국가종합전자조달시스템 신기술 적용 방안연구

忠 南 大 學 校 大 學 院

컴퓨터工學科 데이터 및 소프트웨어工學專攻

안 도 성

2022 年 9 月

목 차

그 림 목 차

표 목 차

제 1 장 서 론

1.1 연구의 배경과 목적

인공지능 시대가 도래 하면서 정형·비정형 데이터 활용에 대한 인식전환과 기계학습을 통한 문제점 개선연구가 더욱 활발하게 이루어 지고 있다. 그 중에 텍스트 분석 분야는 꾸준히 주목 받는 연구 분야 중 하나로 자리하였다. 다양한 매체에서 사용자에게 필요한 물품이나 용역의 정보를 일련의 알고리즘을 통해 가장 알맞게 노출시킬 수 있다는 점은 빅데이터를 활용한 인공지능 분야를 통해서 지향하는 데이터를 통한 가치 창출에 대한 기준에 부합한다.
대용량 데이터를 다룰 수 있게 되면서 다양한 이미지 파일과 텍스트 파일이 생성되고 분석에 활용 되고 있으며, 연구결과에 따라 생활의 편리성향상, 업무의 효율성 증가에 이바지하고 있다.

현대인은 자신이 원하는 정보를 찾는데 점차 많은 어려움을 느끼고 있다. 언제 어디서나 경제적인 부담 없이 편리하게 정보를 습득할 수 있는 인터넷이 가진 장점과는 별개로, 유용한 정보에 접근하는 데에는 물리적인 한계가 존재하기

때문이다. 모르거나 모를 수밖에 없는 정보량이 압도적으로 많이 생산되고 있어 특정 상황과 조건에 따른 답을 파악하기가 쉽지 않다. 이런 이유로 부정확하거나 잘못된 정보를 습득할 가능성도 이전보다 더 높아짐은 물론, 검색 정보를 이해하고 활용하는 수준이 낮아서 발생하는 새로운 형태의 불평등도 야기되고 있다.

그 중에서 텍스트 분석 기법은 분석 대상의 유형에 알맞은 방법을 적용 해야 한다. 대상 파일의 종류가 이미지 형태 유형과 텍스트 유형으로 나눌 수 있다. 이미지 유형의 경우 기존의 이미지파일로부터 텍스트로 변환하여 텍스트를 추출 해야 하고 텍스트 유형의 경우는 바로 텍스트 추출이 가능하기에 서로 다른 점이 있다. 최근 다양한 인공지능 알고리즘이 탄생하고 발전 되고 있음으로써 정확도와 신뢰도가 적용 알고리즘에 따라 큰 폭의 변화가 나타내어 지는 경우도 존재한다.

정확도가 높은 알고리즘을 국가종합전자조달시스템에 적용하기 위해선 알고리즘의 비교를 통한 기법의 적용이 필요하다. 현재까지의 인공지능 연구 동향은 프로젝트에서 적용 하면 계속 사용하면서 보정이 일어나지 않거나 사용률이 현저히 떨어지면서 사용하지 않는 경우도 발생 되어 왔다.

하지만 적용 대상 데이터와 과제에 따라 적용하는 기법이 달라야 하므로 데이터를 활용하는 인공지능에 대해 활발한 연구가 필요하다. 예를 들어, 이미지에서 텍스트를 추출해서 분석을 통해 물품의 세부정보를 더 많이 제공하거나 공고문이나 제안요청서의 텍스트를 분석해서 맞춤형 서비스를 제공하는데 기법을 사용하는 것이 효과적이라고 볼 수 있다

.

텍스트 분석 기법의 대표적인 연구로 전통적인 텍스트 분석 기법인 군집화, 분류 등이 있다. 다만 전통적인 방식의 텍스트 분석은 성능과 정확도면에서 최근 나오고 있는 딥러닝 기반의 모델에 비해 떨어지며, 효율성이 고려가 되지 않는다. 즉, 기존의 전통적인 텍스트 분석 기법에서는 단층이나 다층의 알고리즘으로 분석결과를 제공하기 때문에 단순히 데이터의 양이 많아지면 많아질수록 성능저하나 정확도가 분석하는 과정에서 누락되게 된다. 하지만 기존 연구와는 달리 기계학습(딥러닝)을 이용하여 텍스트 분석을 하는 과정은 학습이라는 강력한 요인이 존재한다. 학습의 경우, 더 많은 데이터와 시간을 반영 할 경우 정확도뿐만 아니라 성능, 데이터 융합 등이 분석에 중요한 부분으로 작용한다. 텍스트 분석을 진행하는 과정의 중요성으로 견주어 볼 때, 전통적인 분석 방식과 기계학습 방식을 상호 비교하는 전략을 사용하였다.

이러한 분석 과정 단계에서 사용되는 임계치의 값을 벗어나거나 도달하지 못한 값을 제외한 평균, 평균, 최소 값, 중간 값 등의 계산 방식은 학습되는 데이터 항목이 같더라도 알고리즘에 따라 달라지게 된다. 데이터의 양이 증가할수록 학습효과가 우수한 모델의 정확성이 늘어날 것이고 이는 전체 구성에 있어서 정량적인 비교를 할 수 있는 요인이 될 수 있다.

본 연구는 텍스트 분석에서 기존의 모델과 기계학습의 모델을 각기 적용 할 경우 나타나는 차이점을 알고자 각 기법의 개념과 정의를 알아보고 우수한 모델이 무엇인지 파악하는 연구를 진행하고자 한다.

1.2 연구과제의 구성

본 연구과제의 구성은 다음과 같다. 2 장에서는 텍스트 분석 개념과 분석과정, 주요 응용분야에 대해 알아보고, 주요 추천 기법과 본 연구에서 수행하고 있는 데이터 전처리, 적용 방법과 기법에 대해 자세히 기술한다.

3 장에서는 다양한 기계학습의 종류와 알고리즘을 확인하고 연구과제의 성격을 고려한 알고리즘 및 학습기법을 적용한 데이터를 통해 구현한 텍스트 분석 구성 및 동작 방법에 대해 서술하고 구현 내용에 대해 기술한다.

4 장에서는 제안한 방법을 이용한 텍스트 분석 시스템과 기존의 연구와의 비교를 통한 성능 분석을 하고, 마지막으로 5 장에서는 결론 및 향후 연구 방향을 기술한다.

제 2 장 관련 연구

2.1 텍스트 분석 개요

서로 다른 여러 텍스트로부터 컴퓨터를 이용해 자동으로 정보를 추출하면서 이전에는 알 수 없었던 새로운 정보를 발견하는 과정으로서 아래와 같이 표현된다.

1. 텍스트 자료들
   - 웹사이트, 도서 자료, 이메일, 평가글, 논문 등
2. 지식 구조의 발견과 표현
   - 사실(facts)
   - 비지니스 규칙,
   - 텍스트 안에서 등장하는 개체간의 관계성
     - 단순 문장 배열로만 남아 있지 않게 되는 것.
     - 이후 자동화 과정으로 통합시킬 수 있는 관계
3. 언어학, 통계학, 기계학습 기법 사용
   - 텍스트 형태 원천 데이터로부터 정보 구조화하고 모델링
   - BI(비지니스 인사이트, Business Intelligence),
   - EDA(탐색적 데이터 분석, Exploratory Data Analysis),

• 기타 연구나 조사로 확장

5. 비슷한 기법

• 정보 추출(Information Extraction)

• “텍스트” 데이터 마이닝

• 학습을 통한 통계적 패턴 인식

• KDD(Knowledge Discovery in Database)

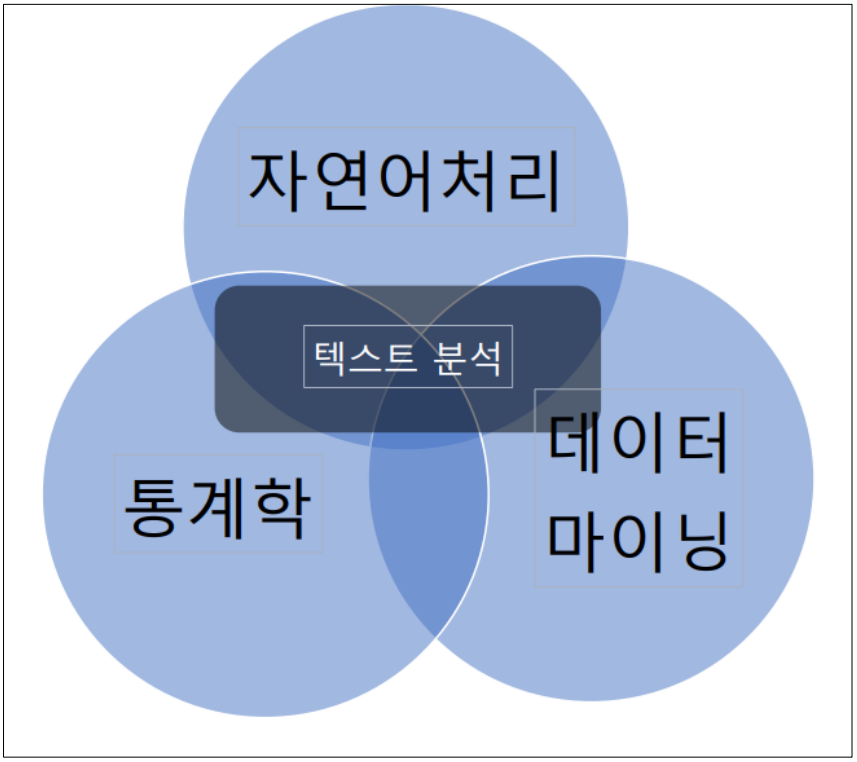

그림 1 텍스트 분석 구성

텍스트 분석과 유사한 텍스트 데이터 마이닝 이라고도 하는 텍스트 마이닝은 텍스트 에서 고품질 정보를 추출하는 프로세스이다 . 여기에는 “다른 문서 자원에서 정보를 자동으로 추출하여 이전에 알려지지 않은 새로운 정보를 컴퓨터가 발견하는 것“이 포함된다. 서면 리소스에는 웹 사이트, 책, 이메일, 리뷰 및 기사가 포함될 수 있다. 일반적으로 통계적 패턴 학습과 같은 수단을 통해 패턴과 추세를 고안하여 고품질 정보를 얻는다. 여기서 텍스트 마이닝의 세 가지 다른 관점을 구분할 수 있다: 정보 추출, 데이터 마이닝 및 데이터베이스의 지식 발견 프로세스. 텍스트 마이닝은 일반적으로 입력 텍스트를 구조화하고(일반적으로 일부 파생된 언어적 특징의 추가 및 다른 요소의 제거와 함께 구문 분석, 데이터베이스 에 후속 삽입) 구조화된 데이터 내에서 패턴을 파생 하고 마지막으로 산출물의 평가와 해석. 텍스트 마이닝의 ’고품질’은 일반적으로 관련성, 참신성의 일부 조합을 나타낸다. 일반적인 텍스트 마이닝 작업에는 텍스트 분류, 텍스트 클러스터링, 개념/개체 추출, 세분화된 분류의 생성, 감정 분석, 문서 요약 및 개체 관계 모델링(즉, 명명된 개체 간의 관계 학습)이 포함된다.

텍스트 마이닝은 다음과 같은 분석 기법을 통해 데이터 속에서 유의미한 정보를 발견한다.

1. 분류(categorization): 주어진 학습 자료로부터 분류 모델을 구축하여 새로운 텍스트 데이터를 분류

2. 군집(clustering) : 주어진 텍스트 데이터에서 속성을 추출하여 유사한 텍스트들끼리 묶는 과정

3. 개념추출(concept extraction): 텍스트의 의미를 대표하거나 요약할 수 있는 개념을 인식하고 추출하는 과정

4. 분류 사전 생성(taxonomy production): 텍스트 데이터에서 유의어 사전을 유도.

5.감성 분석(sentiment analysis): 텍스트에 표현된 긍정적이거나 부정적인 '감성'을 인식

6.텍스트 요약(summarization): 긴 분량의 텍스트를 짧은 텍스트로 요약

7. 개체 관계 모델링(entity relation modeling): 텍스트에서 추출한 개체 간의 관계를 인식하는 분석 기법

텍스트 분석에는 정보 검색 , 단어 빈도 분포를 연구하기 위한 어휘 분석, 패턴 인식, 태깅 / 주석, 정보 추출, 링크 및 연관 분석을 포함한 데이터 마이닝 기술, 시각화 및 예측 분석이 포함 된다. 가장 중요한 목표는 본질적으로 자연어 처리(NLP), 다양한 유형의 알고리즘 및 분석 방법을 적용하여 텍스트를 분석용 데이터로 변환하는 것이다. 이 프로세스의 중요한 단계는 수집된 정보의 해석이라고 할 수 있다.

일반적인 애플리케이션은 자연어로 작성된 문서 세트를 스캔하고 예측 분류 목적으로 문서 세트를 모델링 하거나 추출된 정보로 데이터베이스 또는 검색 색인을 채우는 것이다. 문서는 텍스트 마이닝을 시작할 때 기본 요소이며, 여기에서 문서를 일반적으로 많은 유형의 컬렉션에 존재하는 텍스트 데이터 단위로 정의한다.

텍스트 분석 이라는 용어 는 비즈니스 인텔리전스, 탐색적 데이터 분석, 연구 또는 조사를 위한 텍스트 소스의 정보 콘텐츠를 모델링하고 구조화 하는 일련의 언어적, 통계적 및 기계 학습 기술이다. 이 용어는 대략 텍스트 마이닝과 동일한 말이라고 할 수 있다. 실제로 Ronen Feldman은 “텍스트 분석“을 설명하기 위해 2004년에 “텍스트 마이닝“에 대한 2000년 설명을 수정했습니다. 후자의 용어는 이제 비즈니스 환경에서 더 자주 사용되는 반면 “텍스트 마이닝“은 1980년대로 거슬러 올라가는 일부 초기 응용 분야, 특히 생명 과학 연구 및 정부 정보에서 사용되고 있다.

텍스트 분석이라는 용어는 또한 독립적으로 또는 정형화 된 숫자 데이터의 쿼리 및 분석과 함께 비즈니스 문제에 응답하기 위해 텍스트 분석을 적용하는 것을 설명한다. 비즈니스 관련 정보의 80%가 비정형 형식(주로 텍스트) 에서 비롯된다는 것은 자명한 사실이다. 이러한 기술과 프로세스는 사실, 비즈니스 규칙 및 관계와 같은 지식을 발견하고 제시한다.

2.2 텍스트 분석 기법

전처리(Preprocessing)라고 하는 것은 비정형 텍스트 자료에서 정형 정보를 추출, 분석 가능한 형태로 저장/색인 하는 과정

1. 문서에서 텍스트 추출(“문서 필터”)
2. 토큰 분리
3. 키워드 추출
   ① 형태소 분석 (또는 N-gram)
   ② 품사 태깅
   ③ 가장 높은 점수 후보(1-best) 선정
   ④. (필요하면) 벡터 처리

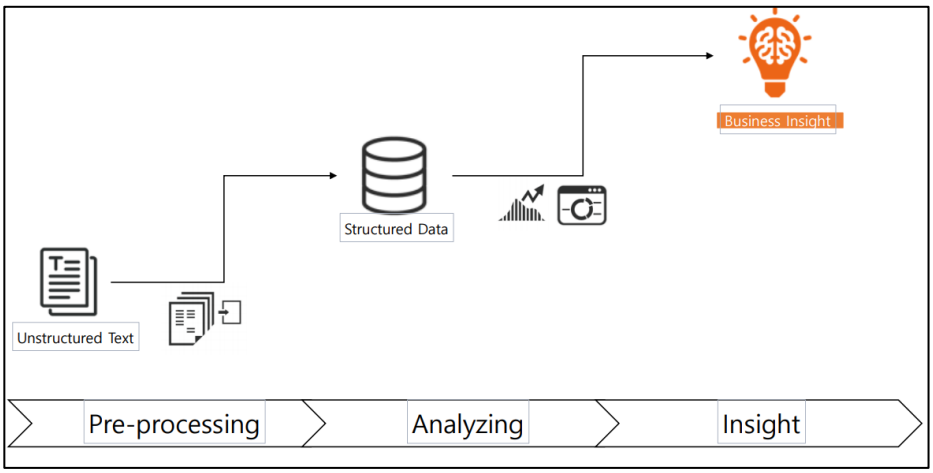

그림 2 전형적인 텍스트 분석 과정

분석단계로서는 여러 가지 절차를 통해 수행하고 있다.

- 차원 축소
- 정보 탐색
- 언어학적 분석
- 통계적 분석
- 자연어 처리
- 문장 분리
- 개체 명 인식
- 인명, 장소 명, 조직/기관 명, 상표
- “쓰기” (write, use, put on), “감기” (cold, roll)
- 대용어 해소
- 패턴 인식
- 스팸 필터링, 감정 분류, ...
- 네트워 분석
- 단어 간/문장 간 연관성 분석
- 시각화

문서 분석단계를 거치면서 가장 유의할 부분은 문서를 구성

하고 있는 단어 표현 방식이다.

보통 단어를 구성하는 집단을 단어사전이라고 표현하며 여러 가지 적용되는 모델이 있지만 주로 단어가방모델과 벡터 모델을 적용한다.

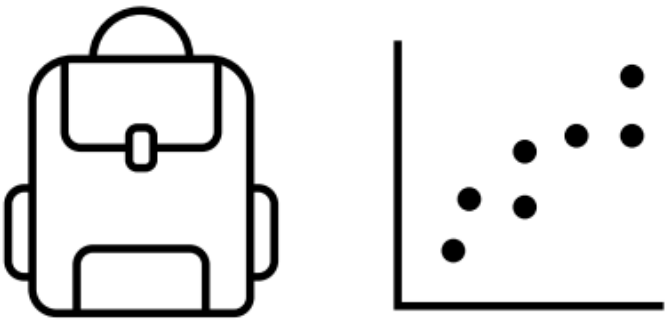

그림 3 단어가방모델(Bag of Words)과 벡터모델(Word Embedding)

단어가방 모델은 단어들의 집합으로 텍스트를 인식한다

예)" 산에는 꽃 피네 꽃이 피네 갈 봄 여름 없이 꽃이 피네

산에서 우는 작은 새여 꽃이 좋아 산에서 사노라네"

1) 단어 인식

{ "산", "꽃", "갈", "봄", "여름", "새" }

2) 출현 횟수 집계

[(“산” ,3), (“꽃” ,4), (“갈” ,1), (“봄” ,1), (“여름” ,1), (“새” ,1)]

3) 불용어 처리 (흔히 출현하는 단어)

[(“산” ,3), (“꽃” ,4), (“갈” ,1), (“봄” ,1), (“여름” ,1), (“새” ,1)

그림 4 색인어 추출 예시

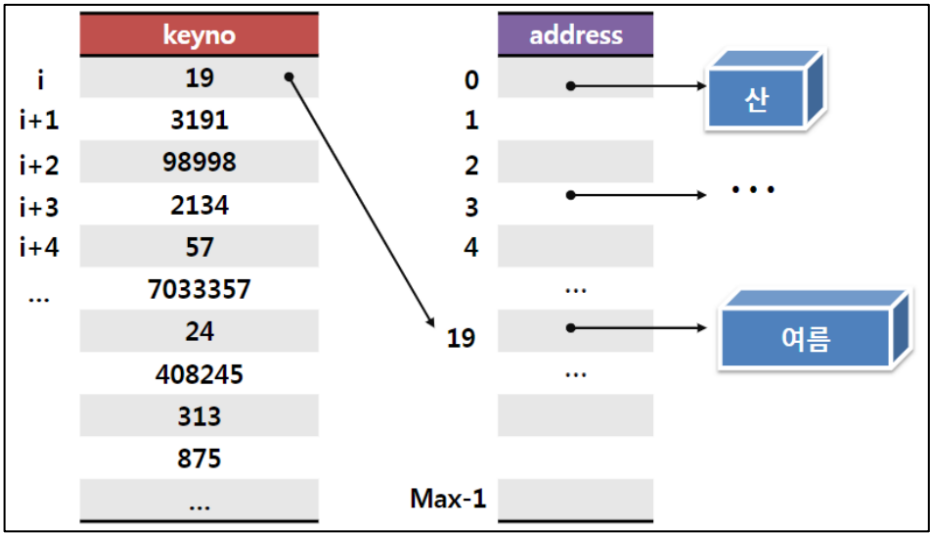

그림 5 숫자화 추출 예시

단어가방 모델에도 한계가 존재한다.

- 단어에 숫자를 대응
- 단순 키워드 검색(매칭)에는 효율적임
- 단어의 의미를 하나의 숫자로 표현하지는 못함
- 중의성(ambiguity) 문제

벡터(분산) 모델은 Zellig Harris (1954)와 Firth (1957)이 정의한 동일한 것을 묶는다는 개념에서 출발한다.

- “You shall know a word by the company it keeps!”

워드임베딩(Word2vec)이 대표적인 적용기법이며

- “비슷한 문맥에는 비슷한 의미가 존재”
- 딥러닝 기반의 자연어처리 시 최초 데이터 처리 영역을 이룸
- 딥하지 않은, 얕은 신경망(shallow neural-nets) 학습을 통해 생성
- “표현 학습(representation learning)”

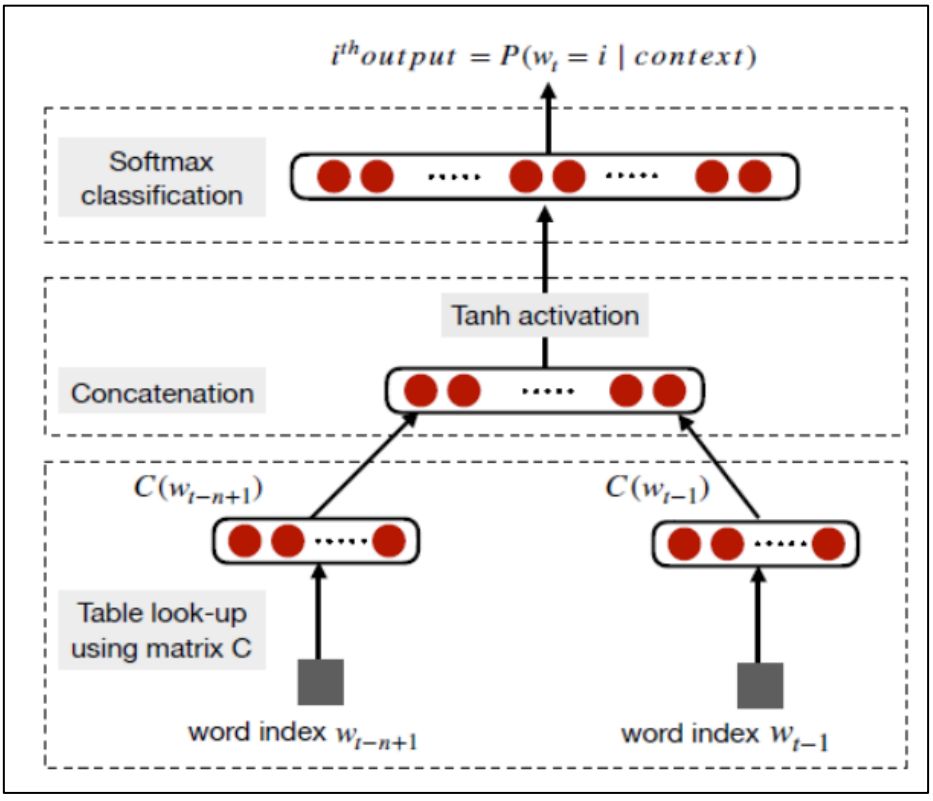

그림 6 워드임베딩 신경망 언어모델(2003)

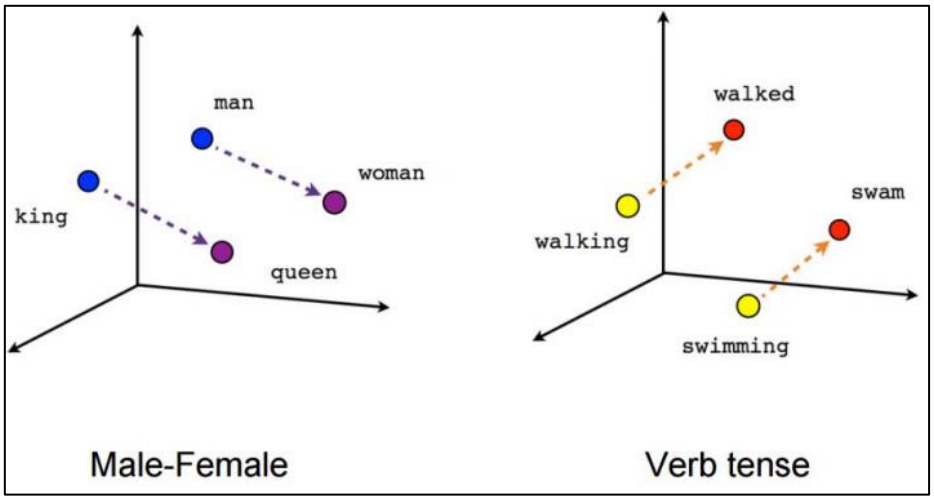

그림 7 워드임베딩 단어 추출

워드임베딩 또한 한계가 존재하는데 바로 중의성 문제이다.

- “아이 머리 **감기**는 엄마 “
- “요즘 **감기**는 무서워..”
- “가오리연에 실 **감기**”

위 예에서와 같이 ‘감기’ 라는 단어가 나타내는 상황 별 표현을 인식하고 파악하는데 있어서 정확한 인식은 쉽지 않은 실정이다.

최근 들어 더욱 정확하고 향상된 성능의 모델이 나왔는데

BERT(Pre-training of Deep Bidirectional Trnasformers for Language Understanding) 이다.

구글에서 개발한 NLP(자연어처리) 사전 훈련 기술이며, 특정 분야에 국한된 기술이 아니라 모든 자연어 처리 분야에서 좋은 성능을 내는 범용 Language Model이다. 11개 이상의 자연어처리 과제에서 BERT가 최첨단 성능을 발휘한다고 하며, BERT는 지금까지 자연어처리에 활용하였던 앙상블 모델보다 더 좋은 성능을 내고 있어서 많은 관심을 받고 있는 언어모델 이다.

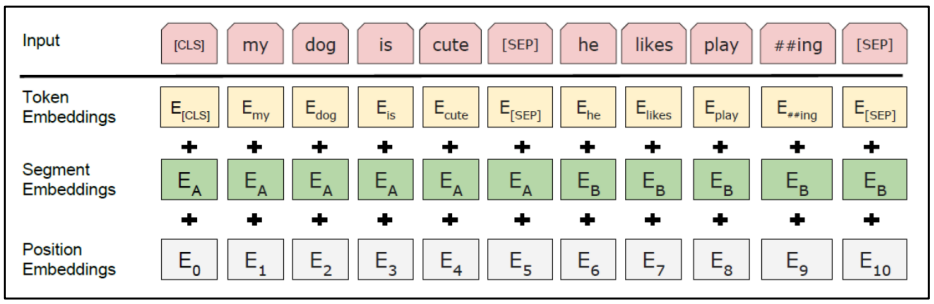

그림 8 BERT모델 (Google, 2018)

텍스트분석은 많은 응용분야가 있으며 대표적인 응용분야로서

- 텍스트 분류(classification, categorization)
- 텍스트 군집화(clustering)

- 개념/개체 추출
- 분류체계(taxonomy) 수립
- 감성 분석(sentiment analysis)
- 문서 요약
- 개체-관계 모델링

와 같은 방법이 존재하며, 보안, 생물, 의학 등 다양한 업무범위를 포함하여, 소프트웨어, 비지니스/마케팅, CRM(Customer Relation Management), VOC(Voice Of Customer) 분석, 주가 예측, 내용 기반 광고에 적용이 가능하다.

제 3 장 기계학습 기법

3.1 기계 학습 개요

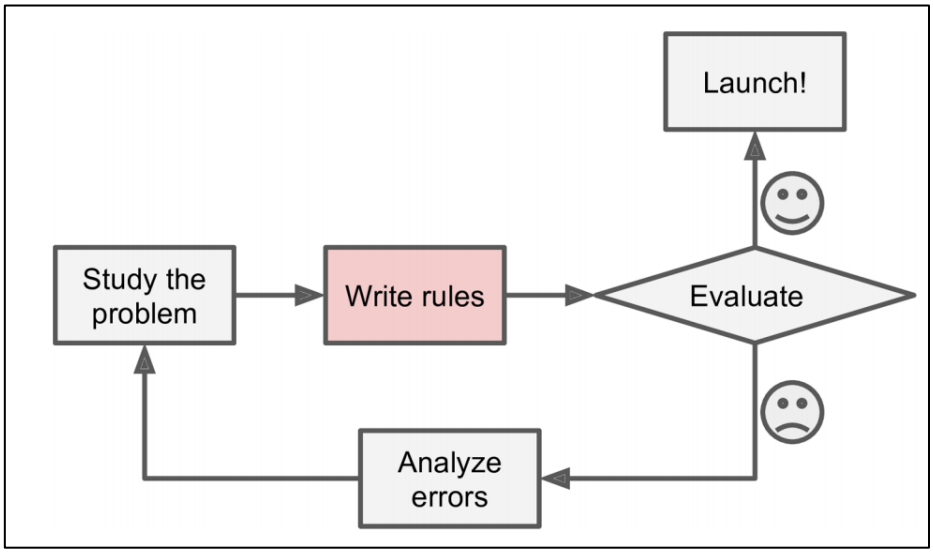

그림 9 전통적인 분석과정

위 그림9에서와 같이 전통적인 분석은 문제를 정의하고 수작업으로 패턴을 찾고, 분석하는 과정에 사람의 주관적인 판단이나 개입하는 것이 주된 과정이었다.

그러나 기계학습이 발전하면서 대상 데이터만 있으면 학습과 훈련을 통해 인공지능이 판단하고 분석하는 고품질의 모델 적용이 가능한 상황이 되고 있다.

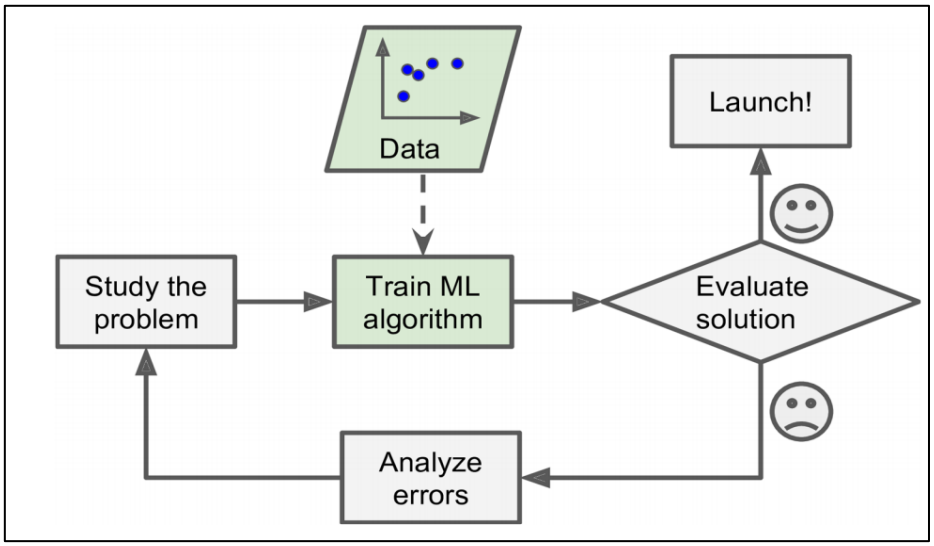

그림 10 기계학습 기반 분석과정

차세대 국가종합전자조달시스템에서도 빅데이터와 인공지능을 도입하여 이미지와 텍스트기반의 비정형화된 데이터를 가공하고 분석하려 하고 있으며 기계학습 기반의 분석을 적용하여 효율성을 향상시키는데 목적을 두고 있다.

이러한 기계학습의 종류에는 크게 3가지가 존재한다.

- 지도학습(Supervised Learning)
- 비지도학습(Unsupervised Learning)
- 기타(Others)

첫 번째로 지도학습은 정답 레이블링이 된 학습셋이 필요하다. 지도 학습(Supervised Learning)이란 간단히 말해 선생님이 문제를 내고 그 다음 바로 정답까지 같이 알려주는 방식의 학습 방법이다. 즉, 여러 문제와 답을 같이 학습함으로써 미지의 문제에 대한 올바른 답을 예측하고자 하는 방법이다.

전형적인 지도학습으로 가능한 분야는 다음과 같다.

- 분류(Classification)

분류는 주어진 데이터를 정해진 카테고리(라벨)에 따라 분류하는 문제를 말하며, darknet의 YOLO, network architecture는 GoodLeNet for image classification을 이용하여 이미지를 분류하고 있다. 분류는 맞다, 아니다 등의 이진 분류 문제 또는 사과다 바나나다 포도다 등의 2가지 이상으로 분류하는 다중 분류 문제가 있다

예) 문자인식(OCR(Optical Character Recognition))

- 회귀분석(Regression)

회귀는 어떤 데이터들의 Feature를 기준으로, 연속된 값(그래프)을 예측하는 문제로 주로 어떤 패턴이나 트렌드, 경향을

예측할 때 사용된다. 즉 답이 분류 처럼 1, 0이렇게 딱 떨어지는 것이 아니고 어떤 수나 실수로 예측될 수 있다.

예) 주가 예측, 날씨/기온 예보

지도학습의 적용 알고리즘으로는 다음과 같으며 현재는 비지도 학습 보다 더 효과적으로 인식되고 있다

- 결정트리/랜덤포레스트(Decision Trees & Random Forests)
- 나이브 베이즈(Naive Bayes)
- Support Vector Machine(SVM)
- 신경망(Neural Networks)
- 선형 회귀(Linear Regression)
- 로지스틱 회귀(Logistic Regression)

구분	분류	알고리즘
지도학습 (Supervised Learning),	Classification	KNN
		Naive Bayes
		Support Vector
		Machine Decision
	Regression	Linear Regression
		Locally Weighted Linear
		Ridge
		Lasso

표 1 지도학습 알고리즘 종류

지도 학습과는 달리 정답 라벨이 없는 데이터를 비슷한 특징끼리 군집화 하여 새로운 데이터에 대한 결과를 예측하는 방법을 비지도학습 이라고 한다. 라벨링 되어있지 않은 데이터로부터 패턴이나 형태를 찾아야 하기 때문에 지도학습보다는 조금 더 난이도가 있다고 할 수 있다. 실제로 지도 학습에서 적절한 피처를 찾아내기 위한 전처리 방법으로 비지도 학습을 이용하기도 한다.

전형적인 비지도 학습의 종류는 다음과 같다.

- 군집(Clustering)
- k-Means

기본적으로 많은 양의 데이터를 효과적으로 처리하기 위한 방법이다. 많은 양의 데이터를 처리하는 과정은 사용자와 아이템 간의 유연한 처리가 힘들게 된다. K-means 군집화 기법은 데이터를 K 개 만큼 군집화하고 해당 평가를 예측하는 시스템이다. 일반적으로 목적함수로 식(1)의 분산 함수가 사용되며, 이 값을 최소화 하는 방향으로 진행된다.

i 번째 클러스터 중심을 μ_i 클러스터에 속하는 점의 집합을 S_i라고 할 때 전체 분산은 다음과 같이 계산된다.

$$\boldsymbol{V = \sum_{i=1}^{k} \sum_{j \in S_i} |x_j - \mu_i|^2}$$

식(1)

위의 수식을 기반으로 하는 K-Means 군집화 과정은 다음과 같이 요약할 수 있다.

① 군집 수 K 개 설정

② 초기 K 개 군집의 중심 값을 설정

③ 각 아이템을 가장 가까운 거리에 있는 군집으로 재할당

④ 할당한 아이템을 포함하여 새로운 군집의 중심을 계산

⑤ ③, ④를 반복하며 식(1)이 최소화 될 때까지 반복 작업

K-Means 군집화는 데이터 개체 수에 선형 비례하는 복잡도를 지니기 때문에 계층적 군집화보다 적은 계산으로 양질의 군집을 발견할 수 있다.

- Hierachical Cluster Analysis(HCA)
- Expectation Maximization
- 시각화/차원 축소(Visualization & Dimensionality Reduction)
- t-distributed Stochastic Neighbor Embedding(t-SNE)
- 주성분 분석(Principal Component Analysis(PCA))
- 워드임베딩
- Word2Vec
- 연관 규칙 학습

구분	분류	알고리즘
비지도학습 (Unsupervised Learning)	-	Clustering
		K Means
		Density Estimation
		Exception Maximization
		Pazen Window
		DBSCAN

표 2 비지도학습 알고리즘 종류

기타학습으로 준-지도 학습(Semi-supervised learning), 강화 학습(Reinforcement learning), 배치 vs. 즉시 학습 (Batch vs. Online learning), 인스턴스 기반 학습 등이 있다.

기계학습이 성공적으로 이루어 지려면 데이터(Data), 학습률(Learning Rate), 테스트와 검증(Test & Validation)이 필수 불가결로 맞물려서 이루어져야 한다.

3.2 기계학습 데이터 적용 요소

기계학습에 있어서 다소 위험성도 있지만 도전적인 요소가 존재하는데 바로 데이터에 대한 탐색이다.

다양한 데이터는 일률적이거나 획일적이 아니므로, 통일성과 특징성을 찾아내는 것도 중요하다. 다만 수집 된 데이터를 의미 있는 데이터로 만들기 위해서는 아래와 같은 데이터의 존재 여부를 충분히 확인 하여야 한다.

- 불충분한 훈련 데이터
- 대표성이 떨어지는 데이터•

  예) 질 낮은 데이터
- 불필요한 특징 “garbage-in garbage-out”
- 과적합(Overfitting) 학습
- 부적합(Underfitting) 학습

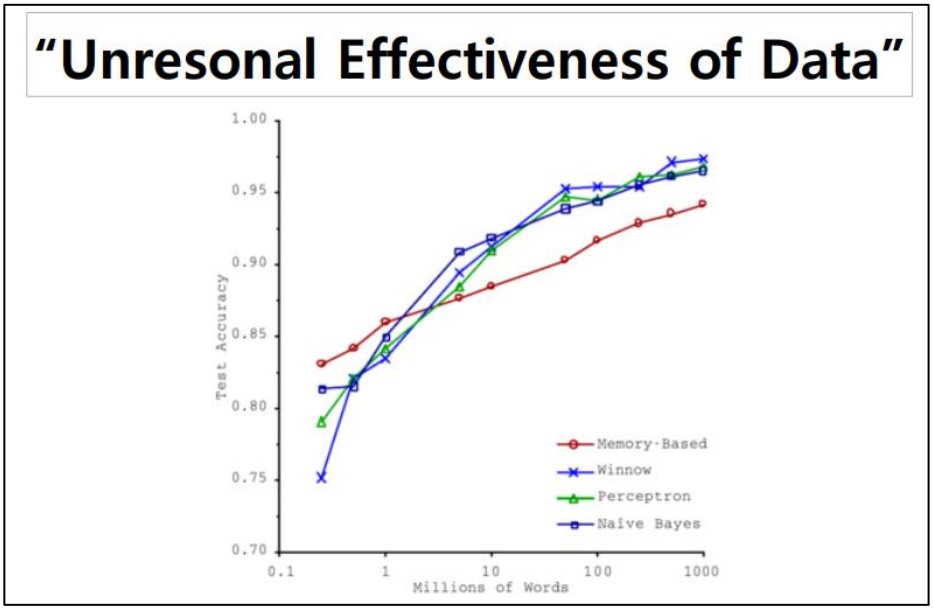

그림 11 데이터 종류에 따른 정확성

cost 를 최소화 하기 위한 gradient descent algorithm 을 구현하는 과정에서 learning rate 의 개념을 배웠다. 이 값을 보통 임의로 설정하게 되는데, 매우 큰 값으로 잡게 되면 그래프에서 하강하는 폭이 매우 크다는 것을 의미한다. 이 경우 학습이 이루어지지 않을 뿐만 아니라 최저점에 도달하는 것을 넘어서 그 반대 방향의 그래프에 도달할만큼 overshooting 이 발생할 수 있다.

반대로 learning rate 가 매우 작을 경우 최저점에 도달하는 데까지 소요되는 시간이 길어지고, global minimum 이 아닌 local

minimum 을 그래프의 최저점으로 인식할 수 도 있는 문제가 발생한다.

따라서 초반에 설정하는 learning rate 값에는 정답이 없다.
보통 0.01 에서 부터 시작하여 cost 함수의 값을 관찰해본다.

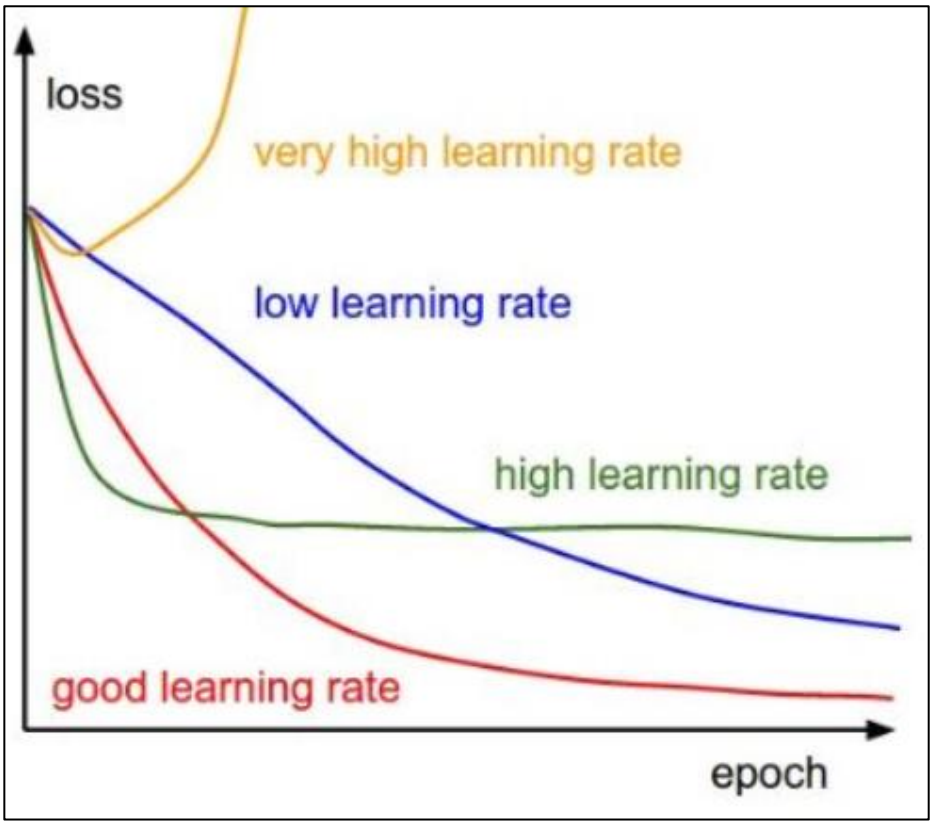

그림 12 학습률(Learning Rate)

마지막으로 훈련셋(training set)과 테스트셋(test set)을 분리하여 학습을 완료시켜야 한다. 진행 과정에서 일반화 에러가 발생 할 수도 있다. 이에 따라 위 셋에 포함되지 않는 새로운 케이스에 대한 에러율이 발생하기도 한다.

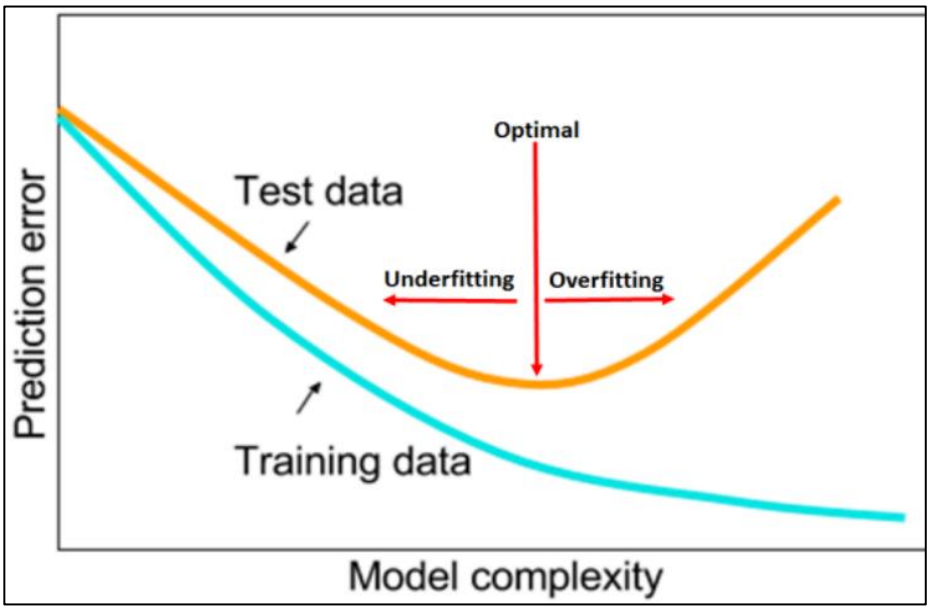

그림 13 훈련 셋과 테스트셋

데이터 셋을 결정하고 평가 값을 정하는 전략으로는 분류 알고리즘 표 1과 같은 알고리즘을 적용하여 연구과제 실습을 진행하고자 한다. 이에 4가지의 알고리즘을 적용하여 평균 값을 결정하였고 이렇게 적용한 데이터의 정확성을 토대로 모델과 알고리즘을 비교 구성하기로 하였다.

• 연구과제 알고리즘

① 결정 트리(Decision Tree)

② 나이브 베이즈(Naive Bayes)

③ 서포트 벡터 머신(Support Vector Machine)

④ 신경망(Neural Net)

3.2.1 Decision Tree

의사결정나무는 데이터를 분석하여 이들 사이에 존재하는 패턴을 예측 가능한 규칙들의 조합으로 나타내며, 그 모양이 '나무'와 같다고 해서 의사결정나무라 불립니다.

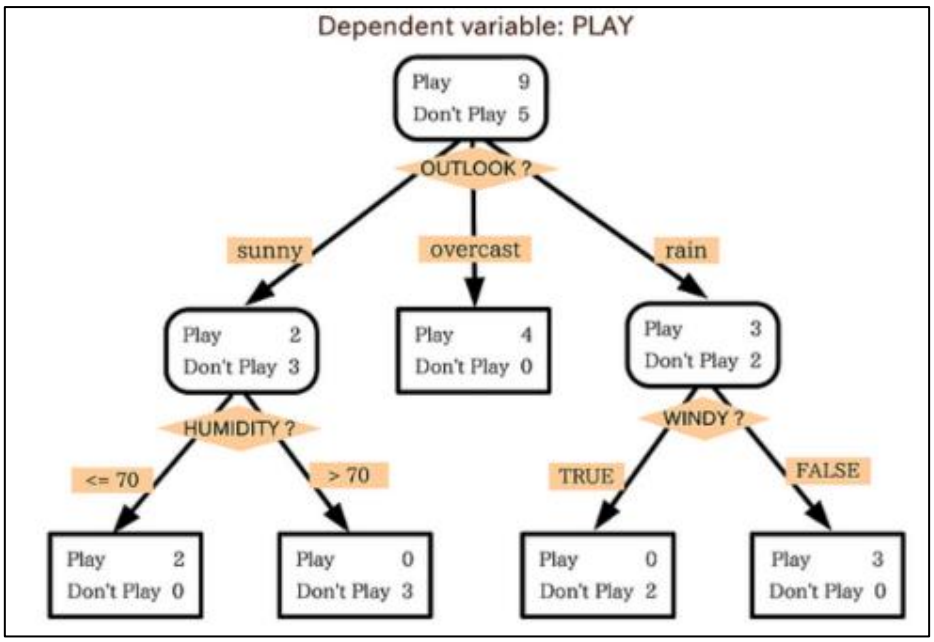

그림 14 결정트리 구성

3.2.2 Naive Bayes

나이브 베이즈는 조건부 확률 모델이다. 분류될 인스턴스들은 N 개의 특성 (독립변수)을 나타내는 벡터 {\displaystyle \mathbf {x}

=(x_{1},\dots ,x_{n})}{\displaystyle \mathbf {x}

=(x_{1},\dots ,x_{n})}로 표현되며, 나이브 베이즈 분류기는 이 벡터를 이용하여 k 개의 가능한 확률적 결과들 (클래스)을 다음과 같이 할당한다.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

우리가 하려는 분류문제에 맞게 변수를 정의해보자면,

y는 분류하고자 하는 class가 된다.

X는 input의 특징벡터가 된다.

X를 각 element를 나타내는 형태로 표현하면 아래와 같다.

$$X = (x_1, x_2, x_3, \cdots, x_n)$$

3.2.3 Support Vector Machine

서포트 벡터 머신(SVM: Support Vector Machine)은 분류 과제에 사용할 수 있는 강력한 머신러닝 지도학습 모델이다

서포트 벡터 머신(이하 SVM)은 결정 경계(Decision Boundary), 즉 분류를 위한 기준 선을 정의하는 모델이다. 그래서 분류되지 않은 새로운 점이 나타나면 경계의 어느 쪽에 속하는지 확인해서 분류 과제를 수행할 수 있게 된다.

만약 데이터에 2개 속성(feature)만 있다면 결정 경계는 이렇게 간단한 선 형태가 될 거다.

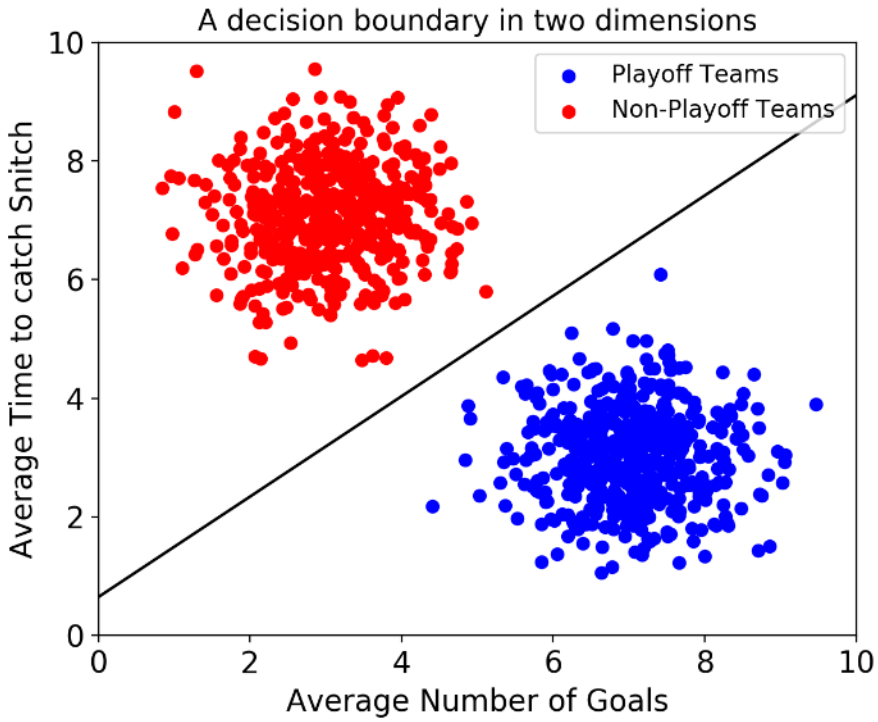

그림 15 SVM 2차원 결정경계 형태

3.2.4 Neural Net

딥러닝에서 가장 기본이 되는 개념은 바로 신경망(Neural Network)이다.

신경망이란 인간의 뇌가 가지는 생물학적 특성 중 뉴런의 연결 구조를 가리키며, 이러한 신경망을 본떠 만든 네트워크 구조를 인공신경망(Artificial Neural Network, ANN)이라 부른다.

인간의 뇌에는 약 1,000억 개의 수많은 뉴런 즉 신경세포가 존재하며, 하나의 뉴런은 다른 뉴런에게서 신호를 받고 또 다른 뉴런에게 신호를 전달하는 단순한 역할만을 수행한다. 하지만 인간의 뇌는 이러한 수많은 뉴런이 모여 만든 신호의 흐름을 기반으로 다양한 사고를 할 수 있게 되며, 이것을 컴퓨터로 구현하도록 노력한 것이 바로 인공신경망이다.

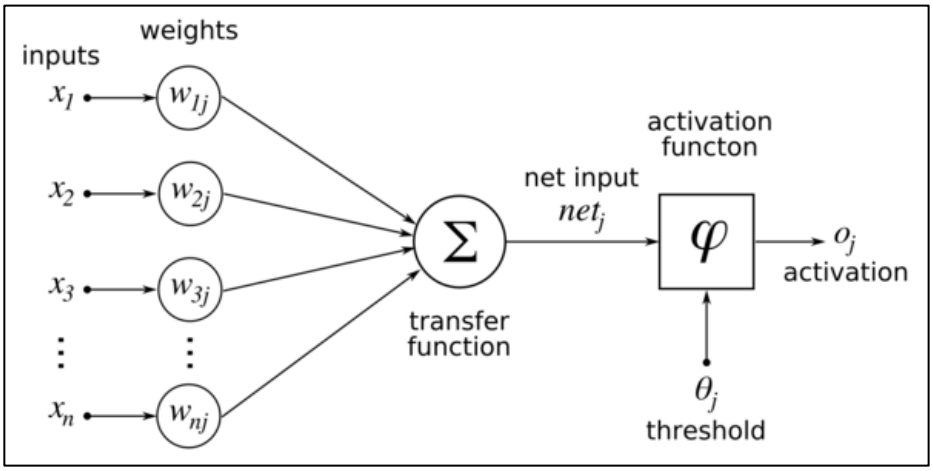

그림 16 신경망 형태

3.2.1.1 퍼셉트론(perceptron)

퍼셉트론(perceptron)이란 1957년 미국의 심리학자 프랑크 로젠블라트(Frank Rosenblatt)에 의해 고안된 인공신경망 이론을 설명한 최초의 알고리즘이라고 할 수 있다. 로젠블라트는 가장 간단한 퍼셉트론으로 입력층과 출력층만으로 구성된 단층 퍼셉트론(single layer perceptron)의 개념을 제안했다.

단층 퍼셉트론(single layer perceptron)이 동작하는 방식은 다음과 같다.

1. 각 노드의 입력치와 가중치를 서로 곱하여 모두 합한다.
2. 이렇게 합한 값을 활성화 함수가 가지고 있는 임계치 (선택의 기준이 되는 값)와 서로 비교한다.
3. 만약 그 값이 임계치보다 크면 뉴런은 활성화되고, 만약 임계치보다 작으면 뉴런은 비활성화 된다.

이러한 단층 퍼셉트론에서 가중치와 임계치를 적절히 변경하면, 상황에 맞는 적절한 의사결정을 내릴 수 있게 된다.

또한, 단층 퍼셉트론을 여러 개 조합하면 더욱 복잡한 문제

도 판단할 수 있게 되며, 이를 다층 퍼셉트론(MultiLayer Perceptron, MLP)이라고 부른다.
다층 퍼셉트론은 단층 퍼셉트론을 사용해서는 풀지 못하는 비선형 문제까지도 풀 수 있다.

일반적으로 인공신경망이란 이와 같은 다층 퍼셉트론의 조합이라 할 수 있다.

다층 퍼셉트론(multi-layer perceptron, MLP)는 퍼셉트론으로 이루어진 층(layer) 여러 개를 순차적으로 붙여놓은 형태다. MLP는 정방향 인공신경망(feed-forward deep neural network, FFDNN)이라고 부르기도 한다. 입력에 가까운 층을 아래에 있다고 하고, 출력에 가까운 층을 위에 있다. 신호는 아래에서 위로 계속 움직이며, MLP에서는 인접한 층의 퍼셉트론간의 연결은 있어도 같은 층의 퍼셉트론끼리의 연결은 없다. 또, 한번 지나간 층으로 다시 연결되는 피드백(feedback)도 없다. 제일 아래 입력 층과 제일 위 출력 층을 제외한 다른 층들은 숨겨져 있다고 해서 은닉층(hidden layer)이라고 하며, 아래 그림의 경우 총 3개 층이 있다.

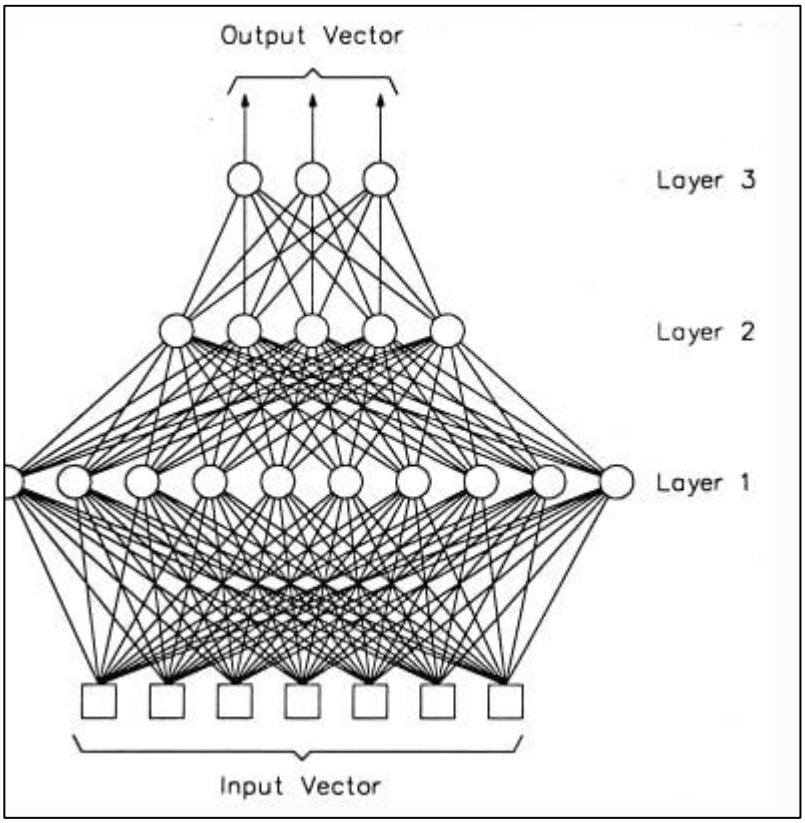

그림 17 다층 퍼셉트론 형태

(1) DenseNet

이미지 분류(classification)와 분할(segmentation), 객체 감지(detection)과 같은 비전 문제에서 탁월한 성능을 내는 CNN은 자연어처리 문제에도 효과적이다. 컨볼루션 연산층(convolution layer)의 필터(filter)가 문맥 파악에 중요한 부분만 도출하는 데

유리한 덕분이기 때문이다. 그 결과, CNN을 통과한 최종 벡터는 문장의 지역 정보를 보존하는 추상화 과정을 거쳐 단어나 표현의 등장 순서를 반영한 문장의 의미 정보(semantic information)를 표현할 수 있게 된다.

(2) 깊이 분리 컨볼루션 연산(Depthwise separable convolution)

형태소와 자모 정보를 반영한 통합 어절 임베딩을 모델에 바로 입력하면 전체 학습 시간이 지나치게 길어지는 문제가 발생할 수 있다. 각 임베딩별로 컨볼루션 연산이 이뤄지다 보니 매개변수(parameter) 수가 늘어나는 만큼 처리 시간이 비례해서 늘어나기 때문이다. 따라서 매개변수가 늘어나는 상황에 대비해 학습 속도나 추론 속도를 높일 필요가 있다.

이를 해결하고자 깊이별 분리 컨볼루션 연산[6]을 이용한다. 2D 이미지 데이터에 대한 깊이별 분리 컨볼루션은 매개변수 수 최적화를 통해 메모리 사용량은 줄이고 학습 속도를 높이고, 채널을 기준으로 각각 [필터 높이, 필터 너비, 1]와 [1, 1, 채널 수]로 분리한 두 종류의 새로운 필터로 각각 깊이별 컨볼루션

(Depthwise convolution)과 포인트별 컨볼루션(Pointwise convolution) 연산을 순차적으로 진행한다.

이 연구에서 깊이별 컨볼루션은 어절 간 관계를 고려한다. 주위 어절을 함께 고려한다는 측면에서 n-gram 확률을 분류에 사용하는 것과 비슷하다고 보면 된다. 포인트별 컨볼루션은 유효 데이터만 추려내고자 전체 채널을 하나로 압축해 컨볼루션 연산을 진행한다. 이렇게 하면 연산 속도를 높이는 데 도움이 된다.

300차원의 통합 어절 임베딩 벡터와 (300*3) 크기의 필터 256개를 이용한 컨볼루션 연산이 있다고 가정. 일반적인 필터를 이용한 1회의 컨볼루션 연산에는 총 (입력 채널 수*필터 높이*필터 너비*출력 채널 수)개의 매개변수가 필요하다. 여기서는 300*1*3*256=230,400개의 매개변수를 연산해야 한다. 반면, 깊이별 분리 컨볼루션에서는 각각 300*1*3*1=900개와 300*1*1*256=76,800개의 매개변수가 필요하다. 이를 합치면 총 77,700개로, 연산 효율이 33.7% 더 좋다는 사실을 확인해볼 수 있다.

그림 18 기존 컨볼루션 연산과 깊이 분리 컨볼루션에 사용하는 필터 예시

(3) 동적 셀프 어텐션(dynamic self-attention)

서로 연관성이 높으나 거리상 멀리 떨어진 어절이 서로 참조할 수 있게 하는 기법으로 어텐션이 있다. 다만 기존의 어텐션 기법은 입력 발화 길이에 제약이 없는 실제 서비스에 적합하지 않을 수가 있다. 어텐션 행렬의 형태가 어절 수에 제약을 받기 때문이다. 이에 입력 어절의 수에 관계없이 어텐션 벡터

의 계산 및 관리 기법인 동적 셀프 어텐션[7]을 적용했다.

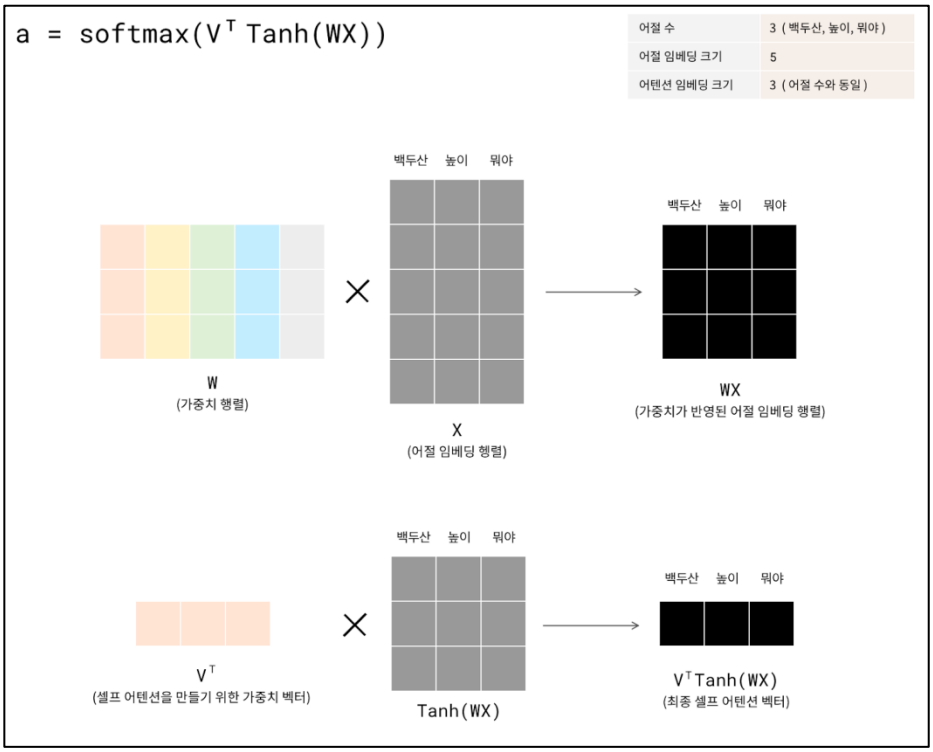

그림 19 기존 어텐션 기법

과정은 다음과 같다. 첫 번째, 각 어절 임베딩 벡터와 셀프 어텐션을 나타내는 동적 가중치 벡터(dynamic weight vector)를 곱해 각 어절의 문맥 점수를 구한다. 두 번째, 각 문맥 점수에 대한 소프트맥스(softmax) 연산을 거치면 중요도 점수(소프트맥스 확률값)를 얻게 된다. 세 번째, 어절 임베딩 벡터에 중요도

점수를 가중치로 두고 선형 결합(linear combination)한다. 네 번째, 이 가중합의 결과는 다시 동적 가중치 벡터로 재정의된다. 이 과정을 거치면 어절 길이에 제약을 가진 가중치 행렬을 사용하지 않으면서도 현재 보는 어절과 관련성이 높은 다른 어절의 중요도도 반영할 수 있게 된다.

3.2.1.2 회귀분석(Regression)

회귀 모델을 한 마디로 정의하면 '어떤 자료에 대해서 그 값에 영향을 주는 조건을 고려하여 구한 평균'이다. 통계학적인 관점에서 보면 모든 데이터는 아래와 같은 수식으로 표현할 수 있다고 가정한다.

$y=h(x1,x2,x3,...,xk;\beta 1,\beta 2,\beta 3,...,\beta k)+\epsilon y=h(x1,x2,x3,...,xk;\beta 1,\beta 2,\beta 3,...,\beta k)+\epsilon$

위 수식에서 h() 가 위에서 말한 조건에 따른 평균을 구하는 함수이며 우리는 이것을 보통 '회귀 모델'이라고 부른다. 이 함수는 어떤 조건(x1, x2, x3, …)이 주어지면 각 조건의 영향력(beta1, beta2, beta3, …)을 고려하여 해당 조건에서의 평균값을 계산한다.

뒤에 붙는 e 는 ‘오차항’을 의미하며,. 측정상의 오차나 모든 정보를 파악할 수 없는 점 등 다양한 현실적인 한계로 인해 발생하는 불확실성이 여기에 포함된다. 이것은 일종의 ‘잡음(noise)’ 인데, 이런 잡음은 이론적으로 보면 평균이 0이고 분산이 일정한 정규 분포를 띄는 성질이 있다.

우리가 회귀 분석을 한다는 것은 이 h() 함수가 무엇인지를 찾는 과정을 의미한다., 여기서 만든 회귀 모델의 예측치와 실측치 사이의 차이인 ‘잔차(residual)’가 정말 우리가 가정한 오차항(e) 의 조건을 충족하는지 확인하는 것이다. 이런 확인 작업을 ‘모델 검정’이라고 부른다. 최대한 실제 h() 에 가깝게 회귀 모델을 만드는 것이 목표이며, 만약 추정을 잘못하면 몇몇 중요한 조건들을 반영하지 못해 h()의 일부분만 회귀 모델로 만들 수 있는데 이것을 ‘underfitting’ 이라고 부른다. 반대로 실제 종속변수에 영향을 주는 조건이 아닌 단순한 ‘잡음’을 평균에 영향을 주는 조건으로 착각하고 모델에 반영할 수도 있는데 이런 것을 ‘overfitting’ 이라고 부른다. 보통 overfitting 문제를 많이 다루고 있지만 사실 현실 세계에서 우리가 만드는 대부분의 회귀 모델은 underfitting 문제도 같이

갖고 있다. 다시 말해, 우리가 만드는 대부분의 회귀 모델들은 h()의 일부분과 e의 일부분을 같이 반영하고 있는 상태이다. 단지 둘 중 어느 쪽이 더 많은 비중을 차지하느냐의 문제이다..

한편 모델을 만드는 이유는 현실을 좀 더 단순한 형태로 표현하기 위해서다. 그리고 이렇게 단순화하려면 불필요하다고 생각하는 정보들을 버리는 것으로 한다. 이때, 회귀 모델을 만들기 위해 버린 정보들이 무엇인지를 설명하는 것이 회귀 모델의 가정(assumption)이다. 즉, 회귀 모델을 만들 때 '실제 데이터는 이러 이러한 특성을 갖고 있다고 가정' 하는 것이다. 따라서 이런 가정이 많아질수록 모델은 좀 더 단순지고. 반대로 가정을 최소화할수록 모델은 복잡해진다.

여기서 설명할 다양한 회귀 모델들은 이렇게 데이터가 어떤 특성을 갖고 있다고 가정했느냐에 따라 나뉘어 진다.

가장 먼저 고려해야 할 가정은 선형성과 비선형성이다.

선형성(linearity)이란 어떤 집합의 원소쌍(아래 수식의 u와 v)에 대해서 함수 f()가 아래 두 가지 성질에 대해 만족함을 뜻한다

$$f(c \times u) = c \times f(u) (\text{여기서} c \text{는 상수})$$

$$f(u + v) = f(u) + f(v)$$

제 4 장 실험 및 분석

4 장에서는 자연어 처리를 적용한 토큰화의 개념을 확인하고 단어사전을 통한 적용 모델과 본 연구에서 제안하는 알고리즘을 고려한 모델의 성능을 평가하고 비교 분석해 본다. 동일한 환경하에서 데이터의 유사성에 대해 비교하고 본 연구에서 제시한 기법을 고려한 텍스트 분석방법의 정확도를 파악하고자 한다.

4.1 연구 과제 실험

본 연구에서는 실험을 위해 토큰화 방식으로 자연어 처리를 활용하였다.

자연어 처리에서 크롤링 등으로 얻어낸 코퍼스 데이터가 필요에 맞게 전처리 되지 않은 상태라면, 해당 데이터를 사용하고자 하는 용도에 맞게 토큰화(tokenization) & 정제(cleaning) & 정규화(normalization)하는 일을 하게 됩니다. 이번에는 그 중에서도 토큰화에 대해서 적용한다.

주어진 코퍼스(corpus)에서 토큰(token)이라 불리는 단위로 나누는 작업을 토큰화(tokenization)라고 한다. 토큰의 단위가 상황에 따라 다르지만, 보통 의미 있는 단위로 토큰을 정의. 여기서는 토큰화에 대한 발생할 수 있는 여러 가지 상황에 대해서 언급하여 토큰화에 대한 개념을 이해한다.

4.1.1. 단어 토큰화(Word Tokenization)

토큰의 기준을 단어(word)로 하는 경우, 단어 토큰화(word tokenization)라고 한다. 다만, 여기서 단어(word)는 단어 단위 외에도 단어구, 의미를 갖는 문자열로도 간주되기도 한다.

예를 들어. 아래의 입력으로부터 구두점(punctuation)과 같은 문자는 제외시키는 간단한 단어 토큰화 작업을 해보면.

구두점이란 마침표(.), 컴마(,), 물음표(?), 세미콜론(;), 느낌표(!) 등과 같은 기호를 말한다.

입력: Time is an illusion. Lunchtime double so!

이러한 입력으로부터 구두점을 제외시킨 토큰화 작업의 결과는 다음과 같다.

출력 : “Time“, “is“, “an“, “illustion“, “Lunchtime“, “double“, “so“

이 예제에서 토큰화 작업은 굉장히 간단하다. 구두점을 지운 뒤에 띄어쓰기(whitespace)를 기준으로 잘라냈다. 하지만 이 예제는 토큰화의 가장 기초적인 예제를 보여준 것에 불과하다.

보통 토큰화 작업은 단순히 구두점이나 특수문자를 전부 제거하는 정제(cleaning) 작업을 수행하는 것만으로 해결되지 않는다. 구두점이나 특수문자를 전부 제거하면 토큰이 의미를 잃어버리는 경우가 발생하기도 한다. 심지어 띄어쓰기 단위로 자르면 사실상 단어 토큰이 구분되는 영어와 달리, 한국어는 띄어쓰기만으로는 단어 토큰을 구분하기 어렵다.

4.1.2 .토큰화 중 생기는 선택의 순간

토큰화를 하다보면, 예상하지 못한 경우가 있어서 토큰화의 기준을 생각해봐야 하는 경우가 발생한다. 물론, 이러한 선택은 해당 데이터를 가지고 어떤 용도로 사용할 것인지에 따라서 그 용도에 영향이 없는 기준으로 정하면 된다. 예를 들어 영어권

언어에서 아포스트로피를(')가 들어가있는 단어는 어떻게 토큰으로 분류해야 하는지에 대한 선택의 문제를 보겠다.

다음과 같은 문장이 있다.

Don't be fooled by the dark sounding name, Mr. Jone's Orphanage is as cheery as cheery goes for a pastry shop.

아포스트로피가 들어간 상황에서 Don't 와 Jone's 는 어떻게 토큰화할 수 있을까?

- Don't
- Don t
- Dont
- Do n't
- Jone's
- Jone s
- Jone
- Jones

이 중 사용자가 원하는 결과가 나오도록 토큰화 도구를 직접 설계할 수도 있겠지만, 기존에 공개된 도구들을 사용하였을 때의 결과가 사용자의 목적과 일치한다면 해당 도구를 사용할 수도 있을 것이다.

케라스의 text_to_word_sequence 는 기본적으로 모든 알파벳을 소문자로 바꾸면서 마침표나 컴마, 느낌표 등의

구두점을 제거한다. 하지만 don't 나 jone's 와 같은 경우 아포스트로피는 보존하는 것을 볼 수 있다.

4.1.3. 토큰화에서 고려해야할 사항

토큰화 작업을 단순하게 구두점을 제외하고 공백 기준으로 잘라내는 작업이라고 간주할 수는 없다. 이러한 일은 보다 섬세한 알고리즘이 필요한데 그 이유는 다음과 같다.

4.1.4. 구두점이나 특수 문자를 단순 제외해서는 안 된다.

갖고 있는 단어들을 걸러낼 때, 구두점이나 특수 문자를 단순히 제외하는 것은 옳지 않습니다. 정제 작업을 진행하다보면, 구두점조차도 하나의 토큰으로 분류하기도 한다. 가장 기본적인 예를 들어보자면, 마침표(.)와 같은 경우는 문장의 경계를 알 수 있는데 도움이 되므로 단어를 뽑아낼 때, 마침표(.)를 제외하지 않을 수 있다.

또 다른 예로 단어 자체에 구두점을 갖고 있는 경우도 있는데, 숫자 사이에 컴마(,)가 들어가는 경우도 있다. 보통 수치를 표현할 때는 123,456,789 와 같이 세 자리 단위로 컴마가 있다.

4.1.5. 줄임말과 단어 내에 띄어쓰기가 있는 경우.

토큰화 작업에서 종종 영어권 언어의 아포스트로피(')는 압축된 단어를 다시 펼치는 역할을 하기도 한다. 예를 들어 what're 는 what are 의 줄임말이며, we're 는 we are 의 줄임말이다. 위의 예에서 re 를 접어(clitic)이라고 한다. 즉, 단어가 줄임말로 쓰일 때 생기는 형태를 말한다. 가령 I am 을 줄인 I'm 이 있을 때, m 을 접어라고 한다.

New York 이라는 단어나 rock 'n' roll 이라는 단어를 보자. 이 단어들은 하나의 단어이지만 중간에 띄어쓰기가 존재한다. 사용 용도에 따라서, 하나의 단어 사이에 띄어쓰기가 있는 경우에도 하나의 토큰으로 봐야하는 경우도 있을 수 있으므로, 토큰화 작업은 저러한 단어를 하나로 인식할 수 있는 능력도 가져야 한다.

4.1.6. 표준 토큰화 예제

이해를 돕기 위해 표준으로 쓰이고 있는 토큰화 방법 중 하나인 Penn Treebank Tokenization 의 규칙에 대해서 소개하고, 토큰화의 결과를 확인해보겠다.

규칙 1. 하이푼으로 구성된 단어는 하나로 유지한다.

규칙 2. doesn't 와 같이 아포스트로피로 '접어'가 함께하는 단어는 분리해준다.

4.1.7.. 문장 토큰화(Sentence Tokenization)

이번에는 토큰의 단위가 문장(sentence)일 경우이다. 이 작업은 갖고 있는 문장 단위로 구분하는 작업으로 때로는 문장 분류(sentence segmentation)라고도 부른다. 보통 갖고 있는 단어사전이 정제되지 않은 상태라면, 문장 단위로 구분되어 있지 않아서 이를 사용하고자 하는 용도에 맞게 문장 토큰화가 필요할 수 있다.

어떻게 주어진 사전으로부터 문장 단위로 분류할 수 있을까? 직관적으로 생각 해 봤을 때는 ?나 마침표(.)나 ! 기준으로 문장을 잘라내면 되지 않을까라고 생각할 수 있지만, 꼭 그렇지만은 않다. !나 ?는 문장의 구분을 위한 꽤 명확한 구분자(boundary) 역할을 하지만 마침표는 그렇지 않기 때문이다. 마침표는 문장의 끝이 아니더라도 등장할 수 있다.

제 5 장 결론 및 향후연구

기존의 전통적인 방식의 텍스트 분석 모델 기법 사용되던 알고리즘은 학습이라는 속성을 고려하지 않았던 때라서 결과에 대한 정확도가 낮았다. 본 연구에서는 기계학습의 최신 알고리즘 기법의 개념과 매칭하여 다양한 알고리즘을 찾아봄으로써 기존에 사용되던 단층 분석 위주의 알고리즘 보다 높은 예측 정확도를 얻을 수 있을 것이다.

텍스트 분석기법에서 예기치 않은 오분류 문장을 완전히 제어할 수 있어야 비로소 사용자에게 충분한 만족감을 선사하는 서비스를 제공할 수 있다고 보고 있다. 지금까지는 전통적 방식과 현대 방식의 모델까지 딥러닝 분류 모델에 적용했을 때 유의미한 성능 개선을 확인했다. 정답 유형 분류에 맞게 미세 조정된(fine-tuning)된 모델을 설정하는 실험에서는 괄목할만한 성능 개선을 이루길 바라며 현재 방식 이외에 좋은 방식에 대한 연구가 필요할 것이다.

참 고 문 헌

[1] Olson, David L.; Delen, Dursun "Advanced Data Mining Techniques" Springer; 1 edition (February 1, 2008) Olson, David L.; Delen, Dursun "Advanced Data Mining Techniques" Springer; 1 edition (February 1, 2008)

[2] Springer-Verlag. 2001. ISBN 978-1-55608-010-4.

[3] Weisstein, Eric Wolfgang. "Regression.

[4] Machine Learning, Tom Mitchell, McGraw Hill, 1997

[5] McCorduck 2004, p. 98, Crevier 1993, pp. 27?28, Russell & Norvig 2003, pp. 15, 940, Moravec 1988, p. 3, Cordeschi & 2002 Chap. 5.

[6] Minsky strongly believes he was misquoted. See McCorduck 2004

[7] 인공지능의 시대에 더 잘 살 수 있는 방법. 이경일

[8] 송주영. 강(强)인공지능과 약(弱)인공지능을 아시나요