

유승희 (June Yoo)

단순 기능을 넘어 유의미한 가치를 고민하며,
AI와 서비스를 연결하는 **개발자 유승희**입니다.



[ABOUT ↗](#)

[PROJECTS ↗](#)

[EXPERIENCES ↗](#)

[CONTACT ↗](#)

ABOUT

ABOUT ↗

PROJECTS ↗

EXPERIENCES ↗

CONTACT ↗

ABOUT ME

AI와 서버를 연결하는 개발자

- AI 기능을 서비스에 효율적으로 적용해 비즈니스 문제를 해결
- 예측 및 자연어 분석 기반 모델 학습 경험 보유
- 학습된 모델을 활용한 API 구현 및 인터페이스 연동 경험 보유

데이터 신뢰와 처리 효율을 고려하는 개발자

- 데이터 무결성과 트랜잭션 안정성을 고려한 시스템 설계 및 개발
- 모델 라이프사이클 체계적 구축 및 관리하며 데이터 기반 의사결정
- 효율적인 데이터 처리를 위한 아키텍처 지속적 개선

명확한 의사소통을 중시하는 개발자

- 모든 과정에서 '선택의 이유'를 끊임없이 되묻는 습관
- 기획, 개발, 운영 등 협업 전반에 명확한 커뮤니케이션 주도
- 팀원들과 열린 소통을 통해 함께 성장하는 개발자

기술 스택

Language

Java (Spring Boot), Python (Flask, FastAPI), SQL

Backend/DevOps

JPA, MyBatis,
MSA, Docker, Jenkins, CI/CD, Kafka, Websocket

Database

MySQL, Oracle, Tiberio

Machine Learning


Scikit-Learn, TensorFlow, LightGBM

Cowork

GitHub/GitLab, Confluence, Jira, Slack

CAREER SUMMARY

COGNET9 | AI 데이터 구축 (인턴)

 2025년 8월(예정)

- AIA 생명 Smart AI Messenger 구축 프로젝트 (GEN AI & Live Chat)
- 라이브챗 채팅 솔루션 LLM 학습 데이터 전처리

KT DS, ICIS Tr 추진담당 고객팀 | Software Engineer (M1)

 2023년 1월 - 2024년 9월

- KT 통합고객정보시스템(ICIS) 차세대 전환
- 모놀리식 아키텍처 → Spring Boot 기반 MSA 환경 전환
- 고객파트 기능 개발 및 테스트 수행

PROJECTS

[ABOUT](#)



[PROJECTS](#)



[EXPERIENCES](#)



[CONTACT](#)



PROJECT SUMMARY

*프로젝트 카드를 누르면 해당 GitHub로 이동합니다.

Monitory

2025.04 - 2025.06

스마트팩토리 모니터링 시스템

kt ds

2023.01 - 2024.09

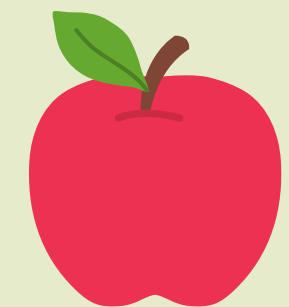
KT ICIS Tr 차세대 전환



STALKER

2022.01 - 2022.07

MBTI별 투자성향 분석 챗봇



2021.09 - 2021.11

농수산물 가격 예측 및 해석

스마트팩토리 실시간 모니터링 시스템

Monitory

2025.04 - 2025.06

스마트팩토리 모니터링 시스템

프로젝트 개요

센서,웨어러블로 수집된 공장 작업 환경, 설비 상태, 작업자 건강 정보를 실시간으로 모니터링 및 관리할 수 있는 시스템

프로젝트 주최

LG CNS AM Inspire Camp 01

개발 기간

2025.04 - 2025.06 (3개월)

사용 기술

Java 17, Python 3.12.3, Spring Boot 3.4.4
Spring WebSocket, Spring Kafka, FlywayDB
FastAPI, scikit-learn 1.7.0, LightGBM 4.5
MySQL8.0, Docker, Jenkins

개발 인원

Service : 3명 / Infra : 3명 / IoT : 2명 / ML : 1명

주요 역할 (기여도 : 60%)

백엔드 개발

- 실시간 히트맵 및 대시보드, 팝업 알림 구현

머신러닝

- 설비 노후 예측 모델 및 재학습 파이프라인 구축

일정 관리

- 문서화 프로세스 및 태스크·이슈 관리

스마트팩토리 실시간 모니터링 시스템

주요 역할

- 백엔드 개발
 - 실시간 기능
 - ERD · FlywayDB
- 머신러닝
- 일정관리

실시간 알림 기능

센서에서 메시지 브로커로 수집된 이벤트를 WebSocket 서버로 전달하여,
클라이언트에 100ms 이내 푸시 알림 제공

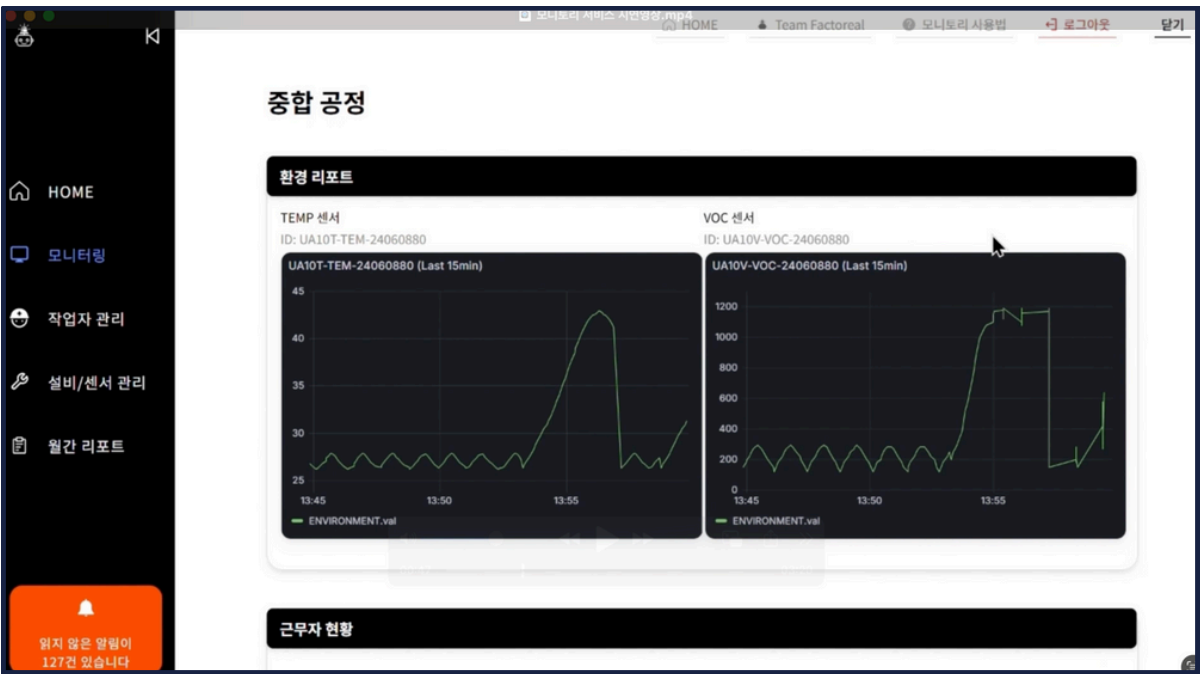


실시간 시계열 대시보드

시계열 데이터베이스 (InfluxDB)에 센서 데이터를
10초 단위로 적재하고, Grafana로 실시간 그래프 구현
운영자는 센서 추이를 실시간으로 확인 가능

공간별 위험도 히트맵 및 팝업 알림

실시간으로 측정되는 공장의 구역별 센서 데이터를
‘정상/주의/위험’ 3단계의 위험도 산출
위험도 기반 히트맵 및 팝업 알림
현장별로 우선 점검 공간을 직관적으로 확인 가능

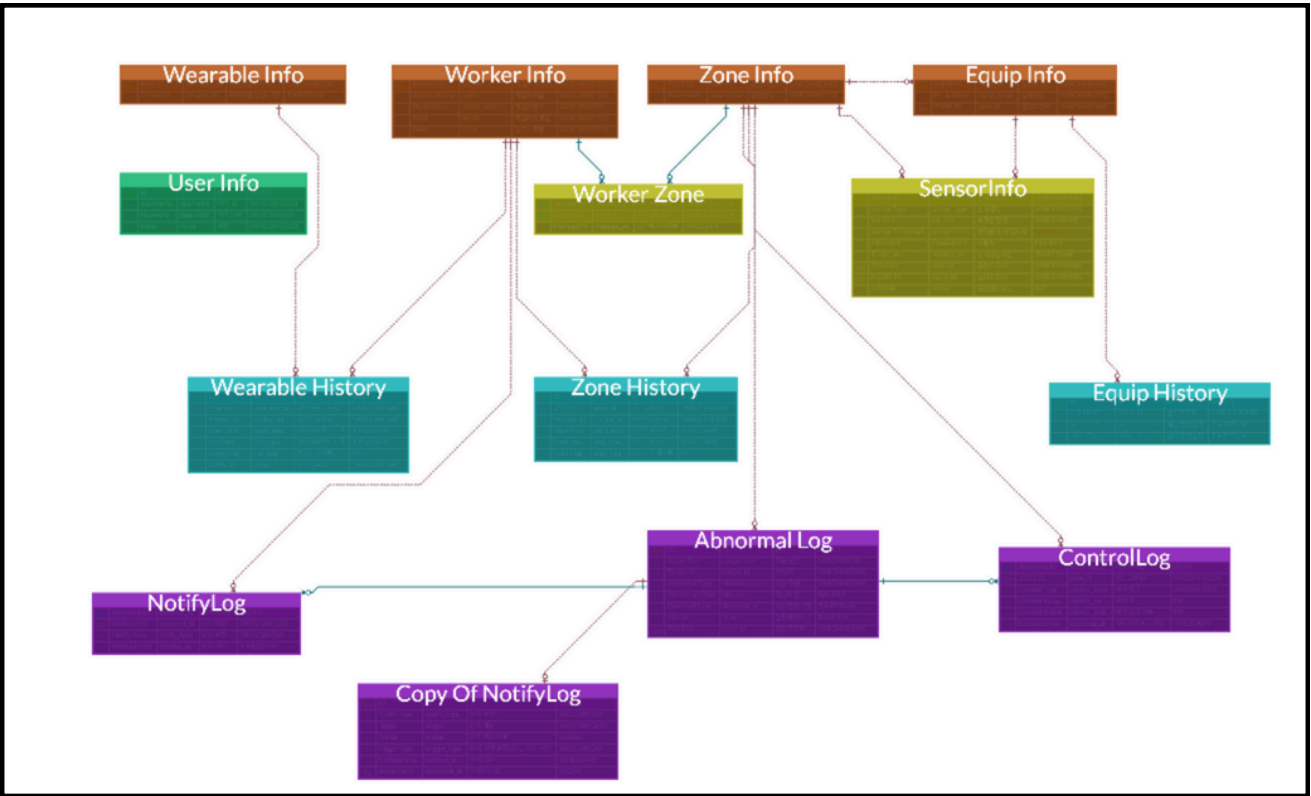


스마트팩토리 실시간 모니터링 시스템

주요 역할

- 백엔드 개발
 - 실시간 기능
 - **ERD · FlywayDB**
- 머신러닝
- 일정관리

ERD 설계 & FlywayDB Migration



```
flyway:  
  enabled: true  
  url: ${spring.datasource.url}  
  user: ${spring.datasource.username}  
  password: ${spring.datasource.password}  
  locations: classpath:db/migration  
  baseline-on-migrate: false  
  
db.migration  
  V1_1__insert_init.sql  
  V1__init.sql  
  V2_1__add_is_read_to_abnormal_log.sql  
  V2__alter__abn_log__add_auto_increment.sql  
  V3__alter_sensor_table_add_allowval_column.sql  
  V4__add_worker_zone_map.sql  
  V5__alter_sensor_table_thres_type.sql  
  V6__insert_worker_data.sql
```

ERD 4계층 구조 설계

Metadata - Relationship - History - Log 계층으로
테이블을 분리해 스키마 복잡도를 낮추고
데이터 정합성·추적성·운영 효율을 동시에 확보

FlywayDB 기반 스키마 버전 관리

버전별 DDL 스크립트 Flyway migrate로 자동 적용,
flyway_schema_history로 변경 이력 기록 및 관리

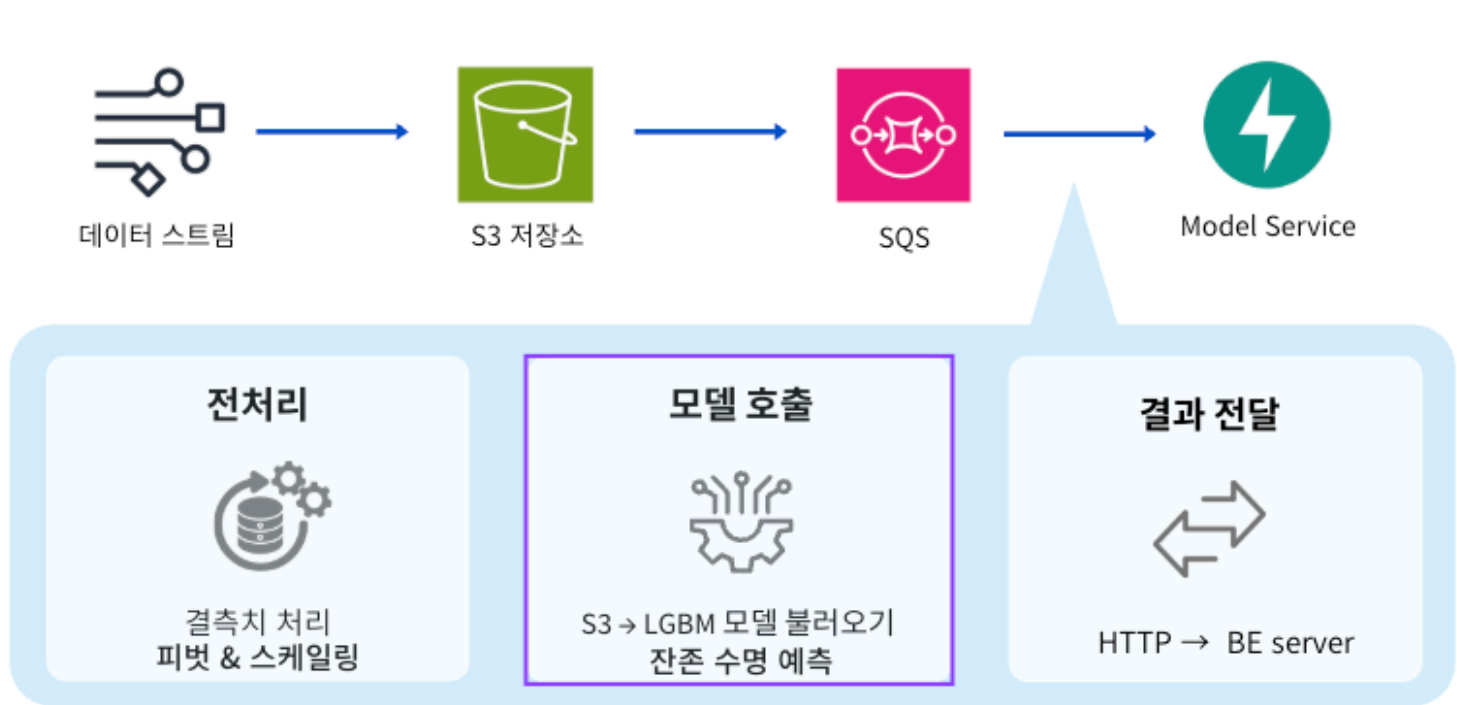
스마트팩토리 실시간 모니터링 시스템

주요 역할

- 백엔드 개발
- 머신러닝
 - 설비 점검일 예측
- 일정관리

설비 점검일 예측 기능

7,000건의 학습 데이터셋 구축 후, LightGBM 회귀 모델 (RMSE ± 3.97 , R^2 0.55) 학습해 설비 점검일을 FastAPI로 제공하고, APScheduler 재학습 자동화 프로세스를 구축
D-Day 대시보드와 Slack 알림을 통해 설비 노후화 선제 대응 가능



모델 선정

LightGBM Regressor

모델 지표

RMSE 3.95
MAE 3.27

설비 현황

종합 설비

예상 점검 일자 (2025-06-25)

예상 점검일까지 D-5

최근 점검 일자 (2025-06-20)

2:06

⚠ 설비 점검 알림

[설비명]

종합 설비

[공간]

종합 공정

[예상 점검일]

2025-06-25

[남은 기간]

🔴 D-5

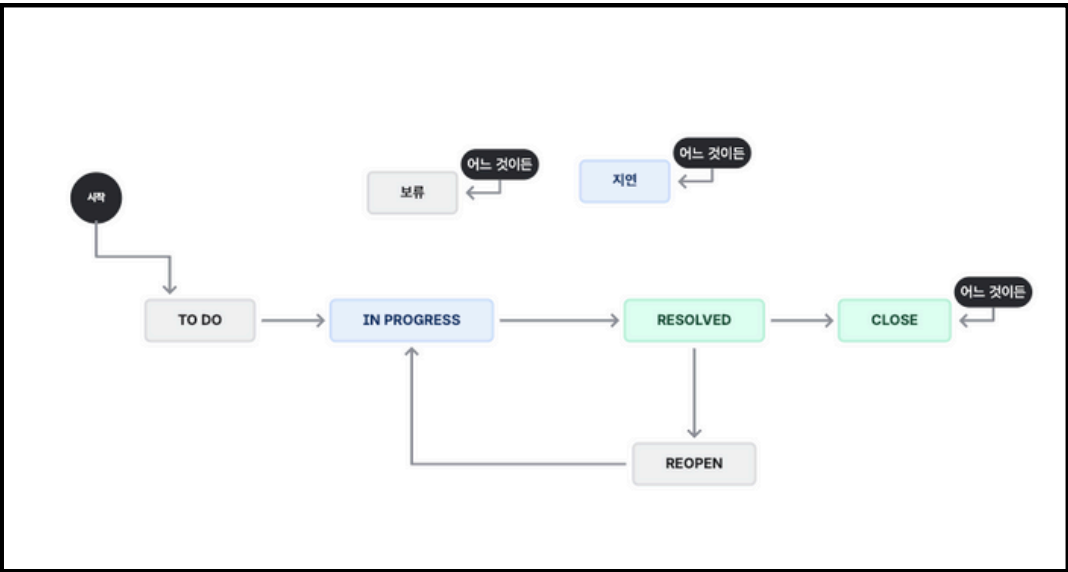
스마트팩토리 실시간 모니터링 시스템

주요 역할

- 백엔드 개발
- 머신러닝
- 일정관리

일정 관리 & 스크럼 운영

프로젝트의 스케줄 관리와 문서화 프로세스를 담당하며,
Jira 기반 워크플로우와 이슈 템플릿 구축 → 일정 준수 및 원활한 협업 환경 체계화



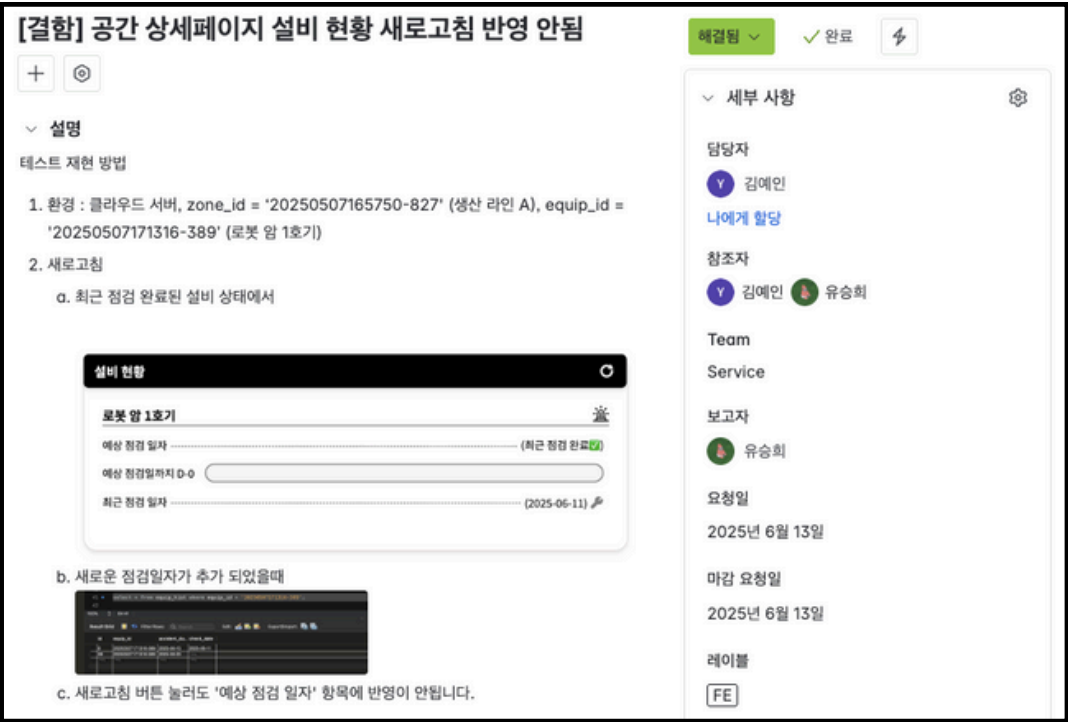
문서화 프로세스 구축

데일리·회고 회의마다 표준 회의록 양식을 작성·공유
Daily Scrum을 10분 이내로 진행해 진행 상황,
장애 요인, 다음 액션 아이템 기록

Jira 이슈 워크플로우 설계 및 결함 관리

백로그·업무 요청·결함 이슈별
입력 필드 (마감일·담당자·담당파트)를 표준화한 템플릿 제작

이슈 워크플로우를 설계해 결함 발생시
우선순위 지정 → 담당자 할당 → 해결 후 회고
결함 대응 속도와 품질 개선



스마트팩토리 실시간 모니터링 시스템

트러블 슈팅

@Async 도입 기반의 실시간 메시지 처리 지연 해결

? Situation

메시지 브로커로부터 수신된 메시지 처리 과정에서 5초 이상의 지연이 발생
처리 경로 : [Message Broker] → [BE 서버 구독] → [로직 처리] → [WebSocket 전송]

📌 Task

모니터링 시스템 특성상 실시간 반영을 위해 처리 속도 개선 필요
“수신 → 처리 → 전송” 파이프라인을 1초 이내로 단축하여 즉시 결과 반영 필요

★ Result

실시간 처리 지연 **5,000ms → 1,00ms 미만**으로 단축
사용자 화면에 즉시 반영되는 실시간 알림 제공으로 시스템 안정성 및 사용자 신뢰성 향상

Action

```
@Configuration 0개의 사용위치  Juyoung Jeong
@EnableAsync // <-- @Async를 사용할 수 있도록 활성화
public class AsyncConfig implements AsyncConfigurer {

    @Bean(name = "websocketExecutor") 1개 사용 위치  Juyoung Jeong
    public Executor websocketExecutor() {
        ThreadPoolTaskExecutor executor = new ThreadPoolTaskExecutor();
        executor.setCorePoolSize(5); // 동시에 실행할 기본 스레드 개수
        executor.setMaxPoolSize(20); // 최대 스레드 개수
        executor.setQueueCapacity(100); // 대기 큐 크기
        executor.setThreadNamePrefix("ws-async-");
        executor.initialize();
        return executor;
    }
}
```

전용 비동기 스레드풀 설정
단계별 비동기 메서드 분리 (메시지 수신, 로직 처리, 결과 전송)
비동기 처리가 필요한 메서드 독립 스레드 실행
Async 예외처리 핸들러 구현 → 예외 발생 시 Slack 알림 구현

KT 통합고객정보시스템 (ICIS) 차세대 프로젝트



2023.01 - 2024.09

KT ICIS Tr 차세대 전환

프로젝트 개요

KT DS에서 진행한 통합고객정보시스템 (ICIS) 차세대 전환 프로젝트로, 기존 모놀리식 아키텍처 시스템을 Spring Boot 기반 MSA 환경으로 전환

사용 기술

Spring Boot 3.0.2, Java 17, JPA, MyBatis, Nexacro, Tiberio DB, Oracle DB, Kafka
GitLab, DevPilot, SVN

개발 기간

2023.01 - 2024.09 (1년 9개월)
- Iteration1 ~ Iteration4 AIT

주요 역할 (기여도 : 45%)

레거시 → MSA 전환

Pro*C/Visual C++ 기반 AS-IS 기능 70여 건
Spring Boot & Nexacro 재구축

고객 서비스 기능 개발

LOB2 기업인터넷 상품
개통·해지·청구 비즈니스 로직 설계 및 구현

시스템 연동

EAI 기반의 타 시스템 연계 인터페이스
설계 및 데이터 전송 기능
“KT API Link” 규격 협의·개발을 담당

배치·데몬 최적화

배치성 데몬 가이드 템플릿 제작
DevPilot 스케줄러 운영

KT 통합고객정보시스템 (ICIS) 차세대 프로젝트

트러블 슈팅

데몬 기능 트랜잭션 범위 설정으로 레코드 락 충돌 해결

```
// 🛑 Situation (기존 구조: 전체 메서드에 트랜잭션)
Time →—————

TX1: |—SELECT R (S lock)—|—UPDATE R (upgrade to X lock)—|—COMMIT (release)—|
TX2: |—SELECT R (S lock 요청)—[블록됨]—...

// 🟢 Task (개선 구조: 조회·처리 분리, 처리만 트랜잭션)
Time →—————

DaemonJob (no TX): findTargets() → [1,2,...] 반환

For each ID:

TX-A: |—UPDATE R1 (X lock)—|—COMMIT (release)—| ← processAB(1)
TX-B: |—UPDATE R2 (X lock)—|—COMMIT (release)—| ← processAB(2)
```

- ? Situation
 - 대용량 트래픽 환경에서 Spring Scheduler + 전체 메소드 @Transactional 적용 구조로 실시간 데몬 작업 중 동일 레코드에 대한 락 충돌 발생하며 무한 대기 발생
- 📌 Task
 - 레코드 락 충돌 현상을 해결하고, 데몬 처리 로직이 개발,테스트,운영 어느 환경에서도 안정적으로 수행 가능하도록 트랜잭션 범위 재설정

KT 통합고객정보시스템 (ICIS) 차세대 프로젝트

트러블 슈팅

데몬 기능 트랜잭션 범위 설정으로 레코드 락 충돌 해결

Action

```
@Service
public class DaemonService {

    // 트랜잭션 없이 조회만
    public void daemonJob() {
        List<Data> list = repository.findPending();
        for (Data d : list) {
            try {
                processAB(d);
            } catch (Exception e) {
                // 로그 기록 후 바로 다음 레코드로
            }
        }
    }

    // a, b를 한 번에 묶는 트랜잭션
    @Transactional
    public void processAB(Data d) {
        a(d); // 첫 번째 처리
        b(d); // 두 번째 처리
        // 둘 다 성공하면 commit, 하나라도 예외면 rollback
    }
}
```

```
Scheduler
|
|
| [Daemon Job Controller] ← 논리 트랜잭션 없음 (조회만 수행)
|
| └─ daemonReceiver() 조회 로직 (NO TX) → 대상 목록 반환
|
| └─ for each item in 목록:
|       |
|       | └─ daemonCud(item) ← 논리 트랜잭션 시작 (@Transactional)
|       |       |
|       |       | └─ Begin Physical TX (DB Connection)
|       |       |
|       |       | └─ repository.updateStatus(item)
|       |       |
|       |       | └─ Commit Physical TX
```

- 데몬 기능 전체 메소드의 @Transactional 제거
- 도메인별 조회로직 처리로직 데몬용 서비스로 분리
- 조회 단계 트랜잭션 없이 Shared Lock 방지
- 처리 로직만 트랜잭션 범위 최소화
- TransactionManager를 활용한 커밋/롤백 제어

★ Result

레코드 락으로 인한 무한 대기 이슈를 해결하고 실시간 데몬 처리 성공률을 확보
파트 내 데몬 기능 개발 가이드 템플릿 작성, 코드 품질 향상에 기여

MBTI별 투자성향 챗봇



STALKER

2022.01 - 2022.07

MBTI별 투자성향 분석 챗봇

프로젝트 개요

카카오 채팅을 통한 MBTI 기반의 투자성향을
분석해주는 대화형 챗봇

프로젝트 주최

BOAZ 17기 BIGDATA Conference

개발 기간

2022.01 - 2022.07 (6개월)

사용 기술

Python, Flask, Konlpy(Okt 토큰나이저),
TensorFlow 2.8.2, Keras 2.8.0 (BiLSTM)

개발 인원

Data&ML : 3명 / ML&Server : 2명

주요 역할 (기여도 : 80%)

머신러닝(자연어 처리)

- 투자성향 분석을 위한 데이터셋 구축 및 모델링
- 서버 개발
- 모델 서빙 및 카카오톡 오픈빌더 연동

MBTI별 투자성향 챗봇

주요 역할

- 머신러닝 (자연어처리)
 - 데이터셋
 - 모델 설계·학습
- 서버개발

데이터셋

MBTI 유형별 투자 성향을 파악하기 위해 수집된 자연어 응답으로 구성된 텍스트 감성 분석용 데이터 셋으로, 다양한 심리·투자 맥락을 포함해 모델 학습에 최적화된 구성을 갖추고 있습니다.

학습용 데이터셋 구축

주관식 10개 문항으로 사용자 응답을 수집,
금부정의 정도 기준으로 라벨링
<부정> 0.0 - 0.3 - 0.5 - 0.7 -1.0 <긍정>
총 3,000개의 답변 데이터 확보



데이터 전처리

Okt 형태소로 토큰화 한 뒤 특수문자·불용어를 제거하고, 최대 길이 기준(Max_len = 75)으로 패딩하여 모델 학습용 입력으로 최적화했습니다.

MBTI별 투자성향 챗봇

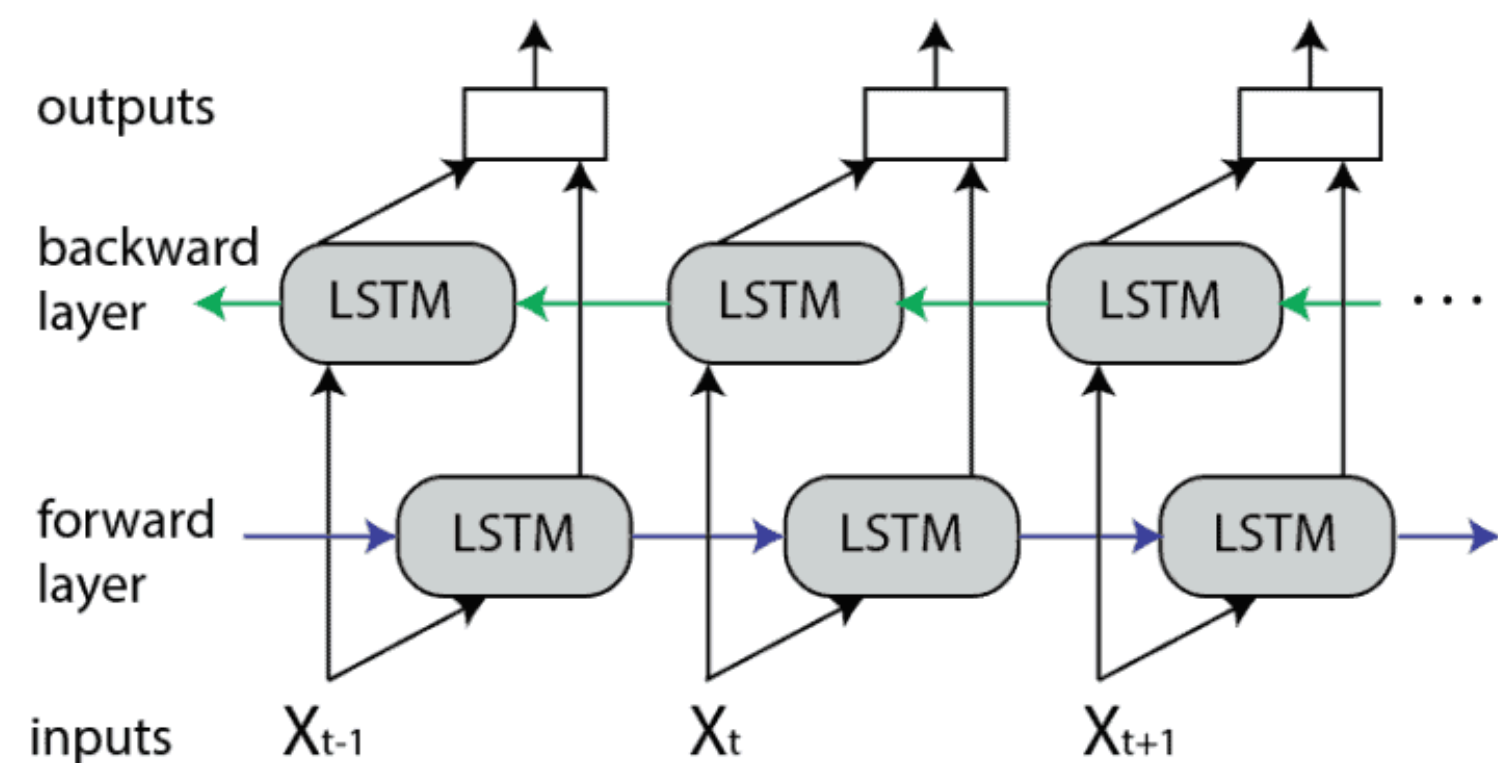
주요 역할

- 머신러닝 (자연어처리)
 - 데이터셋
 - 모델 설계·학습
- 서버개발

모델 설계·학습

주관식 채팅은 응답 길이가 가변적이고

앞뒤 문맥 모두가 중요한 특성 때문에 양방향 정보를 학습하는 BiLSTM을 선택



BiLSTM 아키텍처

입력 토큰 10차원 임베딩 벡터화 →
양방향 LSTM (128 → 64) 로 문맥 학습 →
Dense(2,softmax) 출력하여 긍부정 예측 분류

answer	orig	pred
잘어울린다	1	1
관심가져주고 질문많이 해줘서 고마움	1	1
무조건..?이라고..? 흠 뭔가 의심이 간다	0	0
마음에 들지만 경계한다	0	0
화이팅	1	1
...
와 글썽요^^! 라고 대답하고 집가서 평소처럼 산다 (연락안함)	0	1
날 걱정해주다니..고마워 친구..	1	1
빛지는건 오바다	0	0
그럴 수 있지	1	0
돈줘야하나	1	0

예측 결과

테스트셋 평가 모델 79.45% 정확도 기록
챗봇 서비스 적용 가능 수준

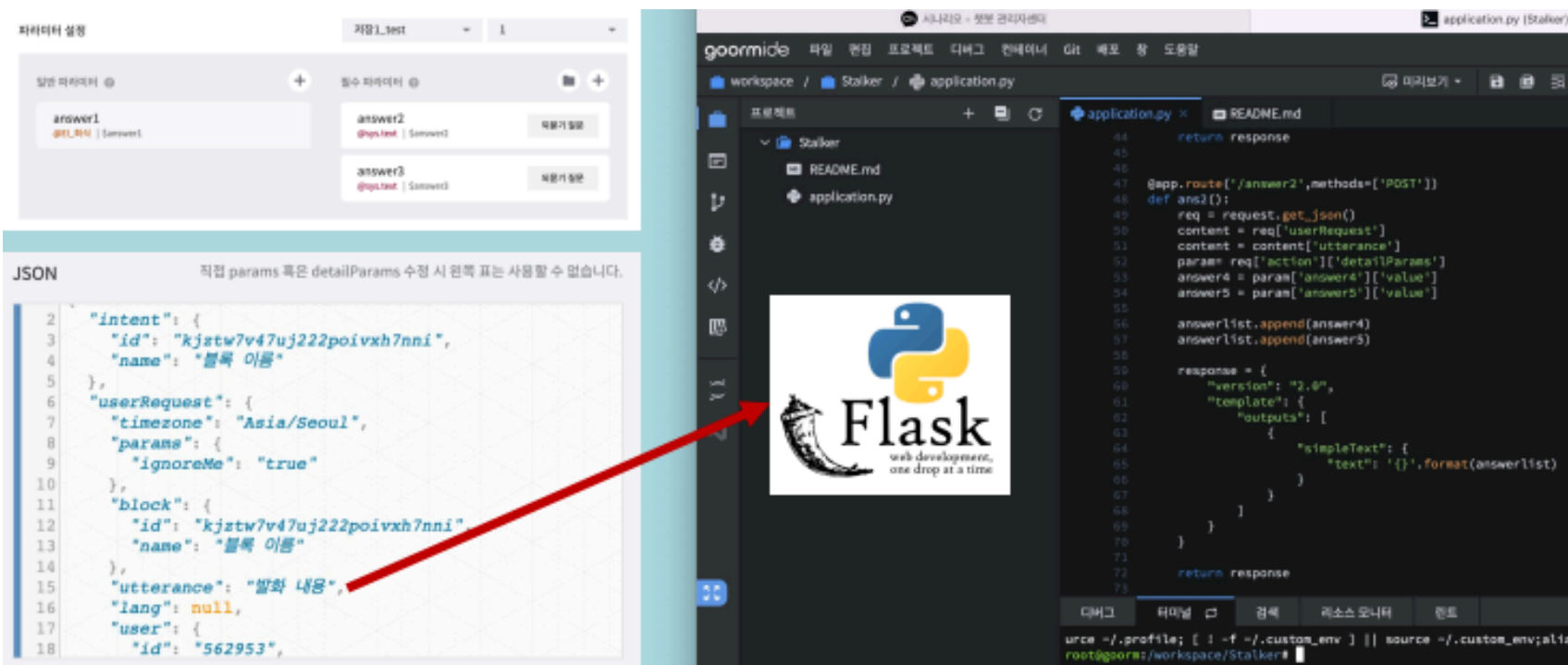
MBTI별 투자성향 챗봇

주요 역할

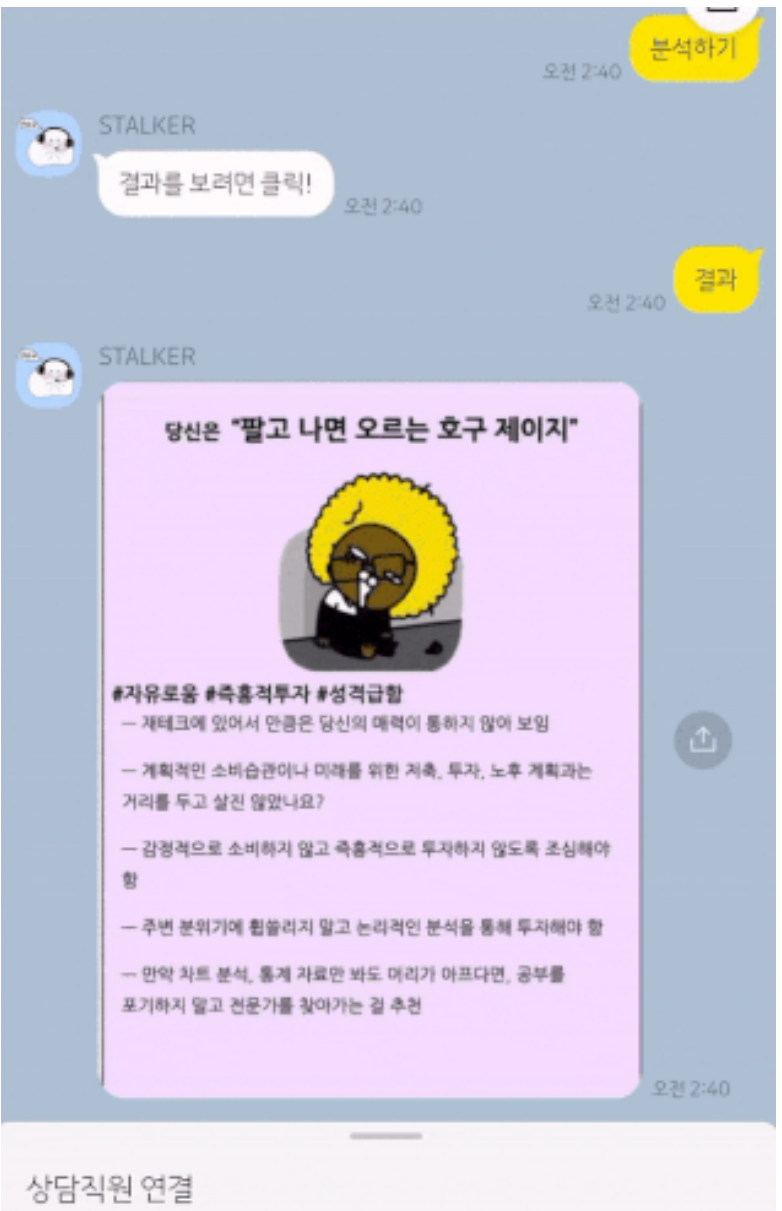
- 머신러닝 (자연어처리)
 - 데이터셋
 - 모델 설계·학습
- 서버개발

서버 개발

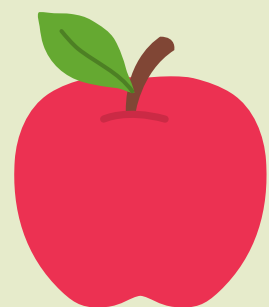
카카오톡 오픈빌더와 ML 감성 분석 모델을 연결해,
사용자별 투자 성향 결과를 실시간으로 제공하는 챗봇 구축



- 카카오 오픈빌더와 연동해 JSON/HTTP 엔드포인트로 메시지 수신·응답을 안정적으로 처리하고, 메모리 기반 세션으로 사용자별 대화 이력을 누적
- Flask 서버 기반 감성분석 모델 추론 엔드포인트 제공
 - 추론 결과 simpleImage 형식으로 포매팅하여 전송



농수산물 가격 예측 및 해석



2021.09 - 2021.11

농수산물 가격 예측 및 해석

프로젝트 개요

기상·공급 불안정으로 변동하는 농산물 가격을
공공·민간 데이터 통합 및 분석으로
가격을 예측하고 인사이트를 제공

프로젝트 주최

중앙대 응용통계학과 공공 데이터 분석 대회
- 대상 수상

개발 기간

2021.09 - 2021.11 (2.5개월)

사용 기술

Python, statsmodels(OLS, VAR)
scikit-learn(ExtraTree),
TensorFlow/keras(Conv1d + LSTM), LIME

개발 인원

Data Analyst : 4명 (EDA / 선형 / 비선형)

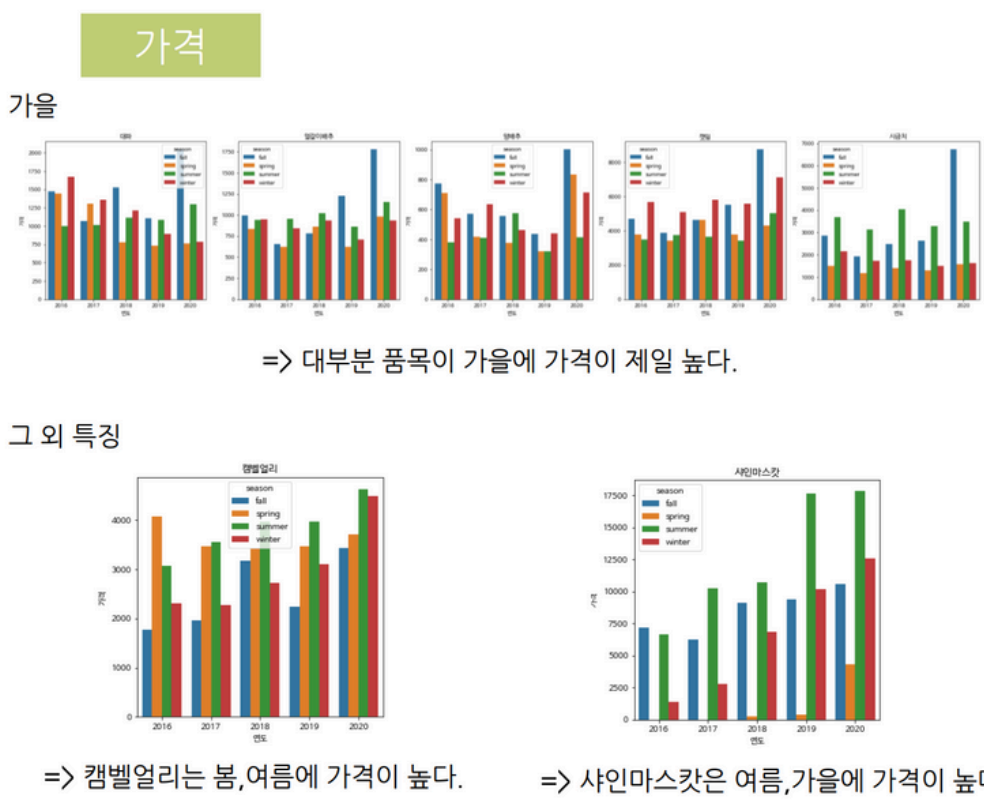
주요 역할 (기여도 : 70%)

데이터 통합·EDA
단계적 모델 개선
모델 해석

농수산물 가격 예측 및 해석

주요 역할

- 데이터 통합·EDA
- 모델 개선

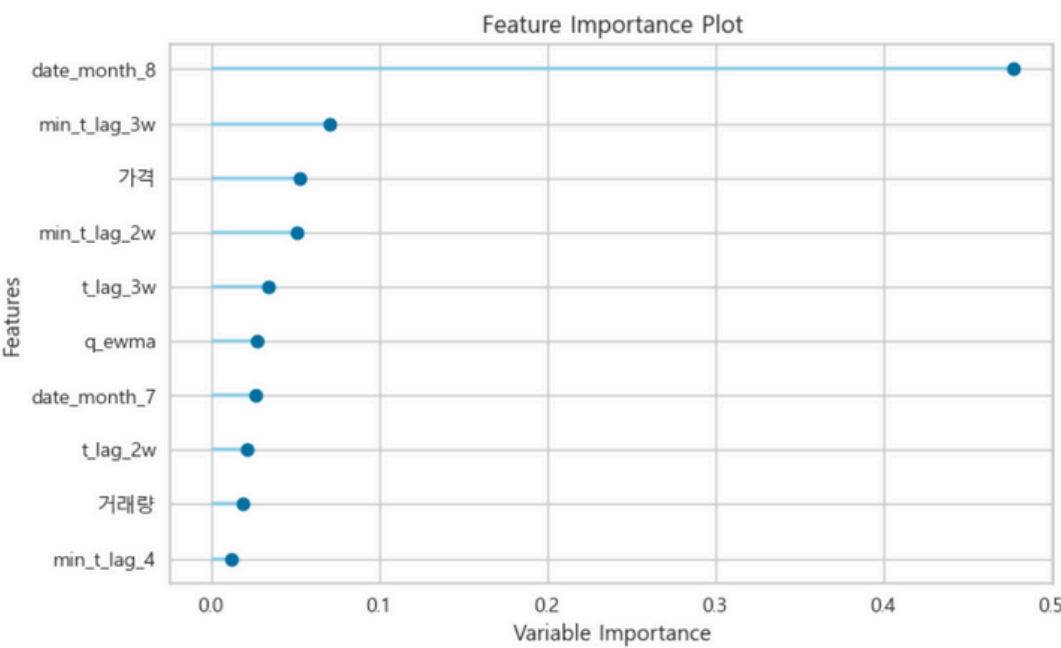


단계적 모델 개선

OLS 회귀 → VAR 적용 →
Extra Trees로 변수 중요도 평가 →
LSTM 기반 시계열 패턴 학습 → LIME 기반 모델 해석

데이터 통합·EDA

농산물 가격, 거래량, 경유 가격, 농산물 물가지수,
기후 데이터를 확보해
계절·지역·요일별 EDA로 데이터 분포 및 주요 패턴 도출



성능이 가장 좋았던 품목이라는 것을 고려해봤을 때 기후 변수 중요
기후 변수가 중요하게 고려되면 모델의 설명력을 높일 수 있을 것

농수산물 가격 예측 및 해석

트러블 슈팅

딥러닝 모델의 블랙박스 한계 극복

? Situation

Extra Trees 트리 모델로 예측 정확도가 충분히 개선되지 않아,
LSTM 모델 도입을 검토해야하는 상황

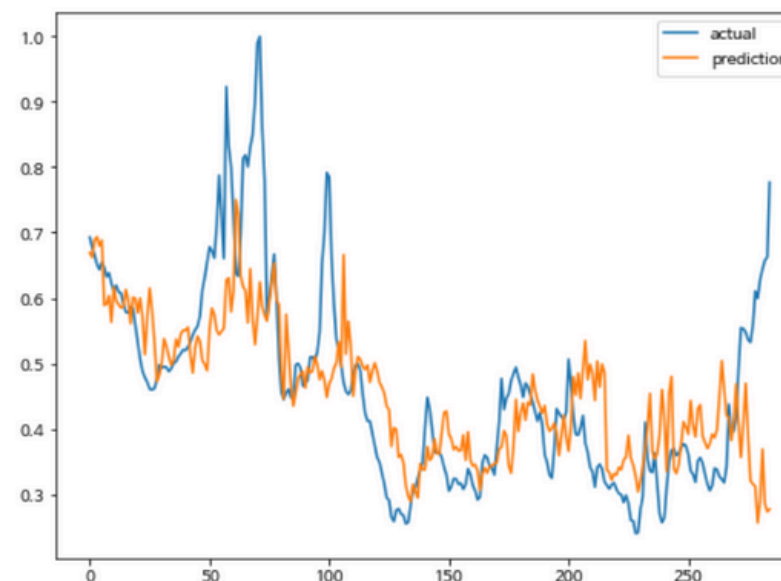
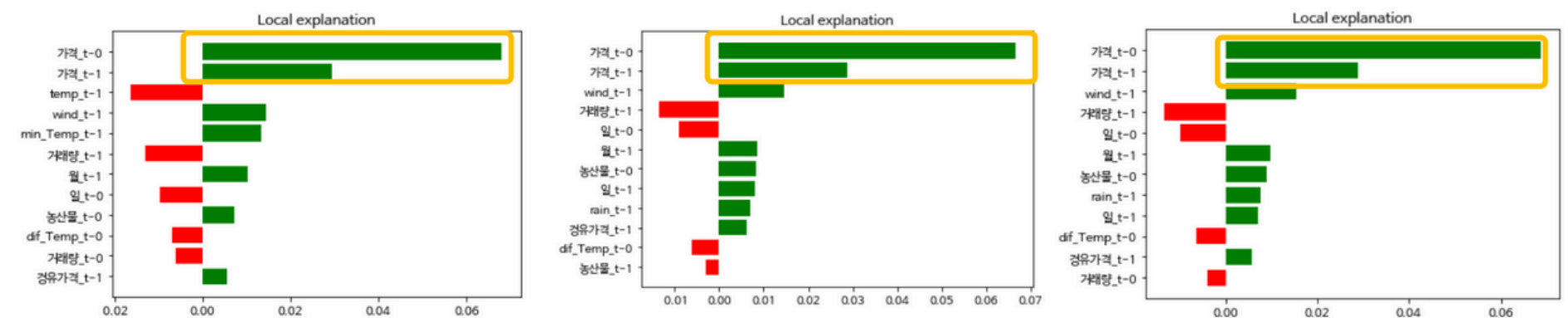
📌 Task

시계열·기후 특성을 반영한 모델 성능 개선
“농산물 가격 예측에서 유의미한 인사이트”를
도출할 수 있도록 해석력 확보가 필요

👤 Action

- LSTM 학습 :
RNN 기반 시계열 흐름과 기후 변수의 복합적 상관관계 학습
- LIME 알고리즘의 Local Explanation :
품목별 변수(기후,거래량 등)의 영향도를 국소적 분석 실시

★ Result



- 트리모델 대비 주요 품목에 대한 성능 향상
평균 **NMAE 0.32 → 0.23** 으로 약 **23%** 개선
- LIME 분석으로 해석력 확보
가격 변동에 대한 기후와 거래량 요인의
영향도 분석 및 인사이트 제공

EXPERIENCES

ABOUT



PROJECTS



EXPERIENCES



CONTACT



EDUCATION

중앙대학교

Applied Statistics

- 학사 2019.02 - 2022.08
- 전체 성적 (4.11 / 4.5)
- 전공 성적 (4.25 / 4.5)

돌마고등학교

- 인문계 졸업 2018.03 - 2019.01

홍콩 한국 국제학교

HK KIS

- Korean Section 2015.09 - 2018.01
- English Section, IGCSE 2013.02 - 2015.09

CERTIFICATES

AZURE FUNDAMENTALS (AZ-900) | 2024.07.06

AICE ASSOCIATE | 2023.07.26

SQLD | 2022.09.30

컴퓨터활용능력 1급 | 2021.07.30

ADSP | 2020.09.29

EXTRACURRICULAR

LG CNS AM Inspire Camp

- 2024.12 - 2025.06
- MSA·AWS·CI/CD 학습 및 프로젝트 수행

중앙대 응용통계학과 학생회장

- 2020.12 - 2021.11
- 코로나 대응 비대면 행사 기획, 공모전 개최

빅데이터 분석 학회 BOAZ

- 2021.07 - 2022.07
- ML/DL/NLP 데이터 분석 과정 학습 및 실습

GitHub : <https://github.com/JuneSHYoo>
Phone : 010-2549-9251
Email : juneyo0823@naver.com

CONTACT

