텍스트 전처리

텍스트 전처리

정제, 불용어 제거, 어간 추출, 토큰화
및 문서 표현 등을 진행하는 과정

cf 파이썬 pytorch의 torchtext는 텍스트 전처리를 위한 파이썬 라이브러리

텍스트 전처리

정제(Cleaning)

- 특수문자 등과 같은 불필요한 노이즈 텍스트 제거 및 대소문자 통일

  예 특수문자 : '!"#$%&₩'()*+,-./:;<=>?@[₩₩]^_`{|}~'

  예 대소문자 통일 : korea, Korea → KOREA

불용어 제거(Stop word elimination)

- 전치사, 관사 등 문장이나 문서의 특징을 표현하는데 불필요한 단어를 제거하는 단계

텍스트 전처리

어간 추출(Stemming)

- 단어의 기본 형태를 추출하는 단계

  예 stem, stemming, stems, stemmed, stemmer ➡ stem

토큰화(Tokenization)

- 코퍼스(corpus)에서 분리자(Separator)를 포함하지 않는 연속적인 문자열 단위로 분리

  예 한글 토큰화 결과입니다. ➡ ['한글', '토큰', '화', '결과', '입니다', '.']

  cf 파이썬에서 영문 토큰화는 nltk를 사용하고 한글 토큰화는 konlpy를 사용

NLP - 형태소분석

형태소

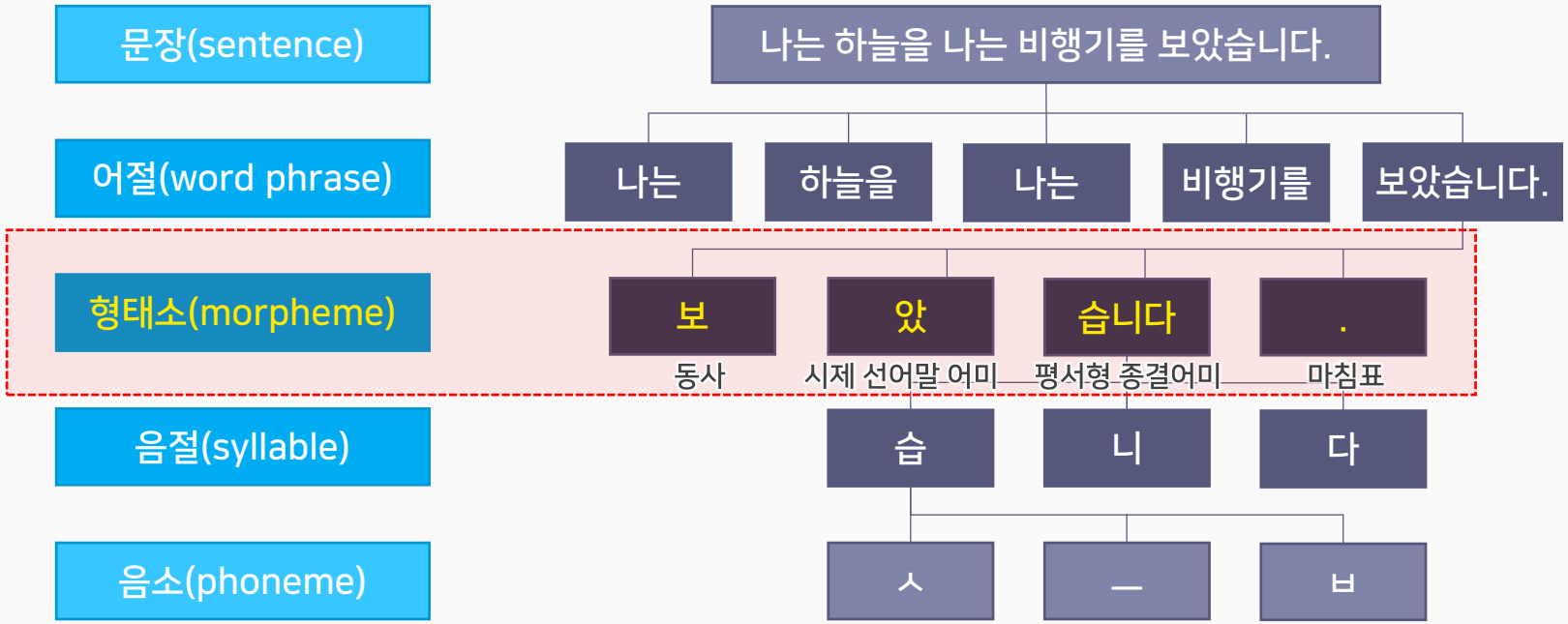
자연어에서 최소의 의미 단위

NLP - 형태소분석

형태소 분석
(Morphological Analysis)

형태소 분석은 자연어 문장에서 의미를 가진 최소 단위인
형태소(명사, 동사, 형용사, 부사, 조사, 어미 등)를 분석

- 자연어 처리를 위해 수행해야 하는 가장 첫 단계의 분석



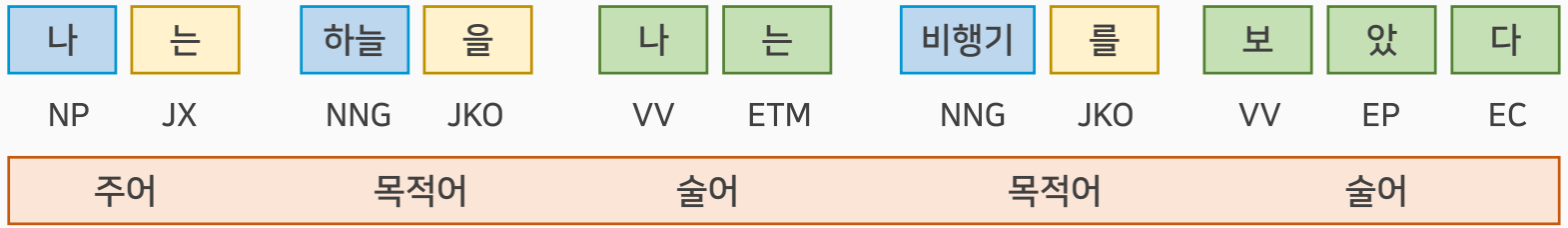
형태소 분석기를 이용하여 형태소를 분석

- 영문 형태소 분석기 : nltk(http://www.nltk.org)
- 한글 형태소 분석기 : konlpy(http://konlpy.org/ko/latest

형태소 분석기	from eunjeon import Mecab tagger = Mecab()
형태소 분석 문장	pos = tagger.pos('나는 하늘을 나는 비행기를 보았다')
형태소 분석 결과	pos [('나', 'NP'), ('는', 'JX'), ('하늘', 'NNG'), ('을', 'JKO'), ('나', 'NP'), ('는', 'JX'), ('비행기', 'NNG'), ('를', 'JKO'), ('보', 'VV'), ('았', 'EP'), ('다', 'EC')]

NLP - 구문 분석

- 문장의 구조적 성질을 규칙화한 문법을 통해 문장의 구조를 분석



NLP - 단어임베딩

임베딩
(Embedding)

범주형 자료를 연속형 벡터 형태로
변환시키는 것

NLP - 단어임베딩

워드 임베딩
(Word Embedding)

인간의 언어를 컴퓨터가 이해할 수 있는 언어로 변환

문자열을 숫자로 변환 하여
벡터(Vector) 공간에 표현

인간이 이해하고 사용하는 언어(문자열)를 컴퓨터로 하여금
효과적으로 인식할 수 있도록 하기 위해 숫자 형태로 변환하는 방법

워드 임베딩(Word Embedding)의 목적

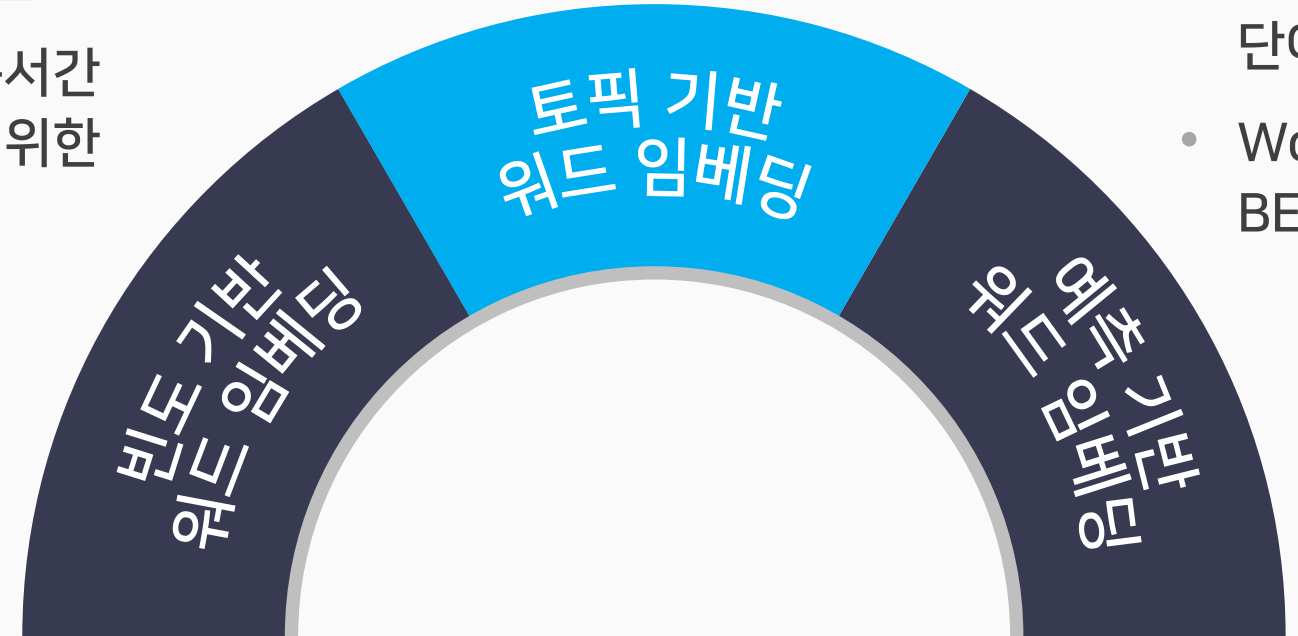
- 인간의 언어를 컴퓨터가 이해할 수 있는 언어로 변환하여 벡터 공간에 표현함으로써 단어와 단어, 문장(문서)과 문장(문서) 간의 유사도 계산 가능
- 벡터간 연산을 통해 의미적 관계 도출 가능
- 사전에 대량데이터로 학습한 모델(pre-trained model)을 재사용하는 전이학습(Tranfer Learning) 가능

워드 임베딩(Word Embedding)의 종류

- 다수 문서에 등장하는 각 단어들의 빈도를 행렬로 표현하거나 가중치를 부여
- 단어의 중요도나 문서간 유사도를 측정하기 위한 임베딩
- DTM, TF-IDF

- 주어진 문서에 잠재된 주제(latent topic)를 추론(inference) 하기 위한 임베딩
- LDA, Latent Dirichlet Allocation

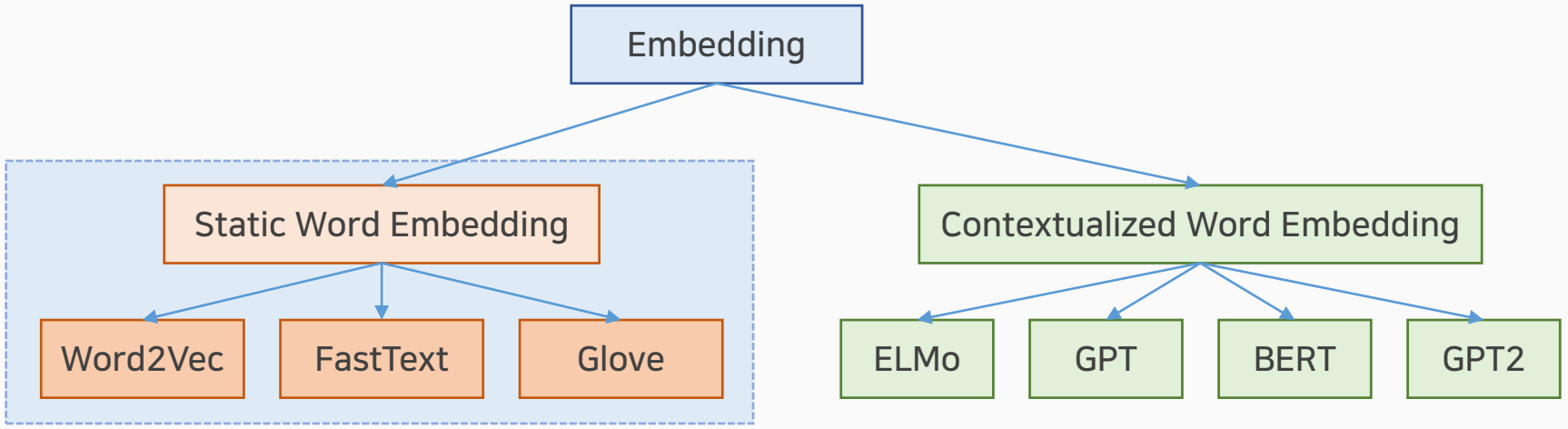
- 주어진 문장이나 단어의 다음 단어 예측, 주변 단어에 대한 예측, Masking 된 단어의 예측등을 위한 임베딩
- Word2Vec, FastText, BERT, ELMo, GPT



워드 임베딩은
단어 단위 워드 임베딩에서 **문장 단위 워드 임베딩**으로
발전

단어 단위 워드 임베딩

- 단어(Word) 기반으로 임베딩을 수행하며 문맥을 고려하지 않은 상태에서 워드 임베딩을 수행
- Word2Vec, FastText, Glove 등의 임베딩 방법
- 서로 다른 문맥의 동음이의어가 동일하게 임베딩되는 문제점



워드 임베딩은
단어 단위 워드 임베딩에서 문장 단위 워드 임베딩으로
발전

문장 단위 워드 임베딩

- 문맥을 고려하여 문장(Sentense) 기반으로 임베딩을 수행
- ELMo, GTP, BERT, GTP2 등의 임베딩 방법
- 문장(Sentense) 기반으로 임베딩을 수행하여 언어 모델(Language Model)로도 불리움