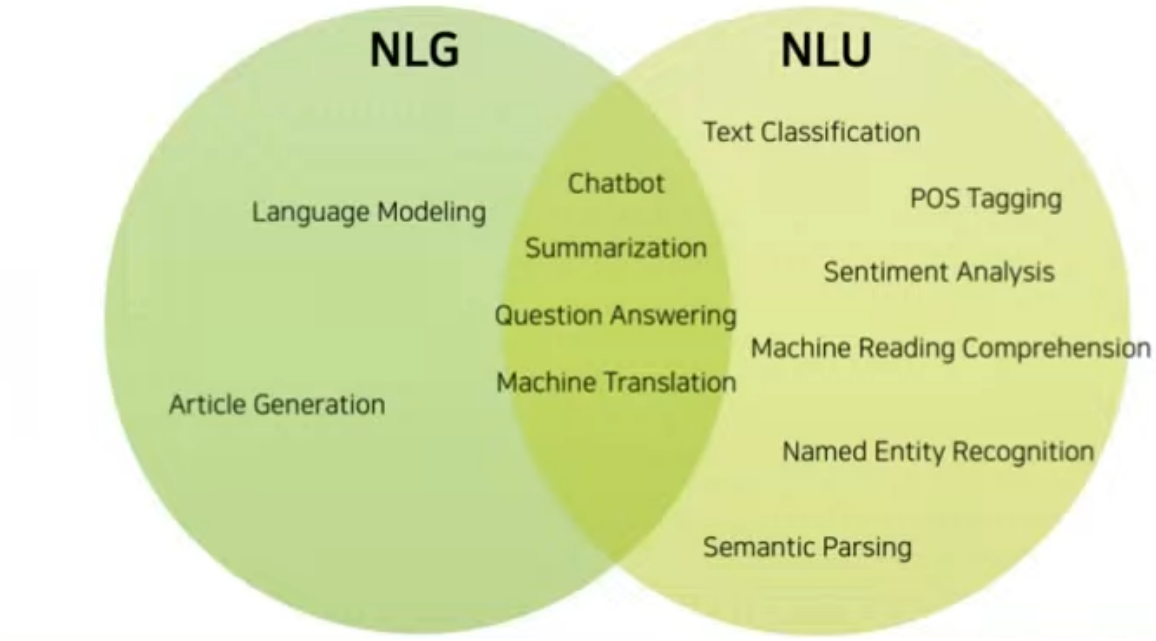
한국어 데이터 전처리

▼ 자연어 처리 (NLP)란

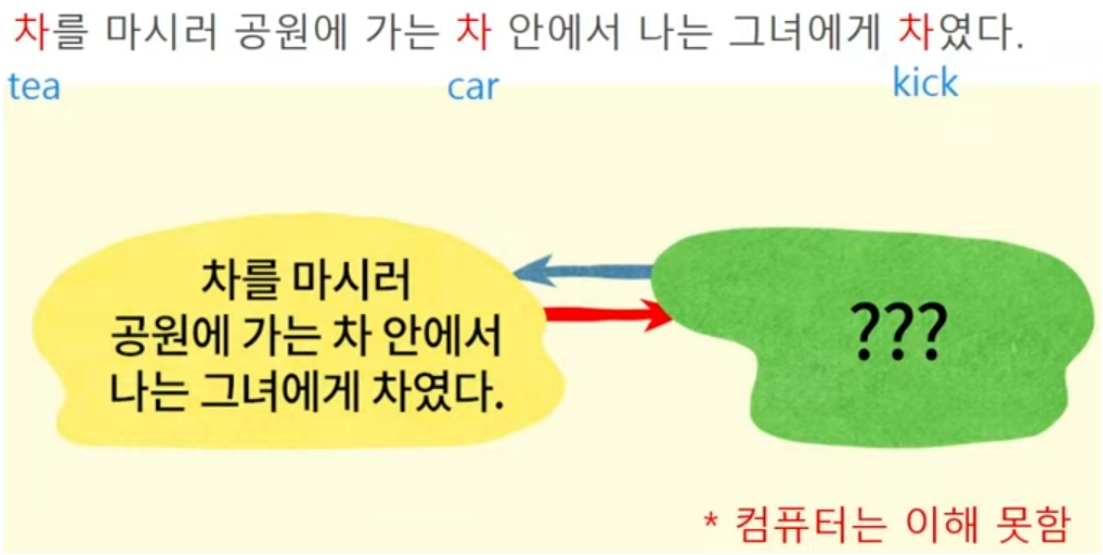
- 사람이 이해하는 자연어를 컴퓨터가 이해할 수 있는 값으로 변환하는 과정
- 컴퓨터가 이해하는 값을 사람이 이해할 수 있도록 다시 바꾸는 과정까지 포함

- 자연어 이해(NLU, Natural Language Understanding)
- 자연어 생성(NLG, Natural Language Generation)



▼ 한국어 자연어 처리가 어려운 이유

▼ 모호성 (Ambiguity)



- 띄어쓰기가 지켜지지 않는다
  - 한국어

    EX) 띄어쓰기를하지않아도읽을수있습니다
  - 영어

    EX) Youcanreadwithoutspacing

- 한국어는 교착어 이다
  - ‘그’ 라는 단어 하나에도 ‘그가’, ‘그를’, ‘그와’, ‘그는’과 같이 다양한 조사가 ‘그’라는 글자 뒤에 띄어쓰기 없이 바로 붙게 됨
  - 같은 단어임에도 서로 다른 조사가 붙어서 다른 단어로 인식이 되면 자연어 처리가 힘들고 번거로워지는 경우가 많음
  - 어간에 접사가 붙어 단어를 이루고 의미와 문법적 기능이 정해짐

▼ 같은 정보를 다르게 표현하기 (Paraphrase)

문장 1	여자가 김치를 어떤 남자에게 집어 던지고 있다.
문장 2	여자가 어떤 남자에게 김치로 때리고 있다.
문장 3	여자가 김치로 싸대기를 날리고 있다.
문장 4	여자가 배추 김치 한 포기로 남자를 때리고 있다.

* 문장의 표현 방식이 다양, 비슷한 단어들이 존재

▼ 교착어, 고립어, 굴절어

교착어

어간과 어미가 명백하게 분리됨

하나의 형태소는 하나의 문법적인 기능을 함

한국어, 일본어, 터키어, 핀란드어, 헝가리어,,,

고립어

문법적인 형태를 나타내는 어미가 거의 없고 어순과 위치만으로 문법적인 형태를 나타냄

중국어, 태국어, 미얀마어, 티벳어

굴절어

단어의 활용 형태가 단어 자체의 변형으로 나타나는 언어로

어간과 접사가 쉽게 분리되지 않음

어휘 자체에 격, 품사 등을 나타내는 요소가 포함됨

인도어, 유럽어, 러시아어

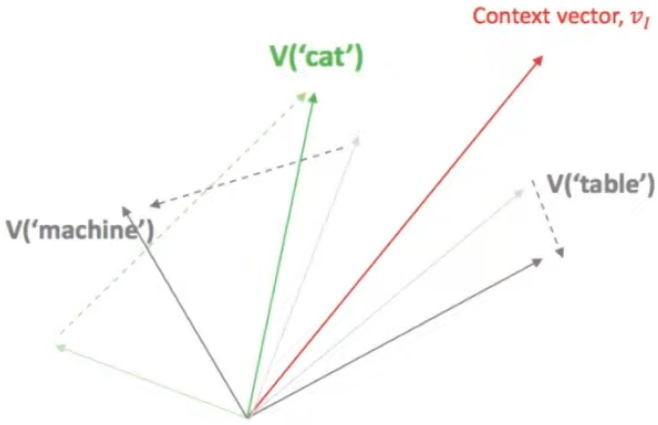
영어는?

- 단어적으로 보면 굴절어 요소가 남아있지만 문장으로 보면 고립어

▼ 전처리는 왜 필요한가

- 한글로 된 데이터를 크롤링 또는 오픈 데이터를 가져다 쓰려고 할 때, 맞춤법이 틀린 것들도 많다 (한글 맞춤법을 잘 지키지 않는 SNS 피드 데이터 등)
- 사소한 차이는 `임베딩 벡터`로 보면 큰 차이일 수 있다

▼ 정제하지 않은 데이터와 정제된 데이터는 분석 결과에서 많은 차이를 보인다



▼ 워드 임베딩 Word Embedding

- 단어를 벡터로 표현하는 방법으로, 단어를 밀집 표현으로 변환
- 밀집 벡터를 워드 임베딩 과정을 통해 나온 결과라고 하여 임베딩 벡터(embedding vector)라고도 함
- 워드 임베딩 방법론으로는 LSA, Word2Vec, FastText, Glove 등이 있음

▼ 전처리 방법

프로젝트에서 한글 데이터를 다룬다면 한글 전처리 방법들을 숙지하고 있어야 한다

▼ Basic

- 기초적인 전처리
- html tag 제거 (크롤링한 html 원문 데이터일 경우)
- 숫자, 영어, 특수문자 등 필요하지 않은 언어 제거
- Lowercasing

- “@%*=()/+ 와 같은 punctuation(문장부호) 제거
- Emoji 및 BMP (유니코드에서 Basic Multilingual Plane(기본 다국어 평면)) 제거

기초 전처리는 데이터를 적재, 전송 등 다른 용도로 사용할 때에도 필요하다

▼ Tokenize

- 자연어 처리는 텍스트를 토큰 단위로 나눈다
- 특히 한국어에서 띄어쓰기는 문맥과 의미를 구분하는데 큰 영향을 준다
- 모든 공백을 없앤 후 문맥에 따라 띄어 쓴 문장을 만드는 것이 좋은 방법

```
너무기대를 안했나봐 (원문)
-> 너무기대를안했나봐
-> 너무 기대를 안 했나봐
```

Spell Check

▼ Pos Tag

- 품사 태깅
- 품사를 붙이는 행위를 PoS Tagging이라고 한다.
- 형태소 분석은 의미있는 가장 작은 단위의 말(형태소)을 분석한다라는 뜻
- Pos Tagging 즉 품사 태깅 행위를 현업에서는 구분없이 동의어로 상당히 자주 사용함
- 형태소 분석은 말 그대로 형태소를 분석하는 모든 행위(어근,접두사/점미사 등 속성 구조 파악)
- konlpy의 형태소 분석기 및, Khaiii 등 여러가지 분석기가 나와 있으며 컨텐츠에 따른 정확도를 확인하여 선택
- 영어는 NLTK는 자연어 처리 및 문서 분석용 파이썬 패키지에서 많이 사용
- 품사가 제대로 태깅 되어야 양질의 분석이 가능하다
- 최근의 ELMo나 BERT 같은 Contextualized Word Embedding 방법에서는 단어 주변의 문맥 정보를 전체적으로 사용하기 때문에 주요 품사만 사용하는 방법

은 효과가 안 좋을 수 있다.

▼ Stemming

- 어간 추출
- 주어진 단어에서 핵심 의미를 담고 있는 부분을 찾는 과정
- 단어의 의미를 담고 있는 어간과 문법적 역할을 하는 접사를 분리하는 방식으로 동작
- 동사를 원형으로 복원한다. (입니다->이다)로 바꾸어 줌

▼ Stopwords

- 불용어처리
- 갖고 있는 데이터에서 유의미한 단어 토큰만을 선별하기 위해서 큰 의미가 없는 단어 토큰을 제거하는 작업

Replacing and Correcting Words

▼ 학습

2-1.

- 경계인식 방식: 머신러닝을 이용한 문장 경계인식

(다중 클래스 분류 모델, 다중 손실을 이용한 공동 학습모델)

- 영역인식 방식: 띄어 쓰는 지점 주변 토큰의 영향을 고르게 받음

2-2. 문장 분리

- 영역인식 방식: 문장 분리의 경우 형태소 분석으로 종결어미를 구분, 문장의 CRF 결과로 판단하는 방법 등
- 한국어 문장분리 파이썬 라이브러리
- 정확도는 약간의 편차들이 있음
- 정확도도 중요하나 대량 데이터처리시 속도도 고려해야 함

4-2. Lemmatisation(표제어)

6. 한국어 전처리 패키지

- PykoSpacing : 띄어쓰기 교정
- Py-Hanspell: 네이버 한글 맞춤법 검사기
- Customized KoNLPy: 영어는 띄어쓰기만 해도 단어가 잘 분리되나 한국어는 경우가 다르다.

형태소 분석기를 사용할 때 이러한 상황을 극복하기 위해 하나의 해결책으로써 형태소 분석기에 사용자 사전을 추가

- 특정 도메인 업종, 특수 명칭 등을 사용하는 텍스트 분석에 유용

7. 실무에서 한국어 전처리

- 기존에 나와 있는 라이브러리르 100% 활용하지는 않음
- 내부에서 사용하는 용어,동의어,불용어 사전을 함께 운용해서 반영함
- 신조어도 주기적으로 반영
- AI 기반 확률적 전처리 방법에는 예외가 종종 발생,하며 대량의 데이터 처리시 발생하는 처리속도 문제도 있음

<결론>

한국어 전치리하는데 어떤 방법들이 있는지, 예외가 발생하는 이유에 대한 이해가 있어야
분석의 정확도를 높일 수 있다.