2026-1 CCIT 최종발표

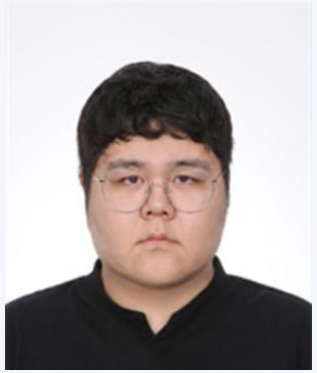
머신러닝 기반 선박 AIS IDS

팀 JB-Pirate-King

Index

TEAM

팀 소개 · 역할 분담



양승권

- 이상 신호 발생기(aivdm_gen) 제작
- 32종 공격 시나리오 설계

임성빈

- AIS 데이터 전처리 · 모델 학습
- OpenCPN IDS 플러그인(C++) 개발

CHAPTER 01

배경과 필요성

선박 사이버 위협과 AIS의 구조적 한계

선박의 디지털 환경

다양한 디지털 기자재

GPS · AIS · ECDIS 등 외부 신호 기반 항법 시스템에 의존

IT / OT 융합 가속

운영기술과 정보기술 결합으로 외부 신호 의존도 · 연결성 증가

NMEA 네트워크 확장

항법 데이터가 단일 네트워크로 통합

선박 사이버 사고 동향

지속 증가

GNSS · AIS 교란 사례가 매년 증가

지역 집중

중동 · 동유럽 등 특정 해역에 집중 발생

안전 직결

항로 왜곡 · 위치 오인이 항해 안전을 직접 위협

사고 사례 — 이탈리아 유령선박 사건

AIS Spoofing 으로 선박 ECDIS 화면 오염

1

AIS Spoofing

위치 신호 교란 발생

2

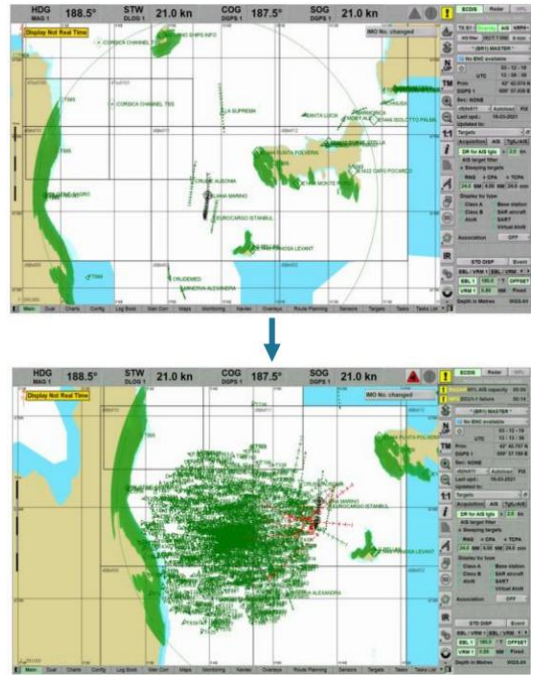
ECDIS 화면 오염

다수 선박이 동일 지점으로 표시

3

운항 판단 오류

충돌 위험 · 선박 정지 · 운임 손실



AIS의 구조적 취약점

평문 브로드캐스트

AIVDM 메시지를 암호화 · 인증 없이 평문으로 송수신

무조건적 신뢰 구조

수신 신호의 진위를 검증하지 않고 그대로 신뢰

가짜 신호 무방비 수신

허위 위치 · 속도 · 식별 정보를 그대로 화면에 반영

AIS Spoofing 위협

허위 신호 주입

가짜 위치 · 속도 · 식별 정보를 AIS 채널에 주입

상황 인식 왜곡

시스템과 항해자의 상황 인식을 의도적으로 왜곡

운항 피해

운항 중단 손실, 최악의 경우 선박 간 충돌

핵심 착안점 — AIS 메시지를 분석해 가짜 신호를 탐지하면 손실 이전에 대응 가능

CHAPTER 02

시스템 설계와 구현

OpenCPN 플러그인 · 머신러닝 탐지 · 신호 발생기

메인 아이디어

AIS 메시지 분석

수신한 AIVDM 시퀀스에서 이상 행동 패턴을 탐지

IDS로 실시간 대응

ECDIS(OpenCPN) 위에서 이상 신호를 즉시 식별 · 시각화

자율운항 대비

무인 자율운항선박에 필수적인 가짜 신호 탐지 기반 확보

프로젝트 구조 — 3개 구성요소

이상 신호 발생기

`aivdm_gen`

정상 · 이상 AIVDM 시퀀스 생성으로 탐지기 검증

AIS IDS 플러그인

`OpenCPN · C++`

AIVDM 수신 → ONNX 추론 → 정상 · 이상 판정

ML + 개선 파이프라인

`Python`

전처리 · 학습 · 평가 · 피처 엔지니어링 · 배포

탐지 흐름

선박(MMSI)당 AIS 신호 10개가 누적되면 한 시퀀스로 묶어 추론을 시작

OpenCPN — 오픈소스 ECS

- Windows · Linux · macOS · Android 멀티플랫폼 지원
- S-57 전자해도 규격과 커스텀 플러그인 구조 제공
- 플러그인은 별도 설치 없이 파일 하나로 배포 가능
- 플러그인 API로 수신 AIVDM 메시지에 직접 접근

IDS 플러그인 동작

1	**MMSI별 시퀀스 누적**	선박당 AIS 신호 10개가 모이면 탐지 시작
2	**ONNX 모델 추론**	번들된 model.onnx로 재구성 오차(MSE) 계산
3	**임계값 판정**	threshold 초과 시 이상 신호로 분류
4	**시각화 · 로그**	이상 신호는 빨간 점, 로그창에 MMSI 출력

머신러닝 기반 탐지 — 원리

비지도 학습

레이블 없는 정상 시퀀스만 학습 (원본 데이터에 정답 없음)

시계열 입력

SEQ_LEN = 10, 시간 흐름에 따른 패턴 반영

재구성 오차 판정

MSE가 임계값을 초과하면 이상으로 분류

학습 데이터 — Marine Cadastre

- NOAA · BOEM 공동 운영 해양 공간 데이터 플랫폼
- 미국 연안 AIS 방송 데이터 (2023 ~ 2025, 3개년)
- 원본(.csv.zst) → 전처리 → 파생 피처 추출 → 학습용 CSV
- 정상 항행 데이터만으로 비지도 모델 학습

초기 입력 피처 12종 (베이스)

sog	cog	heading
status	dt	dist_km
cog_hdg_diff	sog_change	cog_hdg_change
speed_consistency	lat_speed	lon_speed

이상 신호 발생기 (aivdm_gen)

실제 AIS와 동일한 AIVDM 규격으로 정상 · 이상 시퀀스를 합성해 탐지율 · 오탐율을 정량 측정

대표 공격 패턴

속도 이상

순간 · 주기적 SOG 급증

정박 이동

navStatus=1 상태에서 이동

COG / HDG 불일치

진행 · 선수 방향 모순

위치 점프

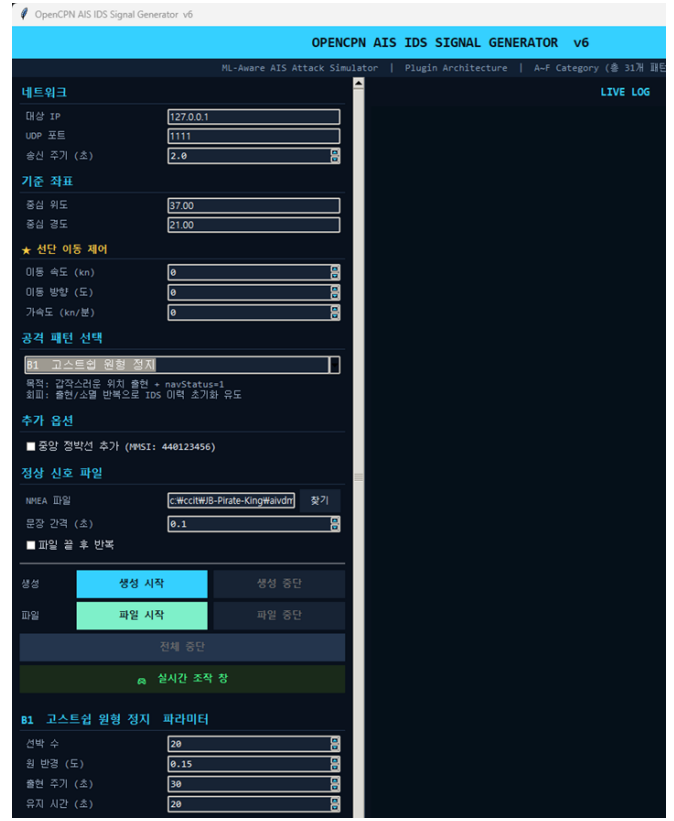
짧은 시간 내 비현실적 이동

원형 순찰

일정 각속도 선회 패턴

집게 협공

중심점으로 양측 수렴



평가 시나리오 — 총 32종

Basic	4종	기본 공격 — COG/HDG · 정박이동 · 속도 · 위치점프
FN	4종	규칙 기반 탐지기 회피용
D	4종	머신러닝 모델 회피 1세대
E	5종	머신러닝 모델 회피 2세대
F	7종	룰 + ML 회피 1세대
G	7종	룰 + ML 회피 2세대

모델 진화

중간발표

LSTM Autoencoder

단일 비지도 시계열 모델로 시작

▶

벤치마크

9개 모델 비교

dcdetect · conv1d · LSTM · tcn 등

▶

검증

앙상블 무이득 확인

탐지율 향상 없음 → 단일 모델 사용 근거 확보

▶

선택

DCdetect 채택

이중 채널 · 패치 어텐션, 위장 공격 탐지에 강점

CHAPTER 03

LLM 활용 수치 개선

LLM 자가개선 파이프라인 · 평가 · 한계

파이프라인 구축

단순 LLM 활용

모델을 학습하고 정확도를 보고하는 단계에서 종료

LLM 기반 프로젝트 부분 자동화

LLM이 개발 · 분석 · 판단 · 피처 발명까지 수행하는 자가개선 파이프라인 구축

LLM 기반 자동화 + 정량 검증

LLM은 제안

새 피처와 다음 전략을 LLM이 제안

데이터 수치 기반 채택

목적점수 +3.0pp 게이트를 통과한 피처만 채택

사람이 승인

배포 · 릴리즈 등 비가역 단계는 사람이 최종 승인

자동화 백본 — 7개 서비스 생태계

모든 서비스를 MCP로 LLM이 직접 조작 — 보고 · 기록 · 배포 무인 자동화

Notion

구조화 기록

Obsidian

로컬 지식 베이스
+ 깃을 통해 동기화

Slack

공식 보고 · 승인 게이트

Discord

실시간 자동 알림

Sheets

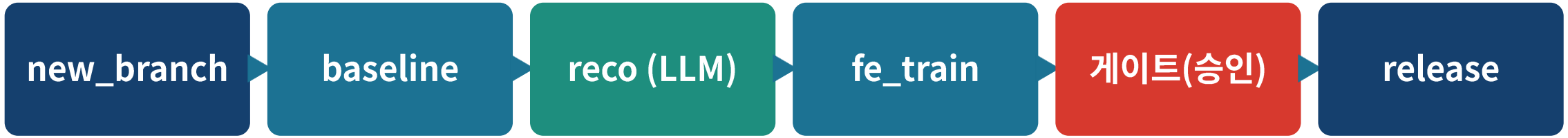
수치 추적 허브 (5탭)

GitHub

자동 브랜치 · 커밋 · 릴리즈

LangGraph 자가개선 파이프라인

브랜치 체이닝 = 그래프 사이클 — run마다 피처 1개 자동 채택 → 재학습 → 배포



StateGraph

노드 + 엣지 그래프, 체이닝은 사이클

Greedy FS

+3.0pp 이상이면 피처 채택, 회귀 가드

HITL 게이트

비가역 단계는 interrupt()로 사람 승인

자동화 백본

Slack · Sheets · GitHub 릴리즈 자동

① 개발 — Claude Code + CLAUDE.md

- ml/ 파이프라인 전체를 LLM이 작성
  — 전처리 · 모델 벤치마크 · 평가 · 피처 엔지니어링 · 오케스트레이터
- CLAUDE.md에 데이터 경로 · 모델 · freeze 수치 · 문서 경로를 컨텍스트로 주입
- 새 세션도 CLAUDE.md만 읽으면 현 상태를 즉시 파악

```
# CLAUDE.md — 프로젝트 컨텍스트
data:  D:/ais_data/.../3yr.csv
model: dcdetect (20 features)
freeze: FP=1%  91.9%
rule:  docs in sync before push

$ claude
> 오케스트레이터에 약세 진단
  노드를 추가해줘

✓ orchestrator.py 작성됨
```

② 분석 — claude_analyze

매 단계 종료 시 LLM이 결과를 5개 섹션으로 자동 분석 (pipeline_steps.py)

1. 결과 평가	탐지율 · 목적점수를 수치로 해석
2. 원인 · 근거	왜 올랐는지 · 어떤 시나리오가 움직였는지
3. 약세 진단	낮은 구간(저속 · 정박 등) 식별
4. 다음 전략	어떤 피처 방향을 시도할지 제안
5. 판정	continue · retry · stop 권고

③ 피처 발명 — n_recommend

- 약세 시나리오를 LLM이 읽고 새 파생피처(lambda)를 발명

핵심 원칙 LLM은 **제안** · 데이터가 **+3.0pp 기준**으로 채택 · 사람이 배포 승인 — 무비판 LLM 신뢰가 아닌 자동화 + 정량 검증 하이브리드

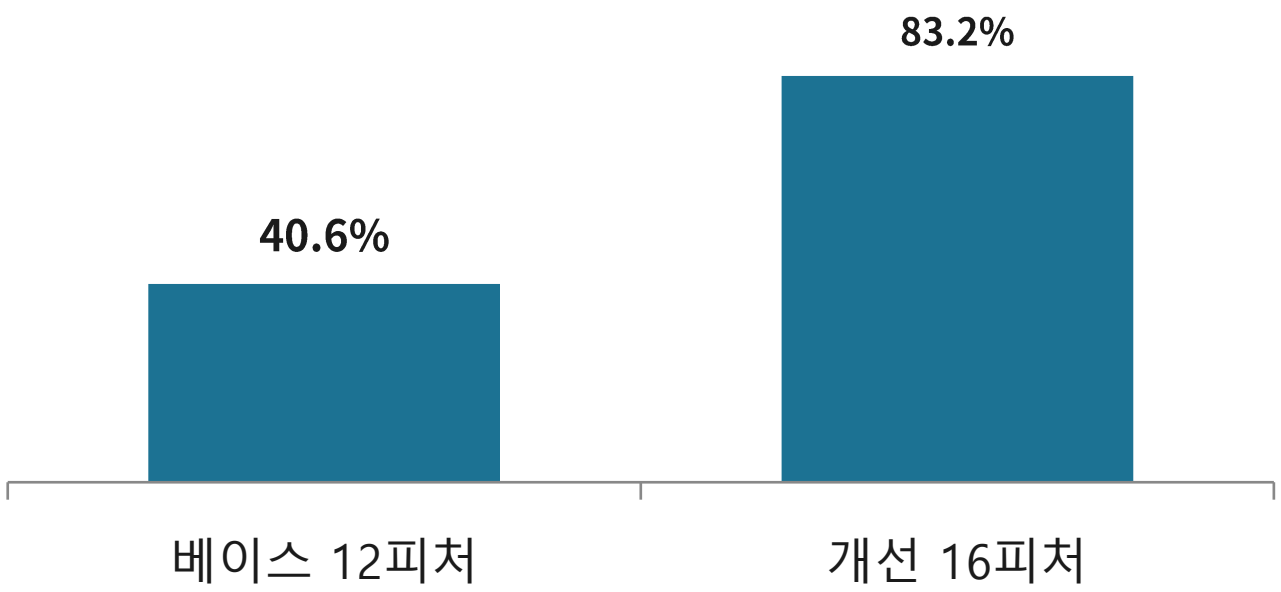
```
# dynamic_candidates.py
# (LLM이 발명한 후보)

"anchor_suspicion":
  lambda s: (
    (s.sog < 1.0)
    * s.heading.diff()
        .abs() )

# 저속 + 선수각 변화 =
# 정박 위장 포착
```

탐지율 개선 LLM 자동화 전 -> 후

FP = 1% 탐지율



평균 탐지율 (전체 시나리오)

40.6%

베이스 12피처

↓ 자동 Greedy 피처 엔지니어링

83.2%

개선 16피처 (배포 모델)

최종 탐지율

FP = 1%

83.2%

결과 요약

- 오탐(FP) 1% 이하 환경에서 공격 86.7% 탐지
- 16개 핵심 피처만으로 경량·실시간 동작
- 32개 공격 시나리오 평가로 안정성 검증

현재 모델 — dcdetect 16피처 (2026-06), 현재 배포 중

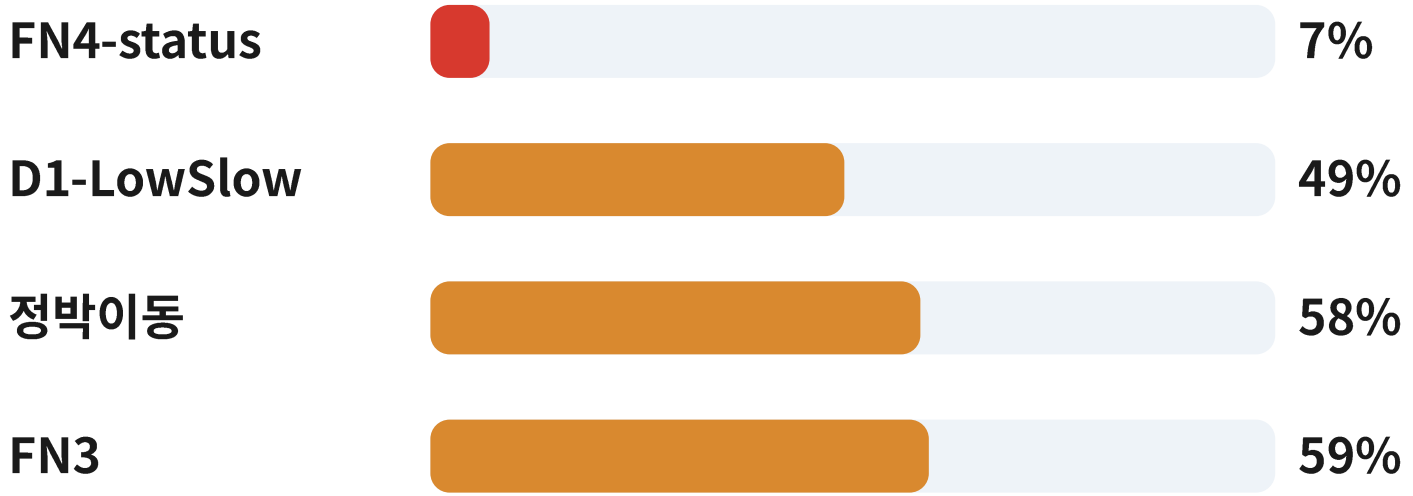
32 시나리오 평가 — 강세와 약세

22 / 32

시나리오에서
89.5% 이상 탐지

holdout(F · G) 포함
대부분의 공격을 강하게 탐지

약세 시나리오 (정직한 한계) — FP = 1%



원인 — 정상 분포와 통계적으로 중첩 (정상과 닮은 이상)

시연

GH-JBPirateKing

현재 한계

분포 중첩 시나리오

저속 · 정박처럼 정상과 통계적으로 유사한 이상은 재구성 오차만으로 분리가 어려움 (D1 · FN4 등)

비지도 단일 레이어의 제약

정상 패턴 학습 기반이라 정상처럼 보이는 공격에 본질적으로 취약 — 모델 결함이 아닌 한계

향후 과제

룰&ML 하이브리드 탐지 기반 개선

약세 시나리오를 룰 기반 탐지 하이브리드로 개선

평가 기준 재설계 및 시나리오 확장

신규 공격 시나리오 추가 및 평가 기준 재정비

향후 과제 – 사업화

1

교내·지역 창업 인프라 활용

교내 창업지원단·창업동아리, 멘토링 연계

2

창업 경진대회로 검증

교외 창업 경진대회로 검증 시도

3

청년 창업 지원사업 탐색

예비창업패키지·창업중심대학 등 청년 대상 사업화 자금·멘토링 신청

감사합니다

Q & A

팀 JB-Pirate-King · 머신러닝 기반 선박 AIS IDS