AI Newsletter (No. 4)

연구관리처 AI연구센터 / (2023년 6월 8일)

K-water연구원 AI연구센터에서 정기적으로 발간하는 뉴스레터입니다.
AI 뉴스, Hands-on 프로젝트, 팁 등을 다양한 내용과 난이도로 담았습니다.

※ 코드 및 뉴스 등 외부링크가 다수 포함되어 있으므로 **인터넷 환경 PC 권장**

>> Hello, world!

#AI연구센터 #Hello, K-water #직접 돌려보기 #AI 2학년 3반

□ 안녕하세요. K-water연구원 연구관리처 AI연구센터입니다.

□ 최근까지도 AI 분야에서는 여전히 ChatGPT을 중심으로 한 거대언어모델이 화두입니다. 그런데, 이탈리아가 최초로 국가단위에서 ChatGPT를 금지하였으며, 국내에서도 언어모델 활용 관련 지침들이 생겨나고 있습니다. 생성AI 기술의 윤리 문제와 보안 이슈에 대해서는 지속적으로 눈여겨봐야 할 것 같습니다.

□ Hands-on 프로젝트 파트에서는 코딩 부담을 덜어주는 로우코드(low-code) 도구 중 대표적인 파이캐럿(pycaret) 패키지 사용방법과 유튜브 영상을 텍스트로 요약하는 프로젝트를 준비하였습니다.

□ 5월에는 한국수자원학회 학술대회를 포함, 3건의 학술대회에서 논문발표 및 AI 워크샵을 주관하였으며, 매월 AI 활용 연구성과 및 정보 공유를 위한 세미나와 심포지엄을 운영하고 있습니다.

모든 실습 예제와 링크는 “AI연구센터” 누리집(Homepage, GitHub)에 게시되어 활용가능합니다.

CONTENTS


```
> print(f“{YOUR_NAME}, Please Have FUN\n :D”);
```

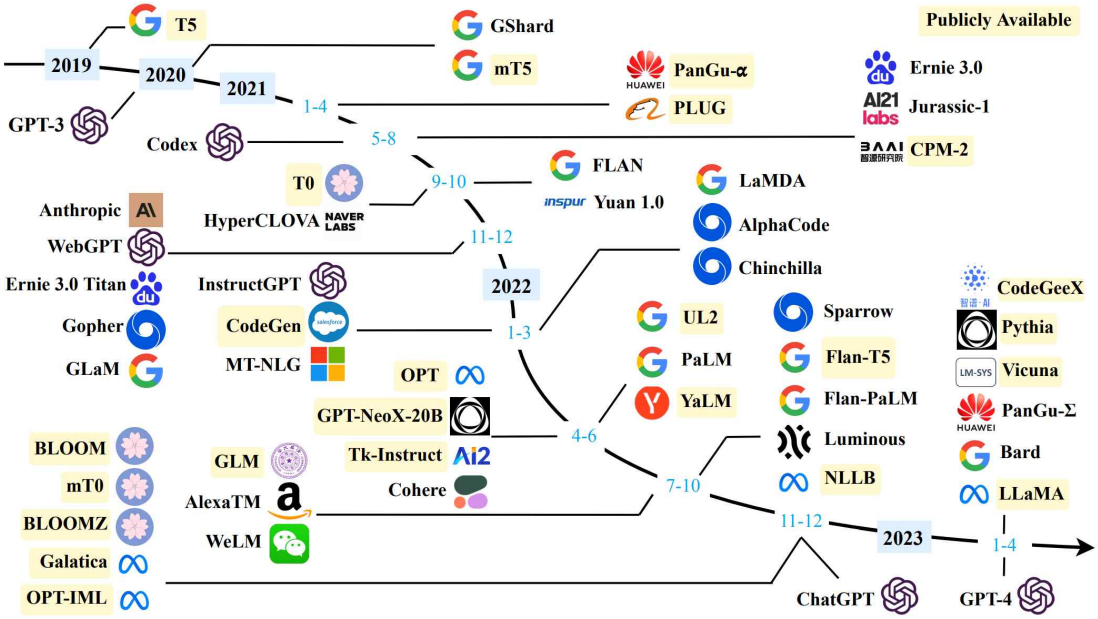
1. What's NEW in AI

#GPT #딥마인드 #러-우전쟁 #Quillbot

□ (논문) GPT와 아이들

○ OpenAI의 ChatGPT가 화제를 일으키며 수많은 언어모델들이 ~~GPT라는 이름을 달고 세상에 쏟아지고 있습니다.

○ 현재 거대언어모델은 사용방법에 있어 크게 오픈소스인지 아닌지로 나눌 수 있습니다. 아래는 최근 발표된 논문*에서 정리한 현존하는 거대언어모델(100억개 이상 매개변수)을 발표시기에 따라 정리한 타임라인입니다.



* (2023) "A Survey of Large Language Models", https://arxiv.org/abs/2303.18223

○ 여기에 각 모델별로 조정(tuning)을 통해 만들어진 모델까지 더하면 셀 수 없이 많은 언어모델들이 경쟁하고 있는 구도입니다.

- MiniGPT, BloombergGPT, BioGPT, HuggingGPT, Code GPT, AutoGPT 등등의 특정 도메인에 특화되거나, 프롬프팅(prompting)을 도와주는 각종 수많은 언어모델이 있습니다.
- 다만 주의하셔야 할 점은 GPT라는 이름이 붙어있다고 해서 꼭 OpenAI의 GPT모델을 사용하는 것은 아닙니다. GPT-3 이후의 모델은 공개되어 있지 않기 때문에 오히려 구글의 T5나 메타의 LLaMA 모델을 기반으로 만들어진 모델이 더 많습니다.

□ (뉴스) Google의 AI 연구팀과 딥마인드 병합

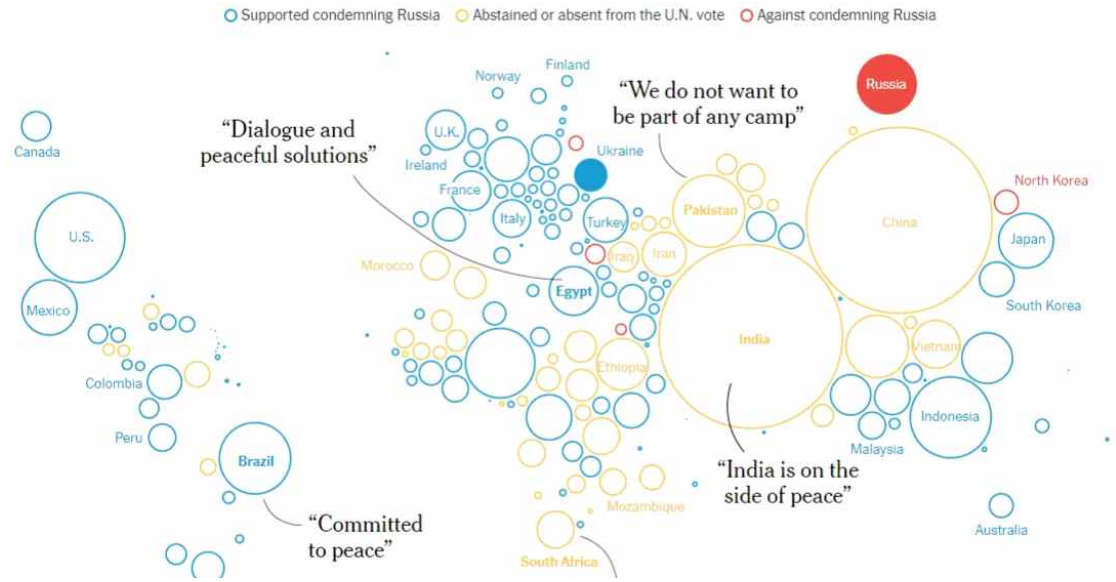
○ 지난 4월 21일 구글의 모회사인 알파벳의 CEO 순다르 피차이는 구글 리서치 브레인 팀과 딥마인드를 합칠 것이라고 발표했습니다.

○ AI 서비스 영역에서 GPT를 필두로 한 MS와 OpenAI사에 구글이 주도권을 빼앗기지 않으려는 움직임으로 여겨지고 있습니다.

□ (시각화) 러시아-우크라이나 전쟁

○ 141개국이 러시아의 무조건적인 철수를 촉구하는 유엔 결의안을 지지하면서 전 세계가 러시아에 대항하는 강력한 글로벌 연합을 형성했습니다.

○ 뉴욕타임스는 각국의 반응을 버블차트 지도를 통해 이를 시각화하였습니다.



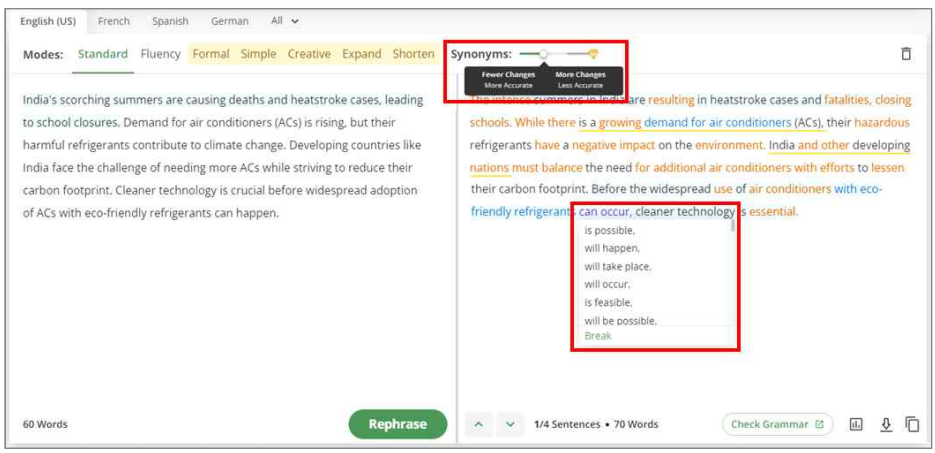
※https://www.nytimes.com/interactive/2023/02/23/world/russia-ukraine-geopolitics.html

□ (어플리케이션) 외국어 작문 도구 Quillbot

○ 외국어 작문 시 ChatGPT와 같이 사용하면 좋은 도구를 소개드립니다. Quillbot은 인공지능을 활용한 작문 도구로, 문장 재구성(paraphrasing), 문법 확인(Grammar Checker), 표절 검사(Plagiarism Checker), AI와 동시작업(Co-worker), 요약(Summarizer), 다양한 형태의 인용문 생성(Citation Generator) 등이 가능합니다.

○ 의역에는 영어, 프랑스어, 중국어, 일본어 등 23개 언어를 지원하며, 영어의 경우 동의어·유의어 추천, 모드 선택(Standard, Fluency, Formal 등)이 가능합니다.

○ 요약은 ChatGPT가 잘하지만, 의역의 경우 Quillbot의 성능이 좋으니 이 둘을 적절히 사용하면 외국어 작문 시 도움이 됩니다.



2. "10줄 코딩" Hands-on AI Project

#로우코드 #Pycaret #회귀 #시계열 #유튜브요약

□ 로우-코드(Low-Code) 머신러닝 (초급)

※Link: https://colab.research.google.com/drive/1tsU2UtZLWfg6AoX6TJfWKrkxB7JJLS6w?usp=sharing
※가독성을 위해 일부 세부적인 라인은 생략되어 있습니다. 전체 코드는 위 링크를 참고해주세요.

○ Python이 아무리 쉬운 언어라고 하지만, 입문자 입장에서는 여전히 어려운 부분들이 있습니다. 최근에는 AI 모델을 만들기 위해 복잡하게 코딩할 필요가 없는 소위 로우(low)코드, 노(no)코드 도구들이 많아지고 있습니다.

○ Pycaret은 기계학습을 쉽게 수행할 수 있는 로우코드 라이브러리입니다.

```
!pip install pycaret
```

○ 데이터 준비

- 대표적으로 회귀분석을 연습해볼 수 있는 미국 건강보험 공개 데이터셋이 있습니다.
- 입력자료로 나이, 성별, BMI, 자녀의 수, 흡연여부, 거주지역을 사용합니다.
- 목표값은 건강보험으로 청구된 의료비(charges)입니다.

```
from pycaret.datasets import get_data
data = get_data('insurace')
```

```
data.info()
```

```
 0   age       1338 non-null   int64
 1   sex       1338 non-null   object
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
 5   region    1338 non-null   object
 6   charges   1338 non-null   float64
```

○ 데이터 셋업

- 탐색적 데이터 분석을 통해, 사전에 변수의 특성을 이해하고, 적절하게 입력자료를 구성해야 하지만, pycaret에 입력하는 것만으로도 이러한 과정을 대부분 해결할 수 있습니다.

```
from pycaret.regression import *
data_setup = setup(data, target = 'charges', session_id = 101)
```

Index	Description	Value	Index	Description	Value
0	Session id	101	7	Ordinal features	2
1	Target	charges	8	Numeric features	3
2	Target type	Regression	9	Categorical features	3
3	Orig. data shape	(1338, 7)	10	Preprocess	True
4	Tran. data shape	(1338, 10)	11	Imputation type	simple
5	Tran. train set shape	(936, 10)	12	Numeric imputation	mean
6	Tran. test set shape	(402, 10)	...	...	...

- setup 함수에 데이터와 목표값만 지정해주었는데도, 알아서 Train/Test 세트를 나누어주고(Index 5~6) 변수의 특성을 파악(Index 7~9)하여 변환(Index 3~4)해줍니다.
- 나이와 자녀의 수는 ordinal 특성으로, 성별, 흡연여부, 거주지역은 categorical 특성으로 처리되었으며, 거주지역은 4개의 지역이 있기 때문에 전체 특성이 늘었습니다.(7개→10개)

○ 기계학습 구현

- 이어서 아래의 한줄만 입력해주면 분석이 끝납니다.

```
best = compare_models()
```

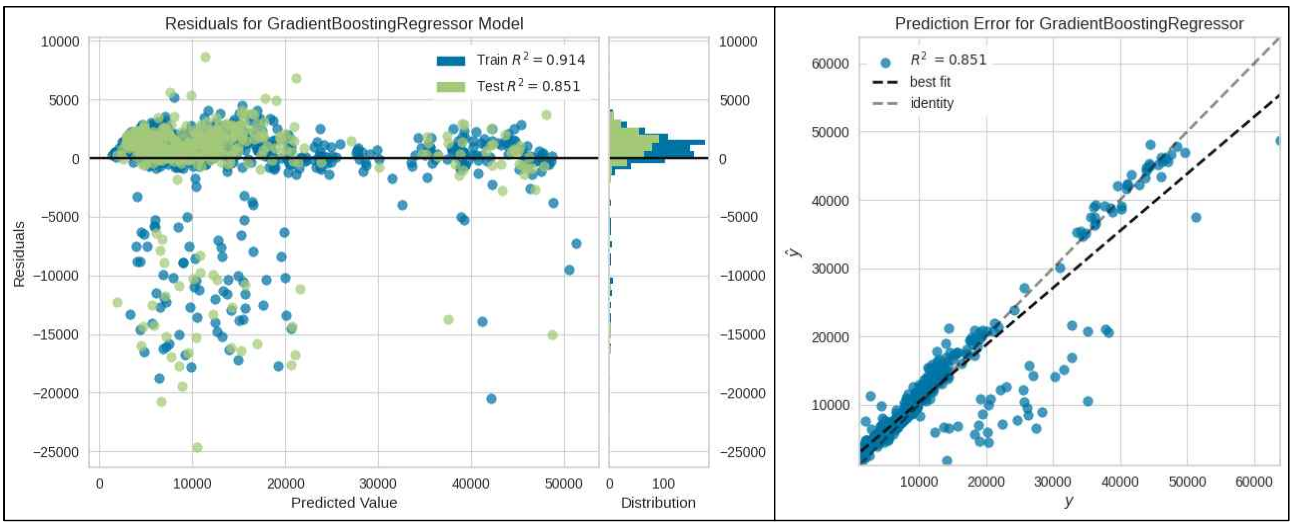
Model		MAE	MSE	RMSE	R^2	...
gbr	Gradient Boosting Regressor	2538.40	21363018	4600.24	0.8528	
rf	Random Forest Regressor	2598.08	22826737	4751.05	0.8439	
lightgbm	Light Gradient Boosting Machine	2809.73	23638463	4843.13	0.8369	
et	Extra Trees Regressor	2565.01	25003638	4987.51	0.8287	
ada	AdaBoost Regressor	3955.97	26874547	5171.00	0.8133	
xgboost	Extreme Gradient Boosting	2917.95	27073283	5193.53	0.8128	
ridge	Ridge Regression	4406.00	38729451	6199.84	0.7334	
br	Bayesian Ridge	4401.87	38727956	6199.73	0.7333	
llar	Lasso Least Angle Regression	4396.45	38724071	6199.43	0.7333	
lar	Least Angle Regression	4396.53	38726185	6199.62	0.7333	
lasso	Lasso Regression	4396.45	38724068	6199.43	0.7333	
lr	Linear Regression	4396.53	38726185	6199.62	0.7333	
...	...	...	...	...	...	...

- MAE, RMSE, R^2와 같은 자주 사용되는 성능지표를 통해 19개의 기계학습 모델의 성능을 비교하고 정렬해줍니다. Gradient Boosting 모델이 가장 좋은 것으로 보이네요.

○ 베스트 모델 검토

- pycaret에 내장되어있는 plot_model 함수를 활용하여 다양하게 시각화할 수 있습니다.
- residual plot을 통해 훈련자료와 테스트 자료의 오차와 실제값과 예측값을 비교해봅니다.

```
plot_model(best, plot='residuals')
plot_model(best, plot='error')
```



○ 신규 데이터 예측

- 'compare_model'을 통해 'best' 변수에는 가장 성능이 좋은 모델이 저장되어 있습니다.

```
predict_model(best, data=new_data)
```

□ 유튜브 요약기 (중급)

※Link: https://colab.research.google.com/drive/12k1yn1y5dRNgEX4KYsc0f-5VcM1up6ng?usp=sharing
※가독성을 위해 일부 세부적인 라인은 생략되어 있습니다. 전체 코드는 위 링크를 참고해주세요.

○ 어떤 정보를 찾다보면 유튜브로만 검색될 경우가 많습니다. 전체적 정보만 빠르게 확인하고 싶을 때는 텍스트로도 정리되어 있으면 합니다.

○ 이를 위해 유튜브의 주소만 입력하면, 영상을 요약하는 프로젝트입니다.

- Colab 환경에서는 두가지 라이브러리만 추가로 설치해주면 됩니다. 유튜브의 자막을 가져오는 API를 활용하는 라이브러리와, Transformers 언어모델을 활용하는 라이브러리입니다.

```
!pip install youtube_transcript_api
!pip install transformers
```

○ 먼저 유튜브 주소를 입력하면 자막 정보를 가져오는 함수를 작성해봅니다.

```
def get_transcript(url, lang='ko'):

    # url을 입력받아 video ID를 추출
    url_data = urlparse(url)
    video_id = parse_qs(url_data.query)["v"][0]

    # 만약 주소가 올바르지 않다면 Video ID가 없다는 메시지를 출력
    if not video_id:
        print('Video ID not found.')
        return None

    try:
        formatter = TextFormatter()

        # 추출한 video ID를 활용하여 앞서 설치한 유튜브 자막 API 호출
        # 기본 언어는 '한국어'로 설정, 다른 언어도 사용 가능(eg. 'en')
        transcript = YouTubeTranscriptApi.get_transcript(video_id, languages=[lang])
        text = formatter.format_transcript(transcript)
        text = re.sub('\s+', ' ', text).replace('--', '')
        return video_id, text

    # 자막을 받아오는 도중 에러가 발생하면 아래와 같이 오류 메시지를 출력
    except Exception as e:
        print('Error downloading transcript:', e)
        return None
```

○ 텍스트를 요약하는 함수를 작성해줍니다.

```
def summarize_youtube(text):
    prefix = "summarize: "
    inputs = [prefix + text]

    # 입력 텍스트의 최대 토큰은 4,096개, 출력 텍스트는 512개(최소 10개)로 설정
    inputs = tokenizer(inputs, max_length=4096, truncation=True, return_tensors="pt")
    output = model.generate(**inputs, num_beams=3, do_sample=True, min_length=10, max_length=512)
    decoded_output = tokenizer.batch_decode(output, skip_special_tokens=True)[0]
    result = nltk.sent_tokenize(decoded_output.strip())[0]

    return result
```

○ 모든 사전작업이 끝났습니다. K-water 유튜브 채널의 가장 최근 영상으로 들어가서 url를 복사하고 아래와 같이 두 개의 함수를 실행시킵니다.

```
url = 'https://www.youtube.com/watch?v=AHa9Ls19O2I'

video_id, text = get_transcript(url)
result = summarize_youtube(text)
```

○ 해당 영상은 '[수(水)타벅스가 간다🚚] 강원지역협력단 편☕'입니다.

○ 자동으로 생성된 자막을 활용하다보니 조금씩은 오타가 보이지만, 아래의 요약된 결과를 확인해보면 나름 잘 된 것 같습니다.

text # 자막 원문

[음악] 안녕하십니까 강원지역 협력단 지역협력부 하선혜 대리입니다 저희 강원지역 협력단은 현대화 사업소 특성상 거리가 떨어진 곳들에 많이 위치하다 보니 다 같이 모일 수 있는 기회가 많이 없습니다 하지만 이번 농촌 일손도끼 봉사활동에서는 각 센터와 사업소들에서 모두 모이는 그런 뜻깊은 자리였는데요 이런 자리에 스타벅스 커피차가 함께해 준다면 더 좋은 추억이 될 것 같아 신청하게 되었습니다 [음악] 강원지역 협력단은 강원도 18개 시군의 거버넌스 역할과 함께 지방산수도 현대화 사업 swm 사업 등 국책사업을 수행하고 있는 조직입니다 특히 올해에는 강릉에서 발생한 대형 산불을 지원하기 위하여 성근 기부 및 병물 등을 지원한 바 있습니다 강원지역 협력단은 강원도 18개 지자체 중 10개의 지자체와 위스탁 협약을 맺어 사업을 수행하고 있으며 올해 사업 목표 이수율 85% 달성이라는 미션을 수행하기 위해 전 직원이 한마음으로 힘을 다하고 있습니다 올해 강원지역 협력단은 7개 현대화 사업소에 성과보증이 게시되는 중요한 시점입니다 성과보증은 한국 상하수도협회에 주관하여 1년 동안 유술 85% 이상 유지를 하는 것인데요 영서 지역은 인재를 시작으로 철원 원주 이렇게 세계사업장 영동지역은 양양을 시작으로 속초 동해 삼척 4개 사업장이 해당됩니다 그리고 우리 협력단의 새로운 사업장이 생겼습니다 작년에 위스탁 협약을 체결한 삼척시 노후 상수관로 정비사업인데요 탄광 지역과 해양건강재라는 어려운 사업대상지지만 그동안 쌓인 현대아사업에 노하우로이 사업도 순항하였음 합니다 그 중에서 핵심적인 강점 두 가지만 말씀드리도록 하겠습니다 첫 번째는 강원지역 협력단에 뛰어난 단합력인데요 협력단 내 어려운 상황이 발생하면 모두가 나서서 돕고자 하는 최고의 협동심을 잡을 수 있습니다 두 번째는 타고난 끼입니다 친숙하고 정확한 업무 수행은 물론이고 그 외에 다양한 취미 활동까지 섭렵하는 다양한 끼와 재능을 가진 저의 협력단의 직원들을 보실 수 있습니다 다방면에서 최고의 능력을 보여주시는 강원제약 협력단의 모든 선배 동료들이 최고의 자랑이십니다 [음악] 이런 날씨에 좋은 음료를 제공해 주셔서 감사했습니다 힘들지만 오늘 너무 뜻깊은 하루였던 것 같습니다 잘 마실게요 오늘 보람이 섰습니다 커피 잘 먹었습니다 아이 신나라 중요한 것은 강원지역 협력단 파이팅 [음악]

result # 요약정보 (실행할 때 마다 조금씩 달라질 수 있음)

강원지역 협력단은 강원도 18개 시군의 거버넌스 역할과 함께 국책사업을 수행하고 있는 조직으로 강원도 18개 지자체 중 10개의 지자체와 위스탁 협약을 맺어 사업을 수행하고 있으며 사업 목표 이수율 85 달성이라는 미션을 수행하기 위해 전 직원이 힘을 다하고 있다.

3. TIPs

☐ set

○ 리스트에 저장되어 있는 변수들 중 중복이 있는지 궁금한 때가 있습니다.

○ set 함수는 중복된 원소들을 제거하기 때문에 이를 활용할 수 있습니다.

```
def all_unique(lst):
    return len(lst) == len(set(lst))

x = [1, 1, 2, 2, 3, 2, 3, 4, 5, 6]
y = [1, 2, 3, 4, 5]

all_unique(x) # False
all_unique(y) # True
```

☐ Wifi QR Code 만들기

○ 카페 같은 곳에서 와이파이 비밀번호를 복잡하게 종이에 적어두면 받아적는 것이 여간 귀찮은 일이 아닙니다. QR Code로 간단하게 해결해봅니다.

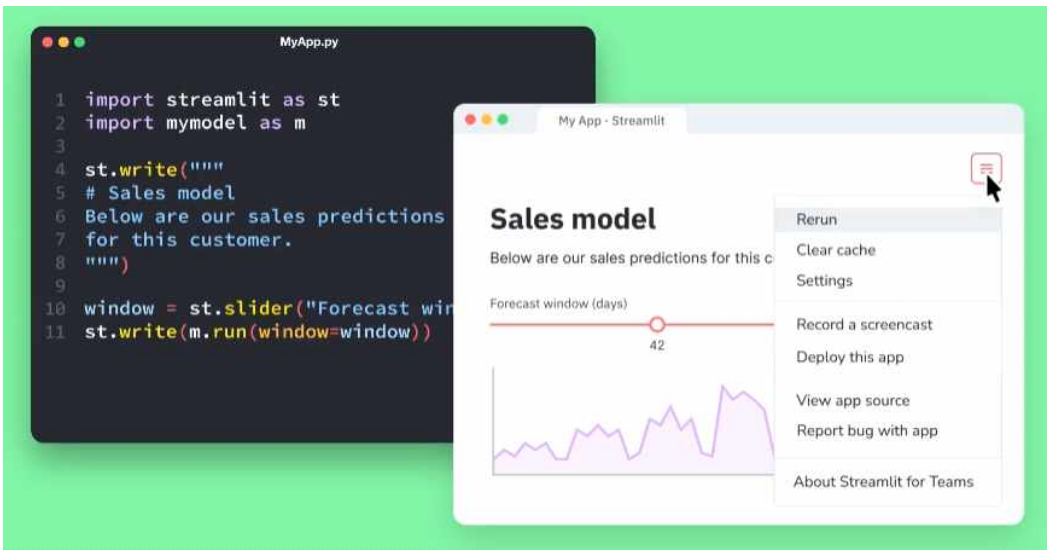
```
!pip install wifi-qrcode-generator
import wifi_qrcode_generator as qr
qr.wifi_qrcode('YOUR_WIFI_AP_NAME', False, 'WPA', 'WIFI_PASSWORD')
```

☐ Streamlit

○ Streamlit은 Python을 통해 쉽게 웹페이지를 만들 수 있는 도구입니다.

○ Python에서 분석한 내용을 빠르게 프로토타입으로 만들 수 있으며, 머신러닝 및 딥러닝 관련 모듈 등을 interactive하게 구현, 배포할 수 있습니다.

○ 실례로 K-water에서 운영하고 있는 '국가상수도정보시스템'의 동파위험정보 서비스는 AI 연구센터에서 Streamlit으로 프로토타입을 만들고, 의견수렴을 거쳐 실제 서비스로 이어진 좋은 예시입니다.



☐ pandas.groupby()

○ 엑셀에서는 간단하게 할 수 있는 필터나 피벗테이블 같은 작업을 python의 pandas 데이터프레임으로 데이터를 옮겨놓고 나면 어려울 때가 많습니다.

○ pandas의 groupby 메소드를 이용하면 이런 작업들을 수행할 수 있습니다.

* https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.groupby.html

```
# 기본적인 사용 방법
dataframe.groupby(분리기준 column).agg({분석할 column: 통계량}, ...)
```

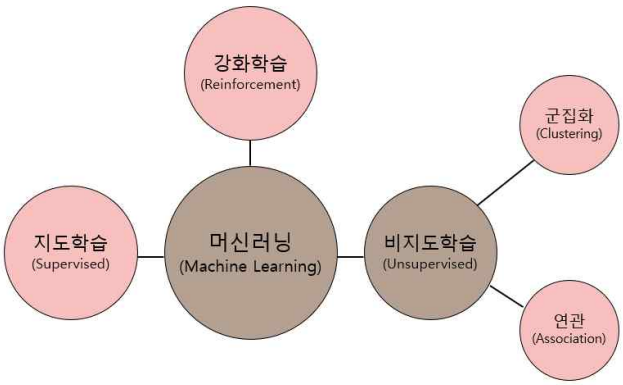
○ groupby 안에 컬럼을 입력하고 해당 컬럼의 값별로 통계량을 산출합니다.

```
dataframe.groupby('column').sum()    # 합계
dataframe.groupby('column').var()    # 분산
dataframe.groupby('column').count()  # 데이터 수
```

○ 엑셀의 피벗테이블처럼 여러 열을 계층적으로 분리할 수도 있습니다.

```
dataframe.groupby(['column1', 'column2']).sum() # column1, column2 계층적 분리 및 합계
```

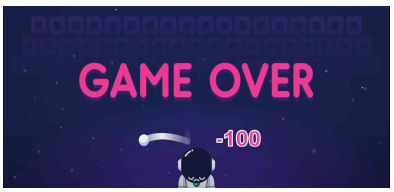
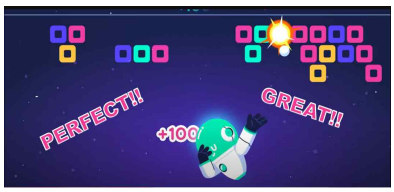
☐ 강화학습 (Reinforcement Learning)



○ 드디어 머신러닝의 끝에 다다랐습니다! 이번 호는 강화학습에 대해 알아보겠습니다.

○ 강화학습은 인공지능이 시행착오를 통해 스스로 학습하는 방법입니다.

○ 아래 그림을 참고해 게임을 예로 들어 보겠습니다. 게임을 못 하면 벌칙을 통해 감점하고 잘하면 보상으로 점수를 준다면, 인공지능은 성공과 실패의 반복을 통해 더 많은 보상을 얻는 방향으로 행동하게 됩니다.

		
못하면 감점	잘하면 보상	**=> 더 많은 보상을 찾아서!**

○ 이렇게 보상과 벌칙만 잘 정해두면 인공지능은 스스로 주어진 환경과 상호작용하며 가장 높은 보상을 받을 수 있도록 학습합니다.

○ 이런 강화학습의 예로는 딥마인드의 알파고 및 알파시리즈, 요즘 핫한 자율주행, 로봇의 행동 알고리즘 훈련방법 등이 있습니다.

4. Meanwhile, in K-water AI Lab.

□ 한국수자원학회 참석 (강원도 고성, 5/25~26)

○ 한국수자원학회 AI응용연구분과와 AI연구센터에서는 'AI 기술 융합을 통한 물관리 혁신 방안'이라는 주제로 기획세션을 개최하였습니다.

- 1부 순서로는 '물관리 기술과 Generative AI 기술의 융합'을 주제로 한국에너지기술연구원의 이제현 박사님, 국민대학교 신주영 교수님, AI연구센터 김성훈 센터장님의 발표와 패널 토론으로 진행되었습니다.
- 2부 순서로는 현재 AI 연구센터에서 적용성 검토 중인 AWS(Amazon Web Service) 클라우드 컴퓨팅을 활용하여 수자원 관련 실무 워크샵을 진행했습니다. 클라우드 상에 있는 고성능 컴퓨터 자원을 활용하여 NeRF, 위성자료, 장기시계열예측 등의 실습을 진행했습니다.

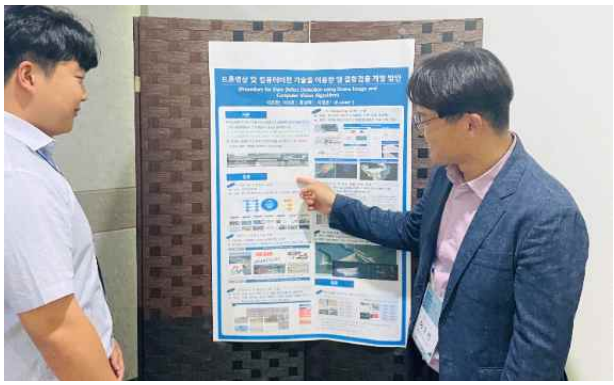
○ 또한 AI 연구센터의 성과 확산을 위해 2건의 논문*을 발표하였습니다.

* 취수원 수질예측을 위한 성층 물리변수 활용 데이터 기반 모델링 연구 (장현준 등)

* AI기반 물공급 시스템내 동파위험 조기경보를 위한 AI모델 개발 연구 (이소령 등)

□ 한국막학회 및 한국정보통신학회 참석

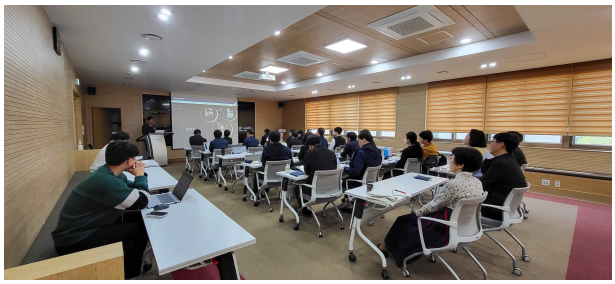
○ 한국막학회(5.18)에서 “막여과 정수장의 AI기술 적용사례”, 한국정보통신학회(5.25)에서 “드론영상의 AI 댐체결함 자동검출”에 대해 발표하였습니다.



□ AI연구센터 세미나 개최

○ AI연구센터에서는 다양한 연구주제로 모든 구성원들이 세미나를 발표하고 있으며, 지난 약 두 달간 5건의 세미나를 진행하였습니다.

회차	발표자	날짜	내용
2	이충성	3.30	수자원 계획관리 분야의 AI 기술 활용방향 모색
3	장현준	4.14	Docker 기초 개념 및 사용법 공유
4	이소령	4.17	스마트한 업무환경을 위한 Notion 소개 사용법
5	김성훈	4.19	연구인을 위한 AI 활용 이해와 실전
6	주경원	5.23	수도수요량 장기시계열 예측 알고리즘 및 코드 리뷰



□ 아르헨티나 공무원 대상 역량강화 워크샵 (5/10)

○ 아르헨티나 공무원을 대상으로 AI 수자원 활용 기술 워크샵을 진행했습니다.