

Deep Learning Week 6

Natural Language Processing

Hanyang Artificial Intelligence Group



What is NLP?

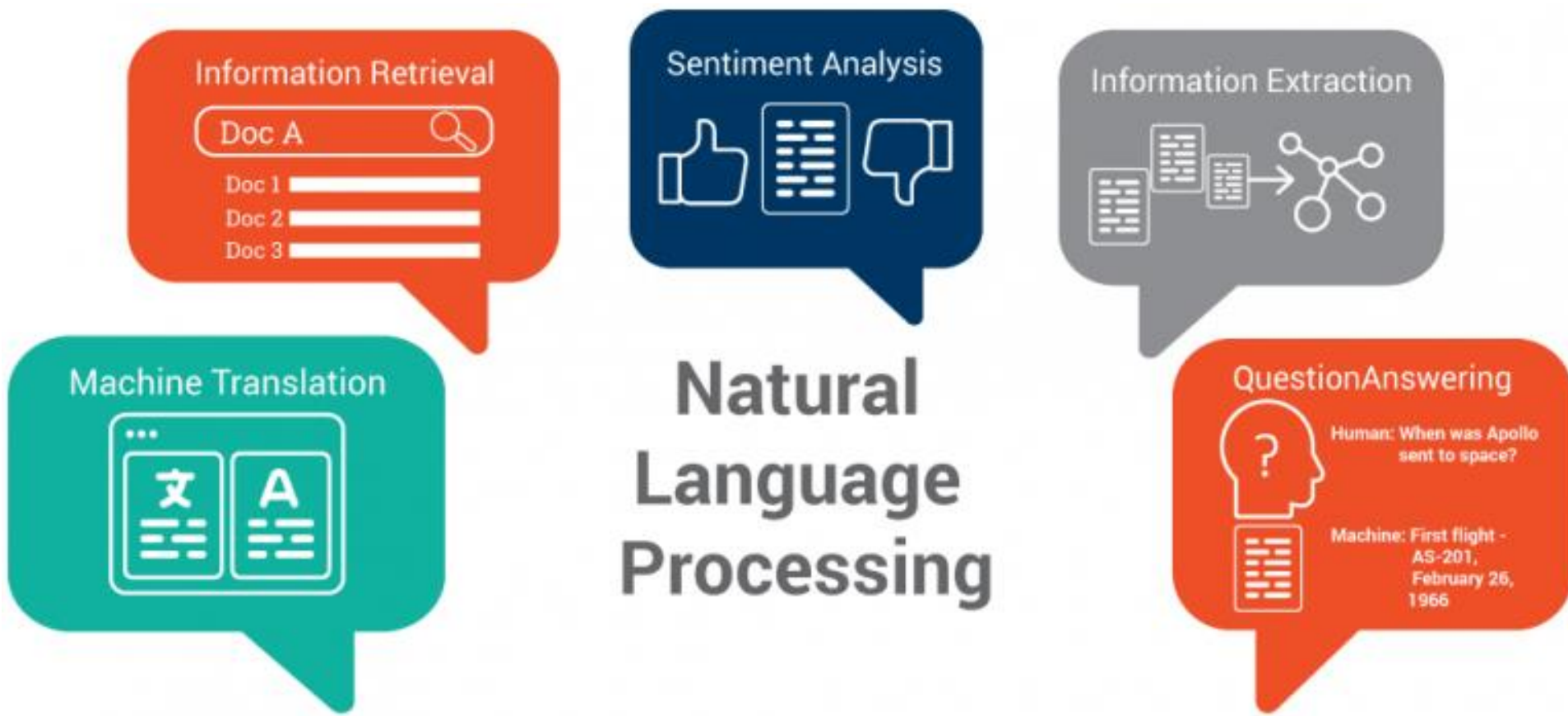
인간의 언어를 이해할 수 있는 AI

- 인간이 일상에서 사용하는 언어의 의미를 분석하여 컴퓨터가 처리할 수 있도록 하는 기술
- 딥러닝을 활용한 방식의 발전으로 최근 크게 떠오르고 있는 분야
- NLP의 핵심 목표는 **자연어를 컴퓨터가 이해할 수 있는 표현(벡터)으로 변환하는 것!**



What can we do with NLP?

인간이 사용하는 언어의 정보를 분석하거나, 원하는 형태로 재가공하는 모든 작업들!



딥러닝 이전 방식의 자연어 처리

규칙 또는 통계 기반의 언어 처리

- 딥러닝이 적용되기 이전 자연어 처리는 단순히 문장 내부의 특정 단어의 등장 여부를 확인하거나 어떤 종류의 단어들이 많이 발생하는지 통계적으로 분석하는 방식을 사용함
- 하지만 인간이 사용하는 언어는 규칙을 기반으로 분석하기에는 너무 복잡하기 때문에, 인간의 언어 능력에 비해 매우 부족한 성능을 보임
- Ex) Bag of Words: 특정 문장에 등장하는 각 단어들의 출현 빈도를 카운트하여 문장을 표현하는 방식으로, 각 단어의 등장 순서는 고려할 수 없음

	the	red	dog	cat	eats	food
1. the red dog →	1	1	1	0	0	0
2. cat eats dog →	0	0	1	1	1	0
3. dog eats food →	0	0	1	0	1	1
4. red cat eats →	0	1	0	1	1	0

자연어 데이터의 전처리(preprocessing) 방법

Tokenizing

- 텍스트 데이터를 토큰 단위로 분할하고, 각각의 토큰에 대한 ID로 변환하여 컴퓨터가 계산할 수 있도록 데이터의 형태를 변환하는 방법

나는 인공지능이 좋다

토큰화

나, 는, 인공지능, 이, 좋다

One-hot encoding

- 단어 집합의 크기를 벡터의 차원으로 하고, 표현하고 싶은 단어의 인덱스에 1의 값을 부여하고, 다른 인덱스에는 0을 부여하는 단어의 벡터 표현 방식

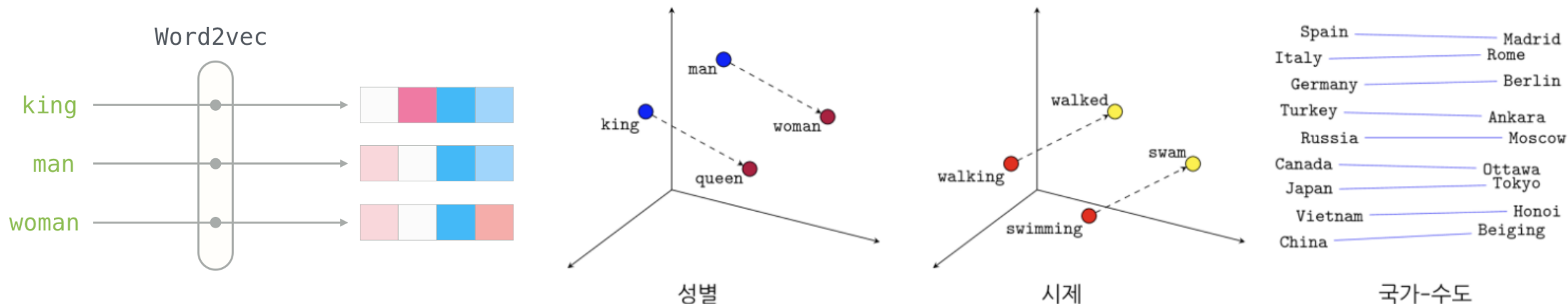
원 핫 인코딩

나	0	1,0,0,0,0
는	1	0,1,0,0,0
인공지능	2	0,0,1,0,0
이	3	0,0,0,1,0
좋다	4	0,0,0,0,1

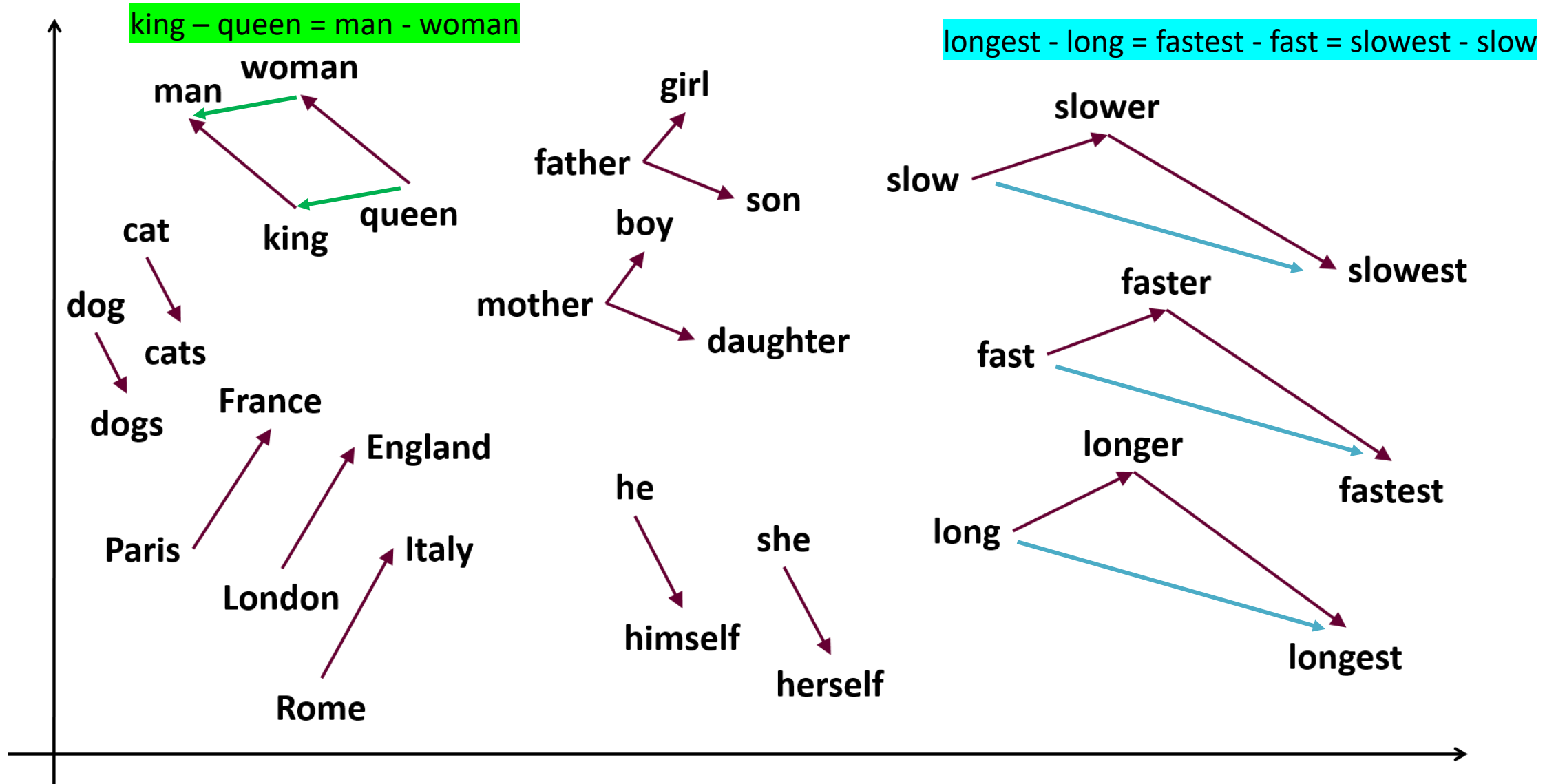
Embedding

임베딩: 단어나 문장을 벡터로 변환!

- One hot encoding 방식은 전체 단어(토큰)의 종류만큼 벡터의 차원이 한없이 커지지만, 오직 한 개의 단어의 값만 1이고 나머지는 모두 0이기 때문에 공간이 크게 낭비됨
- 0과 1로 이루어진 단어 개수 길이의 벡터를 특정한 고정된 길이의 벡터로 변환하여 표현하는 방식을 임베딩(Embedding)이라고 함
- Ex) Word2Vec: 입력된 단어를 고정된 길이의 벡터로 변환 -> 단어 사이의 벡터 계산(사칙연산 등) 가능



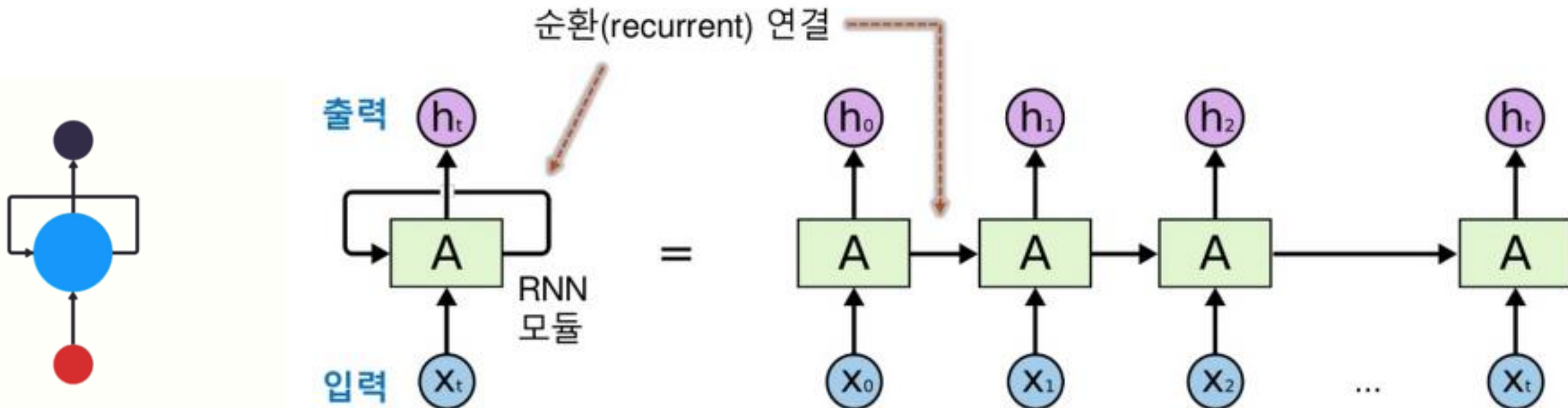
단어 임베딩 시각화 예시(Word2Vec)



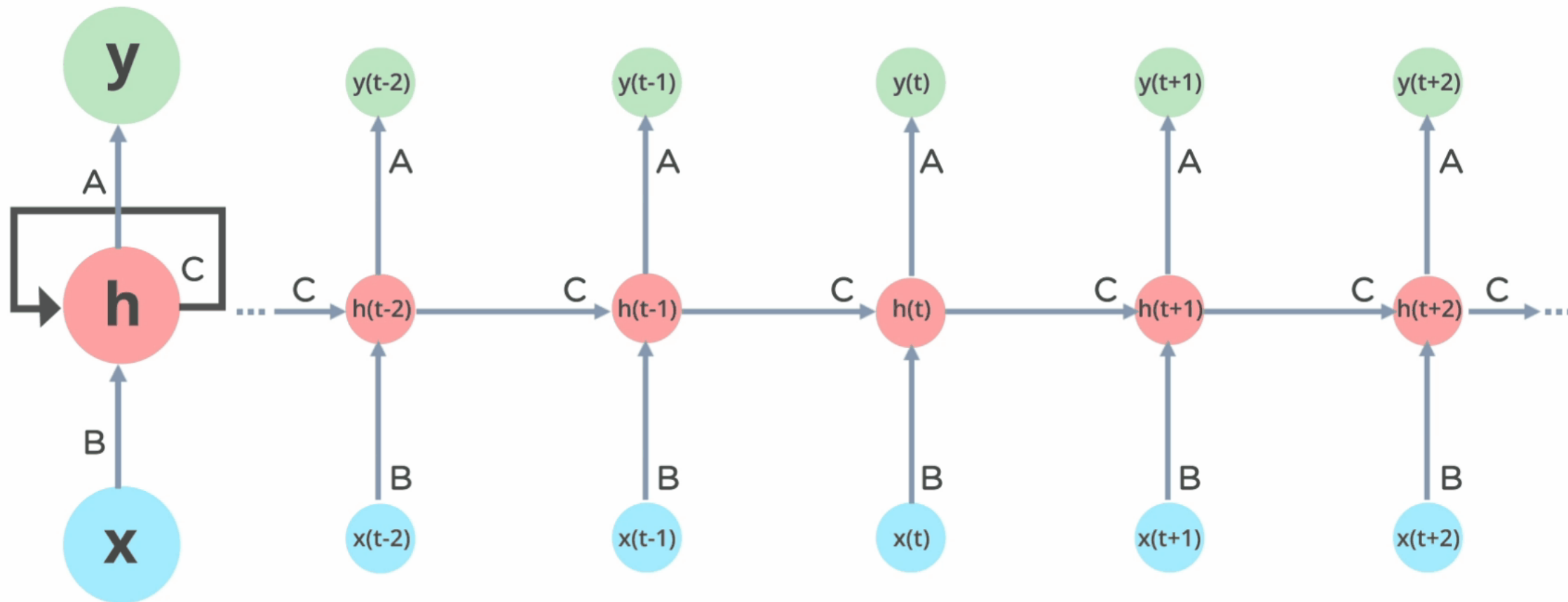
Recurrent Neural Network

입력과 출력을 sequence 기반으로 처리 가능한 딥 러닝 모델

- RNN은 입력 sequence의 각 값을 하나씩 받아 처리하는데, 이전 입력을 받으며 계산된 현재 상태 정보를 다음 입력을 받을 때 함께 고려하여 계산하는 구조
- 기존 방식과 다르게 문장에 등장하는 각 단어의 순서를 고려하여 문장을 표현할 수 있음



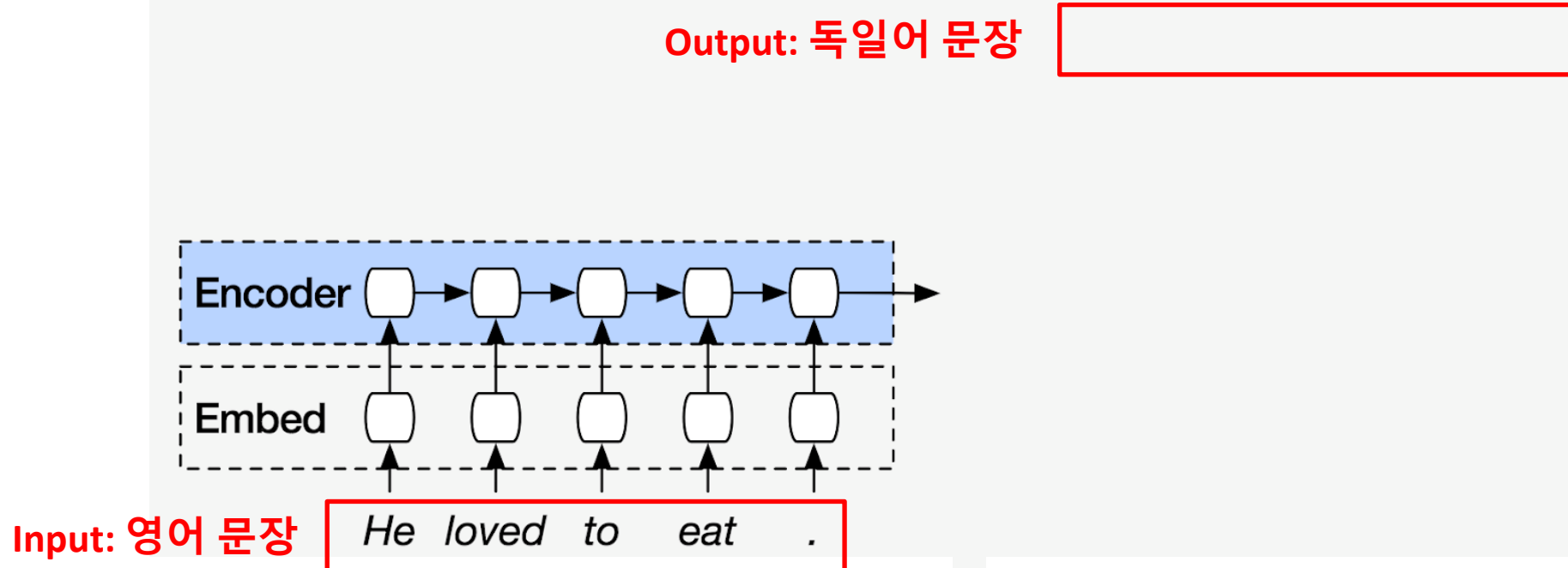
RNN의 작동 방식



Sequence-to-sequence

Seq2Seq: 입력을 문장으로 받아, 다시 문장을 출력할 수 있는 모델

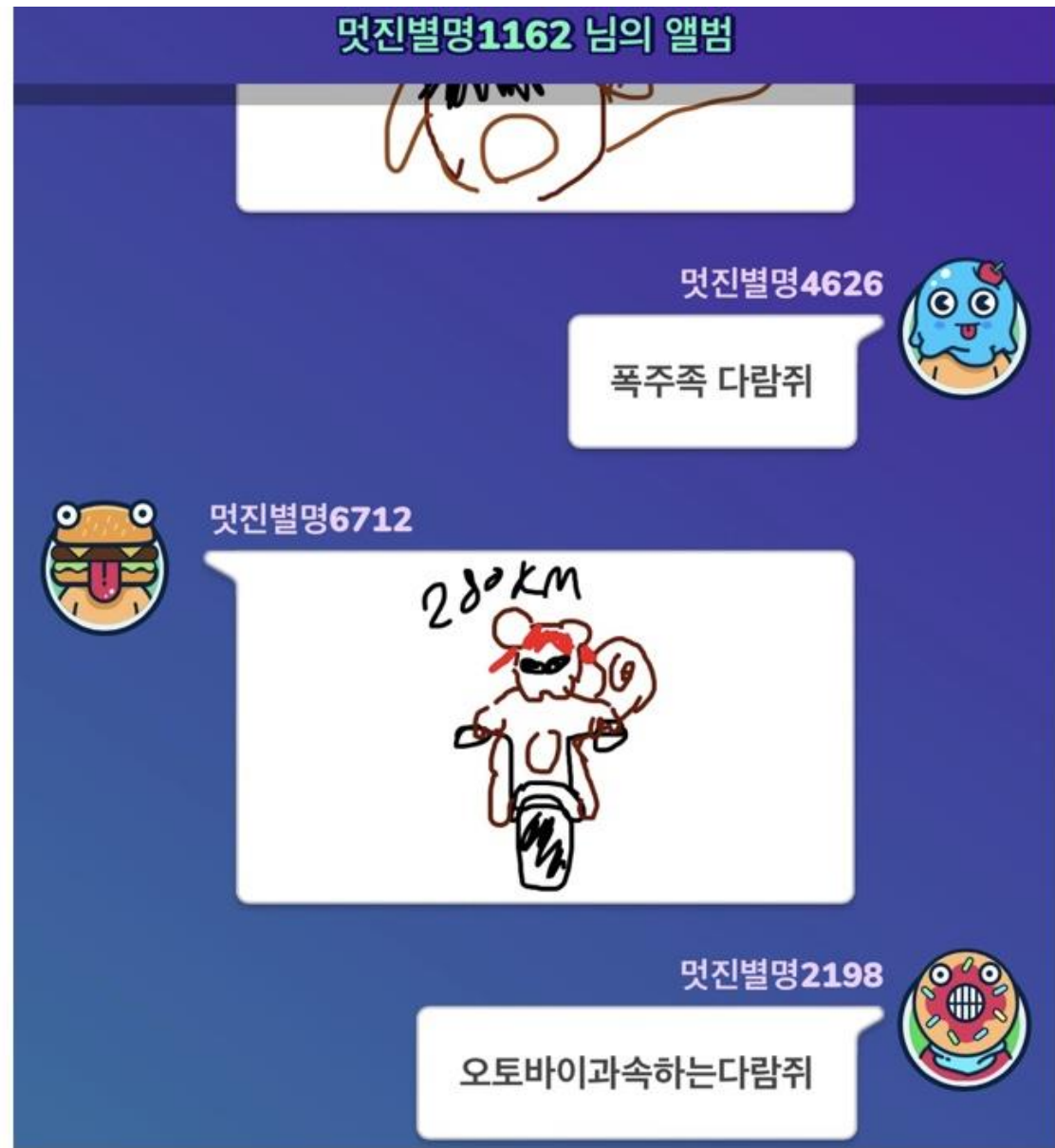
- 입력을 적절한 표현으로 변환하는 인코더(Encoder)와 변환된 정보를 바탕으로 새로운 데이터를 생성하는 디코더(Decoder)모델로 구성
- 기계 번역 등의 작업을 수행하기 위해 고안됨



Seq2Seq == 갈틱폰

갈틱폰 게임 방식에 비유해보자면...

- 출제자가 낸 문제를 그림으로 표현하는 과정
인코딩과 유사함
- 표현된 그림을 바탕으로 원래 문제를 찾아내는 과정
디코딩과 유사함



Recall: SOTA(State-Of-The-Art)

SOTA란?

특정 task를 수행하기 위한 여러 모델이나 방법론 중,
현재 가장 성능이 뛰어난 것!

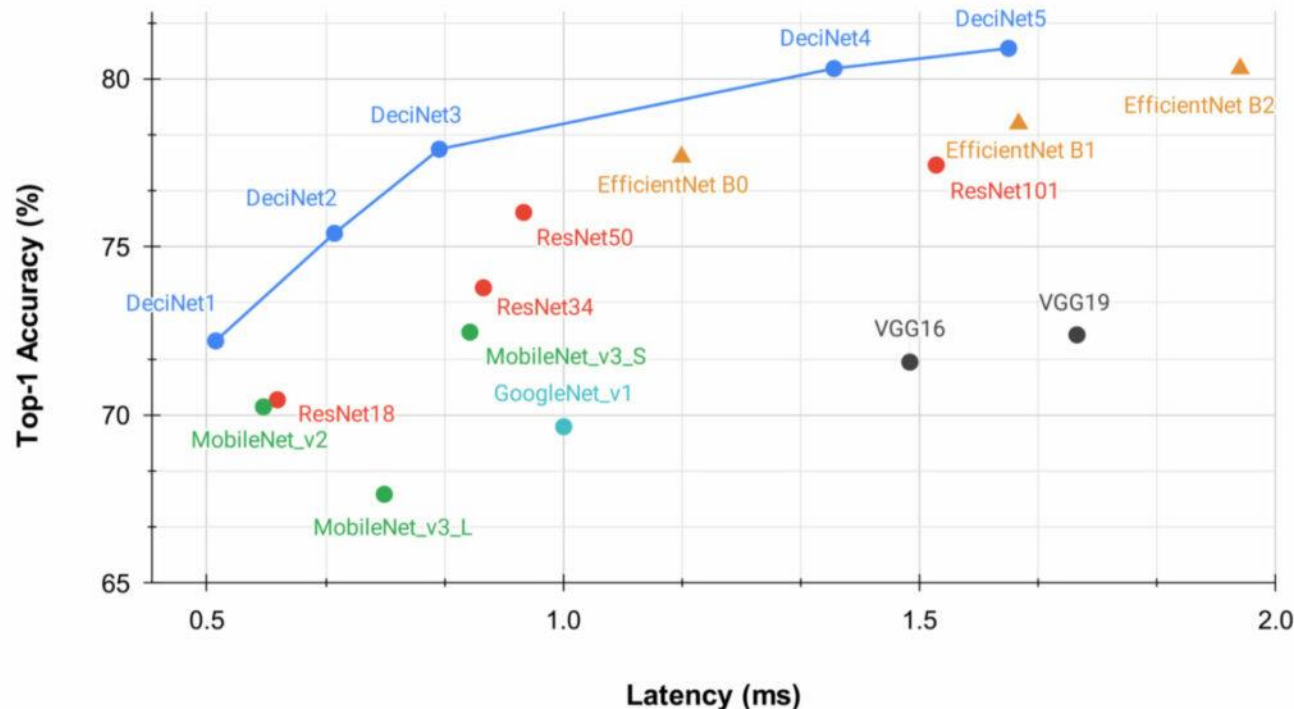
SOTA를 알아야 하는 이유

딥 러닝은 완성된 기술이 아니며, 끊임없이 변화하고,
연구자들과 엔지니어들이 늘 새로운 시도를 하기 때문!

가장 뛰어난 알고리즘도 하루아침에 구식이 되어버릴
수 있는 딥러닝 생태계에서, 최신 연구 결과와
기술들을 follow-up 하는 것이 중요하다!

참고: <https://paperswithcode.com/>

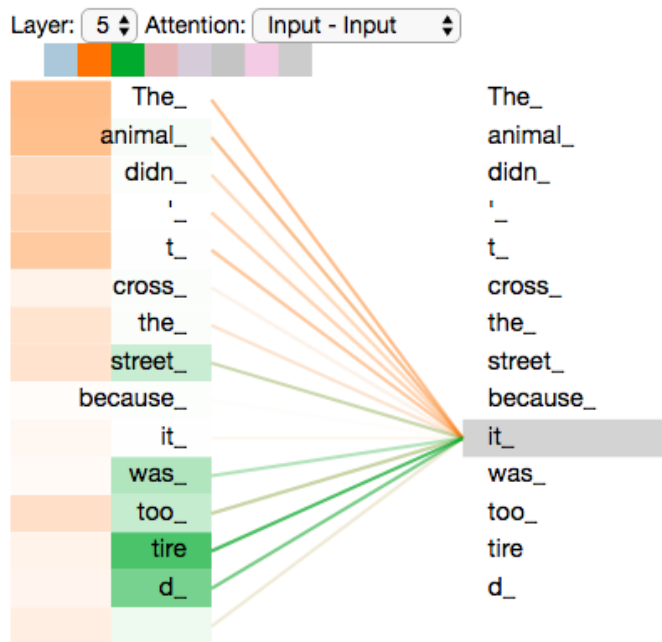
각종 task에서 SOTA를 달성한 ML 알고리즘의 논문,
소스코드 등을 정리한 사이트



Transformer

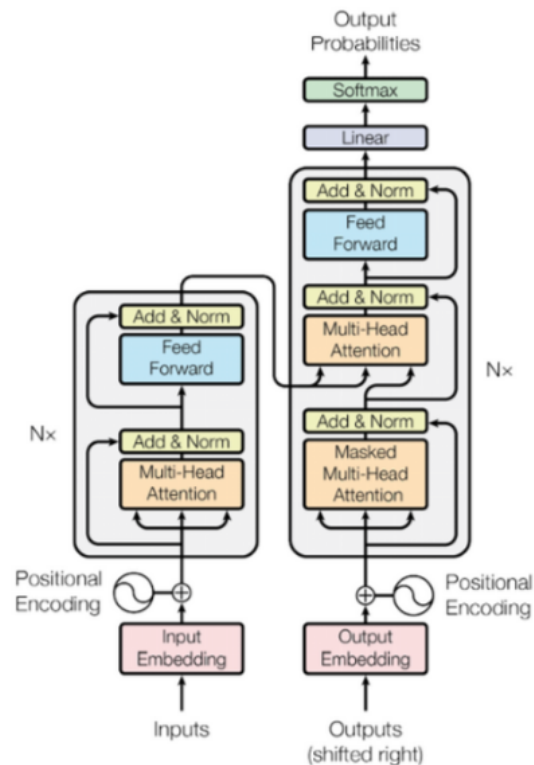
RNN의 단점을 보완한 현 SOTA 아키텍처

- RNN은 입력 sequence의 각 토큰을 하나씩 입력 받기 때문에 문장이 길어질수록 이전 단어에 대한 정보를 잊어버리게 되는 단점이 존재
- 2017년 공개된 논문 Attention is all you need에서 소개된 모델로, 입력을 순차적으로 처리하는 방식이 아닌, 한 번에 모든 단어를 참조할 수 있도록 하는 self attention 방식을 적용



Transformer

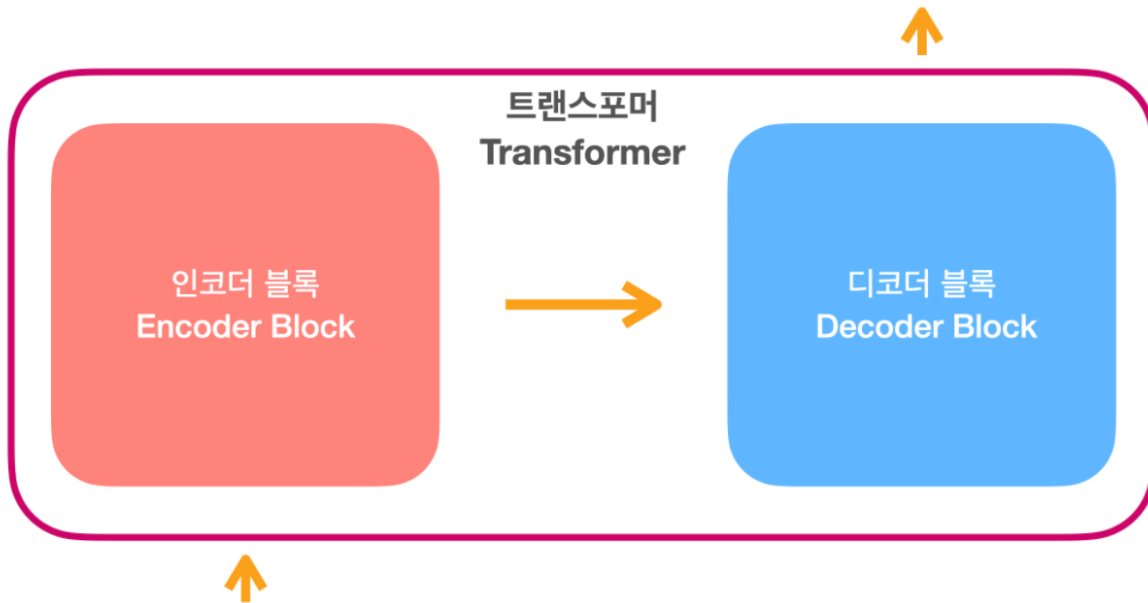
Attention Is All You Need



Transformer를 활용한 기계 번역 과정

1. 한국어 문장을 tokenization 등의 전처리를 거쳐 입력
2. 입력된 데이터는 트랜스포머의 인코더 블록이 컴퓨터가 이해할 수 있는 형태의 특정 크기의 벡터로 변환
3. 디코더 블록은 인코더가 변환한 데이터를 입력 받아 입력된 문장과 같은 의미를 가지도록 영어 토큰들을 출력
4. 출력된 토큰들의 인덱스를 다시 원래 형태로 변환 -> 영어로 번역된 문장

출력: The Golden Age which was longed by every age existed was now.



입력: 각 시대에서 갈망하는 황금시대는 현재였습니다.

Huggingface

- Transformer 기반 모델과 tokenizer를 비롯하여 NLP, CV, Voice 등 다양한 분야의 task를 수행할 수 있도록 하는 오픈 소스 라이브러리를 제공하는 스타트업
- Transformers 라이브러리를 활용하여 필요한 task를 수행할 수 있는 모델을 쉽게 불러올 수 있고, tokenizers 라이브러리를 통해 해당 모델을 사용하기 위한 전처리를 수행할 수 있음
- 다양한 아키텍처를 기반으로 task를 수행하도록 학습된 모델과 토큰나이저 그리고 학습을 위한 데이터셋을 업로드/다운로드할 수 있는 Huggingface hub을 운영하고 있음

• Hub

Host Git-based models, datasets and Spaces on the Hugging Face Hub.

• Transformers

State-of-the-art ML for Pytorch, TensorFlow, and JAX.

• Diffusers

State-of-the-art diffusion models for image and audio generation in PyTorch.

• Datasets

Access and share datasets for computer vision, audio, and NLP tasks.

• Gradio

Build machine learning demos and other web apps, in just a few lines of Python.

Transformers pipeline으로 NLP 맛보기

Pipeline이란?

- Text classification, generation, translation, question & answering 등 다양한 작업을 end-to-end로 실행할 수 있도록 제공하는 API

Tasks



Image Classification



Translation



Image Segmentation



Fill-Mask



Automatic Speech Recognition



Token Classification



Sentence Similarity



Audio Classification



Question Answering



Summarization



Zero-Shot Classification

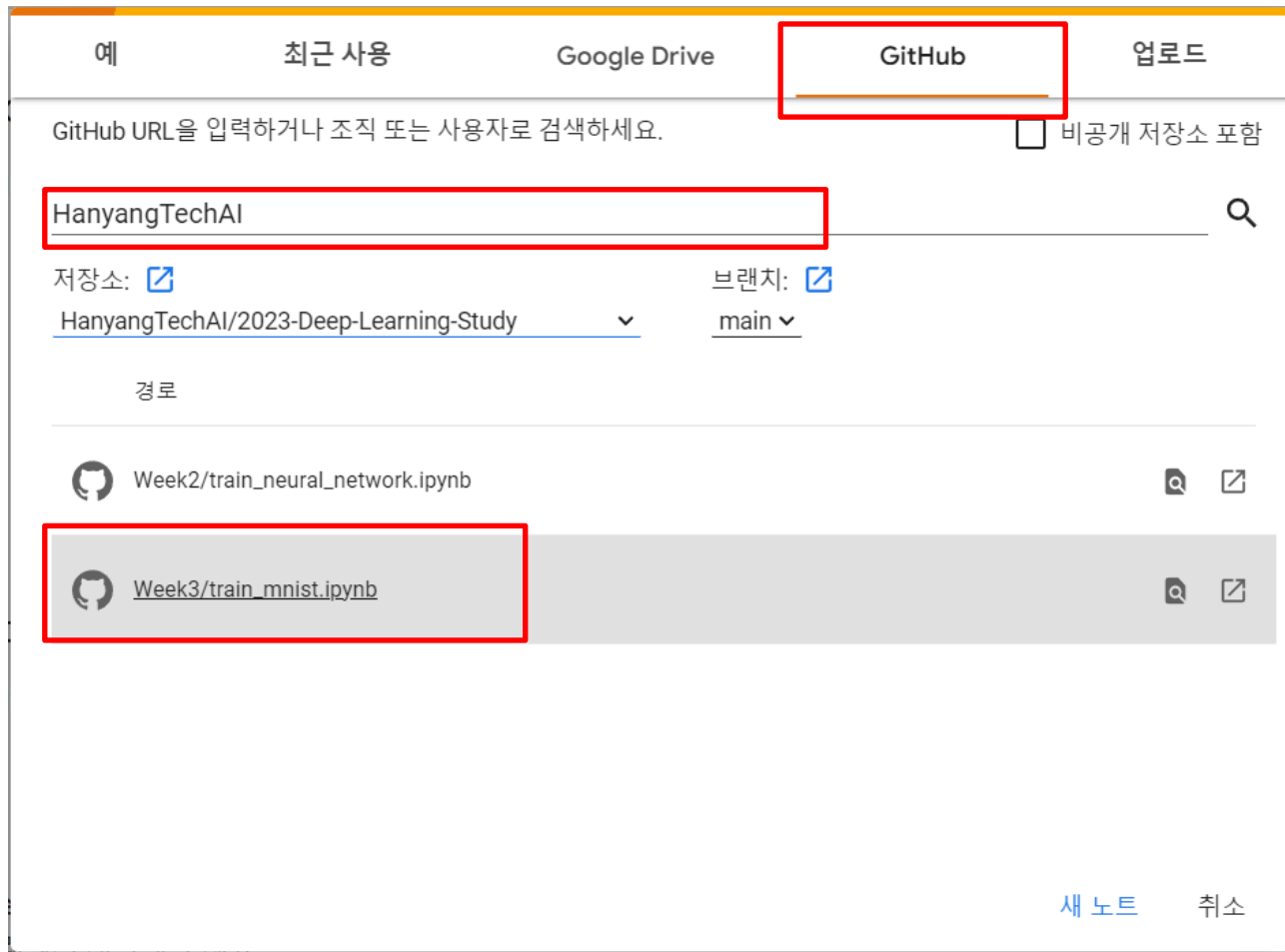
+ 16 Tasks



입력: 오늘은 참 날씨가 좋아.

출력: The weather is so nice today.

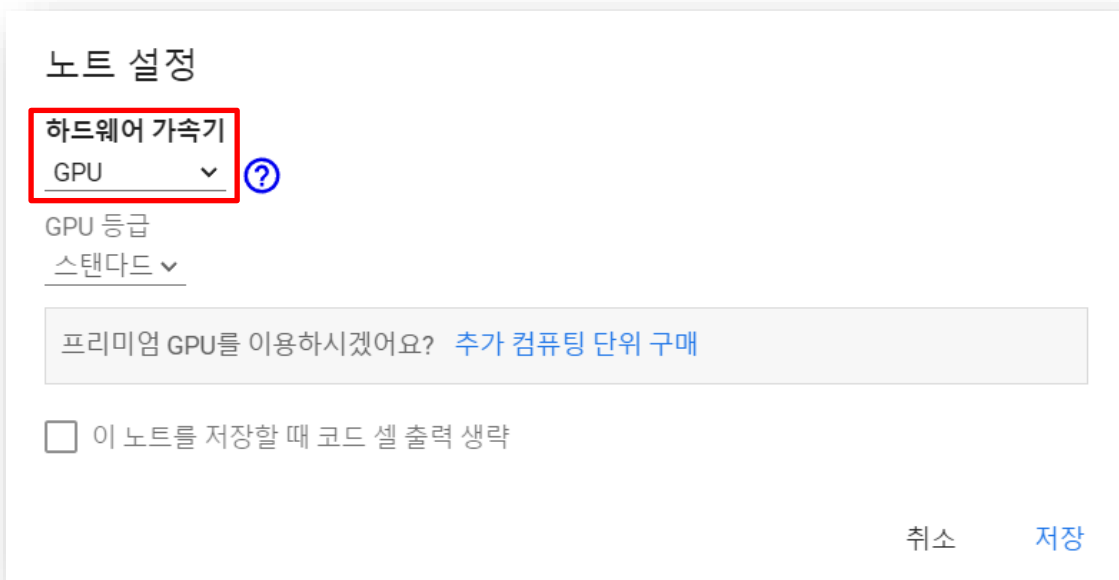
Colab에서 github에 업로드된 노트북 불러오기



Colab에서 GPU 가속 사용하기



The screenshot shows the Google Colab interface for a notebook named 'train_fashion_mnist.ipynb'. The 'Runtime' menu is open, displaying various execution options. The option '런타임 유형 변경' (Change runtime type) is highlighted with a red box. Other options include '모두 실행' (Run all), '이전 셀 실행' (Run previous cells), '초점이 맞춰진 셀 실행' (Run focused cells), '선택항목 실행' (Run selected cells), '이후 셀 실행' (Run subsequent cells), '실행 중단' (Interrupt), '런타임 다시 시작' (Restart runtime), '다시 시작 및 모두 실행' (Restart and run all), '런타임 연결 해제 및 삭제' (Disconnect and delete), '세션 관리' (Manage sessions), '리소스 보기' (View resources), and '런타임 로그 보기' (View logs).



The screenshot shows the '노트 설정' (Notebook Settings) dialog box. The '하드웨어 가속기' (Hardware accelerator) section is highlighted with a red box, showing 'GPU' selected from a dropdown menu. Below this, the 'GPU 등급' (GPU class) is set to '스탠다드' (Standard). A link for '추가 컴퓨팅 단위 구매' (Purchase additional computing units) is visible. At the bottom, there is a checkbox for '이 노트를 저장할 때 코드 셀 출력 생략' (Omit code cell output when saving this notebook) and buttons for '취소' (Cancel) and '저장' (Save).

다섯 번째 과제: Pipeline으로 다양한 NLP task 수행하기

Step 1. 예시 노트북을 실행해보며 transformers의 pipeline을 사용해보기

- 가장 기본적인 기계 번역을 위한 모델을 불러와서 영어로 된 입력이 한국어로 잘 번역되는지 확인해봅시다.

Step 2. Huggingface hub에서 다양한 모델을 찾아 pipeline 실행해보기

- Huggingface Transformers 라이브러리는 기계 번역 뿐만 아니라 문장 분류, 생성, 요약 등 다양한 NLP task를 수행할 수 있는 모델을 지원합니다.
- Huggingface model hub(<https://huggingface.co/models>)에서 원하는 task를 수행할 수 있는 모델의 이름을 잘 찾아서, 해당 모델이 수행할 수 있는 task pipeline을 불러와 실행해 봅시다!
- 노트북(https://colab.research.google.com/github/HanyangTechAI/2023-Deep-Learning-Study/blob/main/Week6/huggingface_pipelines.ipynb)을 바탕으로 조별로 자유롭게 토의해서, 결과를 Github 이슈(<https://github.com/HanyangTechAI/2023-Deep-Learning-Study/issues/6>)에 업로드해주세요!

With HAI, Fly High

Hanyang Artificial
Intelligence