## Word2Vec 과 품사 정보#1

한국어 품사 분류의 가장 중요한 기준은 기능인데, 기능은 분포와 밀접한 관련이 있음.

Word2vec 은 단어벡터를 만들 때 말뭉치의 분포 정보를 학습에 반영함,

Word2Vec 으로 임베딩된 단어 벡터에는 품사 정보가 내재해 있을 것임.

같은 품사에 해당하는 단어 벡터는 서로 유사할 것.

BE 와 CP 는 모두 말뭉치를 글자 단위로 빈도를 세어서 단어일 가능성을 점수로 나타내는 기법

BE 는 단어의 외부 정보, CP 는 단어의 내부 정보를 점수화함

BE 와 CP 를 동시에 고려한 단어 점수표를 만들고, 이를 바탕으로 말뭉치에 tokenize 수행.

토크나이징을 수행한 말뭉치에는 Word2Vec 기법 적용.

종결어미, 조사, 명상, 동사, 형용사, 부사 , 감탄사 등등으로 해당 단어벡터와 코사인 유사도가 가장 높은 단어를 뽑아냄

## Word2Vec 과 품사 정보#2

KoNLPy 의 코모란 형태소 분석기를 적용하여 형태소 분석 실시

Word2Vec 으로 임베딩한 명사 단어벡터와 코사인 유사도가 가장 높은 단어 100 개 우선 뽑음

여기에서 코모란 분석기의 품사 태깅 결과를 참고해 유사 단어의 품사 확인

➔ Word2Vec 으로 임베딩된 단어벡터에 명사라는 품사 정보가 내재해 있다고 볼 수 있음. (명사와 유사한 단어는 명사임)

## NLP 의 기본 절차와 Lexical Analysis

언어학: 말소리를 연구하는 음운론(Phonology), 단어와 형태소를 연구하는 형태론(Morphology), 문법과 맥락/담화를 각각 논의하는 통사론(syntax), 의미론(Semantics)

음성인식-음운, 형태소분석-형태론, 파싱(문장의 문법적 구조 분석)-통사론

## 어휘분석(Lexical Analysis)

포스태깅(PoS)

개체명인식(Named entity recognition)

상호참조(co-reference): 선행 단어를 현재 단어와 비교해 같은 개체인지

의존관계분석(basic dependencies): 성분에 따라 문장구조를 정의하는 구조조문법(생성문법 기반)꽈 달리 단어와 단어가 가지는 의존관계를 중시해 문장 구조를 분석

## 어휘분석 절차

## 문장 분리(sentence splitting): 말뭉치를 문장 단위로 끊음

## 토크나이즈: 토큰: 의미를 가지는 문자열

형태소(뜻을 가진 최소 단위)나 그보다 사우이 개념인 단어(자립하여 쓸 수 있는 최소 단위)

토크나이징은 문서나 문장을 분석하기 좋도록 토큰으로 나누는 작업

영어는 띄어쓰기 만으로도 가능

## Morphological analysis

Text normalization aka

토큰들을 좀 더 일반적인 형태로 분석해 단어수를 줄여 분석의 효율성을 높임

대문자->소문자 Folding

Stemming: 단어를 축약형으로

Lemmatization: 품사 정보가 보존된 형태의 기본형으로 변환

## 포스태깅:

토큰의 품사정보를 할당하는 작업

의사결정나무, 은닉 마코프 모델, 서포트벡터 등이 여기에 해당

KoNLPy 같은 포스태거는 문장분리, 토크나이즈, lemma, 포스테깅까지 한 번에 수행해줌

조사, 어미가 발단한 한국어는 정확한 분석이 어려움. 교착어 성질을 지니는 언어이기 때문.

어근에 파생접사나 어미가 붙어.

어근과 접사, 어미를 적절히 나누는 것이 쉽지 않음.

단어와 형태소 분석이 자연어처리의 기본 중 기본.

## CNN 으로 문장 분류하기

자연언어는 단어나 표현의 등장 순서가 중요한 sequential data

RNN: 입력값을 순차적으로 처리

입력값을 단어로 바꿔놓고 생각해보면 단어의 등장 순서를 보존하는 형태로 학습이 이뤄짐을 알 수 있음

CNN: 텍스트의 지역적인 정보, 즉 단어 등장 순서/문맥 정보를 보존한다는 것

**Yoon Kim (2014) :** 영화 리뷰 사이트에 게시된 댓글과 평점 정보 이용해 각 리뷰가 긍정인지 부정인지 분류하는 모델

N 개의 단어로 이뤄진 리뷰 문장을 각 단어별로 k 차원의 행벡터로 임베딩함

단어를 벡터로 임베딩하기 위해서는 word2vec 이나 GloVe 처럼 distributed representation 을 쓸 수 있고, 또는

단어 벡터의 초기값을 랜덤으로 준 뒤 이를 다른 파라메터들처럼 학습 과정에서 조금씩 업데이트해서 사용할 수 있음

필터의 크기와 개수를 정해 필터 개수만큼 feature map 을 만들고, max-pooling 으로 클래스 개수(긍정, 부정)만큼의 스코어를 출력하는 네트워크 구조

*텐서플로우 코드로 구현된 아키텍처를 한국어 영화 리뷰에 적용*

-단어 벡터의 초기값을 랜덤으로 준 뒤 이를 다른 파라메터들처럼 학습 과정에서 조금씩 업데이트해서 사용할 수 있음 <- 이 방법을 선택

이를 위해서는 텍스트 문장을 숫자들의 나열로 반환해야 함

-단어 숫자를 줄임

**Lookup 테이블 구축**

Word2Vec 을 쓰지 않도 단어 벡터를 만들기로 함. 커다란 Lookup 테이블

지금까지.. 리뷰를 단어 id 들의 나열, id 에 해당하는 단어벡터도, 이제 단어 id 들 각각을 벡터로 바꿔주기만 하면 됨.

왓챠에서 모은 댓글과 평점 데이터 중 80% 학습, 20% 검증 데이터

학습에 쓰지 않은 검증 데이터에 대한 예측 결과(단순 정확도)는 85% 내외로 수렴