더 효과적으로 획득할 수 있는 프로젝트를 기획하였다.

1.2 연구 목표

유튜브 영상의 자막을 추출하여 요약하기 위해 먼저 유튜브 자막을 xml로 추출하는 과정을 거친다. NLP를 통한 형태소 분리와 TextRank 알고리즘을 사용하여 키워드와 핵심문장을 추출하고 LSTM을 통한 딥러닝을 통해 정확한 결과를 얻어내 동영상을 요약할 수 있는 기능을 제공한다.

2. 관련 연구

2.1 NLP 형태소 분석



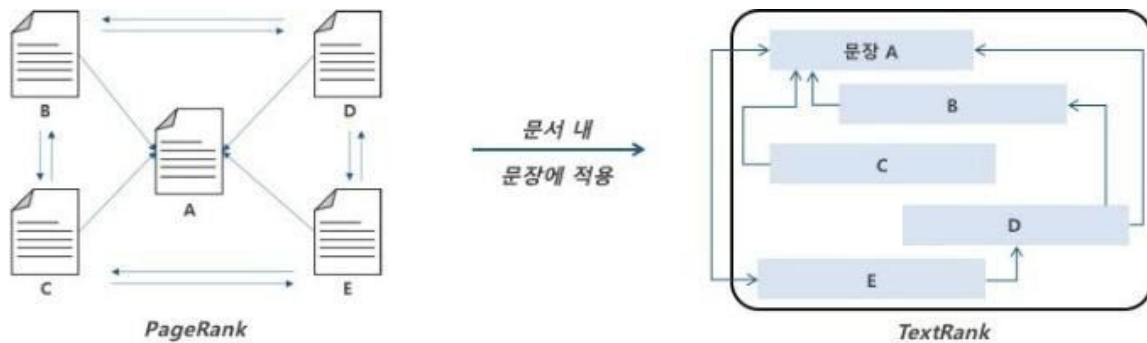
우리가 일생 생활에서 사용하는 언어인 자연어(Natural language)의 의미를 분석하여 처리한다. 입력 받은 자막을 형태소 단위로 분리하여 의미를 학습하고, 분리된 각 형태소들을 graph에서의 node로 놓고, 한 문장에서의 연결을 edge로 놓아서 자막 파일을 graph화 시킨 후, 이에 graph centrality 기법을 사용하여 중요도를 측정한다.

2.2 LSTM

RNN의 한 종류로서 장기 의존성 문제를 해결할 수 있는 알고리즘이다. RNN 방식에서 데이터와 데이터를 사용하는 지점 사이의 거리가 먼 경우에 발생하는 vanishing gradient problem을 해결할 수 있는 알고리즘으로, RNN의 hidden state에 cell-state를 추가한 구조이다. 본 알고리즘을 통해 텍스트 요약 알고리즘의 모델링을 진행한다.

2.3 Text Rank Algorithm

글의 키워드와 핵심문장을 골라내기 위해 가장 많이 사용되는 알고리즘 중 하나인 Text Rank 알고리즘이다. Text Rank 알고리즘은 단어 graph나 문장 graph를 구축하고 구글의 Page Rank 알고리즘을 이용하여 키워드와 핵심문장을 골라낸다. 그리고 이러한 방식으로 골라낸 키워드와 핵심문장을 이용하여 글을 요약하는 방식이다.



3. 기존 연구의 문제점 및 해결방안

3.1 기존 연구의 문제점

한글을 처리하는 알고리즘들은 영어에 비해 발전이 더딘 상황이다. 한국어는 교착어에 속하기 때문이다. 어순이 중요되는 영어와 달리 어근에 접사가 붙어 의미와 문법적 기능이 부여되어, 접사 하나로 테이블이 무수히 많이 발생할 수 있기 때문이다. 또한 평서문과 의문문을 구분하기 힘들고, 주어를 생략하는 경우도 비일비재하다. 따라서 정확한 모델링을 통해 하나 혹은 여러개의 문장에서 핵심을 추출하는 것은 굉장히 어려운 일이다.

이에 따라 한글을 요약해주는 라이브러리 중 제일 잘 알려진 LexRank 같은 경우도, 그리 높은 정확도를 보여주지 못하고 있는 실상이다. 추가 정보 없는 문서를 정확하게 요약하는 것은 힘들다고 판단된다.

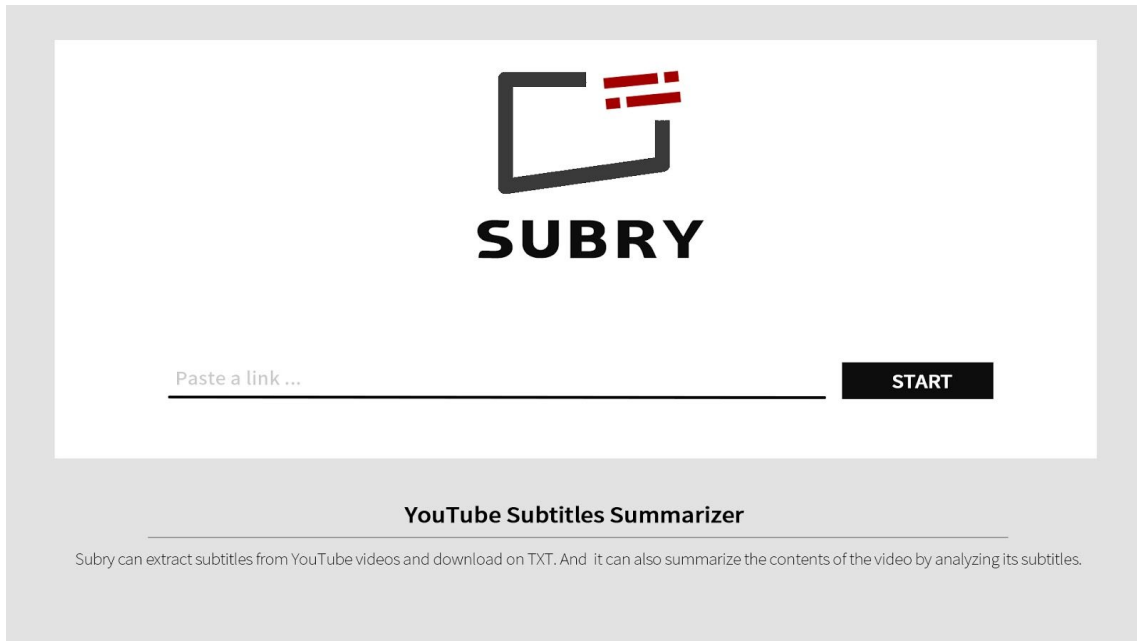
3.2 해결 방안

본 연구의 주 목적은 “텍스트 요약”이므로, 형태소 분석은 가장 모델링이 잘되었다고 판단되는 python의 mecab을 사용할 예정이다. 형태소 분석에 투자하는 시간을 줄이고, 텍스트 요약 알고리즘에 집중한다.

유튜브 영상의 제목, 상세 정보 등의 설명에 나와 있는 단어들은 중요한 단어가 될 확률이 높다고 판단할 수 있다. 따라서 이에 대한 node 들의 중요도를 높은 상태로 놓고 시작하여 전체 자막을 요약한다면, 더 높은 정확도를 가진 결과물을 추출할 수 있을 것이라 기대된다.

4. 프로젝트 내용

4.1 GUI (메인 화면)



The main GUI of the SUBRY application. It features a large logo at the top center consisting of a stylized 'S' made of black and red lines, with the word 'SUBRY' in bold black letters below it. Below the logo is a text input field with the placeholder 'Paste a link ...' and a black 'START' button to its right. At the bottom, there is a section titled 'YouTube Subtitles Summarizer' with a brief description: 'Subry can extract subtitles from YouTube videos and download on TXT. And it can also summarize the contents of the video by analyzing its subtitles.'

4.2 GUI (요약 화면)



The summary GUI of the SUBRY application. It has a header bar with the SUBRY logo on the left. Below the header, there is a text input field with the placeholder 'Paste a link ...' and a black 'START' button to its right. Below the input field, there is a large black rectangular area on the left, and a text input field on the right labeled '동영상 정보' (Video Information). Below these, there is a large text input field on the left labeled '동영상 요약' (Video Summary), and a button on the right with a download icon and the text 'TXT'.

5. 향후 일정

기간	내용
10/19 ~ 10/25	테스트용 유튜브 영상 선정, url을 통해 전체 자막 텍스트로 추출
10/26 ~ 11/01	추출된 자막을 mecab를 통해 형태소로 분리
11/02 ~ 11/08	텍스트 요약 알고리즘 설계
11/09 ~ 11/15	텍스트 요약 알고리즘 설계
11/16 ~ 11/22	텍스트 요약 알고리즘 설계
11/23 ~ 11/29	텍스트 요약 알고리즘 설계
11/30 ~ 12/06	결과물 UI/UX 설계 및 개발
12/07 ~ 12/13	최종 보고서 작성
12/14 ~ 12/20	최종 발표

6. 결론 및 기대효과

유튜브에는 점점 더 많은 동영상이 업로드 될 것이며 그 종류도 더 다양해 질 것이다. 이러한 상황 속에서 우리는 ‘우리가 보고싶은 영상’만이 아니라 ‘어쩔수 없이 봐야하는 영상’에 맞닥뜨릴수도 있다. 한정된 시간 속에서 ‘어쩔수 없이 봐야하는 영상’은 큰 부담이 될 것이나, 이 프로젝트를 활용함으로써 그러한 부담을 덜 수 있고 많은 시간적 이득을 얻게 될 수도 있다. 특히 강의와 같이 정보전달이 주 목적인 긴 영상에서는 내용에 대한 요약본을 미리 읽고 가는 것 만으로도 영상을 이해하는데 있어서 훨씬 더 큰 도움을 줄 수 있을 것이라 예상된다.

7. 참조문헌

[1] TextRank Algorithm - <https://lovit.github.io/nlp/2019/04/30/textrank/>

[2] NLP - <https://ratsgo.github.io/natural%20language%20processing/2017/03/22/lexicon/>