인공지능 인문학

4주차 PART 1
인공지능을 활용한 인문학 연구 설계(1)

김정아

학습 목표

1 인공지능을 활용한 인문학 연구를 어떻게 설계할지 전반적 과정을 살펴 본다

2 주요 머신러닝·딥러닝 방법론을 이해하고 나에게 맞는 방법론을 선택할 수 있다

01

인공지능 인문학 연구 설계

인공지능으로 인문학 연구하기

주제 및 방법론 선정

- 주제 및 연구 방법론과 맞는 인공지능 모델 선정

연구 진행

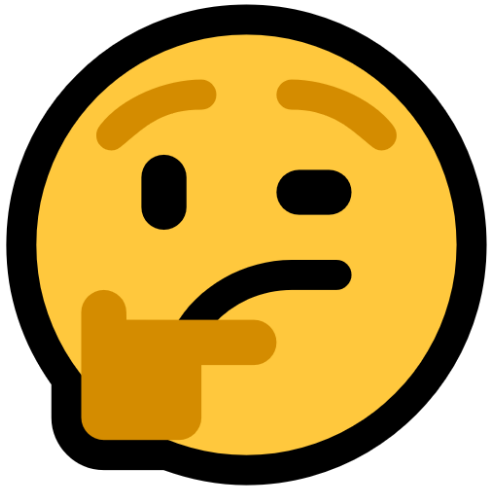
- 학습 데이터 수집 및 전처리
- 인공지능 모델 구축하고 실험

검토

- 연구 주제 및 목적을 기준으로 방법론을 평가하고 수정

0순위: 인공지능 활용 검토

해당 연구에 인공지능 연구가 가능/필요한가?



검토 체크리스트

- ☐ 데이터가 충분한가?
- ☐ 인공지능을 활용하는 것이 다른 방법론에 비해 비교우위가 있는가?
- ☐ 인공지능의 <블랙박스> 특성이 연구에 영향을 주지 않는가?
- ☐ 연구 메인 방법론이 아닌 자료 수집 등 일부 방법론만 활용하는 것은 어떨까?

다른 방법론들과의 비교

장점		단점
- 비정형 데이터도 사용 가능 - 많은 파라미터를 고려하여 분석 가능	인공지능	- 결과의 근거가 분명하지 않음(블랙박스) - 많은 데이터가 필요함
- 결과의 해석 가능 - 역사와 전통	통계	- 데이터 유형, 실험 방법론 등 연구에 제한이 있을 수 있음
- 심층적 연구 가능 - 소수의 데이터로 연구 가능	정성적 연구	- 다량의 데이터 간의 관계 등의 주제에서는 사용하기 어려움 - 연구자의 역량에 크게 영향을 받음

인공지능 방법론 선정

Q **가장 최신의 딥러닝 방법론이 무조건 좋은 거 아닌가요?**

A No
가장 최신의 딥러닝 모델이라고 가장 좋은 건 아니며 고전적인 머신러닝 방법론이 더 빠르고 정확할 수도 있다.

Q **그러면 어떤 기준으로 사용할 방법론을 선정하면 되나요?**

A 명확한 기준은 없지만 단순한 분류 등에는 머신러닝도 충분히 높은 성능을 보인다.
복잡한 데이터를 학습해야 하는 경우는 딥러닝을 활용하는 것이 좋다.
머신러닝, 딥러닝 모델 2~3개를 모두 시도하고 가장 높은 성능을 보이는 모델을 사용하는 것도 방법이다.

데이터 수집

기존 데이터 활용

- 연구 주제와 맞는 데이터가 있고, 약간의 전처리 후에 바로 사용할 수 있는 경우
- 한국어: AI hub, 모두의 말뭉치, 공공데이터포털
- 영어: 구텐베르크 프로젝트

크롤링 등 자동적 방법 활용

- 데이터화 되어 있지 않지만 자동적인 방법을 사용하여 쉽게 구축할 수 있는 방법론
- 기존 데이터에 특정 정보를 결합할 때 자동으로 수행할 수 있는 경우도 이에 포함
- Python 등 코딩을 해야 할 수 있고, 엑셀 등 범용 프로그램을 활용하거나 상용 툴을 사용할 수도 있음

직접 입력

- 고전 자료 등 기존에는 전산화되지 않은 데이터나 이미지 파일로만 존재하는 데이터는 전문가의 직접 입력이 필요
- 기존 자료를 지도 등 다른 데이터와 결합하려 할 때도 자동적으로 처리가 불가능할 때 직접 입력이 필요할 수 있음

중요

모든 데이터는 저작권, 라이선스 등 사용 가능 여부를 반드시 확인해야 하며,
크롤링 데이터 등 개인정보가 남아 있는 데이터는 충분한 익명화 과정을 거쳐야 한다.

데이터 전처리


```
        },
        {
            "id": "SD2200007",
            "age": "20대",
            "occupation": "학생",
            "sex": "남성",
            "birthplace": "서울",
            "principal_residence": "경기",
            "current_residence": "경기",
            "education": "대재"
        }
    ],
{
    "id": "SDRW2200000001.1.1.3",
    "form": "높은 사람이 말하는 게 낫 낫지 않을까?",
    "original_form": "높은 사람이 말하는 게 -낫- 낫지 않을까?",
    "speaker_id": "SD2200007",
    "start": 6.29000,
    "end": 8.43250,
    "note": ""
},
```


- 정규표현식 등을 사용하거나, 파서 등을 활용하여 데이터에서 내가 필요한 부분만 추출
- 내가 필요한 데이터의 요소들을 결합
- 형태소 분석/정규화 등이 필요하다면 수행
- 딥러닝을 활용하여 전처리를 수행할 수도 있음

모델 만들고 학습하기

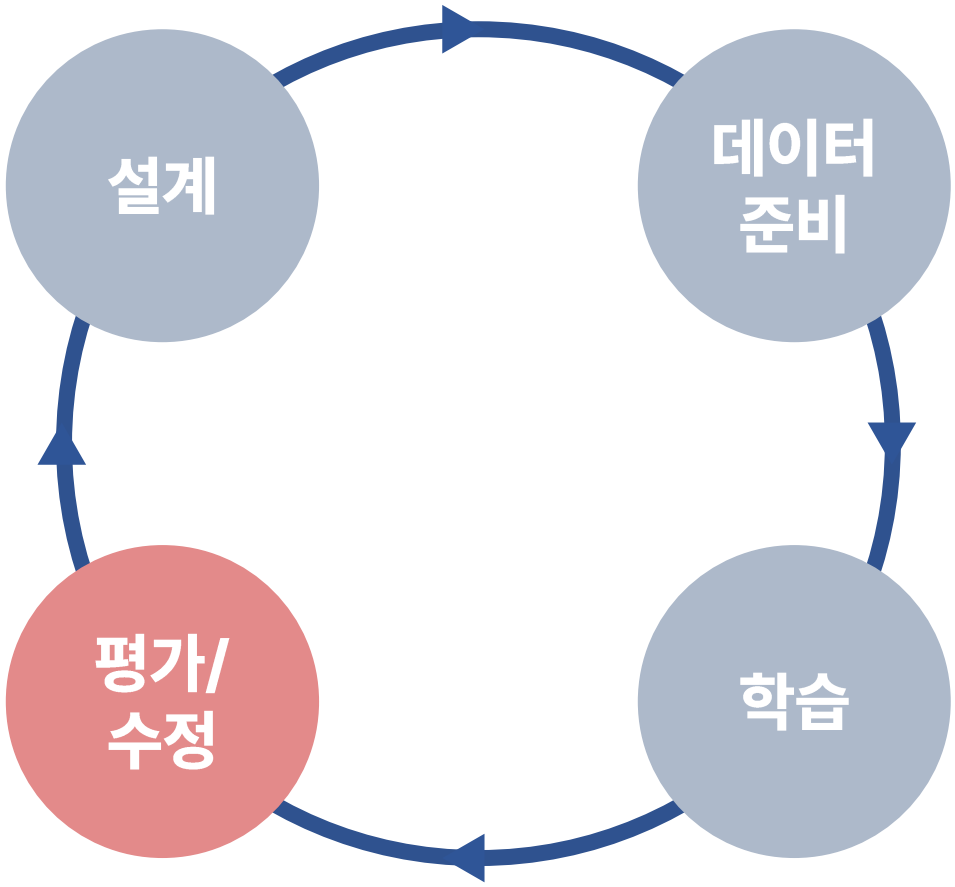
Q **모델 구축을 하는데 배경 지식이 많이 필요한가요?**

A 항상 그런 것은 아닙니다.
점점 No code 도구로 머신러닝을 사용할 수 있도록 하는 경우도 많고, AI 관련된 커뮤니티 등도 발달되고 있어 생각보다 쉬울 수도 있습니다.
사실 그보다 비용이 문제일 수 있습니다.

Q **모델을 만들어도 사용하지 못하는 경우도 있나요?**

A 머신러닝/딥러닝 모델을 만들어도 그 성능이 기대 이하여서 도저히 연구에 활용할 수 없을 수도 있습니다.
데이터부터 만들어가는 연구의 경우 이러한 위험성이 있기 때문에
공개된 데이터를 활용해 비슷한 방식으로 실험을 해 보고 실제 연구를 진행하는 것도 방법일 수 있습니다.
Pre-trained model 등을 최대한 활용해 보는 것도 좋습니다.

평가 및 수정



- 보통 준비한 데이터를 8:2로 나누어 학습과 평가에 각각 사용한다.
- 모델의 성능이 만족스럽지 못하면 결과를 분석하여 모델 및 데이터의 설계에 반영하여 수정
- 다시 학습하고 평가 및 수정을 반복함

02

대표적인 머신러닝 딥러닝 방법론

머신러닝과 딥러닝 비교

	장점	단점
머신러닝	- 딥러닝과 비교했을 때 상대적으로 적은 양의 데이터로도 학습 가능 - 전통적인 머신 러닝 방법론들은 딥러닝과 비교해 더 쉽게 결과를 해석할 수 있음	- 자연어 등 복잡한 데이터를 처리하기 어려울 수 있음 - 모델이 다양하고 각자 특징과 장단점이 다르기 때문에 이를 잘 이해하고 선택하는 것이 중요하며, 데이터 구조 등을 신중하게 구성해야 할 수 있음
딥러닝	- 자연어 등 복잡한 데이터의 미묘한 특성들을 잘 포착하여 학습할 수 있음 - 특히 생성 task에 강함 - 데이터를 가공하지 않아도 특징을 잘 학습할 수 있음	- 많은 데이터가 필요함 - 왜 그런 결과가 나왔는지 해석이 불가능 - 비용 문제 발생할 가능성이 높음

나이브 베이즈(Naïve Bayes)

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

- P(A) A가 일어날 확률
- P(B) B가 일어날 확률
- P(A|B) B가 일어나고서 A가 일어날 확률
- P(B|A) A가 일어나고서 B가 일어날 확률

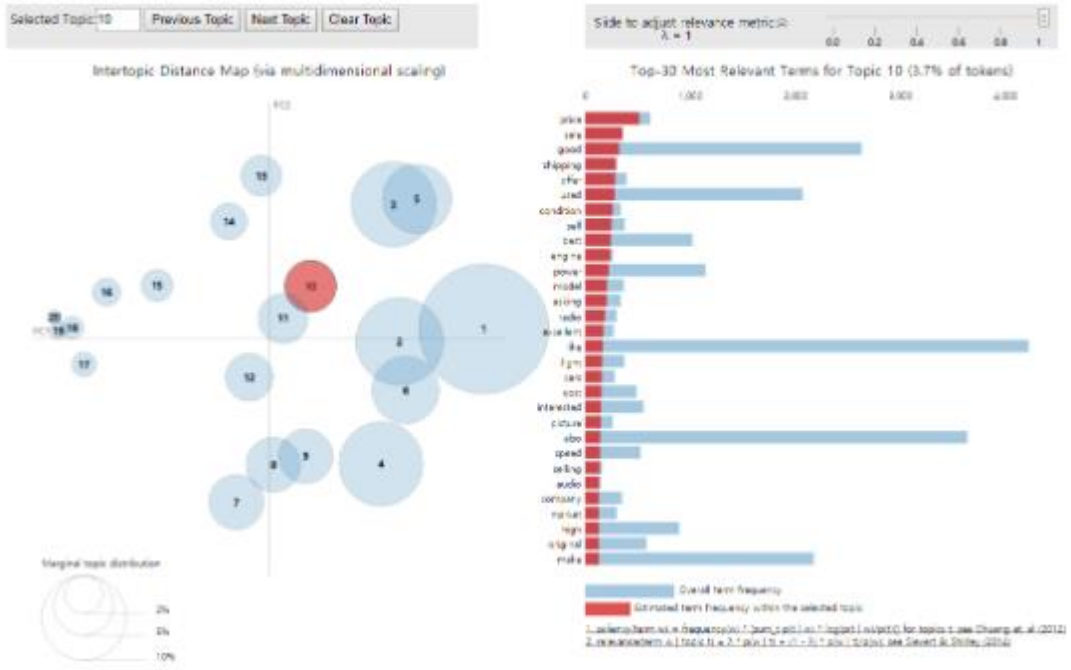
장점

- 속도가 빠르기 때문에 큰 데이터셋에 적합하다

단점

- 어순을 고려하지 않고 빈도만을 고려한다. (각 feature가 독립적이라고 가정한다)

LDA(Linear Discriminant Analysis)



토픽 모델링에 주로 사용

장점

- 각 카테고리 간 구분이 명확한 경우 높은 분류 성능을 보임
- 이해하기 쉬움

단점

- 정해진 토픽 수를 잘 결정해야 하며, 토픽 수가 적절하지 않으면 성능에 영향을 준다.
- 어순을 무시한다.
- 카테고리 간 구분이 쉽지 않은 경우 성능이 떨어진다(해석이 어렵다)

CNN

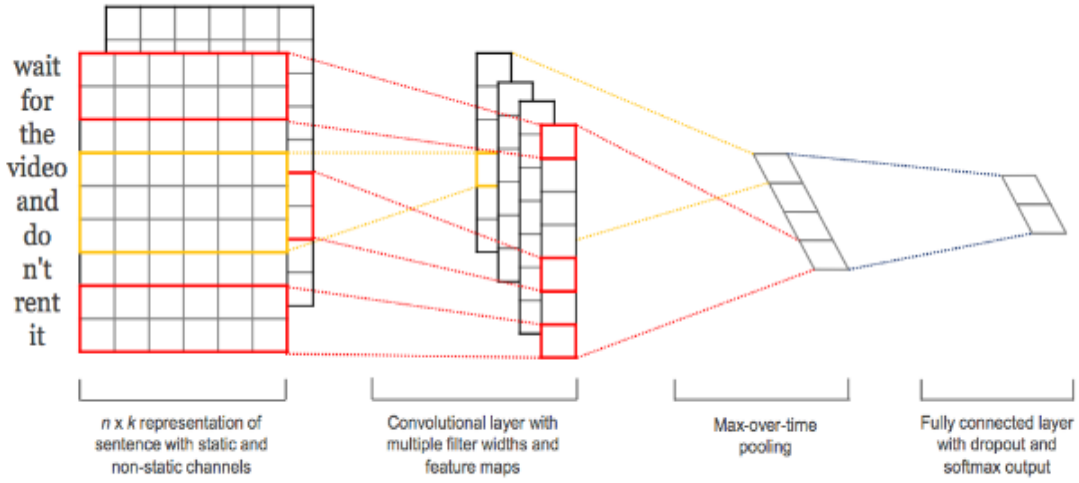


Figure 1: Model architecture with two channels for an example sentence.

Yoon Kim(2014)

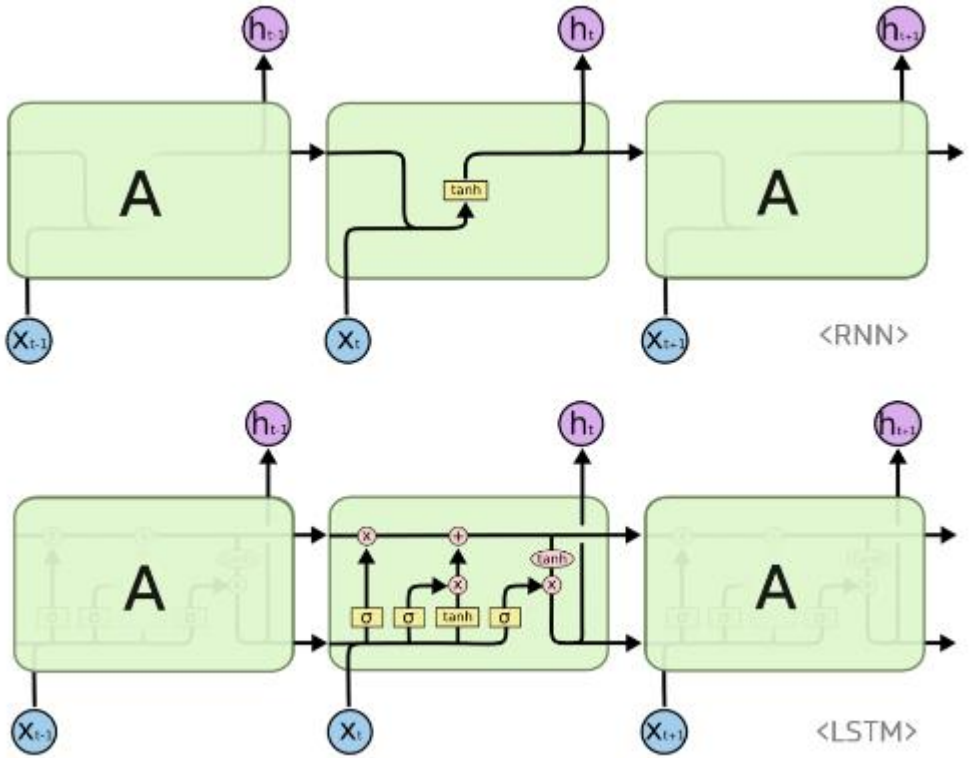
장점

- 다른 딥러닝 모델에 비해 비교적 빠른 편

단점

- 순서가 있는 데이터를 처리하는 데에 한계가 있음

RNN, LSTM



RNN

- 어순 등 순서가 있는 데이터의 특성을 잘 반영
- 관련 정보와 그 정보를 사용하는 지점 사이의 거리가 멀 경우 학습 능력 저하(장기 의존성 문제)

LSTM

- RNN의 단점인 장기 의존성 문제를 상당히 해결
- 많은 학습 데이터가 필요하며 학습에 시간과 컴퓨팅 자원이 많이 소비됨

BERT/GPT

그림9 BERT 구조

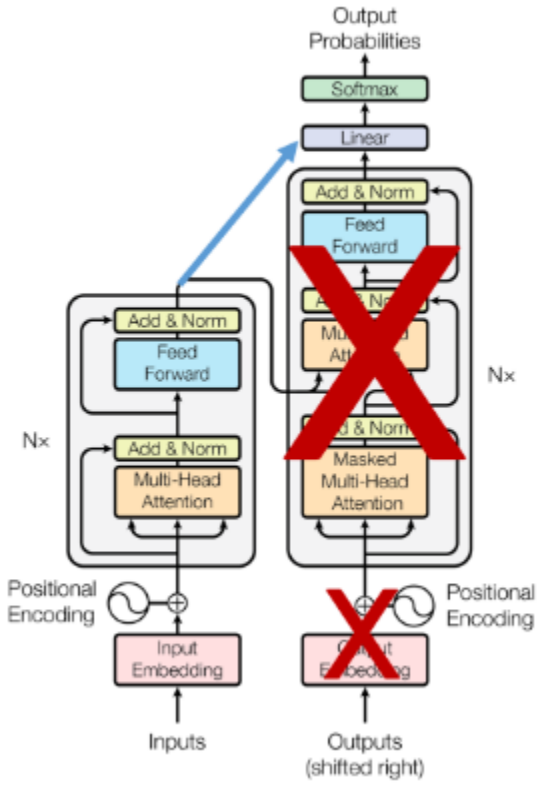
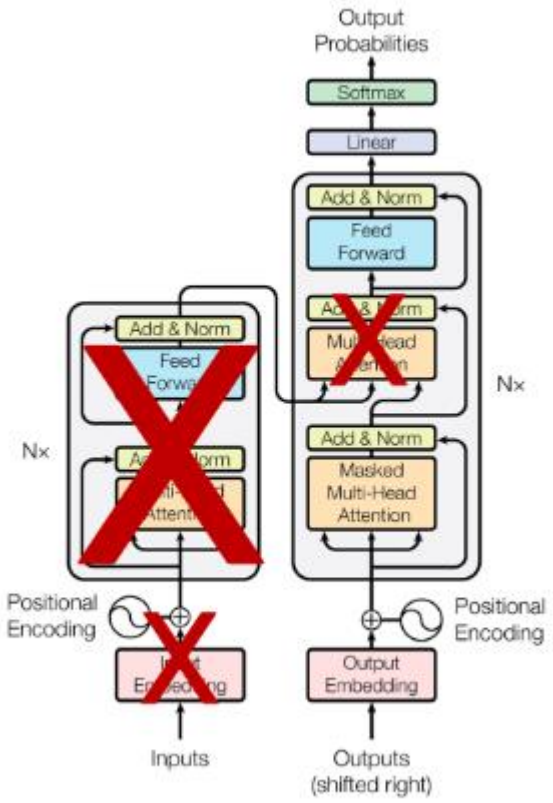


그림2 GPT 구조



BERT

- 분류, 시퀀스 레이블링 등 언어 이해 task에 강점을 보임
- 텍스트 생성에는 GPT보다는 약함

GPT

- 강력한 텍스트 생성 능력
- 텍스트의 맥락을 섬세하게 분석하는 데에는 BERT보다 약함

생성 AI의 프롬프트 엔지니어링

표 7 Few-shot 프롬프트 결과에 대한 항목 별 성능

Tags	ROUGE-L (f1_avg)	Confusion Matrix (f1_avg)
피고인 성명	87.40%	91.98%
피고인 직업	79.94%	20.00%
피해자 성명	92.91%	92.35%
피해자 나이	95.66%	97.11%
피해자 성별	75.49%	75.60%
피해자 직업	78.56%	17.65%
피고인-피해자 관계	**57.10%**	**42.51%**
범행주소	79.07%	71.29%
범행장소	**41.99%**	**31.67%**
범행일시	70.13%	2.90%
범행도구	76.80%	18.10%
범행동기	**42.95%**	해당없음
공격행위	**27.64%**	해당없음
공격부위	**68.37%**	해당없음
공격횟수	**67.90%**	해당없음
피고인 상해	**63.12%**	해당없음
범행결과	**69.01%**	해당없음

박예린·원광재·박노섭(2023), 형사 판결문 정보추출 데이터셋 구축 방안 - GPT-3.5 프롬프트 활용을 중심으로 -

장점

- 몇 개의 데이터를 활용하여 간편하게 task를 수행할 수 있음.
- 모델 제작사(OpenAI 등)에서 프롬프트 플랫폼을 제공하는 경우가 많아 모델 구축이나 fine-tuning 등을 위해 복잡한 과정을 거칠 필요가 없는 경우가 많음

단점

- Fine-tuning에 비해 성능이 떨어짐