**대본**

#1

안녕하세요. 오늘 발표를 맡은 VIVA Pro조의 조원 손승현이라고 합니다.

#2

오늘 설명할 것은 지난주에 발표했던 라벨링 툴과 한글을 활용한 말뭉치에 대한 추가 조사인데, 라벨링 툴은 TagEditor, U-Tagger NE 1.0, KoNLPy 세 가지를 조사했고, 한글을 활용한 말뭉치는 엑소브레인 말뭉치, 울산대 말뭉치 등을 조사했습니다.

#3

첫번째로 조사한 라벨링 툴은 Tag Editor입니다. 첫번째 사진은 ‘나는 광운대학교 새빛관에 있다’라는 문장을 입력하고 Tag Editor를 실행시켰을 때 실행 결과입니다. 두번째 사진은 dependency, 즉 문장을 문맥보다는 문법에 근거해서 나눈 의존구문분석에 근거해서 Tag Editor 지정한 Tag set입니다. 결과화면을 설명해드리면, 새빛관이라는 단어가 ‘ROOT’, 즉 문장의 핵심 단어로 지정이 되어 그 단어에 대해서 나는, 광운대학교에, 있다 라는 단어가 각각 문법에 맞게 태깅된 화면입니다.

정리하자면 Tag Editor는 1,2번과 같은 장점이 있었지만…

#4

다음과 같은 단점이 있었습니다. Tag Editor 자체에서 태깅을 일일이 지정하기 위해 txt 파일을 불러오면 파일이 깨지는 문제가 발생했습니다. 두번째 사진에서 저 동그라미 안에 있는 txt파일을 Tag Editor를 통해서 불러오면 아래화면과 같이 파일이 깨지는 문제가 발생했습니다.

문제를 해결하기 위해 저희는 메모장에서 UTF-8, ANSI 방법으로 인코딩 방법을 바꿔보았으나 파일이 또다시 깨지는 것을 확인할 수 있었습니다.

따라서 텍스트 파일이 깨지는 문제를 해결하지 못했기 때문에 프로젝트에 적합하지 않다고 판단했습니다.

#5

#6

U-Tagger NE를 실제로 작동시켰을 때 화면입니다.

위쪽 사진처럼 저희가 기사에 나온 문장을 메모장에 문장 단위로 나누어서 txt파일로 저장하였고, U-Tagger NE를 사용해서 분석한 결과 형태소 분석이 자동으로 이루어지는 것을 저쪽 동그라미를 통해 확인할 수 있었습니다.

또한, ‘개체명 선택’ 창에서 개체명 분석도 어느정도 이루어지는 것을 확인할 수 있었습니다. 비록저희가 계획했던 분류 태그와는 살짝 다르긴 하지만 어느정도 비슷했고, 또 저희가 따로 개체명을 추가하거나 조정할 수 있다는 것을 알 수 있었습니다.

, 문장 단위로 데이터 셋을 넣기 유용하다는 점, 형태소 분석결과가 파악하기 쉽다는 점, 또 저희가 프로젝트에 직업, 음식, 국가 등 몇몇 개체명 분류 태그를 추가해야 하는데, 태그를 추가하는 것이 상대적으로 쉬웠기 때문에 이번 프로젝트에 적합하다고 생각합니다.

#7

다음은 KoNLPy를 조사해 보았습니다. KoNLPy는 한국어 정보처리를 위한 파이썬 패키지입니다.

자연어처리과정에서 한국어를 즉 형태소 단위로 분리, 즉 토그나이징하는 데이터 전처리가 필요한데, 그 때 많이 사용하는 패키지임을 알 수 있었습니다.

일반적으로 텍스트를 형태소 단위로 분리하는 방법은

1.

2.

의 방법이 있는데, 2번의 방법을 사용할 경우 학습을 시킬 때 너무 많은 시간이 필요하기 때문에 이번 프로젝트에서는 1번의 방법이 필요하다고 판단을 했는데, KoNLPy는 1번의 방법을 사용합니다. 또 윈도우 환경을 기준으로 KoNLPy에서는 4가지의 형태소 분석기를 제공하기 때문에, 저희 프로젝트에서 활용할 수 있을 것 같습니다.

#8

주피터를 이용해서 konlpy를 직접 다운받아서

#9

실행해본 화면입니다.

첫번째 사진 from konlpy.tag import Kkma에서 kkma(꼬꼬마)는 앞에서 언급된 konlpy에 포함되어 있는 형태소 분석기 중 하나로, 저 경우 꼬꼬마 형태소 분석기를 이용해서 형태소 분석을 한다는 의미입니다. 이어서 nouns 명령어를 이용해서 데이터 셋으로 들어온 문장 중 명사만이 추출되어서 콘솔 창에 출력이 됩니다.

또 꼬꼬마 말고 okt 형태소 분석기도 사용을 해봤는데요, morphs 명령어를 이용해서 문장을 형태소 단위로 나눠준 결과를 print명령어를 사용해서 출력하고, nouns, phrases 명령어는 각각 명사와 구를 추출할 것을 의미하는 명령어입니다. 또, pos 명령어를 통해서는 형태소 단위로 분석할 뿐만 아니라 형태소의 종류까지 반환해주는 것을 알 수 있었습니다.

따라서 저희 프로젝트에 활용할 수 있을 것 같습니다.

#10

#11

앞에서 언급한 말뭉치들의 화면입니다. 네이버 영화 말뭉치 같은 경우 저렇게 맞춤법이나 띄어쓰기가 잘못된 경우가 너무 많았습니다. 또 챗봇 데이터 같은 경우는 문장이 평서문이 아닌 문장이 더 많았고, 또 대부분이 저런 짧은 문장들로 이루어져 있어서 충분한 양의 data set을 제공하지 못할 것이라고 판단했습니다.

#12

앞에서 언급한 AI Hub 같은 경우는 저희가 사이트에서 json 파일 형식으로 말뭉치들을 다운받아 봤는데요.

#13

다운받아서 확인을 해봤는데, 데이터 셋의 양이 너무 많아서 그런지 용량이 커서 전체파일을 볼 수가 없었습니다. 화면은 용량이 커서 전체파일을 볼 수 없다는 오류 메시지입니다. 일부 부분은 확인할 수가 있었는데요. 저 동그라미 안에 있는 개체명으로 태깅이 되어있다는 것은 확인해 볼 수 있었습니다.

그러나 용량이 너무 커서, 즉 데이터 셋이 너무 많아서 가공하기가 어려울 것이라는 점, 또 많은시간과 인력이 필요하다는 점 때문에 프로젝트에 적합하지 않습니다.

#14

또 저희 팀은 Exobrain 말뭉치도 조사해 보았습니다. 엑소브레인 사이트의 ‘산출물 다운로드’ 메뉴에 나와있는 결과물들 중, 저희는 형태-의미 주석 말뭉치, 의존관계-의미역 주석 말뭉치, 개체명 주석 말뭉치가 있는 ‘엑소브레인 3세부 데이터 목록’ 이 저희 프로젝트와 적합하다고 생각하여, 지난주에 해당 기관과 연락을 해서 API키를 발급받아 말뭉치 파일을 다운받았습니다.

오른쪽 화면이 말뭉치 파일 화면입니다. 1번 언어분석 통합 말뭉치 같은 경우 뉴스와 위키백과에있는 Q&A 등을 데이터 셋으로 활용한 것을 알 수 있었습니다. 또 2번 세부기술별 말뭉치 같은 경우는 각각 개체명 인식 말뭉치, 의미역 인식 말뭉치, 의존구문분석 말뭉치로 나뉜 것을 알 수 있었습니다.

전체적으로 데이터 셋이 앞의 챗봇과 같이 Q&A 형식으로 되어있으며, 각 말뭉치는 txt, json 형식으로 구성되어 있었습니다.

#15

저희는 2번 세부기술별 말뭉치에 주목해봤는데요. 우선 1번 개체명 인식 말뭉치는 txt파일로 이루어져 있었고 사진과 같이 문장 단위로 데이터셋이 이루어져 있었고 문장 중간중간에 꺽쇠 표시를 이용해서 태그를 분류한 것을 알 수 있었습니다. 또 내용을 분석한 결과 작년 팀이 활용했던 데이터 셋임을 알 수 있었습니다. (작년 팀이 가공했다는 이야기를…)

2번 의미역 인식 말뭉치와 3번 의존구문분석 말뭉치는 모두 자연어 질의 응답을 위한 질문과 정답 쌍에 태깅이 되어있었습니다. 다만 앞의 챗봇과의 차이점은 질문과 답변 문장 모두 평서문으로 이루여져 있기 때문에 형태소 분석을 하기가 더 용이하다는 것입니다.

2번 의미역 인식 말뭉치 같은 경우 이 화면과 같이 어떤 문장에서 문장에서 단어가 가지는 의미위주 로 태깅이 되어있는 것을 확인할 수 있었고, 저 동그라미 안에 있는 것이 태그들의 종류입니다.

#16

2번 의미역 인식 말뭉치에서의 태그들의 종류는 다음과 같았습니다. ARG들을 이용해서 서술어를 기준으로 동작주, 피동작주인지 또 시작점인지 도착점인지에 따라서도 기준을 나눈 것을 알 수 있었습니다.

그 외, 장소, 방향, 조건, 방법, 시간, 범위 등에 대해서도 태그가 이루어진 것을 확인할 수 있었습니다.

#17

3번 의존구문분석 말뭉치는 2번 의미역 인식 말뭉치에 비해 내용보다는 국어문법에 치우쳐서 태깅을 하는 것을 알 수 있었습니다. 태그는 크게 구문 태그 세트와 기능 태그 세트로 나누어져 있었는데, 구문 태그 세트는 ~, 기능 태그 세트는~ .

따라서 2번과 3번 말뭉치를 활용하면 문맥을 파악하는데 용이하여 동음이의어 등의 문제를 잘 해결할 수 있을 것이라고 생각합니다.

#18

앞에서 이야기한 것을 정리하자면 엑소브레인은

1. 2.
2. 또한 앞의 AI hub와 다르게 용량이 다른 말뭉치들에 비해 작았습니다. AI hub같은 경우는 수백 메가였던 반면에, 엑소브레인은 20메가도 안되는 용량이었고, 따라서

…

#19

다음 말뭉치는 울산대 말뭉치를 조사해 보았습니다.

앞의 엑소브레인 말뭉치와 마찬가지로 / / 로 나누어져 있었고, 어느 정도 태깅이 되어 있는데다가 데이터 셋의 양도 충분했습니다.

또한 앞에서 설명한 라벨링 툴 중 UTagger-NE 1.0과 연관시켜서 사용할 수 있는 장점이 있기 때문에 프로젝트의 data set으로 활용 가능하다고 생각합니다.

감사합니다.

**예상질문**

1. Yedda는 왜 내용에 포함되지 않았습니까?

A: 모든 단어가 ‘O’로 표기되기 때문입니다.

1. 소스코드가 확보된 것은? 아무것도 없음
2. 형태소 분석 성능 (고유명사 부분에서 문제가 생기는 것을 확인)
3. 개체명 인식
4. 데이터 셋 크기