

# 자연어처리 Intro



# 강사소개



백혜림

hreeee@yonsei.ac.kr

연세대학교 일반대학원 석사 졸업

신호처리전공 주 연구분야 잡음제거, 음원분리, 음성강화

## 강의이력

모두의연구소 aiffel 강남 1기 & 싹 1기

영우글로벌러닝 2, 3, 4, 5기 자연어처리 강사

서울시교육청 x 네패스 직업계고 ai 기초 & 심화 강사

기상청 AI 프로젝트 강사

엘리스 알고리즘 강사

Upstage x 네이버 커넥트 재단 3기 멘토

그 외 기업강의 다수

(주)비바이노베이션 기술 자문

# 목차



1. 자연어처리 소개
2. NLP vs Others
3. NLP와 Deep Learning
4. NLP는 무엇때문에 어려운가?
5. 한국어 NLP는 무엇이 더 어려운가?
6. NLP History

# 주의사항!

- 수식은 완벽히 이해 못해도 됩니다
  - 나중에 또 보면 됩니다
  - 대신 큰 그림만 잡고 갑시다!
- 실습 코드는 혼자 따로 꼭 구현해 보세요!
  - 같이 할 때에는 이해가 잘가요
  - 하지만, 혼자서 직접 짜지 않으면, 절대 내것이 되지 않습니다 (명심! 명심!)

# Introduction to Natural Language Processing

백혜림 hreeee@yonsei.ac.kr

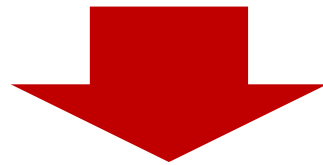


# 언어란 무엇인가?

## - 언어에 대한 다양한 정의

- 1) 사람들이 자신의 머리 속에 있는 생각을 다른 사람에게 나타내는 체계
- 2) 사물, 행동, 생각 그리고 상태를 나타내는 체계
- 3) 사람들이 자신이 가지고 있는 생각을 다른 사람들에게 전달하는 데 사용하는 방법
- 4) 사람들 사이에 공유되는 의미들의 체계
- 5) 문법적으로 맞는 말의 집합(절대적이 아님)
- 6) 언어 공동체 내에서 이해될 수 있는 말의 집합

출처: <https://ko.wikipedia.org/wiki/%EC%96%B8%EC%96%B4>



## 정보 전달

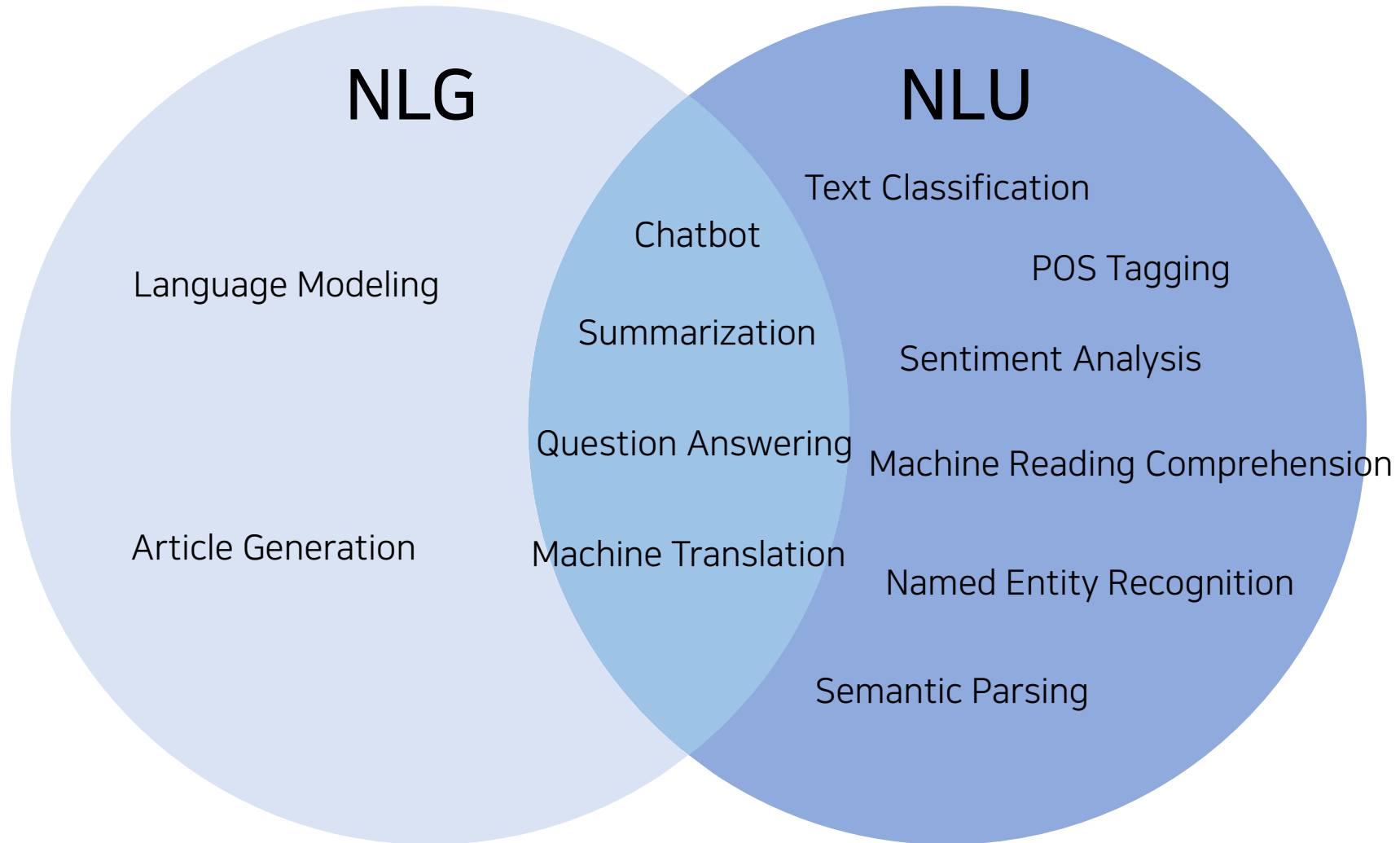
# 인공지능에서 Interface

- 사람의 생각(의도, 정보)를 컴퓨터에게 전달하는 방법
  - Native Interface
    - 사람이 이해할 수 있지만, 엄격한 문법과 모호성이 없는 형태의 전달 방식
    - 인공 언어
      - ex. 프로그래밍 언어
  - Better Interface
    - 사람이 실제 사용하는 형태에 가까운 전달 방식
    - 자연 언어

# 자연어 처리는 무엇인가?

- 자연어(Natural Language)란?
  - 자연어 혹은 자연 언어는 사람들이 일상적으로 쓰는 언어를 인공적으로 만들어진 언어인 인공어와 구분하여 부르는 개념 (출처: 위키피디아)
- Natural Language Processing
  - 사람이 이해하는 자연어를 컴퓨터가 이해할 수 있는 값으로 바꾸는 과정(NLU)
  - 더 나아가 컴퓨터가 이해할 수 있는 값을 사람이 이해하도록 바꾸는 과정(NLG)





# NLP vs Others

백혜림 hreeee@yonsei.ac.kr



# 인공 지능의 다양한 task

- Computer Vision
  - Image Recognition
  - Object Detection
  - Image Generation
  - Super Resolution
- Natural Language Processing
  - Text Classification
  - Machine Translation
  - Summarization
  - Question Answering
- Speech Processing
  - Speech Recognition (STT)
  - Speech Synthesis (TTS)
  - Speaker Identification
- Reinforcement Learning

# 인공 지능의 다양한 task

- Natural Language Processing
  - Discrete value를 다룸
    - 단어, 문장
  - 분류 문제로 접근할 수 있음
  - 샘플의 확률값을 구할 수 있음  
 $P(x = \text{단어})$
  - 문장 생성 (자연어 생성)
    - Auto-regressive 속성을 지님
    - GAN 적용 불가
- Other Fields (ex. Computer vision)
  - Continuous value를 다룸
    - 이미지, 음성
  - 문제에 따라 접근 방식이 다름
  - 샘플의 확률 값을 구할 수 없음  
 $P(x = \text{이미지})$
  - 이미지 생성
    - Auto-regressive 속성 없음
    - GAN 적용 가능

# NLP 연구 요구사항

- Domain Knowledge
  - 언어적 지식 필요
    - ex.한국어는 어떠한 언어적 특성을 지니는가?
- Nasty Preprocessing
  - Task에 따른 정제 (normalization) 과정 필요

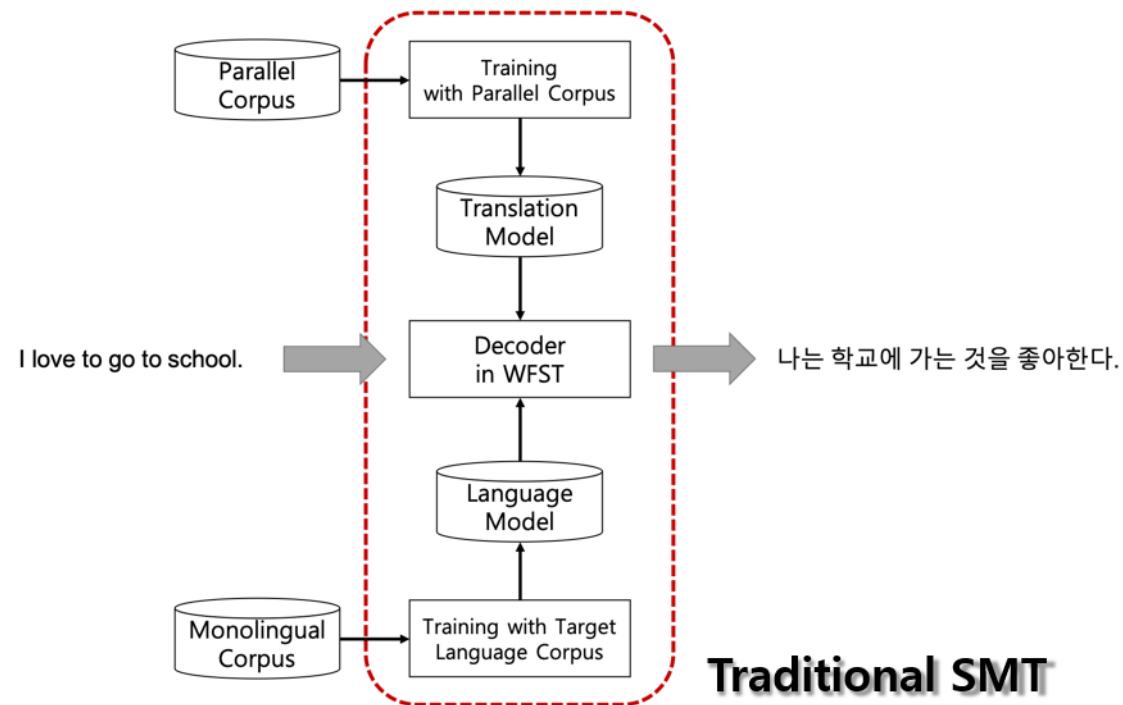
# NLP와 Deep Learning

백혜림 hreeee@yonsei.ac.kr



# 전통적인 NLP와 Deep Learning을 이용한 NLP

- Traditional NLP
  - 단어를 symbolic 데이터로 취급
  - 여러 sub-module을 통해 전체 구성
- NLP with DeepLearning
  - 단어를 continuous value로 변환
  - End-to-End 시스템 추구



# 전통적인 NLP의 특징

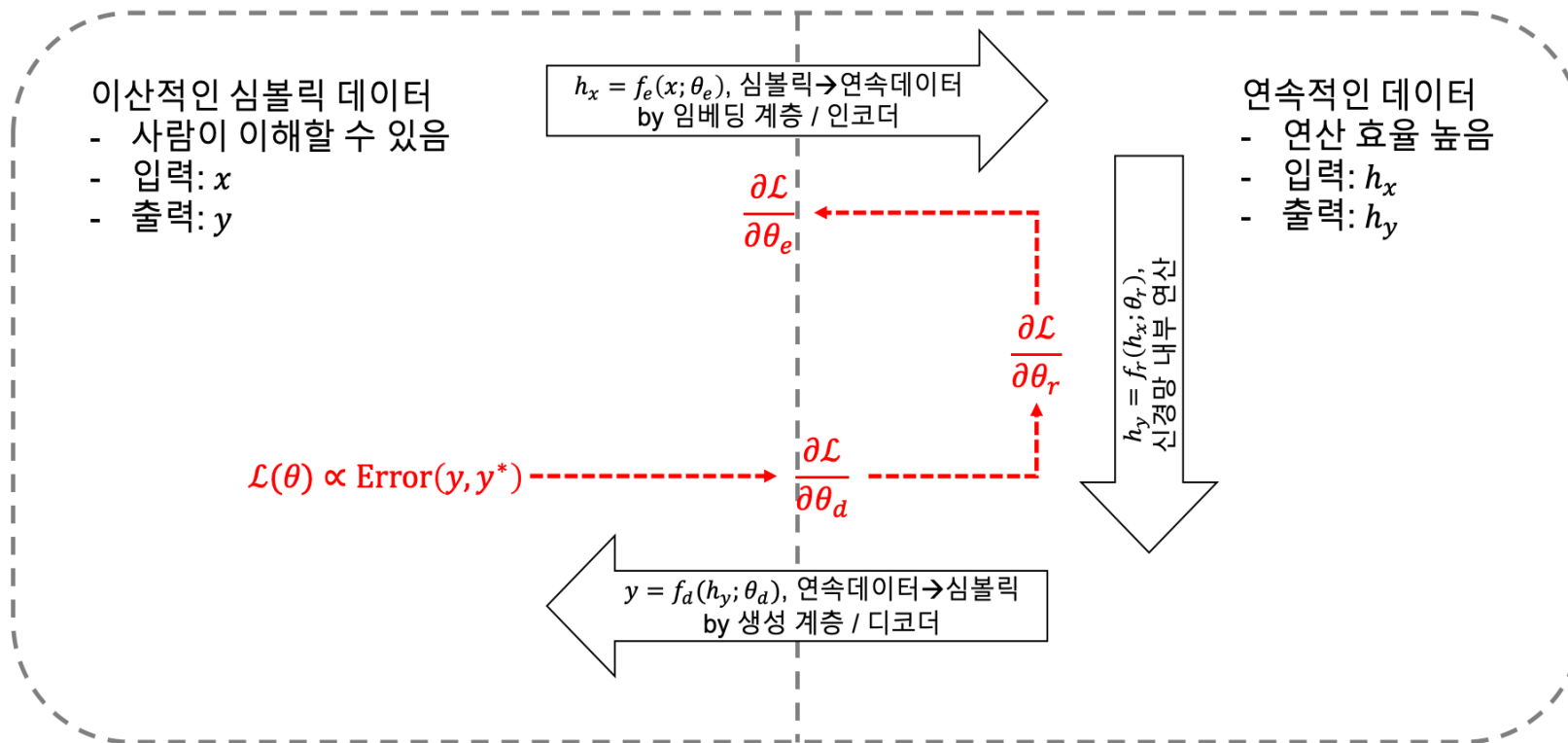
- Symbolic 기반 접근

전통적인 심볼릭 기반 접근 방법	딥러닝 기반 접근 방법
이산적(discrete), 심볼릭 공간	연속적(continuous), 신경망 공간
사람이 인지하기 쉬움	사람이 이해하기 어려움
디버그 용이	디버깅 어려움
연산 속도 느림	연산 속도 빠름
모호성과 유의성에 취약함	모호성과 유의성에 강인함
여러 서브 모듈이 꼭포수 형태를 취하므로 특징 추출에 노력이 필요함	end-to-end 모델을 통한 성능 개선과 시스템 간소화 가능

- 여러 단계의 Sub-module 구성
  - 무거움
  - 각 모듈의 오류가 이후 모듈에 영향을 끼침 (error propagation)



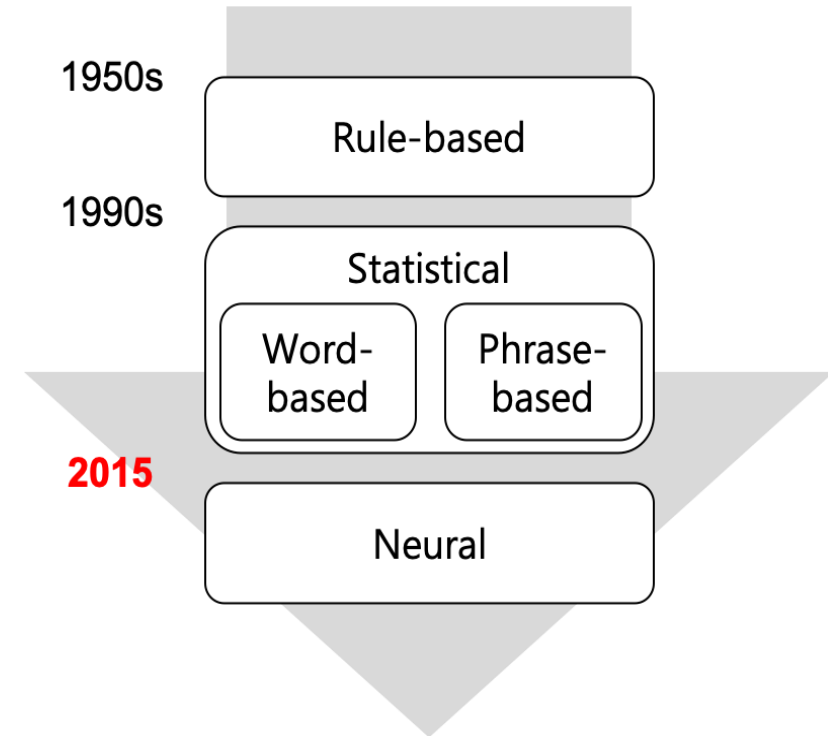
# NLP System with Deep Learning



# Paradigm Shift in Natural Language Processing

- 효율적인 Embedding을 통한 성능 개선
  - 단어, 문장, context embedding
- End to End구성으로 인한 효율/성능 개선
  - 가볍고, 빠르다.

결국, 기계번역의 경우,  
다른 분야에 비해 가장 먼저 성공적인 상용화



# NLP는 무엇 때문에 어려울까?

백혜림 hreeee@yonsei.ac.kr



# Ambiguity (모호성)

원문	차를 마시러 공원에 가던 차 안에서 나는 그녀에게 차였다.
G*	I was kicking her in the car that went to the park for tea.
M*	I was a car to her, in the car I had a car and went to the park.
N*	I got dumped by her on the way to the park for tea.
K*	I was in the car going to the park for tea and I was in her car.
S*	I got dumped by her in the car that was going to the park for a cup of tea.

# Ambiguity (모호성)

## 중의성 해소 (word sense disambiguation)

원문	차를 마시러 공원에 가던 차 안에서 나는 그녀에게 차였다.
G*	I was kicking her in the car that went to the park for tea.
M*	I was a car to her, in the car I had a car and went to the park.
N*	I got dumped by her on the way to the park for tea.
K*	I was in the car going to the park for tea and I was in her car.
S*	I got dumped by her in the car that was going to the park for a cup of tea.

# Ambiguity (모호성)

원문	나는 철수를 안 때렸다.
1	철수는 맞았지만, 때린 사람이 나는 아니다.
2	나는 누군가를 때렸지만, 그게 철수는 아니다.
3	나는 누군가를 때린 적도 없고, 철수도 맞은 적이 없다.

# Ambiguity (모호성)

## 문장 내 정보의 부족으로 인한 모호성이 발생

원문	나는 철수를 안 때렸다.
1	철수는 맞았지만, 때린 사람이 나는 아니다.
2	나는 누군가를 때렸지만, 그게 철수는 아니다.
3	나는 누군가를 때린 적도 없고, 철수도 맞은 적이 없다.

# Ambiguity (모호성)

## 문장 내 정보의 부족이 야기한 구조 해석의 문제

원문	선생님은 울면서 돌아오는 우리를 위로 했다.
1	(선생님은 울면서) 돌아오는 우리를 위로 했다.
2	선생님은 (울면서 돌아오는 우리를) 위로 했다.



# 왜 언어는 모호성을 가질까?

- 언어는 마치 생명체와 같이 진화하며, 특히 **효율성을 극대화**하는 방향으로 진화
- 따라서, 최대한 짧은 문장 내에 많은 정보를 담고자 한다.
  - 정보량이 낮은 내용(context)은 생략
  - 여기에서 모호함(ambiguity)이 발생
- 생략된 context를 인간은 효율적으로 채울 수 있지만, 기계는 이러한 task에 매우 취약하다.

# Paraphrase



# Paraphrase

번호	문장 표현
1	여자가 김치를 어떤 남자에게 집어 던지고 있다.
2	여자가 어떤 남자에게 김치로 때리고 있다.
3	여자가 김치로 싸대기를 날리고 있다.
4	여자가 배추 김치 한 포기로 남자를 때리고 있다.
5	여자가 김치를 사용해 남자를 때리고 있다.
6	남자가 여자에게 김치로 싸대기를 맞고 있다.
7	남자가 여자로부터 김치로 맞고 있다.

# Paraphrase

문장의 표현 형식은 다양하고, 비슷한 의미의 단어들이 존재하기 때문에  
**Paraphrase의 문제**가 존재

번호	문장 표현
1	여자가 김치를 어떤 남자에게 집어 던지고 있다.
2	여자가 어떤 남자에게 김치로 때리고 있다.
3	여자가 김치로 싸대기를 날리고 있다.
4	여자가 배추 김치 한 포기로 남자를 때리고 있다.
5	여자가 김치를 사용해 남자를 때리고 있다.
6	남자가 여자에게 김치로 싸대기를 맞고 있다.
7	남자가 여자로부터 김치로 맞고 있다.

# Discrete, not Continuous

- 이산 값을 갖는 자연어는 사람의 입장에서 인지가 쉬울 수 있으나, 기계의 입장에서 매우 어려운 값
- One-hot encoding으로 표현된 값은 유사도나 모호성을 표현할 수 없다.
  - 서로 다른 One-hot 벡터끼리의 유사도나 거리는 모두 동일하다.
- 따라서, 아래의 질문에 대답할 수 없다.
  - <파랑>과 <핑크> 중에서 <빨강>에 가까운 단어는 무엇인가요?
  - 하지만, 사람의 어휘 체계는 계층적 구조를 띄고 있다.
- 또한 높은 차원으로 표현되어 매우 sparse하게 된다.

# Discrete, not Continuous

- 이산 값을 갖는 자연어는 사람의 입장에서 인지가 쉬울 수 있으나, 기계의 입장에서 매우 어려운 값
- One-hot encoding으로 표현된 값은 유사도나 모호성을 표현할 수 없다.
  - 서로 다른 One-hot 벡터끼리의 유사도나 거리는 모두 동일하다.
- 따라서, 아래의 질문에 대답할 수 없다.
  - <파랑>과 <핑크> 중에서 <빨강>에 가까운 단어는 무엇인가요?
  - 하지만, 사람의 어휘 체계는 계층적 구조를 띄고 있다.
- 또한 높은 차원으로 표현되어 매우 sparse하게 된다.

**딥러닝에서는 Word Embedding을 통해 해결**

# 한국어 NLP는 무엇이 더 어려울까?

백혜림 hreeee@yonsei.ac.kr



# 교착어

종류	대표적 언어	특징
교착어	한국어, 일본어, 몽골어	어간에 접사가 붙어 단어를 이루고 의미와 문법적 기능이 정해짐
굴절어	라틴어, 독일어, 러시아어	단어의 형태가 변함으로써 문법적 기능이 정해짐
고립어	영어, 중국어	어순에 따라 단어의 문법적 기능이 정해짐



# 교착어 : 접사 추가에 따른 의미 파생

원형	피동	높임	과거	추측	전달	결과
잡						+다 잡다
잡	+히					+다 잡히다
잡	+히	+시				+다 잡히시다
잡	+히	+시	+었			+다 잡히셨다
잡			+았(었)			+다 잡았다
잡				+겠		+다 잡겠다
잡					+더라	잡더라
잡		+히	+었			+다 잡혔다
잡		+히	+었	+겠		+다 잡혔겠다
잡	+히	+었	+겠		+더라	잡혔겠더라
잡			+았(었)	+겠		+다 잡았겠다
...						...
잡	+히	+시	+았(었)	+겠	+더라	잡히시었겠더라

# 교착어 : 유연한 단어 순서 규칙

번호	문장	정상여부
1.	나는 밥을 먹으러 간다.	O
2.	간다 나는 밥을 먹으러.	O
3.	먹으러 간다 나는 밥을.	O
4.	밥을 먹으러 간다 나는.	O
5.	나는 먹으러 간다 밥을.	O
6.	나는 간다 밥을 먹으러.	O
7.	간다 밥을 먹으러 나는.	O
8.	간다 먹으러 나는 밥을.	O
9.	먹으러 나는 밥을 간다.	X
10.	먹으러 밥을 간다 나는.	X
11.	밥을 간다 나는 먹으러.	X
12.	밥을 나는 먹으러 간다.	O
13.	나는 밥을 간다 먹으러.	X
14.	간다 나는 먹으러 밥을.	O
15.	먹으러 간다 밥을 나는.	O
16.	밥을 먹으러 나는 간다.	O

# 모호한 띄어쓰기

- 근래 이전까지 동양권 언어에는 띄어쓰기가 존재하지 않았음
  - 서양에서는 중세시대에 띄어쓰기가 확립됨
- 따라서, 아직 우리나라 말은 여전히 띄어쓰기와 궁합을 맞추는 중
  - 전 국립언어원장님도 어려워하시는 띄어쓰기

참고: [https://news.chosun.com/site/data/html\\_dir/2013/05/21/2013052103173.html](https://news.chosun.com/site/data/html_dir/2013/05/21/2013052103173.html)

- 왜? 띄어쓰기가 어지간히 틀려도 잘 알아듣기 때문



# 평서문과 의문문의 차이 부재

언어	평서문	의문문
영어	I ate my lunch.	Did you have lunch?
한국어	점심 먹었어.	점심 먹었어?

# 주어 부재

언어	평서문	의문문
영어	I ate my lunch.	Did you have lunch?
한국어	점심 먹었어.	점심 먹었어?

# 한자 기반의 언어

- 표의 문자인 한자를 표음 문자인 한글로 wrapping
  - 표의 문자 : 의미 또는 사물의 형상을 글씨로 나타냄
  - 표음 문자 : 사람이 말하는 소리, 음성을 글씨로 나타냄
- Wrapping 과정에서 정보의 손실 발생

茶 vs 車

# 단어 중의성으로 인한 문제 발생 사례

- '차'의 hidden representation

# NLP의 History

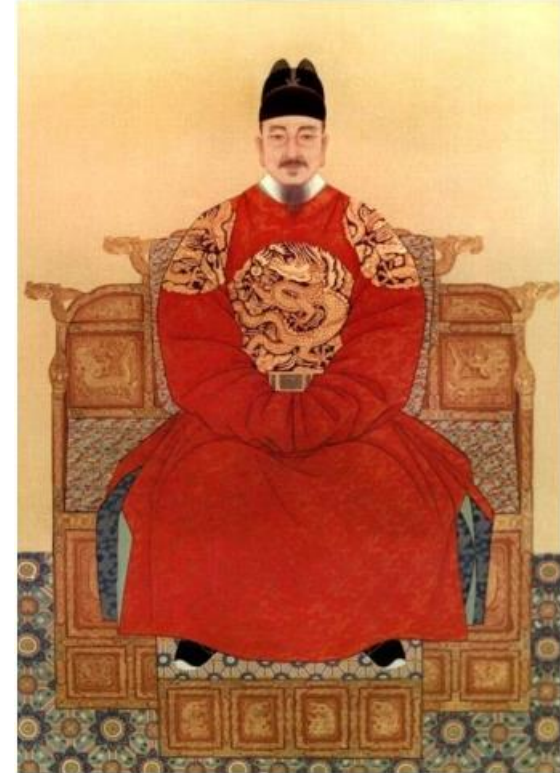
백혜림 hreeee@yonsei.ac.kr





# 극악 난이도 한국어 NLP

- 한글은 굉장히 늦게 만들어진 문자
  - 따라서 기존 다른 문자들의 장점을 흡수
  - 굉장히 과학적으로 만들어짐
- 효율이 극대화 되었기 때문에 더욱 어려운 것
- 앞으로 우리는 자연어처리 전반 뿐만 아니라, 한국어에 적용하였을 때의 특성과 문제도 다룰 것



# Neural NLP의 역사

Before Deep Learning

After Sequence to Sequence

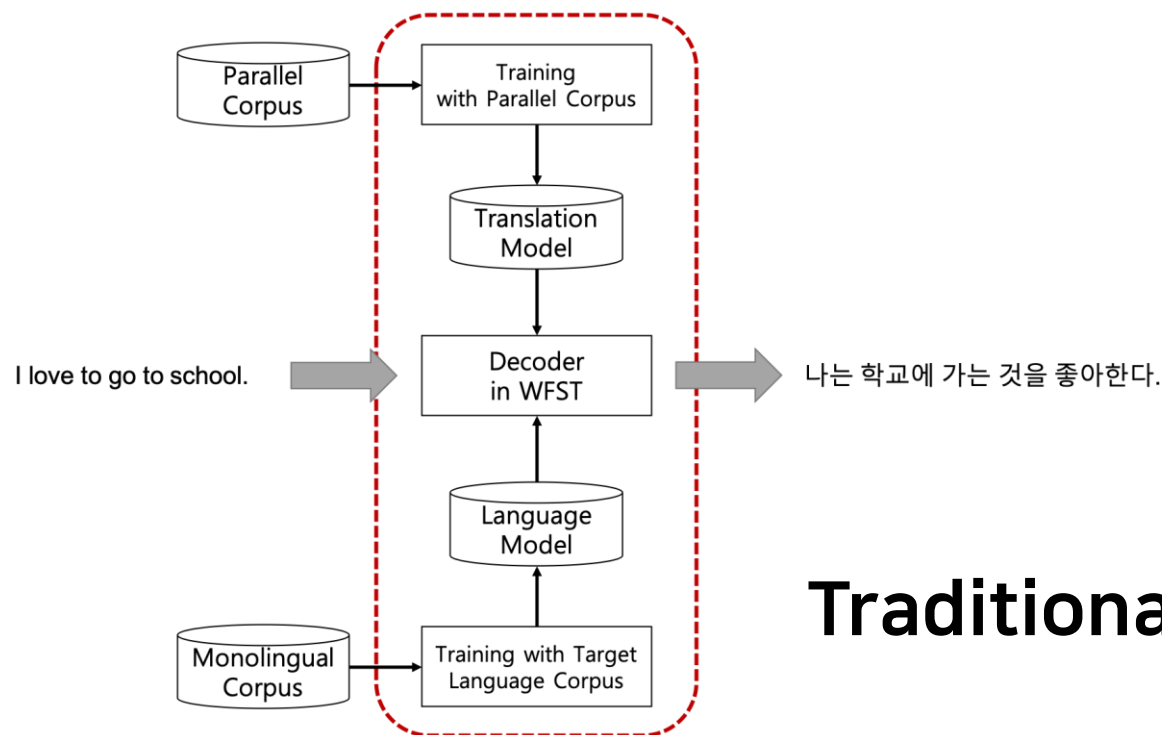
After Sequence to Sequence with Attention

Era of Attention

Pretraining and Fine-tuning

# Befor Deep Learning

- 전형적인 NLP application 구조
  - 여러 단계의 sub-module로 구성되어 복잡한 디자인을 구성
  - 매우 무겁고, 복잡하여 구현 및 시스템 구성이 어려운 단점
  - 각기 발생한 error가 중첩 및 가중되어 error propagation

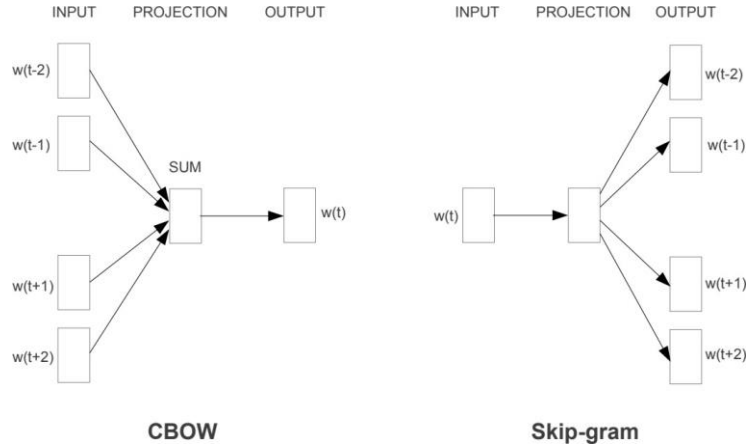


**Traditional SMT**

# Before Sequence to Sequence

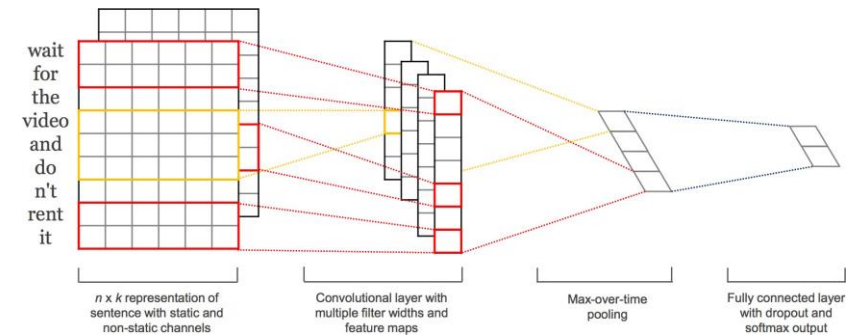
## Word Embedding

- [Mikolov et al., 2013]



## Text Classification

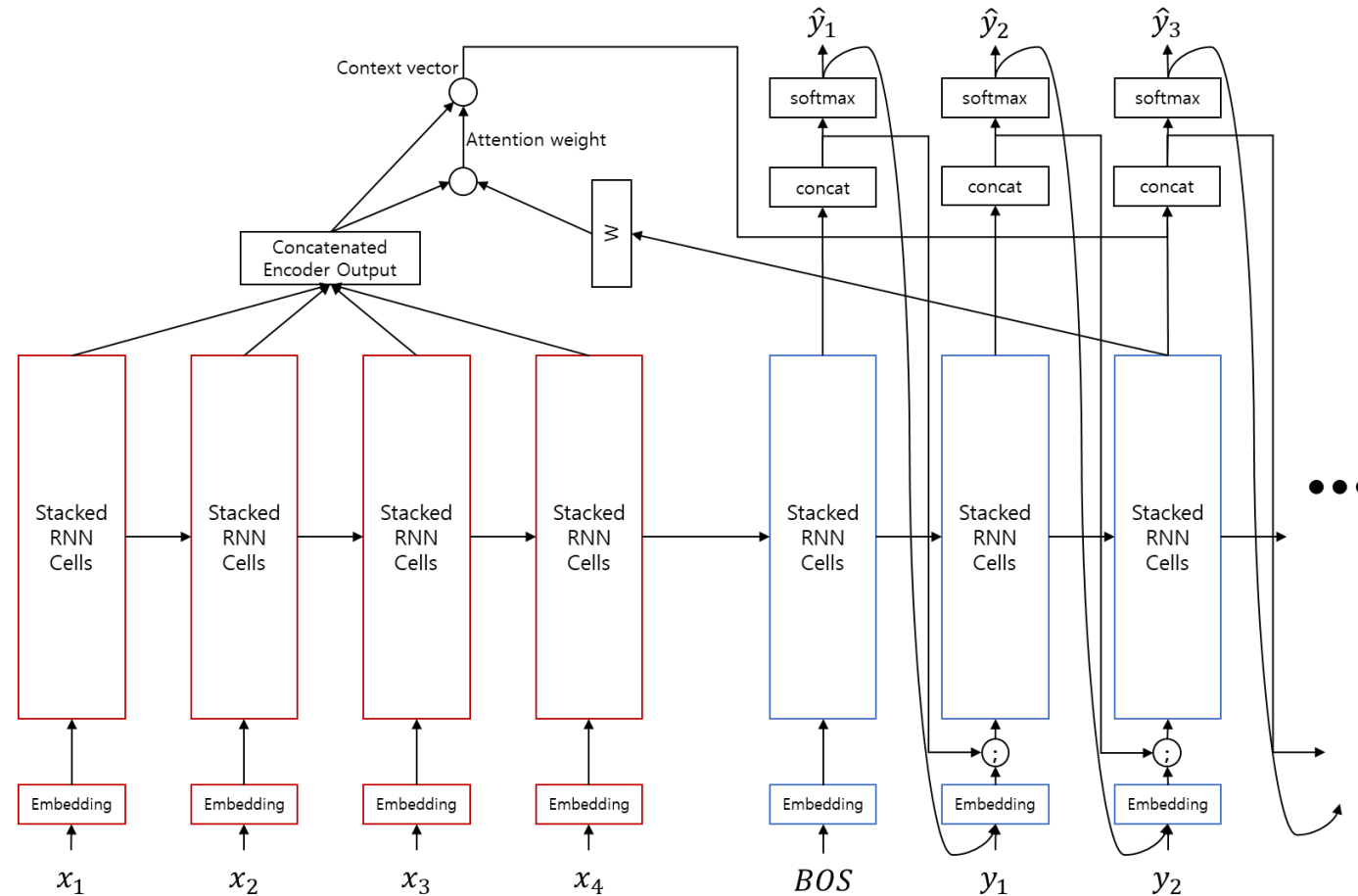
- [Kim, 2014]



Text > 수치 값

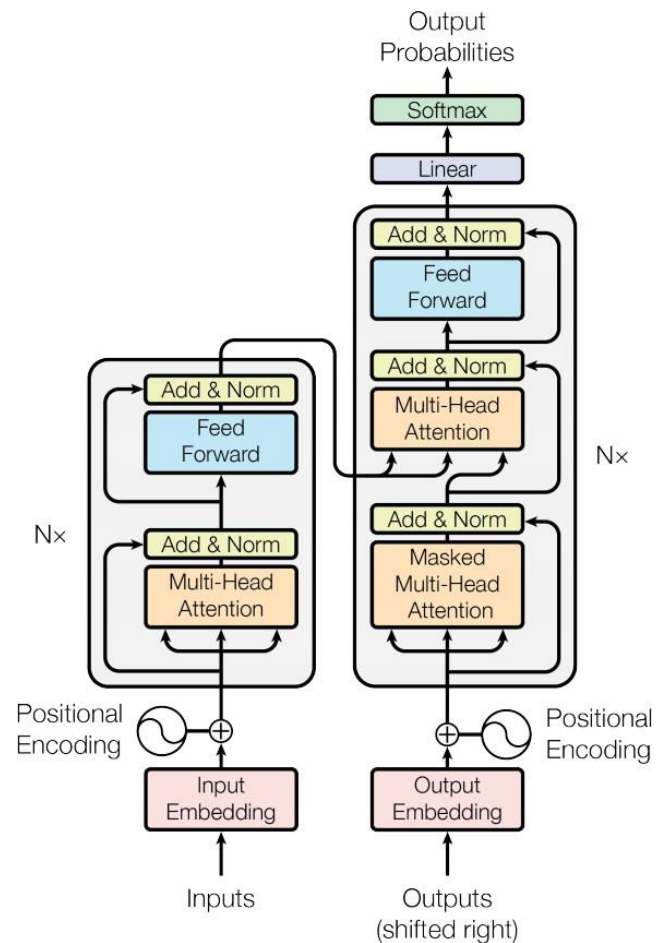
# After Sequence to Sequence with Attention

Beyond "text to numeric"

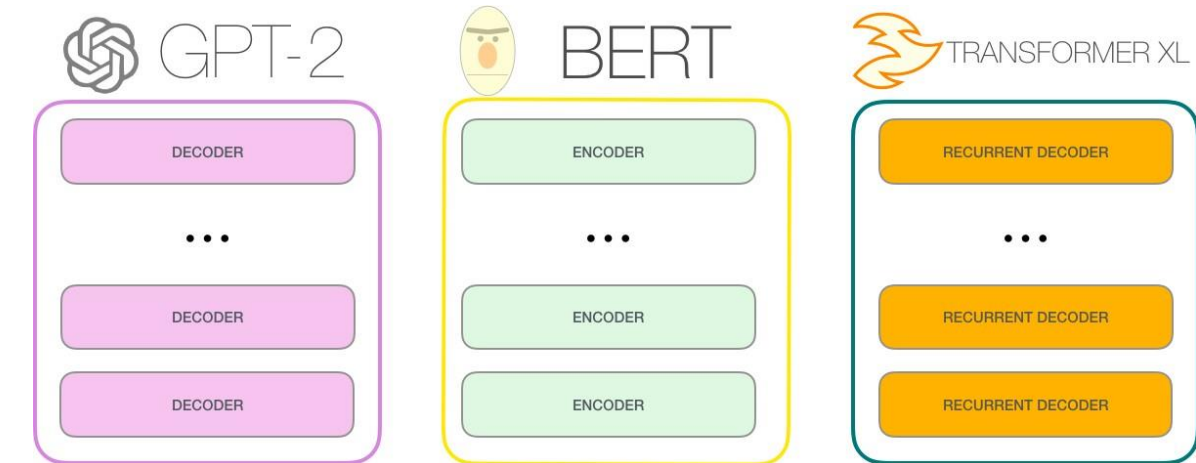


# Era of Attention

## Transformer by End to End Attention



Everything can be done by Attention.

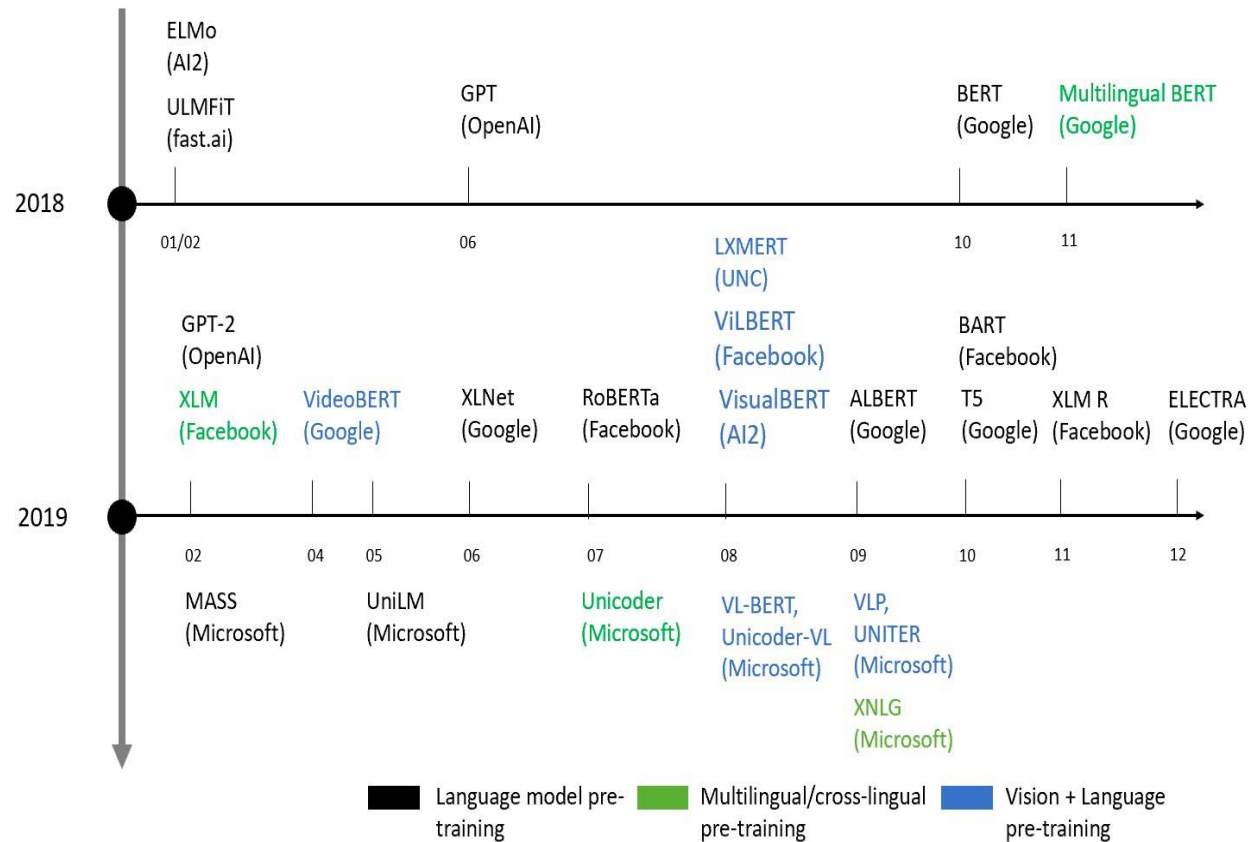


출처: <http://jalammar.github.io/illustrated-gpt2/>

Figure 1: The Transformer - model architecture.

# BERTology : Pretraining and Fine-tuning

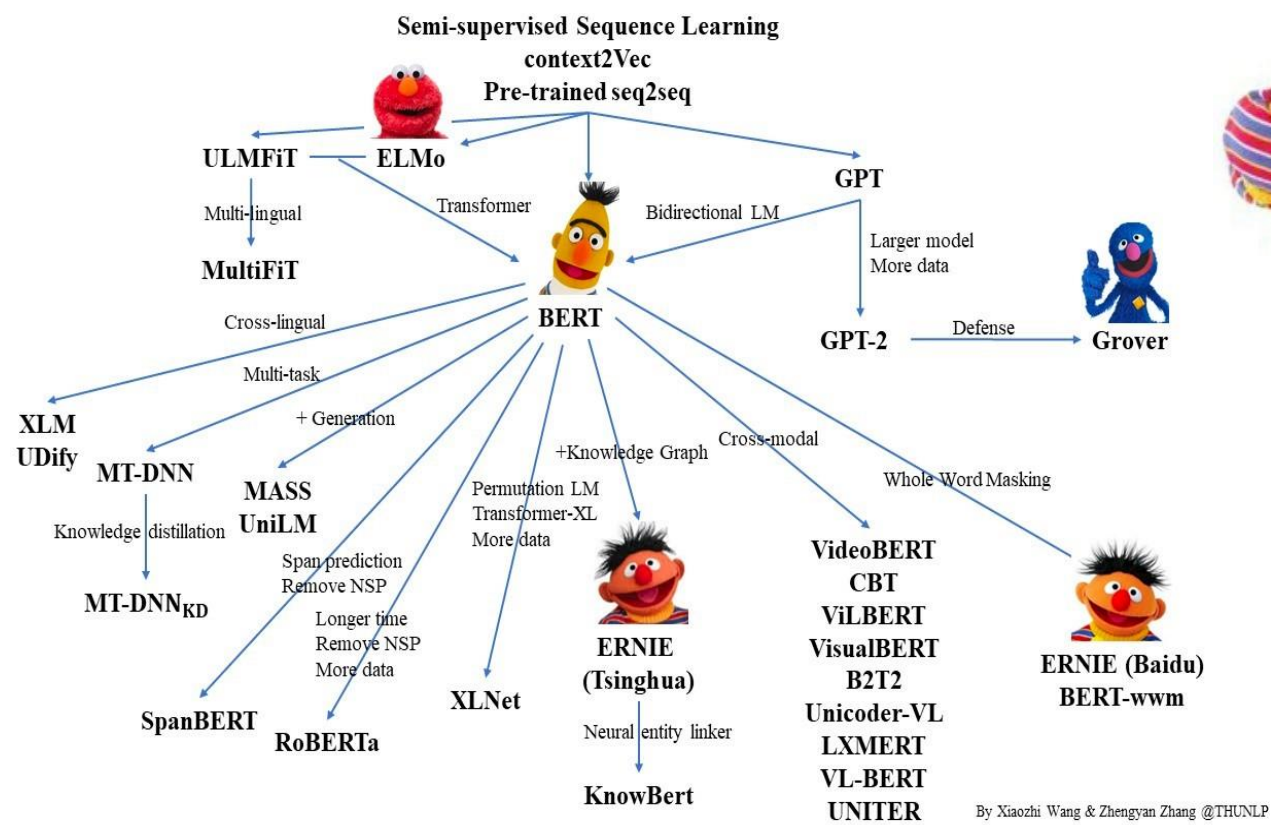
Big Language Models mainly based on Transformer



출처: <https://github.com/amyxie361/Blog/issues/20>

# Appendix

Another version..



출처: <https://github.com/thunlp/PLMpapers>



**수고하셨습니다.**