자연어 처리 개요

언어란 무엇인가

- 언어는 다양하게 정의할 수 있습니다.
  - 사람들이 자신의 머리 속에 있는 생각을 다른 사람에게 나타내는 체계
  - 사물, 행동, 생각, 그리고 상태를 나타내는 체계
  - 사람들이 자신이 가지고 있는 생각을 다른 사람들에게 전달하는 데 사용하는 방법
  - 사람들 사이에 공유되는 의미들의 체계
  - 문법적으로 맞는 말의 집합(절대적이 아님)
  - 언어 공동체 내에서 이해될 수 있는 말의 집합

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

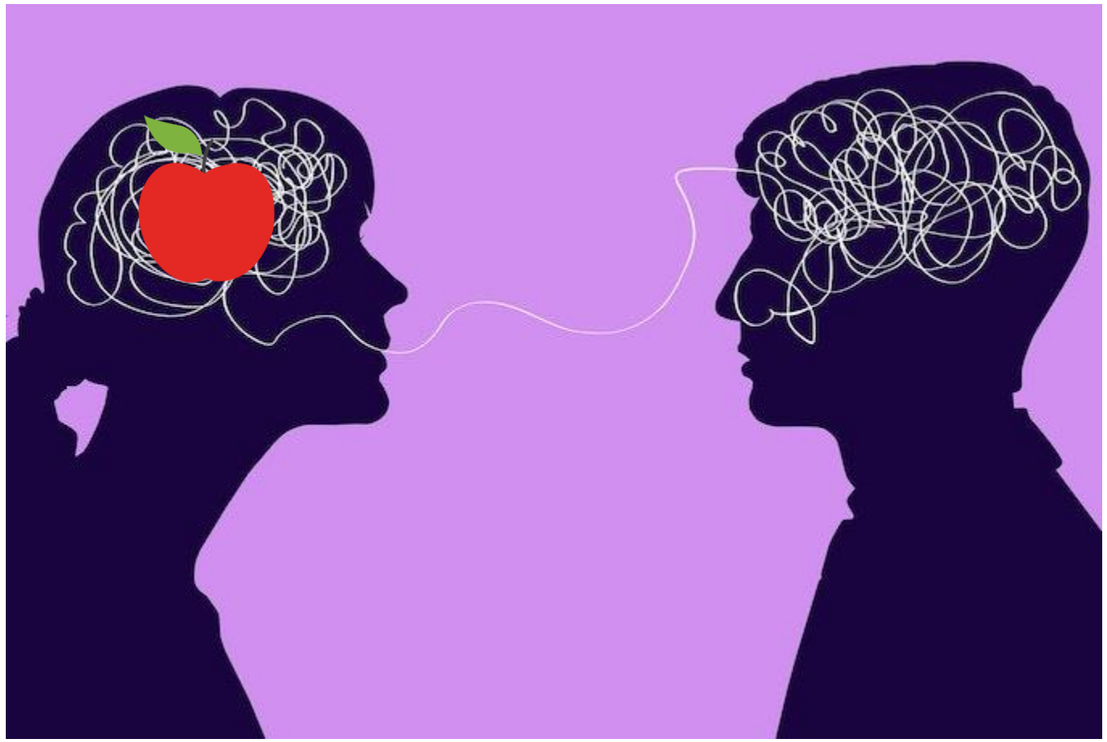
언어란 무엇인가

- 언어는 다양하게 정의할 수 있습니다.
  - 사람들이 자신의 머리 속에 있는 생각을 다른 사람에게 나타내는 체계
  - 사물, 행동, 생각, 그리고 상태를 나타내는 체계
  - 사람들이 자신이 가지고 있는 생각을 다른 사람들에게 전달하는 데 사용하는 방법
  - 사람들 사이에 공유되는 의미들의 체계
  - 문법적으로 됫
  - 언어 공동체

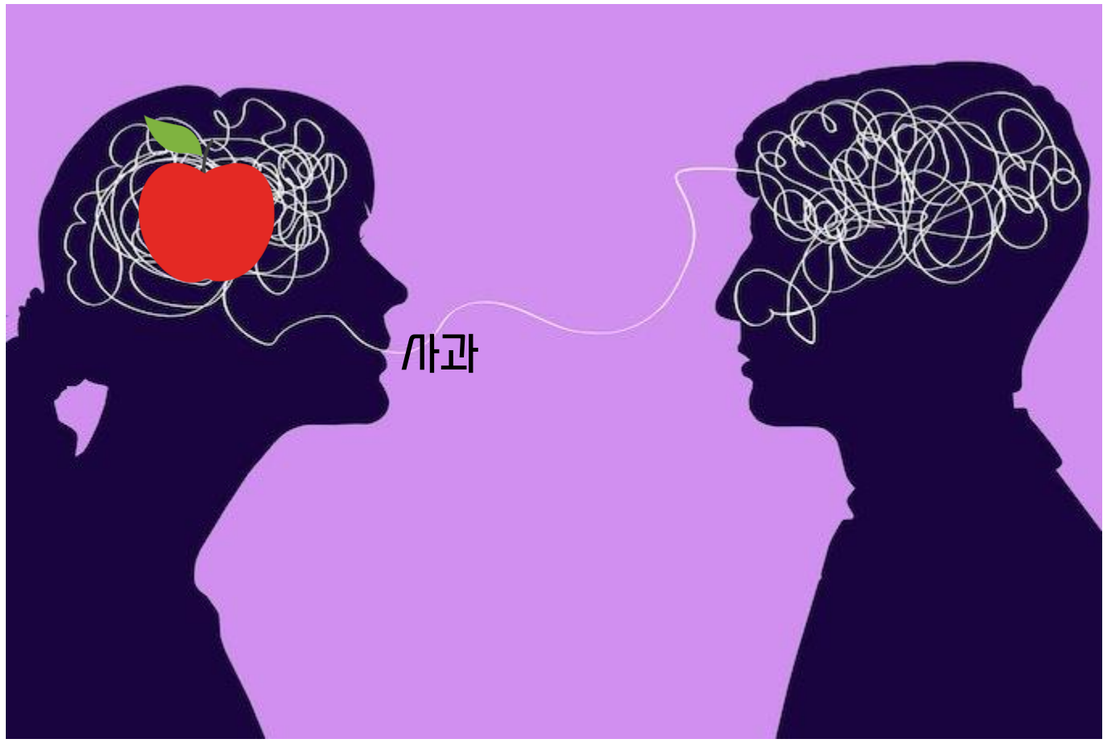
정보 전달

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

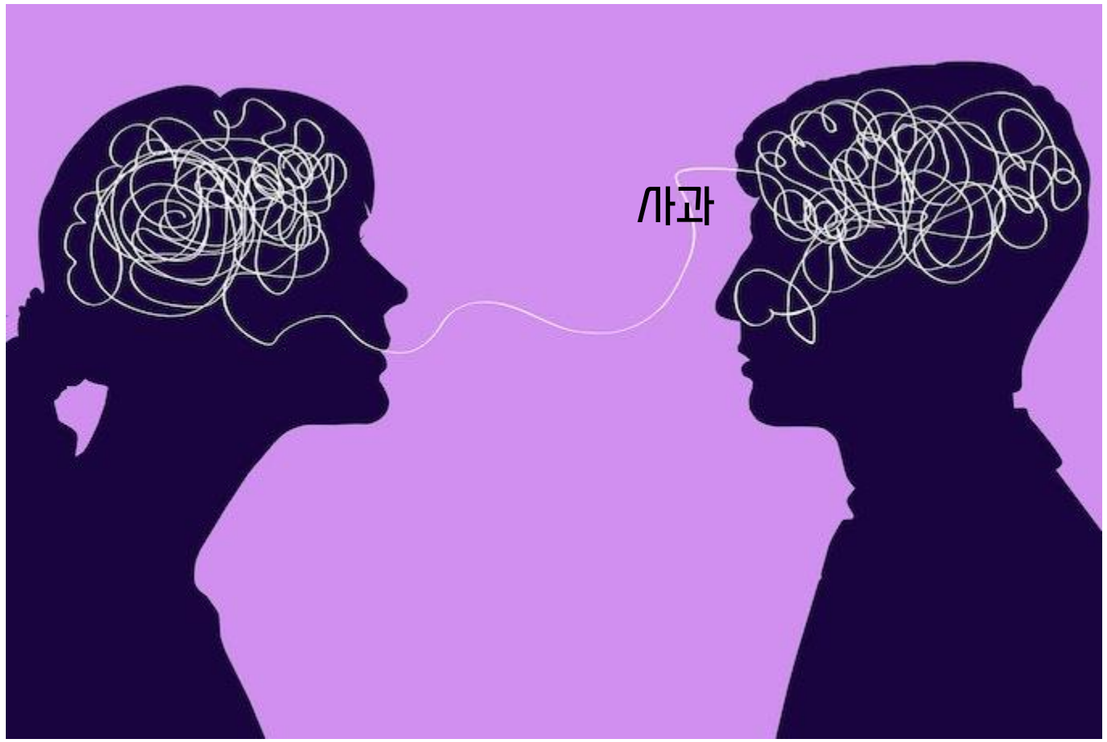
정보 전달이란?



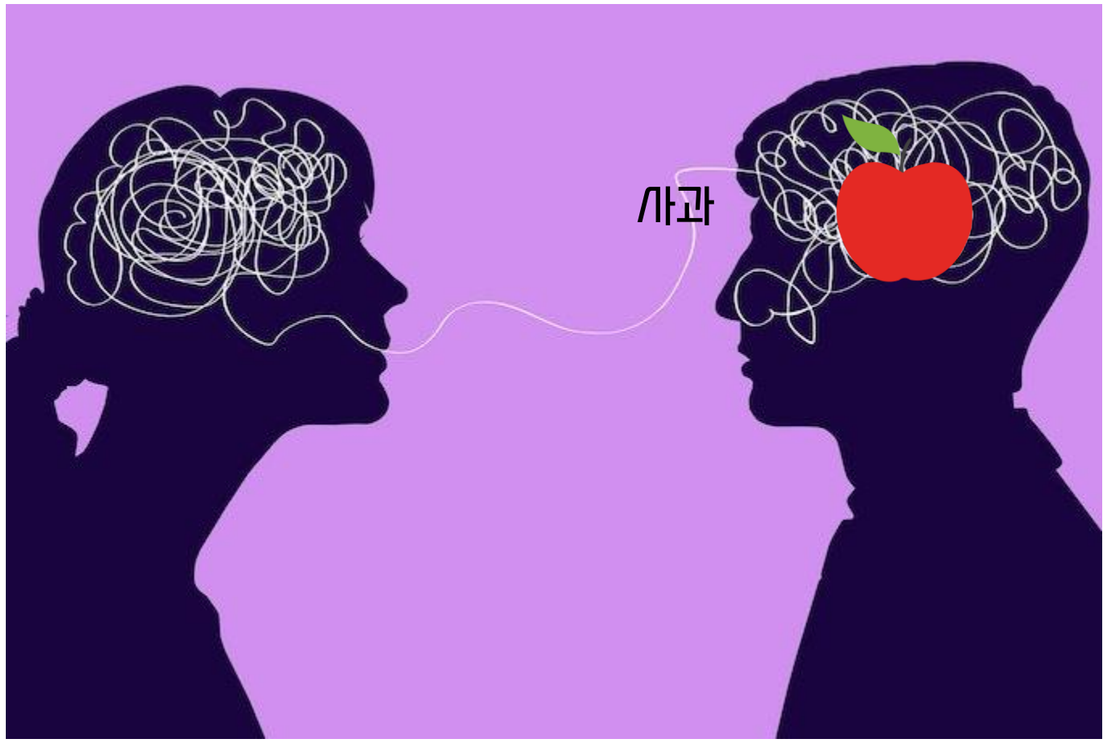
정보 전달이란?



정보 전달이란?



정보 전달이란?



사람과 컴퓨터를 이어주는 방법

- 사람의 생각(의도, 정보)을 컴퓨터에게 전달하는 방법

- Naïve Interface
  - 사람이 이해할 수 있지만, 엄격한 문법을 적용시키며 모호성이 없는 형태의 전달 방식입니다.
  - 일반적으로 사람이 컴퓨터에게 명령을 하기 위해 만들어진 프로그래밍 언어가 해당됩니다.

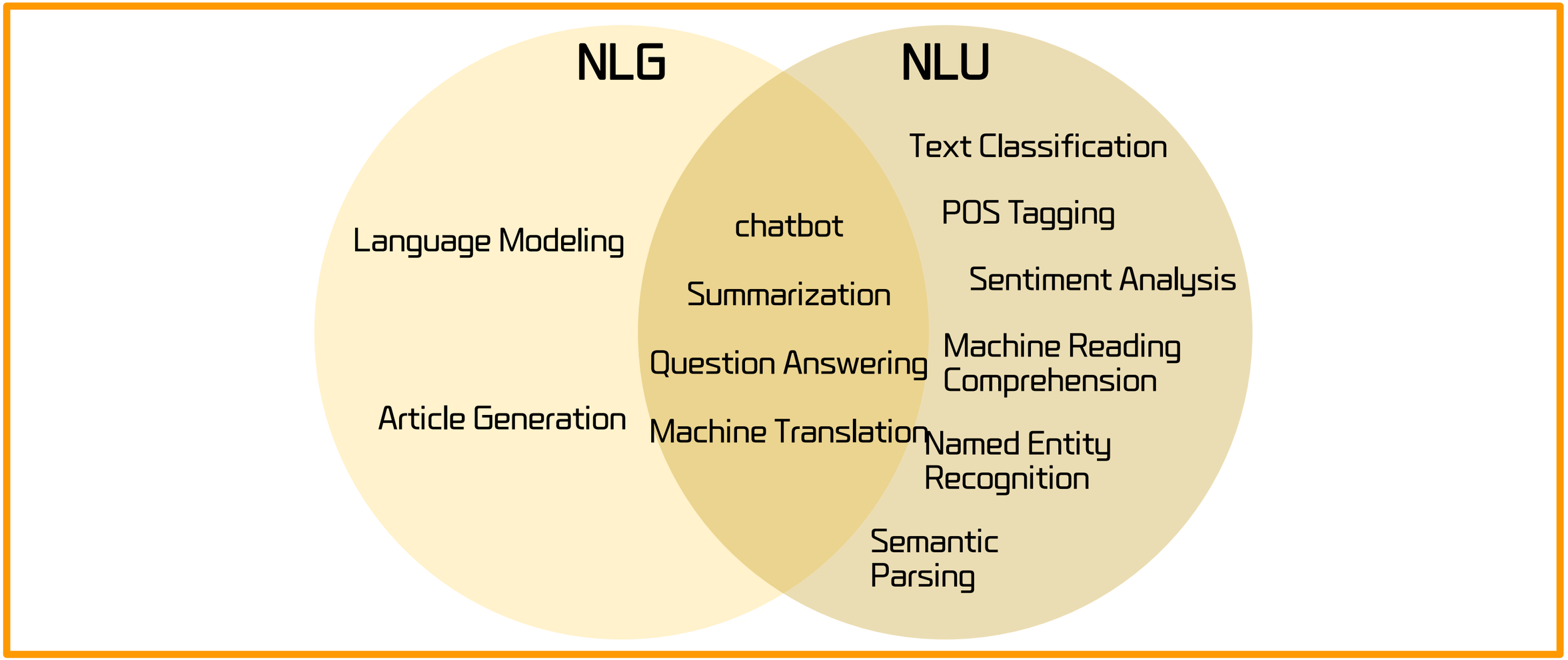
- Better Interface
  - 사람이 실제 사용하는 형태에 가까운 전달 방식입니다. 따라서 느슨한 문법, 모호성이 있는 형태가 됩니다.
  - 사람과 사람 사이에서 사용하는 자연 언어가 해당됩니다.

자연어 처리란?

- 자연어(Natural Language)
  - 자연어 혹은 자연 언어는 사람들이 일상적으로 쓰는 언어를 인공적으로 만들어진 언어인 인공어와 구분하여 부르는 개념

- 자연어 처리(Natural Language Processing)
  - 사람이 이해하는 자연어를 컴퓨터가 이해할 수 있는 값(벡터)으로 바꾸는 과정 (NLU – Natural Language Understanding)
  - 더 나아가 컴퓨터가 이해할 수 있는 값을 사람이 이해하도록(자연어로) 바꾸는 과정 (NLG – Natural Language Generation)

자연어 처리 연구 영역


NLG
NLU
Text Classification
POS Tagging
chatbot
Language Modeling
Sentiment Analysis
Summarization
Machine Reading Comprehension
Question Answering
Article Generation
Machine Translation
Named Entity Recognition
Semantic Parsing

자연어 처리와 딥러닝

전통적인 자연어 처리

- 전통적인 NLP는 단어를 Symbolic 데이터로 취급합니다. 즉 하나의 단어에 상징적인 의미가 있다고 생각합니다
  - symbolic 데이터의 특성 상 원-핫 벡터로만 표현이 가능하고, 이는 단어의 유사성과 모호성을 해결하지 못합니다.

빨강	파랑	주황
1	0	0
0	0	1
0	1	0

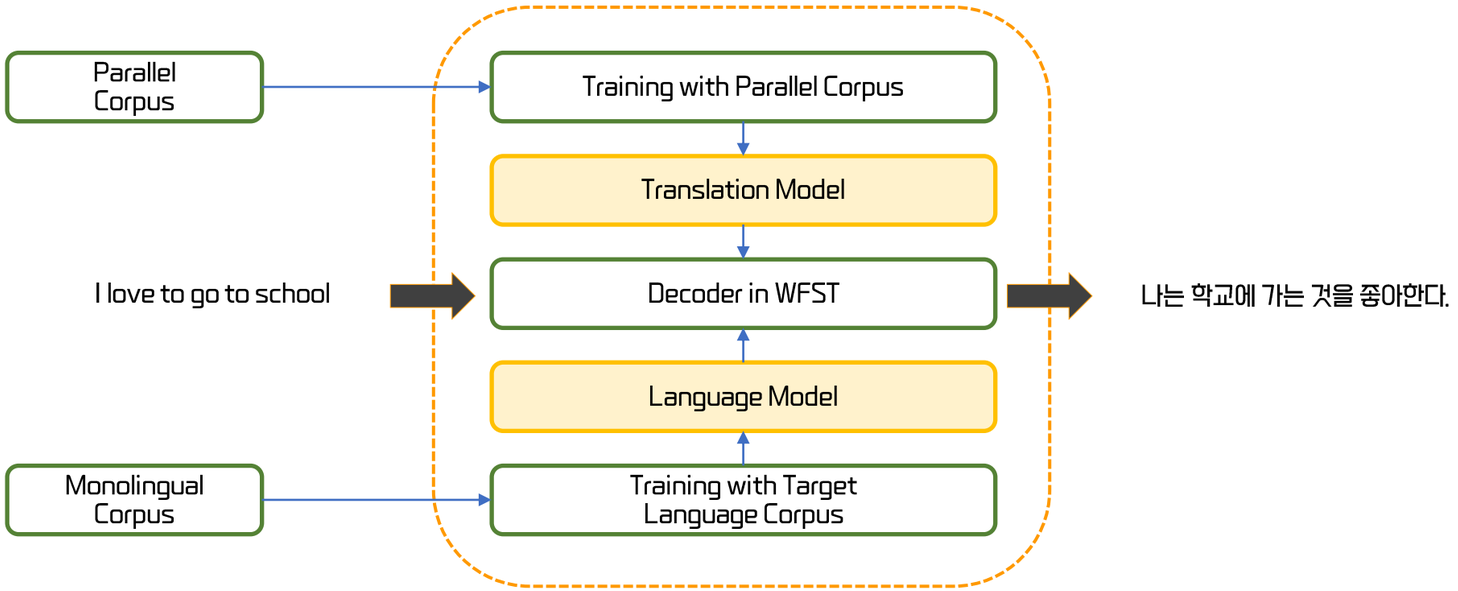
전통적인 자연어 처리

- 전통적인 NLP는 단어를 Symbolic 데이터로 취급합니다. 즉 하나의 단어에 상징적인 의미가 있다고 생각합니다
  - symbolic 데이터의 특성 상 원-핫 벡터로만 표현이 가능하고, 이는 단어의 유사성과 모호성을 해결하지 못합니다.
  - Word2Vec, FastText, Glove, Embedding 등 단어를 벡터로 표현하여 유사성과 모호성을 해결할 수 있습니다.

	dim1	dim2	dim3
빨강	0.7	-1.5	1.8
파랑	-0.8	1.2	-1.1
주황	0.8	-1.1	2.1

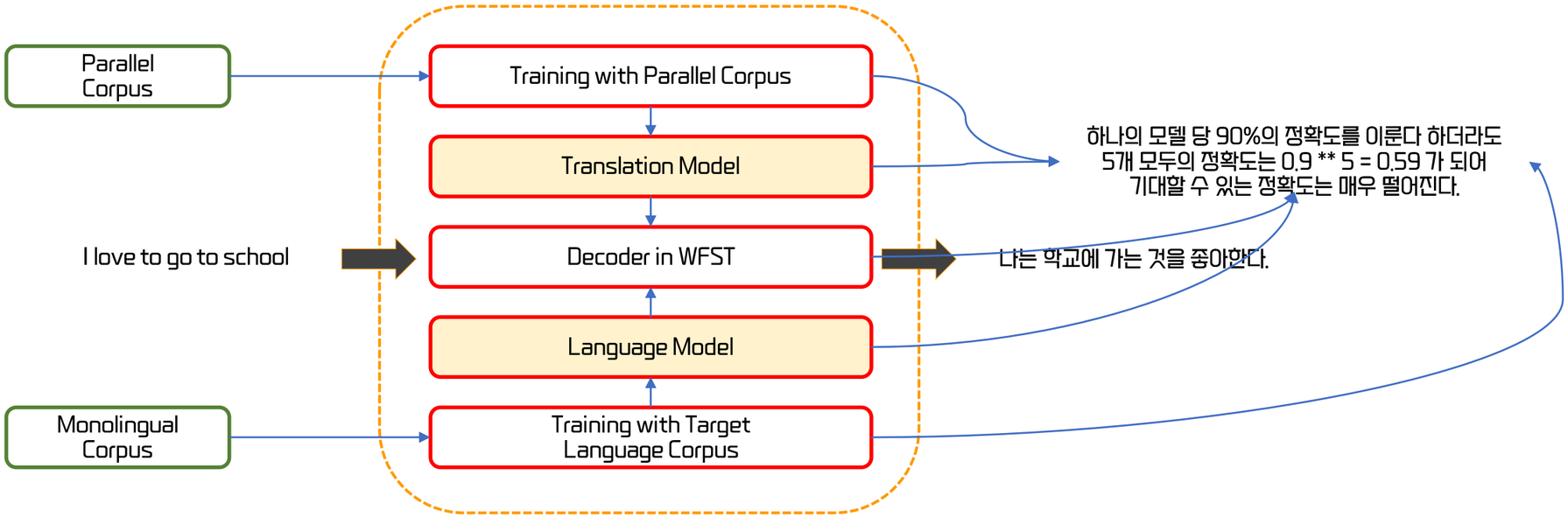
전통적인 자연어 처리

- 전통적인 NLP는 단어를 Symbolic 데이터로 취급합니다. 즉 하나의 단어에 상징적인 의미가 있다고 생각합니다
  - symbolic 데이터의 특성 상 원-핫 벡터로만 표현이 가능하고, 이는 단어의 유사성과 모호성을 해결하지 못합니다.
  - 전통적인 자연어 처리 방법은 여러 Sub module을 통해 전체를 구성하게 됩니다.



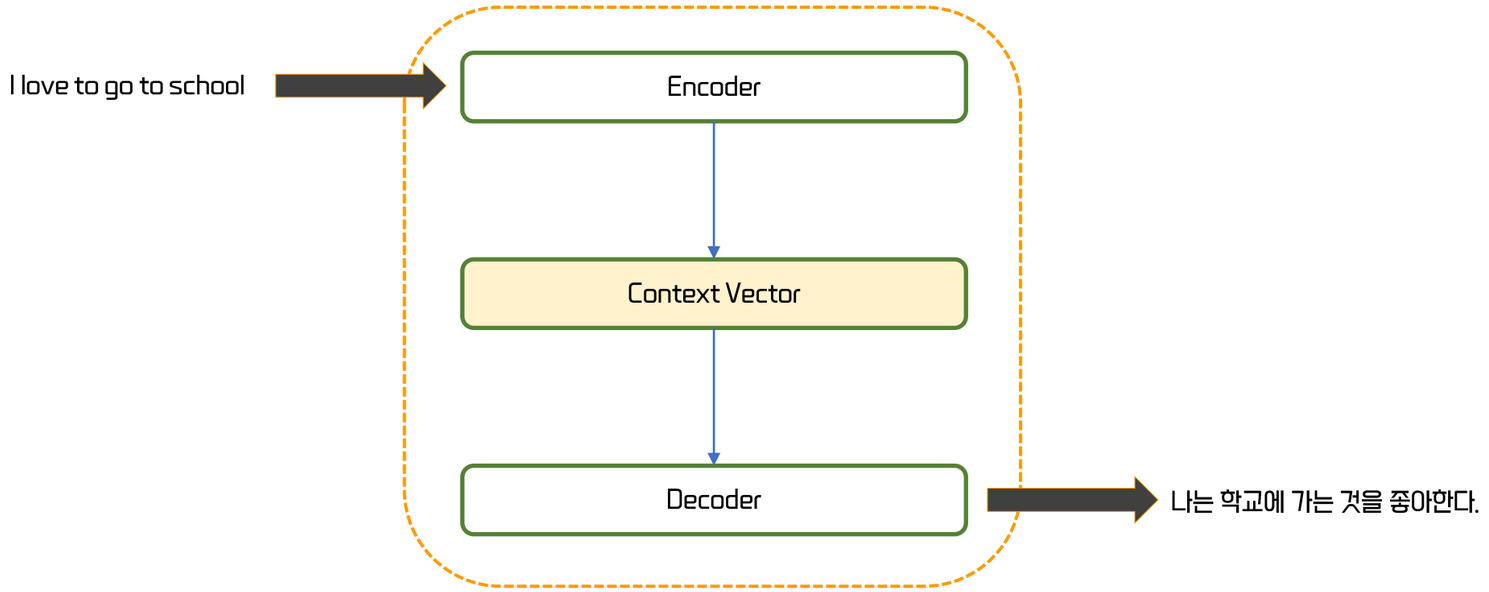
전통적인 자연어 처리

- 전통적인 NLP는 단어를 Symbolic 데이터로 취급합니다. 즉 하나의 단어에 상징적인 의미가 있다고 생각합니다
  - symbolic 데이터의 특성 상 원-핫 벡터로만 표현이 가능하고, 이는 단어의 유사성과 모호성을 해결하지 못합니다.
  - 전통적인 자연어 처리 방법은 여러 Sub module을 통해 전체를 구성하게 됩니다.



딥러닝 기반의 자연어 처리

- 단어를 연속적인 값으로 변환하여 단어와 단어 사이의 유사성과 모호성을 벡터의 유사도로 나타낼 수 있게 합니다.
- End-to-End 시스템을 추구합니다.



전통적인 NLP vs 딥러닝 기반 NLP

전통적인 심볼릭 기반 접근 방법	딥러닝 기반 접근 방법
이산적(Discreate), 심볼릭 공간	연속적(Continuous), 신경망 공간
사람이 인지하기 쉽습니다.	사람이 이해하기 어렵습니다.
디버그가 용이합니다.	디버깅이 어렵습니다.
연산 속도가 느립니다.	연산 속도가 빠릅니다.
모호성과 유의성에 취약합니다.	모호성과 유의성에 강인합니다.
여러 서브 모듈이 폭포수 형태를 취하므로 특징 추출에 노력이 필요합니다.	end-to-end 모델을 통한 성능개선과 시스템 간소화가 가능합니다.

딥러닝 기반 NLP 시스템

사람이 이해할 수 있는 이산적인 심볼릭 데이터

입력 : x

출력 : y

$h_x = f_e(x; \theta_e)$, 심볼릭 → 연속데이터
(임베딩 계층 및 Encoder)

$h_y = f_r(h_x; \theta_r)$
신경망 내부 연산

$y = f_d(h_y; \theta_d)$, 연속데이터 → 심볼릭
(분류 계층 및 Decoder)

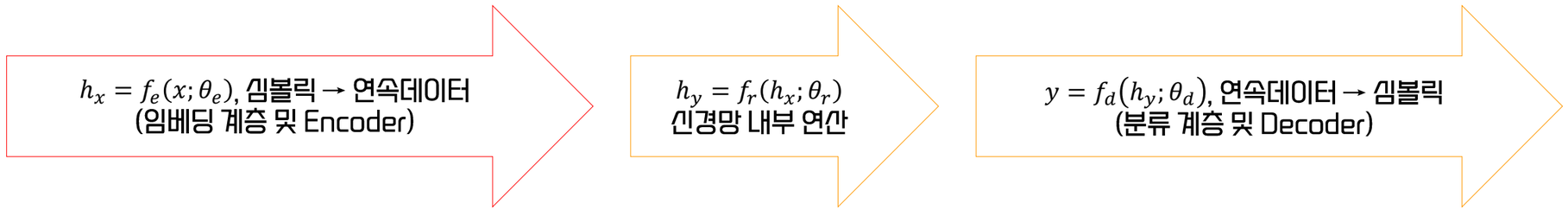
$Loss(\theta)$

딥러닝 기반 NLP 시스템

사람이 이해할 수 있는 이산적인 심볼릭 데이터

입력 : x

출력 : y



$Loss(\theta)$

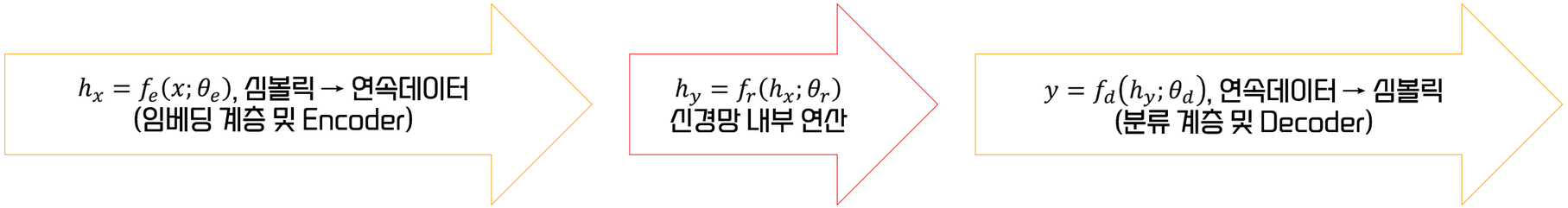
사람이 이해할 수 있는 단어. 즉 현실 세계

딥러닝 기반 NLP 시스템

사람이 이해할 수 있는 이산적인 심볼릭 데이터

입력 : x

출력 : y



$Loss(\theta)$

사람이 이해할 수 있는 데이터를 컴퓨터가
내부적으로 연산했습니다. 사람은 이해하지 못하고,
컴퓨터는 이해할 수 있는 가상 세계에 해당합니다.

딥러닝 기반 NLP 시스템

사람이 이해할 수 있는 이산적인 심볼릭 데이터

입력 : x

출력 : y

$h_x = f_e(x; \theta_e)$, 심볼릭 → 연속데이터
(임베딩 계층 및 Encoder)

$h_y = f_r(h_x; \theta_r)$
신경망 내부 연산

$y = f_d(h_y; \theta_d)$, 연속데이터 → 심볼릭
(분류 계층 및 Decoder)

$Loss(\theta)$

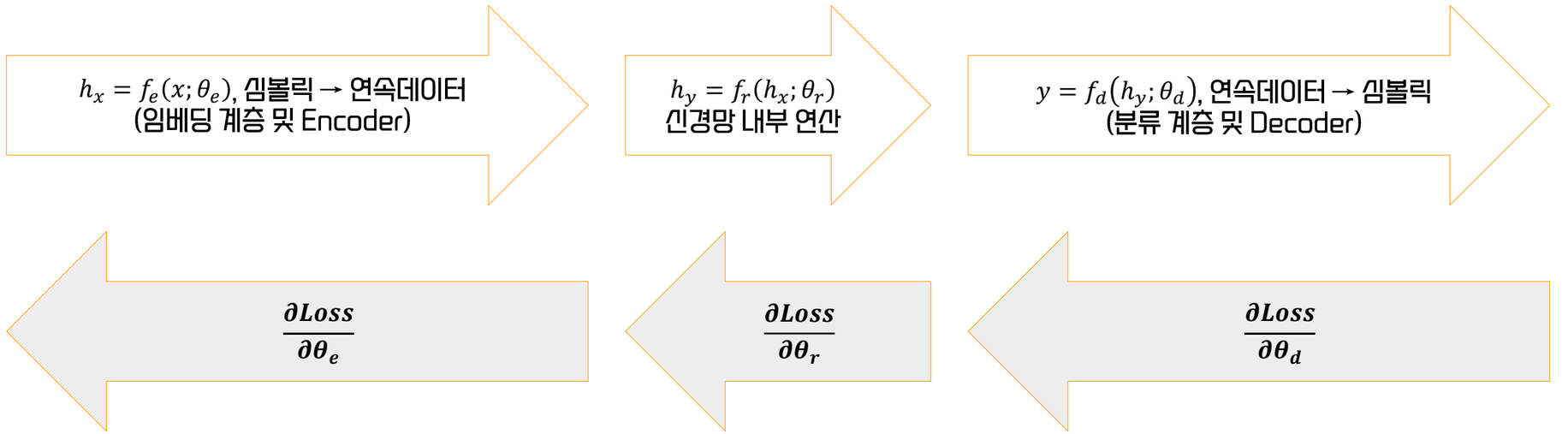
컴퓨터가 이해한 데이터를 사람이 이해할 수 있도록
다시 돌려 놓습니다. 즉 현실 세계의 데이터 입니다.

딥러닝 기반 NLP 시스템

사람이 이해할 수 있는 이산적인 심볼릭 데이터

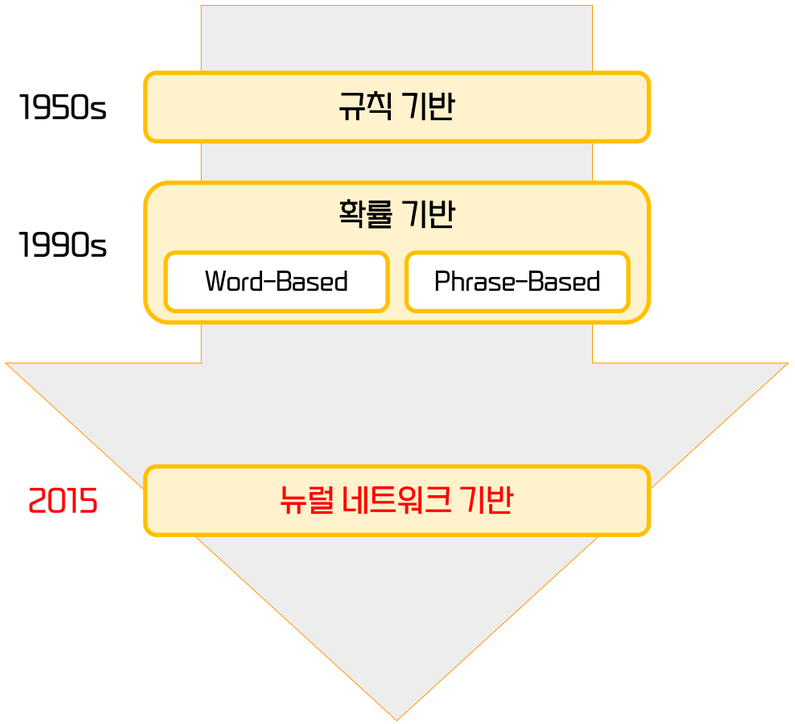
입력 : x

출력 : y



자연어 처리 패러다임의 변화 과정

- 효율적인 임베딩을 통한 성능 개선을 이루어 냈습니다.
  - 단어, 문장의 유사성
  - 컨텍스트 임베딩
- End-to-End 구성으로 인한 효율과 성능 개선
- 기계번역의 경우 다른 분야에 비해 가장 먼저 성공적인 상용화를 이루어 냈습니다.



다른 분야와 차이점

인공지능 삼대장(+1)

- 컴퓨터 비전(Computer Vision)
  - 이미지 인식 및 분류 모델(Image Classification / Image Recognition)
  - 객체 탐지 모델(Object Detection)
  - 이미지 생성 모델 (Image Generation)
  - 해상도 업 스케일링 모델(Super Resolution)
- 자연어 처리(Natural Language Processing)
  - 텍스트 분류(Text Classification)
  - 기계 번역(Machine Translation)
  - 텍스트 요약(Text Summarization)
  - 질의응답(Question Answering)
- 음성 처리(Speech Processing)
  - 음성 인식(Speech Recognition – STT)
  - 음성 합성(Speech Synthesis – TTS)
  - 화자 인식(Speaker Identification)

인공지능 삼대장(+1)

- 컴퓨터 비전(Computer Vision)
  - 이미지 인식 및 분류 모델(Image Classification / Image Recognition)
  - 객체 탐지 모델(Object Detection)
  - 이미지 생성 모델 (Image Generation)
  - 해상도 업 스케일링 모델(Super Resolution)
- 자연어 처리(Natural Language Processing)
  - 텍스트 분류(Text Classification)
  - 기계 번역(Machine Translation)
  - 텍스트 요약(Text Summarization)
  - 질의응답(Question Answering)
- 음성 처리(Speech Processing)
  - 음성 인식(Speech Recognition – STT)
  - 음성 합성(Speech Synthesis – TTS)
  - 화자 인식(Speaker Identification)

여기에 강화학습 추가!
Reinforcement Learning

NLP와 다른 분야의 차이점

- NLP
  - **Discrete Value를 다룹니다.**
    - 이산적인 값을 다루게 됩니다. 단어, 문장 들은 연속적인 값이 아닌, 하나의 단어나 문장이 의미하는 바가 확실히 있는 이산적인 값입니다.
  - **분류 문제로 접근할 수 있습니다.**
    - 문장이 생성 되는 작업은 곧 단어를 생성하는 작업이고, 이는 곧 이산적인 값(단어)를 생성하게 됩니다.
    - 즉 이전 단어들이 주어졌을 때 다음 단어가 어떤 것이 나와야지 자연스러운가?를 판단하기 때문에 분류 문제에 속합니다.
- Others (ex. Computer Vision)
  - **Continuous Value를 다룹니다.**
    - 이미지의 픽셀 하나를 따로 놓고 보면 의미 하는 바를 알아 낼 수 없고, 이 픽셀 데이터들을 연속적으로 이어 붙어야 이미지 데이터로써 인식이 가능합니다.
  - **문제에 따라 접근 방식이 다를 수 있습니다.**
    - 이미지 생성, 음성 합성 등의 경우 기본적으로 Regression Task를 기본으로 삼습니다.
    - 이미지 생성의 경우 얼마나 표현하고자 하는 이미지와 흡사한지, 음성 합성의 경우 얼마나 자연스러운 지 등등을 판단하기 때문입니다.

NLP와 다른 분야의 차이점

- NLP
  - **샘플의 확률 값을 구할 수 있습니다.**
    - 어떤 단어가 주어졌을 때 이 단어가 나올 확률을 구해낼 수 있습니다.
    - $P(x = \text{단어}, \text{문장})$
    - 두 문장이 존재할 때 더 자연스러운 문장을 판단할 수 있습니다. 확률적으로 봤을 때. 즉 확률이 높은 문장을 판단할 수 있습니다.
      - “나 먹는다 밥” vs “나는 밥을 먹는다.”
    - 이러한 성질을 이용해 자연어 생성이 가능합니다.
  - **문장 생성(자연어 생성)**
    - auto-regressive 속성을 지닙니다.
      - 이전의 생성된 단어를 토대로 다음 단어를 예측해서 생성합니다.
    - GAN(적대적 생성망) 적용이 불가능 합니다.
- Others (ex. Computer Vision)
  - **샘플의 확률 값을 구할 수 없습니다.**
    - 픽셀 하나가 주어졌을 때 그 이미지의 확률을 구해 내는 것은 불가능 합니다.
    - $P(x = \text{이미지})$
  - **이미지 생성**
    - auto-regressive 속성이 없습니다.
    - GAN 적용이 가능합니다.

NLP 연구의 요구사항

- 도메인 지식이 필요합니다.
  - 언어적인 지식을 매우 많이 필요로 합니다. 예를 들면 한국어는 어떠한 언어적 특성을 지니는가?에 대한 이해가 필요합니다. 사람은 언어를 어떻게 발전시켜왔는지, 이 언어가 왜 이렇게 표현되어야 하는지 등등을 배경지식으로 가지고 있는 것이 좋습니다.

- 전처리의 어려움(Nasty Preprocessing)
  - Task에 따른 정제(Normalization) 과정이 필요합니다.
  - 텍스트 데이터는 크롤링이나 각종 데이터 수집기법을 이용해 굉장히 쉽게 얻어낼 수 있는데, 이를 전처리 하기 위해서는 많은 노력이 필요합니다.

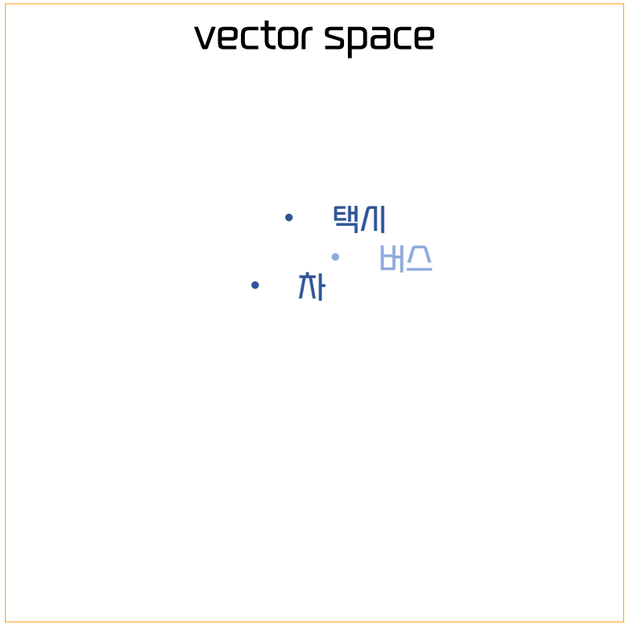
자연어 처리가 어려운 이유

NLP가 어려운 이유. 모호성(Ambiguity)

- 단어의 중의성 (Word Sense Ambiguation)
  - 하나의 단어가 여러 뜻을 가지고 있는 경우입니다.
  - 예를 들어 “차” 라는 하나의 단어에도 타는 “차”, 마시는 “차”, 발로 찰 때의 “차” 가 모두 다릅니다. 이러한 중의성을 해소하는 것을 중의성 해소(Word Sense Disambiguation)라고 합니다. 주변 단어를 보고 중의성이 있는 단어를 판단해야 합니다.

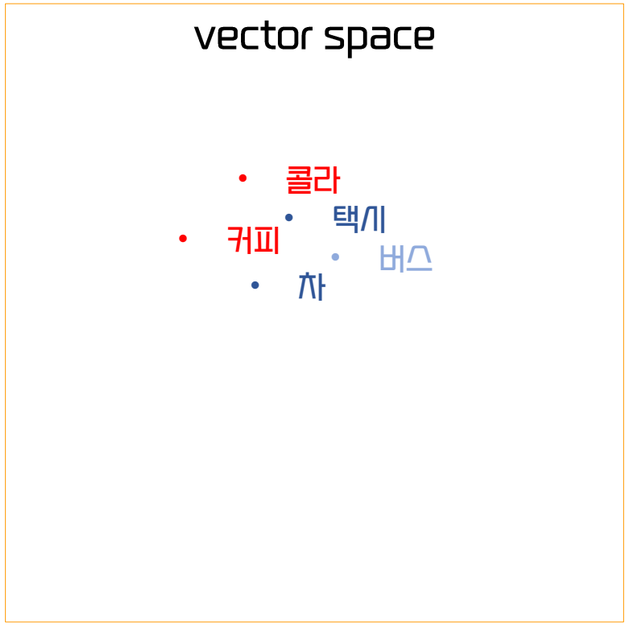
NLP가 어려운 이유. 모호성(Ambiguity)

- 단어 중의성으로 인한 문제 발생 사례



NLP가 어려운 이유. 모호성(Ambiguity)

- 단어 중의성으로 인한 문제 발생 사례



NLP가 어려운 이유. 모호성(Ambiguity)

- 문장의 중의성 (Sentence Sense Ambiguation)
  - 문장이 다양하게 해석될 수 있는 성질을 의미합니다. 이를 중의문 이라고 합니다.

문 2. 중의성이 드러나지 않는 문장은?

① 그들은 자신의 이익을 위해 이번 문제를 축소하고 은폐하려 하였다.

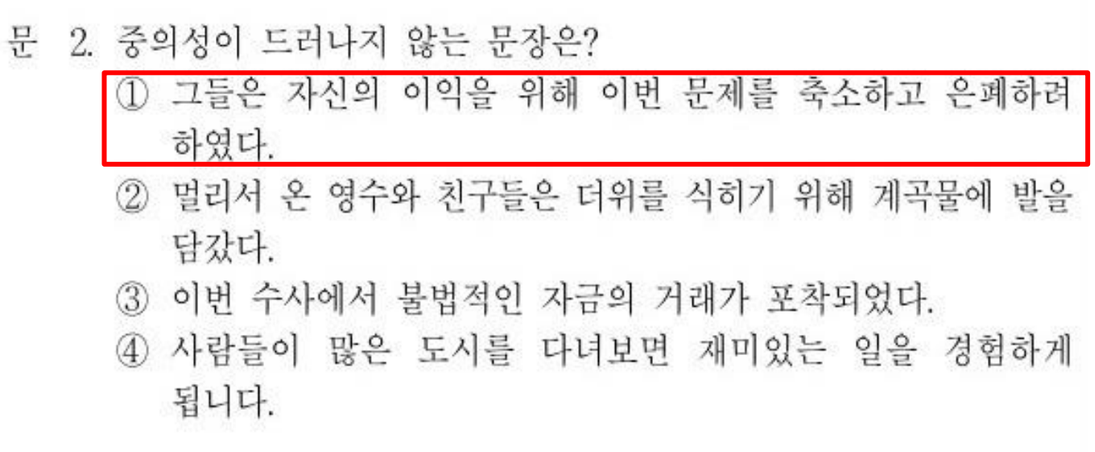
② 멀리서 온 영수와 친구들은 더위를 식히기 위해 계곡물에 발을 담갔다.

③ 이번 수사에서 불법적인 자금의 거래가 포착되었다.

④ 사람들이 많은 도시를 다녀보면 재미있는 일을 경험하게 됩니다.

NLP가 어려운 이유. 모호성(Ambiguity)

- 문장의 중의성 (Sentence Sense Ambiguation)
  - 문장이 다양하게 해석될 수 있는 성질을 의미합니다. 이를 중의문 이라고 합니다.



문 2. 중의성이 드러나지 않는 문장은?

① 그들은 자신의 이익을 위해 이번 문제를 축소하고 은폐하려 하였다.

② 멀리서 온 영수와 친구들은 더위를 식히기 위해 계곡물에 발을 담갔다.

③ 이번 수사에서 불법적인 자금의 거래가 포착되었다.

④ 사람들이 많은 도시를 다녀보면 재미있는 일을 경험하게 됩니다.

NLP가 어려운 이유. 모호성(Ambiguity)

- 문장의 중의성 (Sentence Sense Ambiguation)
  - 문장의 중의성이 발생하는 대표적인 이유는 언어를 통해 정보를 압축하려고 하는 성질 때문입니다.
  - 아래의 원문에 대한 세 가지 해석은 모두 맞는 말입니다. 사람도 마찬가지고, 컴퓨터도 마찬가지로 전후상황을 이해하고 있지 않으면 원문의 문장을 정확하게 해석하는 것이 쉽지 않습니다.

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

NLP가 어려운 이유. 모호성(Ambiguity)

- 문장의 중의성 (Sentence Sense Ambiguation)
  - 언어는 효율성을 극대화 하기 위해 정보의 생략을 필수적으로 포함합니다. 정보가 생략 될 수록 효율성은 증대됩니다. 즉 짧은 길이의 문장으로 정확한 정보를 전달해 낼 수 있어야 합니다.
  - 생략되는 정보는 보통 모두가 알고 있는 뻔한 정보들이 됩니다.
  - 문장의 생략 때문에 모호성이 극대화 됩니다.
  - 사람은 전후상황을 매우 쉽게 인지할 수 있으나, 컴퓨터는 이를 인지하기가 힘들기 때문에 모호성을 해결하기가 어렵습니다.

NLP가 어려운 이유. 모호성(Ambiguity)

- 문장의 중의성 (Sentence Sense Ambiguation)
  - 아래 문장은 문장 내 정보의 부족이 야기한 구조 해석의 문제를 나타냅니다.

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

언어는 왜 모호성을 갖는가?

- 언어는 마치 생명체와 같이 진화하며, 특히 효율성을 극대화 하는 방향으로 진화합니다.
- 따라서 최대한 짧은 문장 내에 많은 정보를 담고자 합니다.
  - 즉 정보량이 낮은 내용(context)은 생략합니다.
  - 여기에서 모호함(ambiguity)이 발생합니다.
- 생략된 context를 인간은 효율적으로 채울 수 있지만, 기계는 이러한 task에 매우 취약합니다.

NLP가 어려운 이유. 의역(Paraphrase)

- 문장의 표현 방식은 매우 다양하고, 비슷한 의미의 단어들이 존재하기 때문에 paraphrase의 문제가 존재합니다.

NLP가 어려운 이유. 의역(Paraphrase)

- 문장의 표현 방식은 매우 다양하고, 비슷한 의미의 단어들이 존재하기 때문에 paraphrase의 문제가 존재합니다.

- 남자의 입에서 주스가 흘러나오고 있다.
- 남자의 입을 타고 노란 주스가 흘러나오고 있다.
- 입에서 나온 주스가 남자의 턱을 타고 흘러내리고 있다.
- 남자는 충격 받은 표정으로 주스를 입에서 내보내고 있다.
- 예나는 선정이 딸이라는 이야기를 듣고 남자의 입에서 주스가 흘러나온다.

...

NLP가 어려운 이유. 연속적이 아닌 이산적인 값

- 이산 값을 갖는 자연어는 사람의 입장에서 인지가 쉬울 수 있으나, 기계의 입장에서는 매우 어려운 값입니다.
- One Hot Encoding으로 표현된 값은 유사도나 모호성을 표현할 수 없습니다.
  - 서로 다른 One Hot Vector끼리의 유사도나 거리는 모두 동일합니다.
- 따라서 컴퓨터는 아래와 같은 질문에 대답할 수 없습니다.
  - 파랑과 핑크 중에 빨강과 가까운 단어는 무엇인가?
  - 하지만 사람의 어휘 체계는 계층적 구조를 띄고 있습니다.
- 또한 높은 차원으로 표현되어 매우 희소(sparse)하게 됩니다.

NLP가 어려운 이유. 연속적이 아닌 이산적인 값

- 이산 값을 갖는 자연어는 사람의 입장에서 인지가 쉬울 수 있으나, 기계의 입장에서는 매우 어려운 값입니다.
- One Hot Encoding으로 표현된 값은 유사도나 모호성을 표현할 수 없습니다.
  - 서로 다른 One Hot Vector끼리의 유사도나 거리는 모두 동일합니다.
- 따라서 컴퓨터는 아래와 같은 질문에 대답할 수 없습니다.
  - 파랑과 핑크 중에 빨강과 가까운 단어는 무엇인가?
  - 하지만 사람의 어휘 체계는 계층적 구조를 띄고 있습니다.
- 또한 높은 차원으로 표현되어 매우 희소(sparse)하게 됩니다.

딥러닝 에서는 Word Embedding으로 해결합니다!

한국어 자연어 처리의 어려움

한국어 자연어 처리가 특히 어려운 이유. 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 더해짐
고립어	영어, 중국어	어순에 따라 단어의 문법적 기능이 정해짐

- “나는 밥을 먹었습니다” 를 여러 나라 언어로 하면?
  - 일본어 : 私はご飯を食べました。
  - 독일어 : Ich aß Reis.
    - aß 가 독일어로 “먹었다.”라는 뜻입니다. 원형은 Essen (먹다) 입니다.
    - 굴절어의 특징상 단어의 형태가 바뀜으로써 문법적 기능(과거형)이 부여 되었습니다.
    - 영어는 굴절어이자 고립어 입니다.
  - 영어 : I did eat rice.
    - 영어는 고립어이기 때문에 Did I eat rice? 라고 어순을 바꾸면 의문문으로 바뀌게 됩니다.

교착어. 접사 추가에 따른 의미 파생

교착어. 접사 추가에 따른 의미 파생

원형(어간)	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

순번	문장	정상 여부
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

한국어 자연어 처리가 특히 어려운 이유. 띄어쓰기

- 근대 이전까지 동양권 언어에는 띄어쓰기가 존재하지 않았습니다.
  - 서양에서는 중세시대에 띄어쓰기가 확립되었습니다.
- 따라서 아직 우리나라 말은 여전히 띄어쓰기와 궁합을 맞추고 있습니다.
- 그럼에도 불구하고 우리가 한국어를 사용함에 있어서 전혀 불편함이 없는 이유는 무엇일까요!?

한국어 자연어 처리가 특히 어려운 이유. 띄어쓰기

- 근대 이전까지 동양권 언어에는 띄어쓰기가 존재하지 않았습니다.
  - 서양에서는 중세시대에 띄어쓰기가 확립되었습니다.
- 따라서 아직 우리나라 말은 여전히 띄어쓰기와 궁합을 맞추고 있습니다.
- 그럼에도 불구하고 우리가 한국어를 사용함에 있어서 전혀 불편함이 없는 이유는 무엇일까요!?
  - 한국어는띄어쓰기가지켜지지않아도읽기가매우쉽기때문입니다.

한국어 자연어 처리가 특히 어려운 이유. 띄어쓰기

- 근대 이전까지 동양권 언어에는 띄어쓰기가 존재하지 않았습니다.
  - 서양에서는 중세시대에 띄어쓰기가 확립되었습니다.

- 따라서 아직 우리나라 말은 여전히 띄어쓰기와 궁합을 맞추고 있습니다.

- 띄어쓰기에 따라 의미가 달라질 수도 있는 경우

한국어 자연어 처리가 특히 어려운 이유. 평서문과 의문문 차이 부재

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

한국어 자연어 처리가 특히 어려운 이유. 주어 부재

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

- 영어는 주어가 존재하는 형태로 문장이 완성됩니다.
- 하지만 한국어는 주어가 없이 문장이 완성될 수 있으며, 주어는 주변 상황(context)를 보고 예측해야 합니다.

- 한국어를 영어로 번역할 때 컴퓨터는 주어가 없는 상황에서 주어를 예측해야 합니다.

한국어 자연어 처리가 특히 어려운 이유. 한자 기반의 언어

- 표의 문자인 한자를 표음 문자인 한글로 wrapping 하였습니다.
  - 표의 문자 : 의미 또는 사물의 형상을 글씨로 나타낸 것
  - 표음 문자 : 사람이 말하는 소리, 음성을 글씨로 나타낸 것

- Wrapping 과정에서 정보의 손실이 발생됩니다.

車 vs 茶

- 한자는 문자 하나만 보고 어떤 차인지 알 수 있으나, 한국어는 주변 단어와 context를 파악해야 어떤 차 인지 알 수 있습니다.

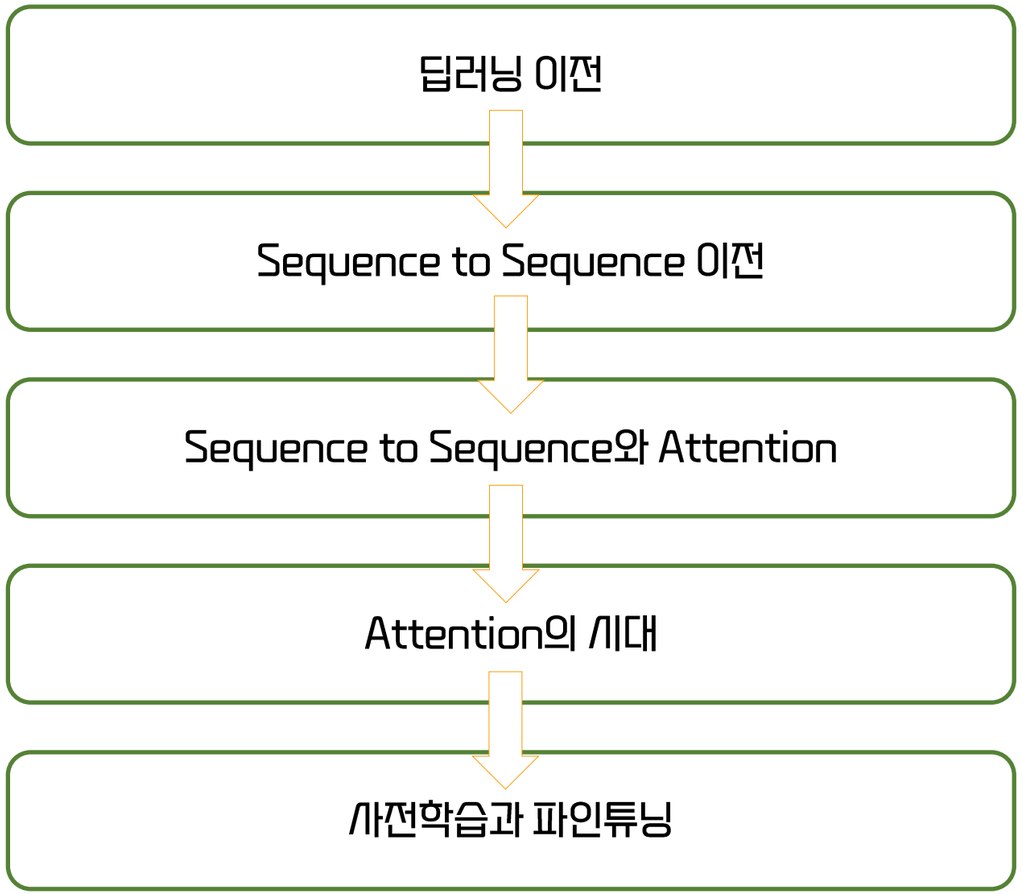
극악 난이도 한국어 NLP

- 한글은 굉장히 늦게 만들어진 문자입니다. 따라서 기존 다른 문자들의 장점을 흡수하였고, 굉장히 과학적으로 만들어진 문자입니다.

- 사용하기 쉽게 만들어졌기 때문에 효율이 극대화 되어 한국어에 대한 자연어 처리가 더욱 어려운 것입니다.

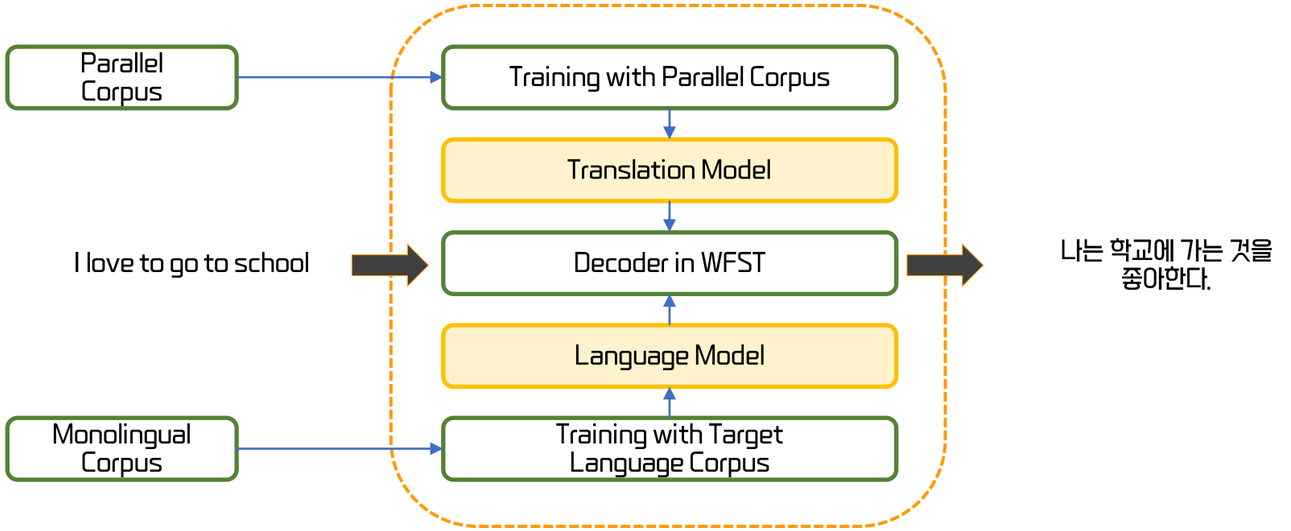
딥러닝 기반 자연어 처리 역사

NLP의 흐름


딥러닝 이전
Sequence to Sequence 이전
Sequence to Sequence와 Attention
Attention의 시대
사전학습과 파인튜닝

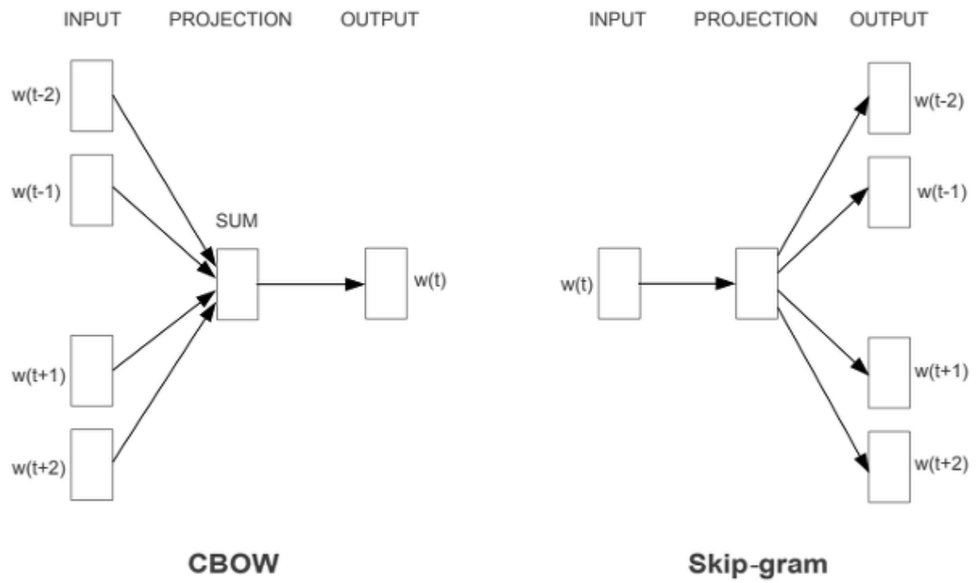
딥러닝 이전의 자연어 처리. Traditional SMT

- 전형적인 NLP Application 구조
  - 여러 단계의 서브 모듈로 구성되어 복잡한 디자인 구성 되었습니다.
  - 여러 모델들이 각각 존재하기 때문에 구현 및 시스템 구성이 어려운 단점이 있었습니다.
  - 각기 발생한 에러가 중첩 및 가중되어 에러 전파가 일어납니다.



Sequence to Sequence 이전의 자연어 처리

- Word Embedding



Sequence to Sequence 이전의 자연어 처리

- Text Classification

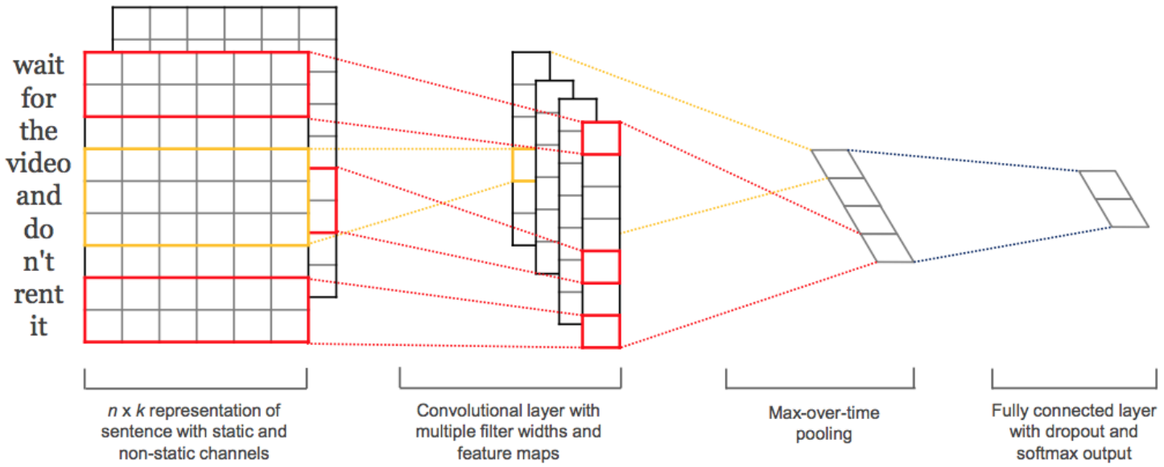


Figure 1: Model architecture with two channels for an example sentence.

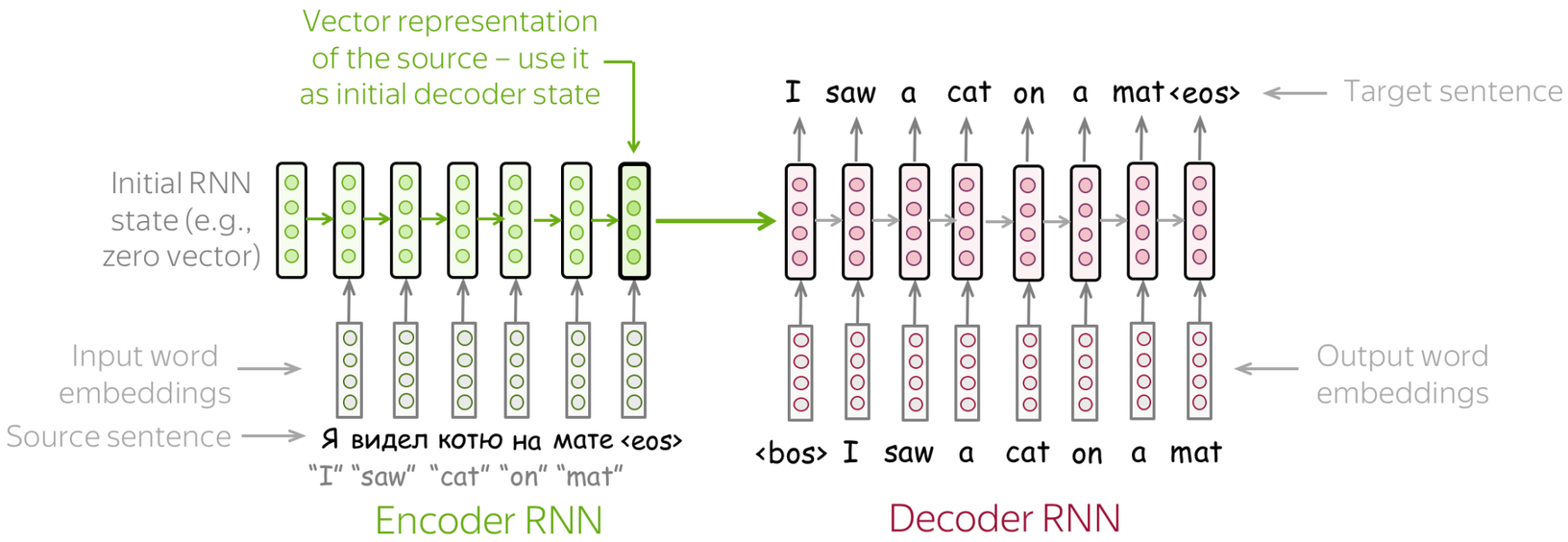
Sequence to Sequence 이전의 자연어 처리

- Word Embedding, Text Classification 모두 텍스트 데이터를 수치형 값(Numerical Values)으로 바꾸는 데에 그쳤습니다.

- 단어가 주어지면 어떠한 벡터로 변환하고, 문장이 주어지면 해당 문장이 긍정인지, 부정인지를 벡터로 나타내는 기법이 발전 하였습니다.

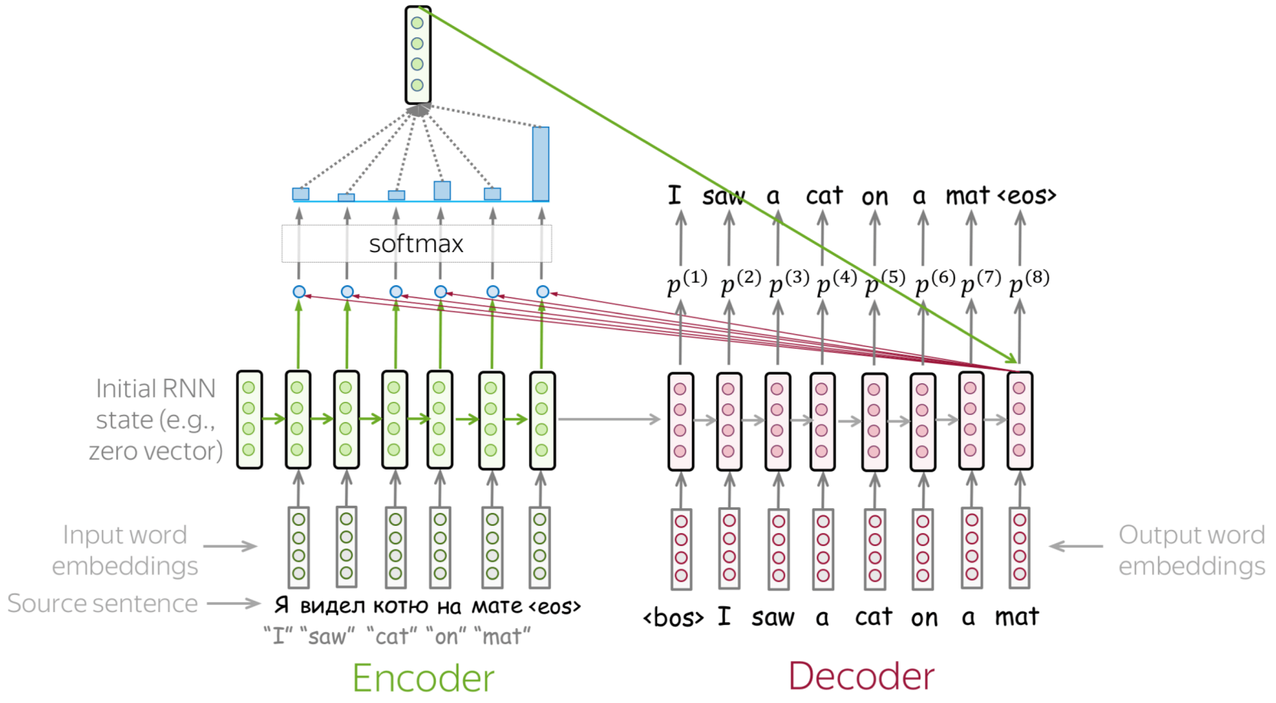
Sequence to Sequence와 Attention

- 텍스트를 수치형 벡터로 만드는 것을 넘어 텍스트를 입력 받아 텍스트를 출력하는 모델이 발전 되었습니다.



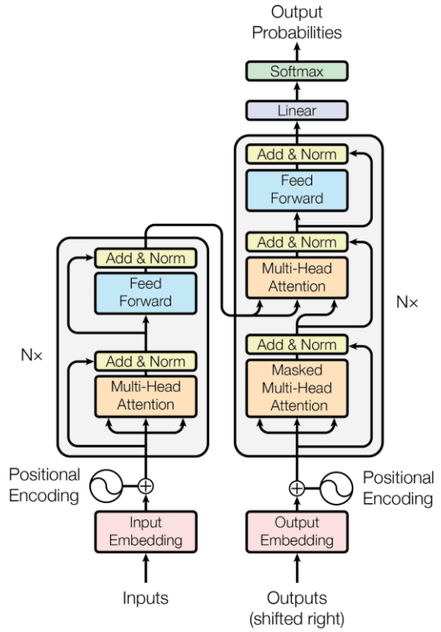
Sequence to Sequence와 Attention

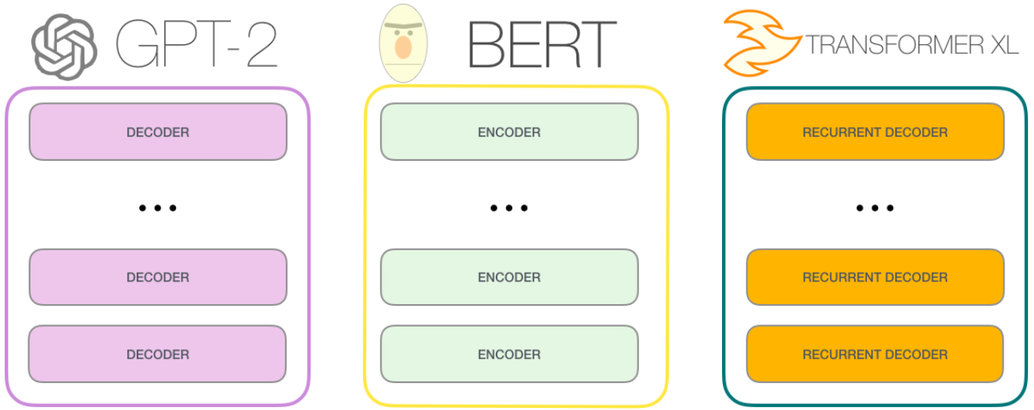
- Attention 모델이 등장함에 따라 자연어 처리는 엄청난 발전을 이룰 수 있었습니다.



Attention의 시대

- 기존의 Sequence To Sequence 를 뛰어 넘어 2017년 구글에서 발표한 Attention 모델 기반의 Transformer는 Attention 기법만을 이용해 딥러닝 네트워크가 구성되었고, 성능이 월등히 뛰어나고, 확장성도 굉장히 용이하였습니다.





사전학습과 파인 튜닝 (BERT)

- Transformer 모델을 굉장히 넓고 깊게 쌓아 대량의 인터넷에 쌓여 있는 텍스트 데이터들을 이용해 거대한 네트워크를 만들어 낼 수 있습니다. 기존의 레이블링이 필요 없이 인터넷의 문장을 활용해 만들어 냅니다.

- 이러한 데이터를 이용해 학습된 거대한 네트워크를 이용해 다양한 Downstream task를 파인 튜닝하는 작업들을 하게 될 수 있게 되었습니다.