**한국어의 문법적 특성을 반영한**

**사전학습 언어모형 구축**

**톡위드 ( 강혜빈, 구채원, 조소연)**

**통계학과 201911501 강혜빈**

**통계학과 201911503 구채원**

**통계학과 201911536 조소연**

목차

1. **요구 조건 및 제약 사항 분석**

**1-1. 과제 목표 및 요구조건**

**1-2. 제약사항 분석 및 수정사항**

1. **설계 상세화 및 변경 내역**

**2-1. 입력 생성**

**2-2. Pre-training과 Finetuning**

1. **갱신된 과제 추진 계획**
2. **구성원별 진척도**
3. **보고 시점까지의 과제 수행 내용 및 중간 결과**

# **1.요구 조건 및 제약 사항 분석**

1-1. 과제 목표 및 요구조건

1. 과제목표

한국어의 의존 구문분석에 특화된 사전학습 언어모델을 구축한다.

* 현재 배포된 사전학습 언어모델은 영어의 언어단위 구성방법(Toknizer)를 사용하기에 한국어의조사의 구조적 역할을 인식하지 못하여 구문분석이 온전히 이루어 지지 못한다.
* 이를 해결하기 위해 형태소 임베딩을 추가하여 형태소 단위의 사전학습 언어모델을 개발하고자 한다.

1-2. 제약사항 분석 및 수정사항

1. 기존 제약사항에 대한 수정사항

* Python 상의 파일 용량 제한
* Input 데이터 파일을 2기가 단위로 분할한 뒤 모델 학습에 필요한 임베딩 벡터 생성

1. 추가 제약사항에 대한 대책

* KoBigBird Toknizer와 형태소 Kkma tagger 상의 형태소 차이
* 형태소 분석기 mecab 을 이용하여 우선 형태소 pos\_ids를 생성한 뒤 추후 Toknizer와 맞춰출 방법 연구 예정

# **2. 설계 상세화 및 변경 내역**

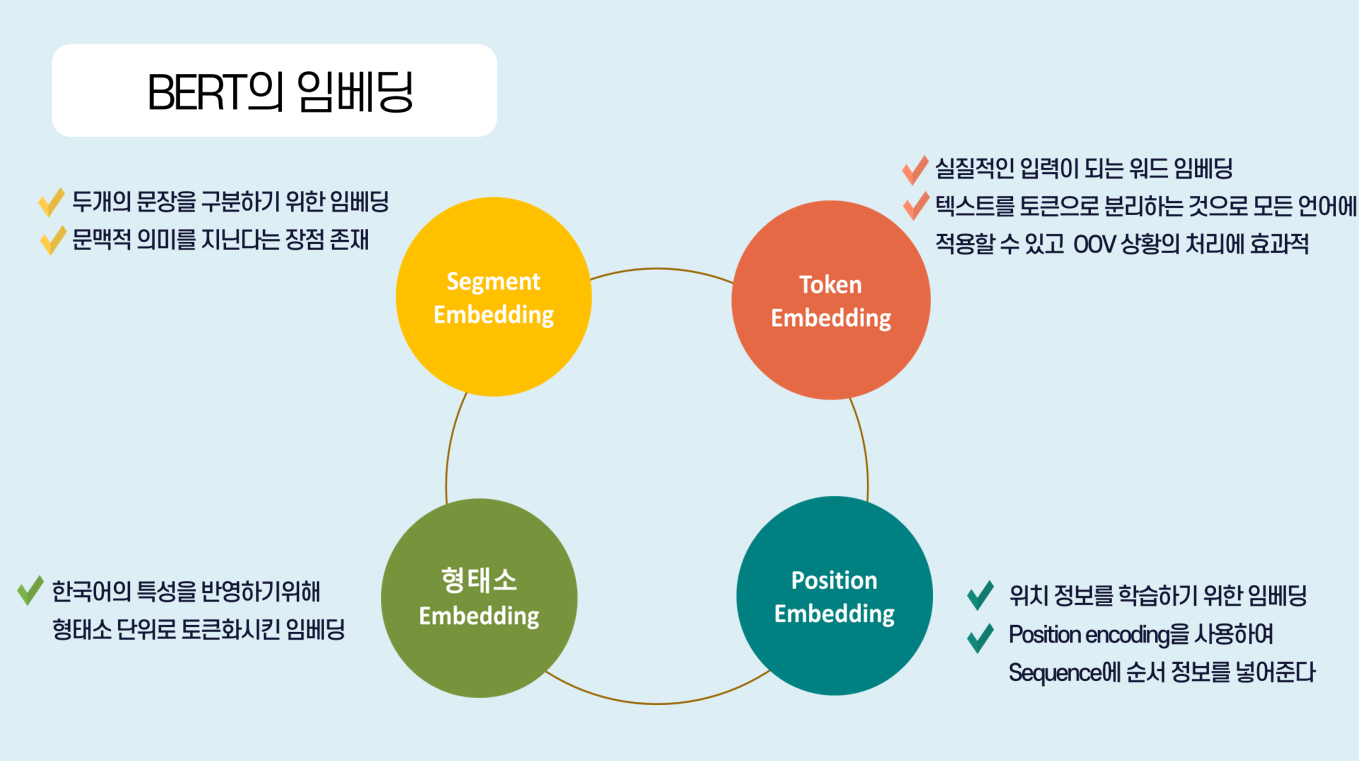


그림 1 사전학습을 위한 임베딩

2-1. 입력 생성

* 30GB가량의 input txt파일 이용
* 토큰화를 위한 토크나이저로는 KoBigBird모델 형태소 태거로는 Konlpy패키지의 Kkma를 사용
* 실질적인 입력이 되는 Token Embedding과 두 개의 문장을 구분하기 위한 Segment Embedding 파일 생성
* KoBigBird Toknizer와 형태소 Kkma tagger 상의 형태소 차이로 인해 임시적으로 mecab 태거를 이용하여 형태소 Embedding을 생성

2-2. Pre-training과 Finetuning

* 입력한 임베딩 벡터로 사전학습 진행
* KorNLI, KorSTS, NSMC 등의 데이터셋을 이용하여 각각 추론, 유사문장, 감성분석 등의 task를 진행하고 Finetunning 할 예정
* 기존의 사전학습 모델인 BERT, HanBERT 등과 결과를 비교하며 성능을 측정

# **3. 갱신된 과제 추진 계획**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5월 | | | 6월 | | | | | 7월 | | | | | 8월 | | | | | | | 9월 | | |
| 3주 | 4주 | 5주 | 1주 | 2주 | 3주 | 4주 | 5주 | 1주 | 2주 | 3주 | 4주 | 5주 | 1주 | 2주 | 3주 | | 4주 | | 5주 | 1주 | 2주 |
| 착수보고서 작성 | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  | |  |  |  |
| 딥러닝, NLP  스터디 | | |  |  |  |  |  |  |  |  |  |  |  |  |  | |  | |  |  |  |
|  |  | 데이터 전처리 | | |  |  |  |  |  |  |  |  |  |  |  | |  | |  |  |  |
|  |  |  |  | 형태소 분석기 사용법 익힘 | | |  |  |  |  |  |  |  |  |  | |  | |  |  |  |
|  |  |  |  |  |  |  | 입력 입베딩 생성 | | |  |  |  |  |  |  | |  | |  |  |  |
|  |  |  |  |  |  |  |  |  |  | 중간보고서 작성 | |  |  |  |  | |  | |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  | 언어모델 학습 | | | | |  | |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | 테스트 및 보완 | | | | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  | | 최종 보고서 작성 및 마무리 | | | |

# **4. 구성원별 진척도**

|  |  |
| --- | --- |
| 이름 | 역할분담 |
| 구채원 | * 데이터 전처리, * token embedding,seg\_embedding 생성 * 중간보고서 작성 |
| 강혜빈 | * 데이터 전처리 * token embedding,seg\_embedding, 생성 * 형태소 임베딩 생성 |
| 조소연 | * 데이터 전처리 * token embedding,seg\_embedding, 생성 * 형태소 임베딩 생성 |

# **보고 시점까지의 과제 수행 내용 및 중간 결과**

6-1. 형태소 분석기 이해와 실행

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 2 형태소 분석기 실행 예시

* 2MB단위로 input txt파일을 분할한 뒤 파이썬 상으로 형태소분석기(KL Tagger)를 실행

6-2. input 임베딩 생성

* 약 30GB의 학습용 input.txt파일을 2GB단위로 분할한 뒤 KobigBird을 이용하여 두 문장 을 이어 붙인 뒤 토큰화
* 실질적인 입력인 Tokem Embedding과 Segment Embedding 벡터를 npy로 저장하였다.

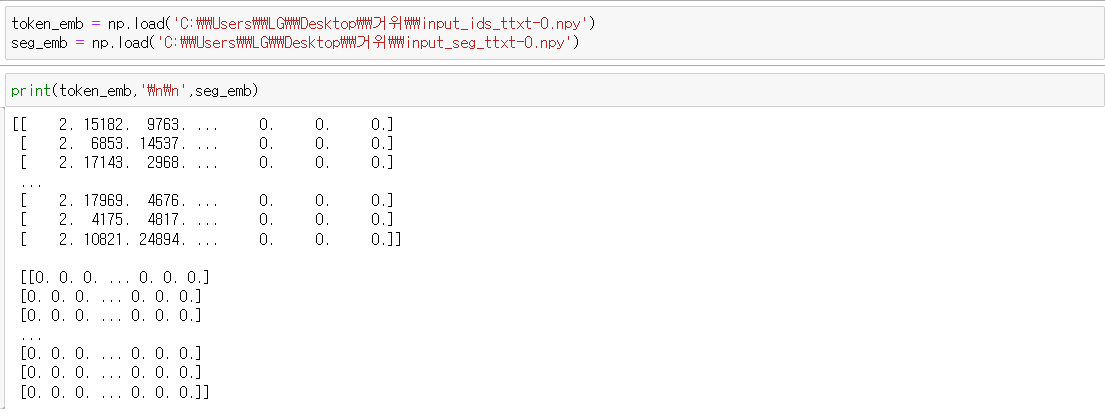


그림 3 Token Embedding, Segment Embedding

* KoBigBird Toknizer와 형태소 Kkma tagger 상의 형태소 차이로 인해 임시적으로 mecab 태거를 이용하여 형태소 Embedding 벡터를 생성한 후 npy로 저장