NLP 과정

텍스트 전처리, 피처 벡터화, 텍스트 분석		
텍스트 전 처리	텍스트 정규화 : 클랜징, 토큰화, 필터링(Stop words) 피처 벡터화 : BOW, 희소행렬, Count/TF-IDF	NLTK 설치
한글 텍스트 처리	형태소, 형태소 분석기(okt), Word2Vec(Skip-gram,CBOW), toji.model 활용 Word2Vec 모델 활용 (위키피디아 한국어)	KoNLPy 설치 Gensim 설치
감성분석	네이버 영화평점 감성분석 : 150k, 200k dataset IMDB 영화평 감성분석 : 지도 학습, 비지도 학습 ML(지도), NLTK WordNet, VADER(비지도) IMDB 신경망 모델(임베딩 학습)	DT, LR, RF, SVM, KNN GridSearchCV, Pipeline Keras
텍스트 분류	20뉴스그룹 : 텍스트 정규화, 피처 벡터화, ML 학습/평가 네이버 뉴스 : TF-IDF 사전 활용, MLP 학습/평가	DT, LR, RF, SVM, KNN Pipeline GridSearchCV 결합 Keras
토픽 군집화 유사도	토픽 모델링(20 뉴스그룹) : 연관도 높은 Word 추출 문서 군집화, 군집 별 핵심 단어(Opinion Review Dataset) 문서 유사도 측정 : 코사인 유사도(Opinion Review Dataset)	LDA Clustering, cluster_centers_ Cosine_similarity

NLP

기계가 자연어를 이해하는 방법

‘Apple’은 사과인가, 회사인가?
- 사람은 문장의 컨텍스트(Context)를 통해 이해
- 기계는 단어를 수치적으로 표현해야 이해
- 이를 Word Embedding이라고 표현
- 텍스트를 숫자로 표현하는 것은 같은 문자라고 하더라도 다른 수치로 표현

“the cat sat on the mat”을 기계가 이해하려면 ..
- [the, cat, sat, on, mat] 단어별 분류 후 One hot encoding을 통해 0 ,1로 표현
- ‘cat’는 [0, 1, 0, 0, 0, 0], ‘the’의 경우 [1, 0, 0, 0, 0]으로 나타냄

단어를 사전을 사용하여 매핑하고 이를 벡터로 변환
- 벡터화를 시켜 숫자로 표현하면 분류나 회귀 등의 다양한 분석이 가능

NLP

자연어 관련 용어

- Corpus(말뭉치): 텍스트(문서)의 집합

- Token(토큰): 단어처럼 의미를 가지는 요소

- Morphemes(형태소): 의미를 가지는 언어에서 최소 단위
  형태소는 언어학에서 일정한 의미가 있는 가장 작은 말의 단위로 발화 체 내에서 따로 떼어낼 수 있는 것을 의미. 즉, 더 분석하면 뜻이 없어지는 말의 단위.

- POS(품사): ex) Nouns, Verbs

- Stopword(불용어): 조사, 접미사와 같이 자주 나타나지만 실제 의미에 기여하지 못하는 단어들

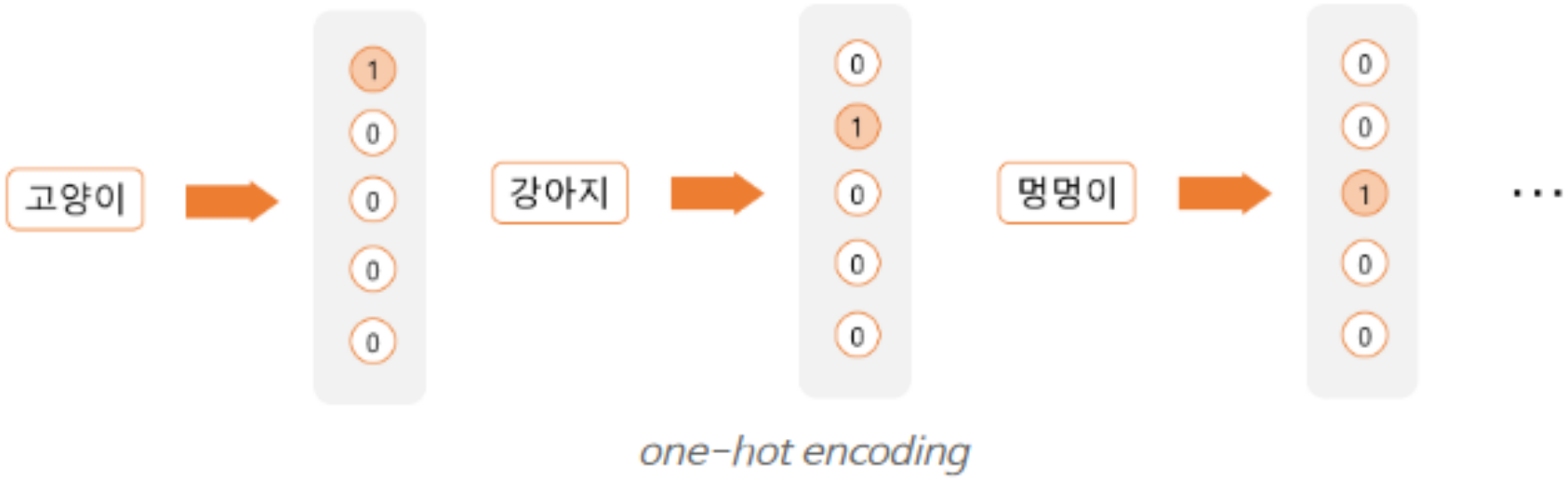
- Stemming(어간 추출): 어간만 추출하는 것을 의미(running, runs, run -> run)

- Lemmatization(음소표기법): 앞뒤 문맥을 보고 단어를 식별하는 것

NLP - 텍스트 벡터화

one-hot encoding(Sparse representation)

- 단어를 벡터로 바꾸는 가장 단순한 방법은 단어에 번호를 매기고, 그 번호에 해당하는 요소만 1이고 나머지는 0을 갖는 벡터로 변경
- 총 5개의 단어가 있는데 '강아지'라는 단어에 2번을 매겼다고 하자. 그러면 '강아지'는 2번째 요소만 1이고 나머지는 모두 0인 5차원의 벡터로 표현
- N개의 단어가 있다면 각 단어는 한 개의 요소만 1인 N차원의 벡터로 표현
- **단점은 벡터 표현에 단어와 단어 간의 관계가 전혀 드러나지 않는다는 점**



one-hot encoding

NLP - 텍스트 벡터화

Word Embedding(Dense representation, distributed representation)

- 단어를 벡터로 바꿀 때, 좀 더 똑똑하게 바꿔서 벡터에 단어의 의미를 담을 수 있다면 어떨까?
- 비슷한 의미의 단어들은 비슷한 벡터로 표현이 된다면?
- 단어와 단어 간의 관계가 벡터를 통해서 드러날 수 있다면?
- 벡터 간의 덧셈 뺄셈이 해당하는 단어 간의 의미의 합과 의미의 차로 반영
- 자연어 처리의 경우 대상은 텍스트이고, 이 텍스트의 속성을 표현해 놓은 것이 데이터
- 각각의 속성을 독립적인 차원으로 표현하지 않고 우리가 정한 개수의 차원으로 대상을 대응시켜서 표현
- '강아지'란 단어는 [0.16, -0.50, 0.20. -0.11, 0.15]라는 5차원 벡터로 표현

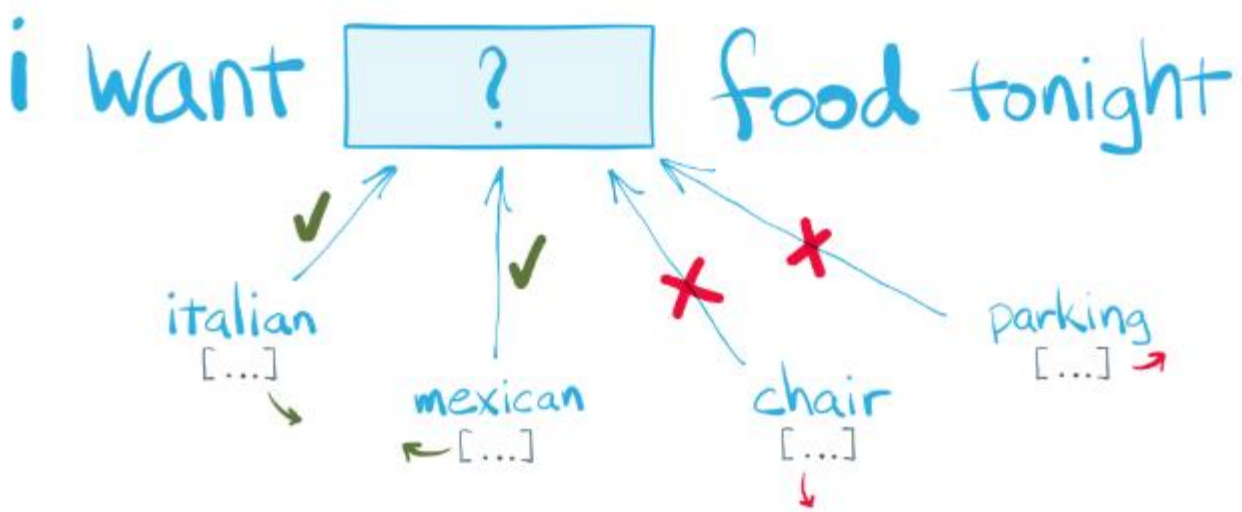
NLP - 텍스트 벡터화

Dense Representation의 장점

- 적은 차원으로 대상을 표현. 차원이 높으면 차원의 저주(curse of dimensionality)라는 문제 발생
- 더 큰 일반화 능력(generalization power). '강아지'와 '멍멍이'가 서로 비슷한 벡터로 표현이 된다면, '강아지'에 대한 정보가 '멍멍이'에도 일반화

아이디어

- predictive method란 지도학습을 통하여 맥락으로 단어를 예측하거나 단어로 맥락을 예측
- 이 예측 모델을 학습하면서 단어를 어떻게 표현해야 할지를 배우게 되고, 이 과정에서 비슷한 단어가 비슷한 벡터로 표현



NLP - 텍스트 벡터화

알고리즘

맥락으로 단어를 예측하는 CBOW(continuous bag of words) 모델

- 주변 단어, 다른 말을 맥락(context)으로 타겟 단어(target word)를 예측
- 주변 단어란 타겟 단어의 직전 몇 단어와 직후 몇 단어이며 타겟 단어의 앞 뒤에 있는 단어들을 타겟 단어의 친구들이라고 간주. 이 주변 단어의 범위가 window size
- 주위에 있는 단어가 입력이 되고 타겟 단어가 우리가 예측해야 하는 출력 값이 되는 문제를 푸는 과정에서 모델의 파라미터를 학습하고, 이렇게 학습된 파라미터가 단어들의 벡터 표현
- 파라미터가 학습되는 방식은 일반적인 머신 러닝, 딥 러닝 모델이 학습되는 방식과 같음
- CBOW에서 모델의 입력은 주변 단어인데 입력이 비슷하면 어떻게 될까? 출력도 비슷해질 것이다. 즉 주위에 있는 단어가 비슷하면 그 단어의 벡터 표현 역시 비슷해짐

단어로 맥락을 예측하는 skip-gram 모델

중심단어 업데이트 기회가 많기 때문에 말뭉치 크기가 동일하더라도 학습량의 차이로 CBOW에 비해 성능이 좋음(window=2이면 4배)

※ window : 학습할 단어를 연관시킬 앞뒤의 단어 수

NLP - 라이브러리

자연어 처리 Python 라이브러리

1. NLTK
   - 파이썬의 대표적인 자연어 처리 라이브러리
   - 방대한 데이터 세트와 서브 모듈을 가지고 있으며 NLP의 거의 모든 영역을 커버

2. KoNLPy
   - 우리나라 한글에 특화된 자연어 처리 라이브러리 :
   - 단어 품사 별 분류 : hannanum, kkma, okt, komoran, mecab

3. Gensim
   - 문서 사이의 유사도를 계산, 텍스트 분석을 돕는 라이브러리(Word2Vec 제공)
   - 토픽 모델링(Topic Modeling) 분야에서 가장 두각
   - Word Embedding : word2Vec

NLP - 라이브러리

자연어 처리 Python 라이브러리

4. Spacy
  - 연구보다는 생산용으로 고안된 고급 NLP를 위한 오픈 소스 라이브러리. 심도 깊은 데이터마이닝 가능

5. Scikit-learn
  - feature_extraction 서브패키지와 feature_extraction.text 서브 패키지는 전처리용 클래스 제공
  - DictVectorizer : 각 단어의 수를 세어놓은 사전에서 BOW 벡터 생성
  - CountVectorizer : 문서 집합에서 단어 토큰 생성, 각 단어의 수를 세어 BOW 인코딩한 벡터 만듬
  - TfidfVectorizer : TF-IDF 방식으로 단어의 가중치를 조정한 BOW 벡터 생성
  - HashingVectorizer : 해시 함수(hash function) 사용, 적은 메모리와 빠른 속도로 BOW 벡터 생성

NLP

Trend

- NLP 시장은 기계 번역, 정보 추출, 자동 요약, 질문 답변, 텍스트 분류, 감정 분석 및 기타(스팸 인식 및 언어 감지 등)로 분류

- NLP 솔루션의 채택이 증가함에 따라 지원 및 유지 보수와 같은 새로운 서비스의 필요성도 증가

- NLP의 진화는 기업과 소비자 모두에게 여러모로 중요한 영향을 미칠 것이며, 인간 언어의 의미와 뉘앙스를 이해할 수 있는 알고리즘으로 진화하면서 의료 산업이나 법률, 교육계 등 다양한 분야에서 어떤 파급 효과를 가져올 것인지 상상이 가능

- 특히 시장과 NLP 응용 솔루션에 기반이 되는 텍스트를 컴퓨터가 읽을 수 있는 형태로 모아 놓은 언어 자료 '말뭉치(말모둠, 글모둠)' 양이 클수록 AI가 인식(이해)할 수 있는 자연어의 정확도가 높아지며, AI가 얼마나 많이 학습 하느냐에 그 성능이 좌우

NLP

사례 : 카카오

- 카카오는 2018년 말부터 딥러닝 기반 형태소(形態素, morpheme) 분석기 '카이(khaiii)'를 오픈소스로 제공. 딥러닝을 통해 학습한 데이터를 활용해 형태소를 분석하는 모델

- 딥러닝 기술 중 하나인 콘볼루션 신경망(CNN, Convolutional Neural Network)을 이용해 음절기반으로 형태소를 분석하는 방법을 채택

- khaiii는 "Kakao Hangul Analyzer III"의 첫 글자들만 모아 만든 이름으로 카카오에서 개발한 세 번째 형태소분석기. 두 번째 버전의 형태소분석기 이름인 dha2 (Daumkakao Hangul Analyzer 2)를 계승

※ 형태소분석기는 단어를 보고 형태소 단위로 분리해내는 소프트웨어. 이러한 형태소분석은 자연어 처리의 가장 기초적인 절차로 이후 구문 분석이나 의미 분석으로 나아가기 위해 가장 먼저 이루어져야 하는 과정(한국어 위키피디아에서 인용)