

- 새로 알게 된 기술을 빠르게 익히고 알맞게 활용하여 멋진 것을 만들기 좋아합니다.
- 혼자보다는 같이 만드는 걸 좋아합니다.
- 좋아하는 일을 오래 하려면 돈을 벌어야 한다는 것도 압니다.
- 그래서 혼자 생각하지 않고 혼자 말하지 않고 혼자 만들지 않습니다.

이력

- 2022.01.- **Senior Software Engineer**, *ailys*, Seoul
웹 기반 no-code ETL 어플리케이션인 'SSDP'¹개발을 위한 cross-functional 팀 리딩
- FastAPI 기반 API gateway 도입을 포함한 백엔드 시스템 개편으로 요청 처리시간 최대 92% 감축 효과
 - 비즈니스 담당자와의 긴밀한 협업을 통한 요구사항 파악 및 개발 우선순위 관리
- 2021.02.- **Software Engineer**, *ailys*, Seoul
- 2021.12. 자사 머신러닝 플랫폼을 위한 Spark 기반 ETL 라이브러리 담당 개발
- SSDP 신규 구상, 설계 및 개발로 본사 제품의 ETL 기능성 15배 확장
- 2018.06.- **Machine Learning Scientist**, *ailys*, Seoul
- 2021.01. 웹 기반 no-code 머신러닝 플랫폼 'DAVINCI LABS'의 Python 백엔드 개발
- 비지도학습 모듈 기능 연구, 신규 개발 및 UX 설계 기여
 - 가중치 기반 지도학습 기능 개발 및 사고발생률 예측 문제 실사례 연구. 예측 성능 최대 17% 개선 확인
 - 외부 데이터 분석 및 R&D 프로젝트 다수 진행하여 건당 최대 3억원대 매출 기여
DB손해보험(KR), AEON Financial Services (JP), KEB하나은행 (KR), 한국원자력안전기술원 (KR) 등
- 2013.06.- **Private Instructor**, *Sehan Academy/Accel Education/Paul Academy/freelance*, Seoul
최대 20명 대상으로 미국 SAT/AP/IB 과정 Math, Physics, Statistics, Computer Science 및 Data Science 강의

학력

- 2008.08.- **B.S. in Mathematics with Distinction**, *Harvey Mudd College*, Claremont, CA, USA
- 2012.05. 이수과목: Data Structures (CS), Computability and Logic (CS), Stochastic Processes, Operations Research

기술

Skills

- 언어 **Python, SQL**
- 도구 **FastAPI, Pydantic, Spark, Dask**
- 이론 **MLOps, Clean Architecture, OOP, Functional Programming, Machine Learning**
- 환경 **Docker, Linux/Shell, Git, GitHub Actions**

특허

- 2021.10. 해석이 용이한 모델생성방법 및 그 장치와, 이를 이용하여 생성된 모델을 이용한 데이터분류방법 및 그 장치, 국내특허 제10-2314848호, 공동발명자
- 2021.10. 최적모델탐색방법 및 그 장치, 국내특허 제10-2314847호, 공동발명자
- 2021.06. 비지도학습에서의 사용자의도 반영 방법 및 그 장치, 국내특허 제10-2273868호, 공동발명자
- 2021.06. 비지도학습방법론에 기인한 지도학습모델 생성 방법 및 그 장치와, 이를 이용한 비지도학습모델 해석 방법 및 그 장치, 국내특허 제10-2273867호, 공동발명자

¹Self-Service Data Preparation

이력 상세

ETL 서비스 개편

역할/기간 리드, 2022.03. -

기술 Python, FastAPI, Spark, Docker, Apache Livy

- 내용
- 기존의 Apache Livy 서버 앞에 FastAPI 서버를 배치하여 API gateway 패턴 도입
 - Gateway 서버 API 설계 및 Swagger UI를 이용한 자동 문서화
 - Stateful 작업 효율화를 위해 기존의 batch script 지원 외에 interactive session 지원을 추가 구현
 - SSDP 관련 작업 로직을 batch script 기반에서 interactive session 기반으로 마이그레이션
 - 최대 실행 session 개수 제한 설정으로 CPU 및 메모리 과부하 방지 장치 (backpressure) 도입

- 효과
- ETL 작업 요청 처리시간 최대 92% 감축 (12초 → 1초)
 - 단일 client 서비스에서 multi-client 서비스로 사용성 확장

Spark 기반 no-code ETL 어플리케이션 (SSDP) 개발

역할/기간 리드, 2021.02. -

기술 Python, Spark, Pydantic, Docker, Apache Livy

- 내용
- SSDP: 사용자가 GUI 상에서 정의한 ETL 파이프라인에 따라 데이터를 처리하는 어플리케이션
- DAG 기반으로 Pipeline 개체 모델 설계 및 JSON Schema 표준에 맞춰 DSL로 구현
 - Lazy evaluation을 적극 활용하여 Spark로 개별 기능 구현 및 Apache Livy로 REST API 운용
 - UI/UX 디자인을 위한 cross-functional 논의 리딩; 현실 도구에서의 비유를 활용하여 디자인 직관화
 - 제품의 목표 시장을 담당하는 영업팀원을 인터뷰하여 현장 고객 요구사항 파악 및 우선순위 반영

- 효과
- 자사 머신러닝 제품의 ETL 기능성 15배 이상 확장
 - 전처리 지원 보강으로 자사 제품에 대한 일본 고객사 관심 증대

비지도학습 모듈 연구 및 개발

역할/기간 팀원, 2019.09. - 2021.01.

기술 Python, Dask, scikit-learn, TensorFlow, Clean Architecture

- 내용
- 지도/비지도 이상탐지 시스템과 데이터 탐색을 위한 manifold learning 기법에 대한 리서치 수행
 - Clean Architecture 방법론을 참고하여 Dask/scikit-learn/TensorFlow 기반 코어 라이브러리 구현
 - 사용자 workflow에 대한 고민을 바탕으로 UI/UX 디자인 논의 주도


- 효과
- 자사 머신러닝 제품의 지원 범위 확장
 - 모듈을 이용한 이상탐지 및 클러스터링 프로젝트 진행으로 매출 기여

개인 프로젝트

2020.11.- 2021.06.  **From Nand to Tetris: Building a Modern Computer from First Principles**, online course,

Python, Jack, Hack assembly, HDL

NAND 게이트에서 시작하여 실제 작동하는 컴퓨터 시스템(일명 'Hack computer system')을 만들기까지의 과정에서 주어지는 12가지 과제 전부 구현 완료

- HDL²을 사용하여 NAND 게이트로부터 ALU와 CPU 설계
- Stack 구조 기반 VM 바이트코드를 Hack의 어셈블리 언어로 번역하는 VM translator를 Python으로 구현
- Jack이라는 객체 기반 고수준 언어를 VM 바이트코드로 변환하는 컴파일러를 Python으로 구현
- Hack 시스템 상의 Jack 플랫폼 운영을 위한 간단한 OS를 Jack으로 구현
- 2048³ 게임을 Jack으로 구현 후 직접 만든 OS 상에서 작동 확인 

²Hardware Description Language

³<https://play2048.co/>