자연어 처리의 시작

1. 자연어 처리 활용 분야와 트렌드

1) **학회**

1. Natural language processing (자연어 처리)

- 학회 : ACL, EMNLP, NAACL
- 분야
  - Low-level parsing : Tokenization, stemming
  - Word and phrase level : NER(Named Entity Recognation), POS(Part-Of-Speech) tagging
  - Sentence level : 감성 분류(Sentiment Analysis), 기계 번역(Machine Translation)
  - Multi-sentence and paragraph level : 논리적 내포 및 모순관계 예측(Entailment Prediction), 독해 기반 질의응답(question answering), 챗봇(dialog systems), 요약(summarization)

2. Text mining (텍스트 마이닝)

- 학회 : KDD, The WebConf(前 WWW), WSDM, CIKM, ICWSM
- 분야
  - Extract useful information and insights from text and document data
  - 문서 군집화(Document clustering)
  - Highly related to computational social science : 통계적으로 사회과학적 인사이트 산출

3. Information retrieval (정보 검색)

- 학회 : SIGIR, WSDM, CIKM, Recsys

- 분야
  - Highly related to computational social science
  - 정보 검색 분야, 추천 시스템

2) **자연어 처리 분야의 트렌드**

- 컴퓨터 비전 혹은 영상 처리 분야, 인공지능과 딥러닝 기술이 가장 활발히 적용
- 기존 머신러닝과 딥러닝 기술로 자연어 처리 문제를 해결

→ 주어진 텍스트 데이터를 숫자로 변환하는 '워드 임베딩(Word Embedding)' 과정을 거치게 됨.

- 텍스트 데이터는 문장을 구성하는 순서 정보가 중요하기 때문에 이를 받아들일 수 있는 특화 모델에 대한 연구가 필요
  - ex) 'RNN(**Recurrent Neural Network)**' , LSTM, GRU 모델
- 최근 발표된 대부분의 모델들: 트랜스포머 모델을 기반
  - 영상/신약개발/시계열 예측 등에서도 다양하게 사용
- 자가지도 학습(self-supervised Learning)이 가능한 BERT, GPT 와 같은 모델 유행

2. 기존의 자연어 처리 기법

1) Bag-Of-Words (단어 가방 모형)

- 단어들의 순서는 전혀 고려하지 않고, 단어들의 출현 빈도에만 집중하는 텍스트 데이터의 수치화 표현 방법
- 단어를 벡터로 표현하기 위해서는 주어진 문장에 쓰인 단어들을 사전(Vocabulary) 형태로 저장하→ 단어들의 중복을 허용 X
- 저장된 단어들: 유니크한 카테고리 변수(Categorical variable)
  - 원-핫 인코딩(One-hot Encoding)를 이용해 벡터로 표현 가능.
  - 주어진 문장을 원-핫 벡터의 합(숫자)로 표현 가능

- '가방'이라는 개념을 통해 문장을 구성하고 있는 단어들을 가방에 순차적으로 정리하는 것.

- 가방에 있는 각 단어들은 각각 원-핫 벡터를 통해 숫자로 변환하며, 주어진 문장은 벡터의 합으로 표현

2) **Naive Bayes Classifier for Document Classification**

- 나이브 베이즈 분류기는 인공 신경망 알고리즘에 포함X
  - 머신러닝의 주요 알고리즘으로 분류에 있어 준수한 성능을 보여줌.
  - 베이즈의 정리(Bayes' theorem)를 이해할 필요가 있음.
    - 베이즈 정리: 조건부 확률을 계산하는 방법

Data	Doc(d)	Document (words, w)	Class (c)
Training	1	Image recognition used convolutional neural networks	CV
	2	Transformers can be used for image classification task	CV
	3	Language modeling uses transformer	NLP
	4	Document classification task is language task	NLP
Test	5	Classification task uses transformer	?

- 학습 데이터로 주어진 Traing 1~4 번 문장을 통해 Test data(5번 문장)을 CV, NLP 두 클래스 중에 한 곳으로 분류.
  - 5번 문장에 있는 각 단어들이 1~4번 문장에 몇 번 등장했는지를 조건부 확률로 계산
  - 단점: 다른 단어들이 분류하고자 하는 문장에 많이 등장했을지라도, Training data에서 1번이라도 등장하지 않았다면 모든 단어들의 확률 곱으로 인해 0으로 수렴

3. Work Embedding - Word2Vec

1) **Word Embedding**

: 각 단어를 좌표공간에 최적의 벡터로 표현하는(임베딩하는) 기법

EX)

- kitty : 아기 고양이
- cat : 고양이
- hamburger : 햄버거

위 단어들을 벡터를 통해 좌표공간으로 표현 → kitty'와 'cat'은 비슷한 위치

,'hamburder'는 꽤 먼 거리에 표현

    - 유사한 단어는 가까이, 유사하지 않은 단어는 멀리 위치하는 것을 '**최적의 좌표값**'으로 표현 가능

2) **Word2Vec Idea**

- "문장 내에서 비슷한 위치에 등장하는 단어는 유사한 의미를 가질 것이다" 에서 출발
- 주변에 등장하는 단어들을 통해 중심 단어의 의미가 표현될 수 있다
- 워드를 토크나이징(Tokenizing)해준 후, 유니크한 단어만 모아서 사전(Vocabulary) 만들기.
- 그 이후 문장에서 중심 단어를 위주로 학습 데이터를 구축.
    - EX) “I study math”라는 문장의 중심단어가 study 라고 한다면, (I study), (study I) , (study math) 와 같은 단어쌍을 학습 데이터로 구축
- **토크나이징(Tokenizing)** : 말그대로 문자(Text)를 컴퓨터가 이해할 수 있는 Token이라는 숫자 형태로 바꿔주는 행위

3) **Word2Vec의 계산**

- 문장의 단어의 갯수만큼에 Input, Output 벡터 사이즈를 입력/출력.
- 연산에 사용되는 히든 레이어(hidden layer,은닉 층)의 차원(dim)은 사용자가 파라미터로 지정 가능
- Tensorflow나 Pytorch와 같은 프레임워크에서는 임베딩 레이어와의 연산은 0이 아닌 1인 부분,
    - ex) [0,0,1]의 벡터인 경우는 3번째 원소와 곱해지는 부분의 컬럼(column)만 뽑아서 계산
- 결과값으로 나온 벡터는 softmax 연산을 통해 가장 큰 값이 1, 나머지는 0으로 출력.
- 연산이 반복되면서, 같이 등장하는 단어들 간의 벡터 표현이 유사해짐.

4) **Word2Vec의 특성**

- 워드투벡터를 통해 단어를 임베딩
  - queen - king, woman - man aunt - uncle 의 벡터가 비슷.

  → 여성과 남성의 관계성을 잘 학습했다는 것을 의미

5) **Application of Word2Vec**

- Word2Vec은 그 자체로도 의미가 있지만, 뿐만 아니라 다양한 테스크에서 사용
- Machine translation : 단어 유사도를 학습하여 번역 성능을 더 높여줌
- Sentiment analysis : 감정분석, 긍부정분류를 도움
- Image Captioning : 이미지의 특성을 추출해 문장으로 표현하는 테스크를 도움

4. Word Embedding - GloVe

1) **Glove : Global Vectors for Word Representation**

- Glove는 Word2Vec과 다르게 사전에 미리 각 단어들의 동시 등장 빈도수를 계산
- 단어간의 내적값과 사전에 계산된 값의 차이를 줄여가는 형태로 학습
- Word2Vec는 모든 연산을 반복하지만, Glove는 사전에 계산된 Ground Truth를 사용해 반복 계산을 줄일 수 있음.

→ **Word2Vec보다 더 빠르게 동작하며, 더 적은 데이터에서도 잘 동작.**

2) **사전 학습된 Glove 모델**

- 사전에 이미 대규모 데이터로 학습된 모델이 오픈소스로 공개 되어 있음.
  - 위키피디아 데이터를 기반으로 하여 6B token만큼 학습 되었으며, 중복 제거 시에도 단어의 개수가 무려 40만개(400k)
- 학습된 모델을 나타낼 때 뒤에 붙는 "uncased": 대문자 소문자를 구분하지 않는다는 의미
- "cased: 대소문자를 구분한다는 의미.
  - ex) Cat과 cat이 uncased에서는 같은 토큰으로 취급되지만, cased에서는 다른 토큰으로 취급