

< 모델 정의서 >

작성일자 : 2026-01-05

---

# 모델 정의서

## 용왕의 한 수

이미지 기반 낚시 환경 분석과 음성 기반 조행 기록을 결합한 스마트 낚시 플랫폼

소속 : LG U+ why not sw camp 8기

팀명 : NAVIS

팀장 : 권혜린

팀원 : 박시은, 오원석, 윤예빈, 조승아

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

---

<b>1. 프로젝트 개요 (통합)</b>	<b>3</b>
1.1 전체 시스템 구성도	3
<b>2. 개발 및 학습 환경 (통합)</b>	<b>4</b>
2.1 하드웨어 및 OS	4
2.2 사용 언어 및 라이브러리	5
<b>3. 데이터 정의 및 처리 (데이터 유형별 그룹화)</b>	<b>6</b>
3.1 이미지 객체 탐지(yolo)	6
3.1.1 데이터 구성	6
3.1.2 이미지 분류 데이터셋 (for CNN - 물색/예기색)	8
3.2.1 데이터 구성	8
3.2.2 데이터 전처리 및 파이프라인 (Preprocessing & Pipeline)	9
3.2.3 데이터 라벨링 (Labeling)	9
3.3 텍스트 데이터셋 (sLLM용)	10
3.3.1 데이터 출처 및 수집	10
3.3.2 데이터 전처리 및 구축 파이프라인 (Data Pipeline)	10
<b>4. 모델 구조 및 기술 사양 (핵심 - 모델별 서술)</b>	<b>11</b>
4.1 객체 탐지 모델 (YOLO)	11
4.2 물색 분류 모델 (Vision-Centric)	14
4.3 예기색 추천 모델 (Hybrid Multi-modal)	16
4.4 LoRA튜닝(Fine-tuning) 방식을 채택한 sllm	20
<b>5. 향후 계획 (통합)</b>	<b>23</b>

---

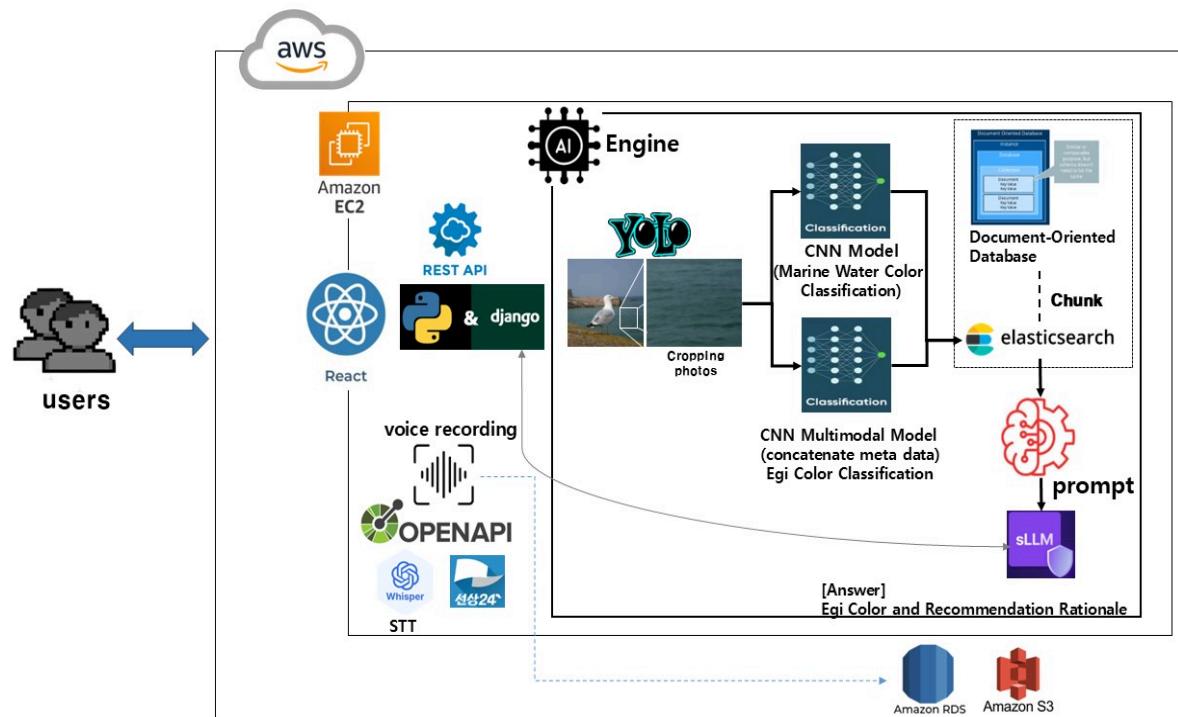
&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

## 1. 프로젝트 개요 (통합)

### 1.1 전체 시스템 구성도

- 예: [입력] -> YOLO(탐지) -> CNN(색상분류) -> [데이터 결합] -> sLLM(추천) -> [출력]



< 모델 정의서 >

작성일자 : 2026-01-05

---

## 2. 개발 및 학습 환경 (통합)

### 2.1 하드웨어 및 OS

- **Operating System:** Windows x64 (PowerShell Environment)
- **GPU (Graphic Processing Unit):** NVIDIA GeForce RTX 4060 Laptop GPU
  - **Architecture:** Ada Lovelace
  - **VRAM:** 8 GB (GDDR6)
- **CUDA Environment:**
  - **NVIDIA Driver Version:** 566.14 (Supports up to CUDA 12.7)
  - **CUDA Toolkit (Runtime):** 11.8 (Applied via PyTorch Internal)

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

## 2.2 사용 언어 및 라이브러리

- **Language:** Python 3.9.25 **Virtual Environment:** Conda ([navisenv](#))
- **Key Libraries:**
  - **Deep Learning & AI:**
    - **PyTorch:** [2.7.1+cu118](#) (Computed on CUDA 11.8)
    - **Transformers:** [4.57.3](#) (Hugging Face)
    - **Ultralytics (YOLO):** [8.3.241](#)
    - **Torchvision:** [0.22.1](#) / **Torchaudio:** [2.7.1](#)
  - **Backend & API:**
    - **Django:** [4.2](#) (LTS Version)
    - **Django REST Framework:** [3.16.1](#)
    - **Gunicorn:** [23.0.0](#) (WSGI Server)
  - **Data Processing & CV:**
    - **OpenCV-Python:** [4.12.0](#) (Image Preprocessing)
    - **Pandas:** [2.3.3](#) / **NumPy:** [2.0.2](#) (Data Analysis)
    - **Scikit-learn:** [1.6.1](#)
  - **Infrastructure & Database:**
    - **Boto3:** [1.42.2](#) (AWS SDK for S3/EC2)
    - **Elasticsearch:** [9.1.2](#) (Search Engine)
    - **MySQL Client:** [2.2.7](#)

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

---

### 3. 데이터 정의 및 처리 (데이터 유형별 그룹화)

#### 3.1 이미지 객체 탐지(yolo)

##### 3.1.1 데이터 구성

- 원천 데이터

- 수집 대상: 공개된 유튜브 라이브 영상 56건
- 선별 기준 (**Data Mining**): 유튜브에 공개된 라이브를 20분 간격으로 시간의 흐름 반영, 유튜브 라이브 조황 위치를 서해, 남해에서도 다양하게 하여 다양한 환경 변수 고려함.
- 데이터 규모: 약 800장 확보 → 200장은 배경 노이즈 제거 및 데이터 검수 과정에서 제외되어 총 600장 데이터 확보
- 데이터 전처리: YOLOv8 표준 입력 해상도인 640x640 규격으로 통일

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

## ■ 데이터 라벨링

- Bounding Box 사용하여 600장의 다양한 낚시 환경에서 촬영된 사진을 라벨링
- water 클래스를 라벨링한 600장의 데이터를 yolo 학습을 위해 dataset split의 비율은 train data : 7, valid set : 2, test set : 1 비율로 **Hold-Out** 방식 사용

[데이터 구조]

```
water_dataset
├── test
│   ├── images
│   └── labels
├── train
│   ├── images
│   └── labels
└── valid
    ├── images
    └── labels
    data.yaml
```

## ■ 데이터 증강

- **epoch:** 100
- **batch\_size:** 16
- **img\_size:** 640
- **lr0(초기 학습률):** 0.01

## ■ 주요 하이퍼파라미터 (Technical Specs):

- **Mosaic:** 1.0
- **Mixup:** 0.0
- **Flip Left-right:** 0.5
- **HSV-H:** 0.015

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

### 3.2 이미지 분류 데이터셋 (for CNN - 물색/에기색)

#### 3.2.1 데이터 구성

- 원천 데이터
  - 수집 대상: 쭈꾸미, 갑오징어 낚시 유튜브 라이브 스트리밍 영상 41건
  - 선별 기준 (Data Mining): 전체 영상 중 사전 정리한 '조과 성공 키워드(잡았다, 하트 등)' 필터링과 '문맥 분석'을 통해 유효 타임스탬프를 추출, 해당 시점의 프레임만을 선별하여 구축함.
  - 데이터 규모: 바다 물색 이미지 및 사용 예기 색상 쌍 총 1,000세트
- 학습용 데이터셋 분할:
  - (1) 물색 분류용: YOLO가 탐지하여 Crop한 바다 이미지 + 물색 레이블 (1,424장)
  - (2) 에기색 분류용: YOLO가 탐지하여 Crop한 바다 이미지 + 에기색 레이블 (1,000장)
  - 데이터 구조: [이미지 파일 1개] ↔ [레이블 쌍 (물색: 탁함, 예기: 빨강)] 매핑 구조

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

### 3.2.2 데이터 전처리 및 파이프라인 (Preprocessing & Pipeline)

- **파이프라인 연계 (Key Point):**
  - **물색 모델:** 전체 이미지 리사이징 (224x224) 및 정규화(Normalization) 적용
  - **예기색 모델: YOLO 객체 탐지 모델을 선행 수행하여** 검출된 예기 영역(BBox)을 **Auto-Crop** 후 학습 데이터로 가공 (배경 노이즈 제거 효과)
- **이미지 증강 (Augmentation):** 데이터 수량 한계(1,000장)를 극복하기 위해 학습 시 실시간 증강 적용 (Rotate, Random Brightness, Horizontal Flip)

### 3.2.3 데이터 라벨링 (Labeling)

- **형식:** 이미지 파일명과 매칭되는 CSV 메타데이터 파일 관리
- **구조 예시:**

Image_ID	File_Path	Label_Water (물색)	Label_Lure (예기색)
img_001	./data/raw/001.jpg	탁함 (Murky)	고추장 (Red/Green)
img_002	./data/raw/002.jpg	맑음 (Clear)	수박 (Green/Black)

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

### 3.3 텍스트 데이터셋 (sLLM용)

#### 3.3.1 데이터 출처 및 수집

- **출처:** 낚시 전문 유튜브 채널 및 관련 커뮤니티
- **수집 대상:** 현장 낚시 가이드, 특정 상황에서의 추천하는 에기색, 조황 시 확인해야 할 날씨 항목이 포함된 영상 스크립트 및 게시글
- **데이터 규모:**
  - **원시 데이터(Raw):** 스크립트 파일 60건
  - **처리 데이터:** 벡터화 가능한 문단 단위 청크(Chunk) 약 3,062개

#### 3.3.2 데이터 전처리 및 구축 파이프라인 (Data Pipeline)

- **청킹(Chunking):** 검색 정확도 높이기 위해 의미 단위로 텍스트 분할 (예: 500 토큰 단위)
- **동의어 전처리:** 명사 기반으로 색인어를 구분하기에, 동일한 의미이지만, 다르게 사용되는 단어(은어) 정리
- **Elasticsearch:** 원본 데이터를 문장 단위로 토큰화(Tokenization) 및 청킹하여 Elasticsearch에 3,620개 문장을 텍스트 색인

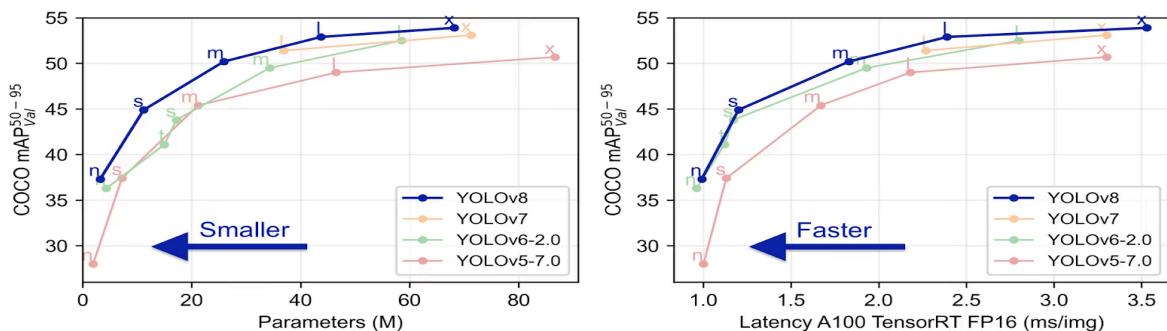
&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

## 4. 모델 구조 및 기술 사양 (핵심 - 모델별 서술)

### 4.1 객체 탐지 모델 (YOLO)

- 모델: YOLOv8n(Nano)
- 선정 이유 :** 실시간으로 입력되는 선상(배)에서 찍은 사진은 하늘, 배, 사람, 파도 거품 등 다양한 요소들이 함께 포함되어 있어 물색을 예측할 때 **배경 노이즈**가 발생함. 이 상태에서 CNN을 적용하면 Pooling 과정에서 중요한 물색 특징까지 함께 손실 될 수 있어 **분류 정확도가 떨어질 수 있음**. 이를 방지하고자 CNN 과정을 하기 전에 water 영역만 잘라내는 과정을 **crop**이라고 하며, crop된 이미지를 입력으로 사용하면 불필요한 배경이 제거되어 물색에 대한 중요 정보가 보존됨.
- YOLOv8 모델 중 Nano 모델 선정 이유:** water(물)은 형태가 단순하고, 특징이 명확하여 최신 기술인 YOLOv11의 복잡한 연산 레이어보다 YOLOv8n은 가벼운 모델이기에 **단순한 특징 추출 능력**에 적합함.  
또한 YOLOv8모델 중에서 Nano모델은 YOLOv8s(Small), YOLOv8m(medium)과 비교했을 때 가장 **작고 가벼운 모델**이라 압도적인 속도와 모바일 기기와 같은 저사양 임베디드 장치에서 **실시간 추론이 가능해** 핵심기능인 휴대폰을 이용해 실시간으로 사진을 찍는 과정과 유사해 YOLOv8n을 사용함.



&lt; 모델 정의서 &gt;

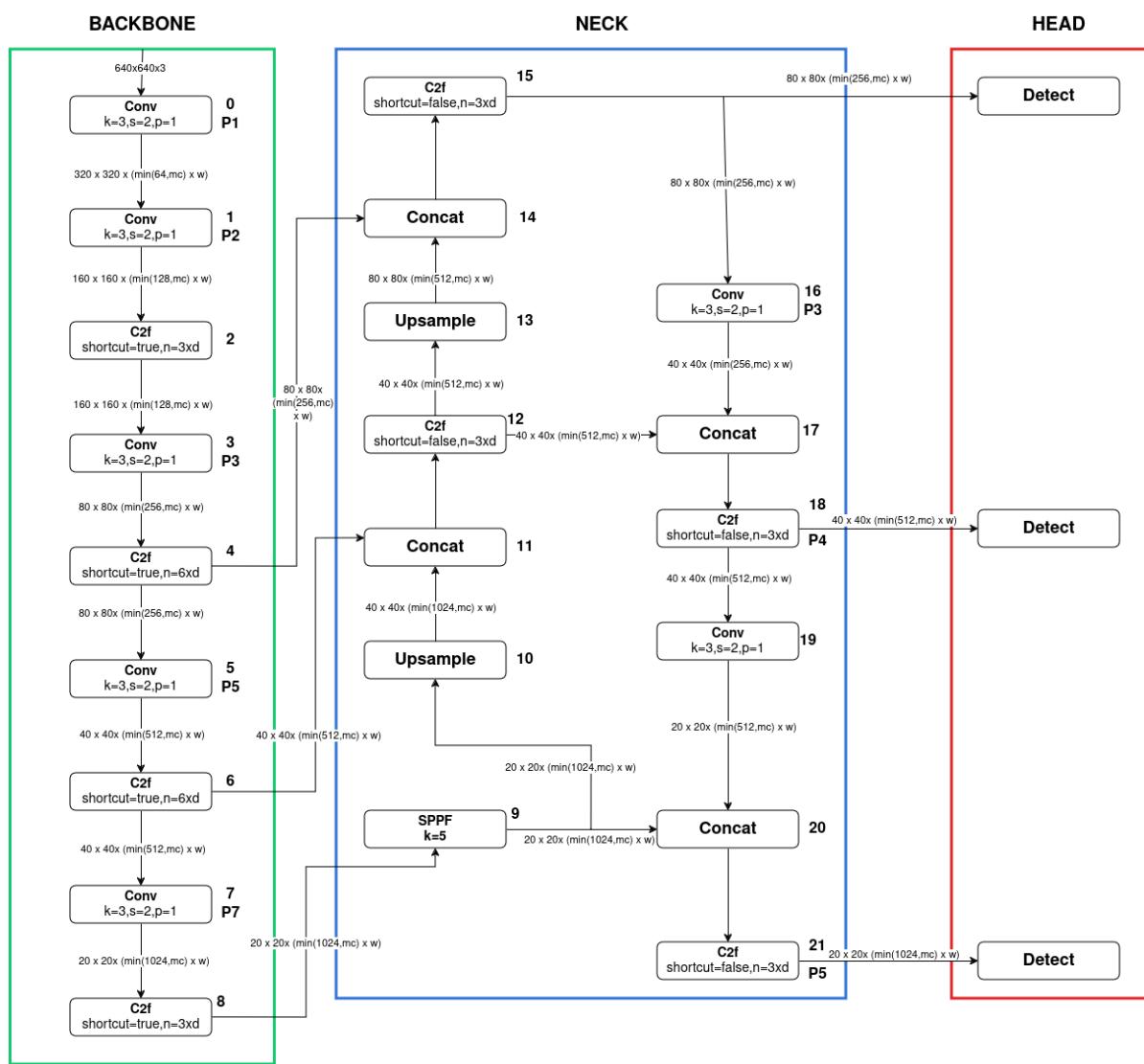
작성일자 : 2026-01-05

- 
- 구조: BACKBONE → NECK → HEAD

구성 요소	역할	역할
Backbone	특징 추출(SPPF 포함)	사진 전체를 훑어 객체 고유의 색상, 물결, 투명도 같은 <b>핵심 정보</b> 찾기
Neck	특징 결합	멀리 있는 객체 정보 <b>추출</b>
Head	위치 및 종류 판별	분석된 데이터 바탕으로 화면에 Bound Box 통해 <b>객체 탐지</b>

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05



&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

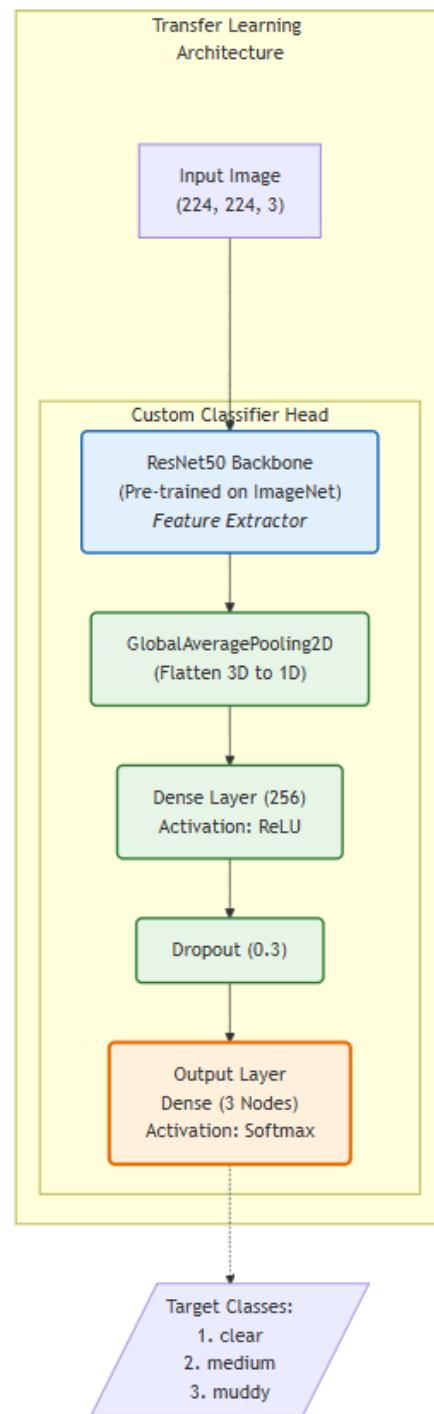
## 4.2 물색 분류 모델 (Vision-Centric)

- **모델:** ResNet50 (Pre-trained on ImageNet)
- **선정 이유:** 물색 판별은 미세한 질감과 색조의 차이를 구분해야 하므로, 깊은 레이어(Deep Layer)와 통해 기울기 소실을 최소화하여 고수준의 **특징 추출**이 가능한 ResNet50을 선택함.
- **구조:** [Input 224x224] → [ResNet50] → [GAP] → [Softmax]

계층(Layer)	구성 및 사양	주요 역할
<b>Input Layer</b>	(224, 224, 3), RGB Image	입력 이미지의 해상도를 표준화하여 모델 연산 효율성 확보
<b>Backbone</b>	<b>ResNet50 (Pre-trained)</b>	<b>Feature Extractor</b> 으로 선행 학습된 가중치를 활용하여 수면의 질감, 예기의 형태 등 고차원 특징 추출
<b>Pooling Layer</b>	<b>GlobalAveragePooling2D(GAP)</b>	3D 특징 지도를 1D 벡터로 압축(Flatten)하여 공간 정보의 핵심 요약
<b>Dense Layer</b>	<b>256 Nodes (ReLU)</b>	추출된 시각적인 특징(색상, 투명도, 물결 등)들을 256개의 뉴런이 상호 결합하여 분석 후 추론
<b>Regularization</b>	<b>Dropout (0.3)</b>	학습 시 뉴런의 30%를 무작위로 제외하여 <b>과적합(Overfitting)</b> 방지
<b>Output Layer</b>	<b>3 Nodes (Softmax)</b>	최종 타겟 클래스(clear, medium, muddy)별 확률값 산출

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05



&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

---

### 4.3 예기색 추천 모델 (Hybrid Multi-modal)

- **모델:** Custom Multi-input CNN (Image + Tabular)
- **선정 이유:** 최적의 예기는 단순히 예기 사진만으로 결정되는 것이 아니라, **당일의 '풍향'**, **'수온'**, **'물때'** 등 환경 변수와의 복합적인 상관관계에 의해 결정됨. 따라서 정형 데이터(CSV)와 비정형 데이터(이미지)를 동시에 학습할 수 있는 하이브리드 구조를 자체 설계함.
- **네트워크 구조 다이어그램:**
  - [이미지(64px)] → [Conv Block 3단] ↴
  - [환경변수(14개)] → [Dense Block 2단] → [Concatenate] → [Dense] → [Output]

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

순서	구분	레이어 및 구성	설정 값 (Hyperparameter)	역할 및 세부 설명
1	비정형 데이터 (Image Branch)	<b>Input Image</b>	(64, 64, 3)	64x64 해상도의 RGB 컬러 이미지(ROI)를 입력으로 받음.
2		<b>Conv2D 32</b>	Filter: 32개 BN(Batch Normalization) MaxPool	이미지의 선, 점 등 기초적인 저수준 특징(Low-level) 추출 및 연산 최적화.
3		<b>Conv2D 64</b>	Filter: 64개 BN MaxPool	물결 패턴, 사물의 윤곽 등 중수준 특징(Mid-level) 추출.
4		<b>Conv2D 128</b>	Filter: 128개 BN MaxPool	물색의 깊이, 복잡한 질감 등 고수준 특징(High-level) 완성.
5		<b>Flatten</b>	3D(Tensor) → 1D(Vector)	추출된 공간 특징 지도를 1차원 수치 리스트로 변환하여 논리 연산 준비.
6		<b>Dense 128</b>	128 Nodes Dropout(0.3)	시각 정보를 128개 차원으로 종합하며, 드롭아웃을 통해 과적합(Overfitting) 방지.
7	정형 데이터 (Meta Branch)	<b>Input Meta</b>	Env Data (수온, 수심 등)	기상 API 등으로부터 획득한 수치형 환경 데이터를 입력으로 받음.

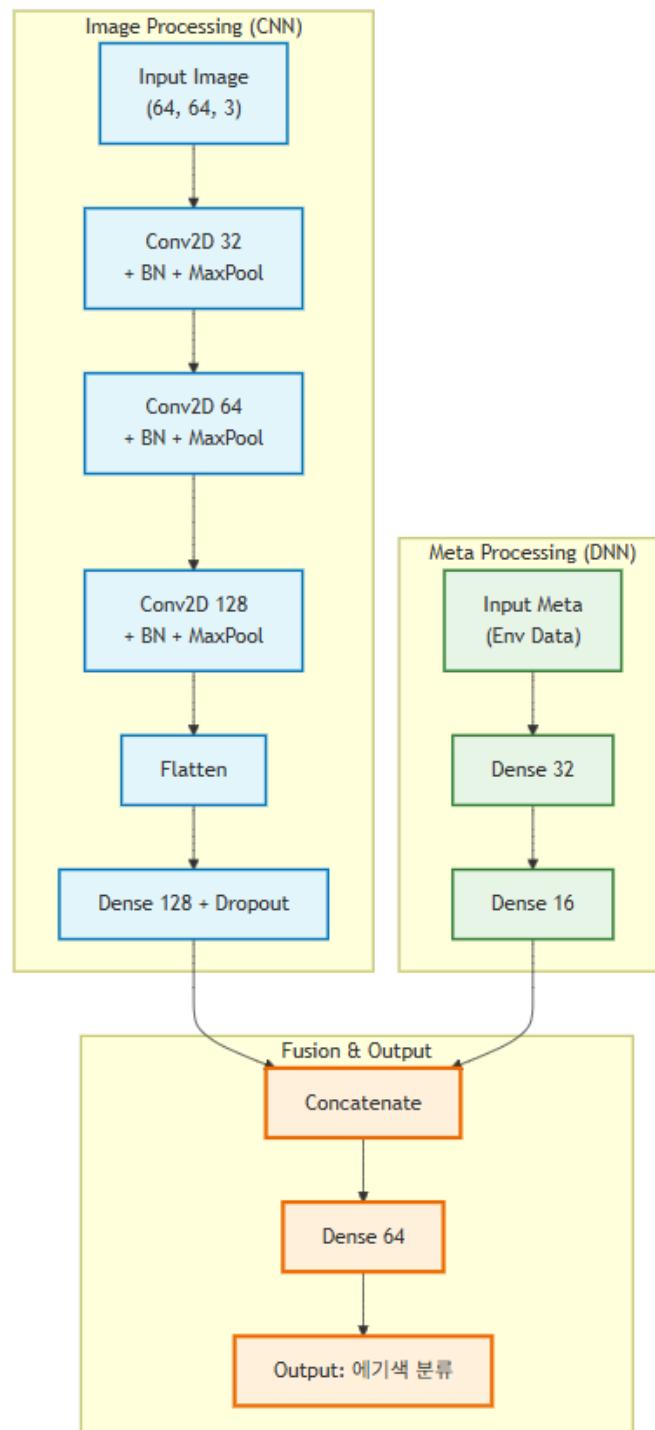
&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

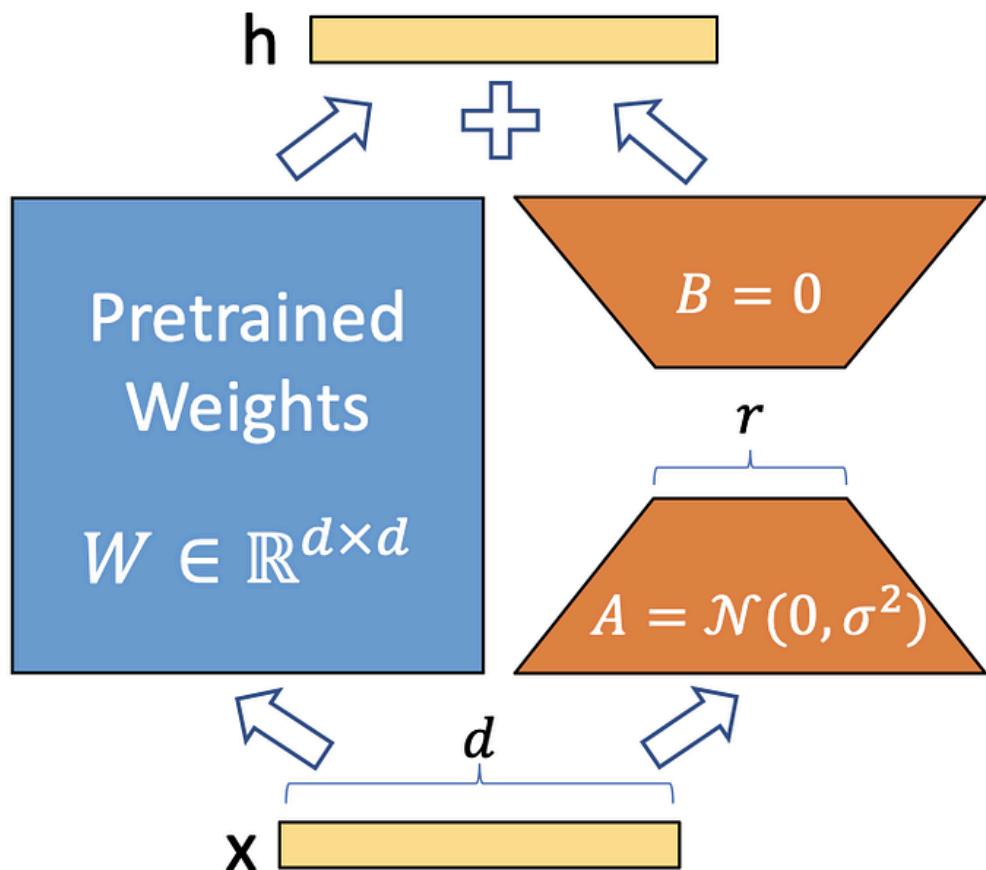
<b>8</b>		<b>Dense 32</b>	32 Nodes	환경 변수 간의 상관관계를 분석하여 첫 번째 특징 벡터 추출.
<b>9</b>		<b>Dense 16</b>	16 Nodes	데이터 핵심 요약 및 이미지 정보와의 결합을 위한 차원 최적화.
<b>10</b>	데이터 융합 (Fusion Layer)	<b>Concatenate</b>	128(CNN) + 16(DNN)	[융합] 시각 특징과 환경 컨텍스트를 하나로 합쳐 144차원의 통합 정보 생성.
<b>11</b>		<b>Dense 64</b>	64 Nodes (ReLU)	[추론] 결합된 다중 정보를 바탕으로 현재 상황에 최적인 에기색을 논리적으로 추론.
<b>12</b>		<b>Output</b>	Dense (Softmax)	[결론] 최종 에기색 분류 확률값을 산출하여 가장 적합한 색상을 제안함.

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05



#### 4.4 LoRA튜닝(Fine-tuning) 방식을 채택한 sllm



&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

- **Base Model(Pretrained Model):** [EleutherAI/polyglot-ko-1.3b](#)

- **선정 이유:** EleutherAI/polyglot-ko-1.3B는 저사양 임베디드 장치 및 모바일 환경에서의 실시간 추론 필요로 함. 한국어 특화 모델이기에 효율성이 높고, 3B, 8B 모델에 비해 토큰 생성 속도가 빨라 응답 지연 시간을 최소화 할 수 있음.

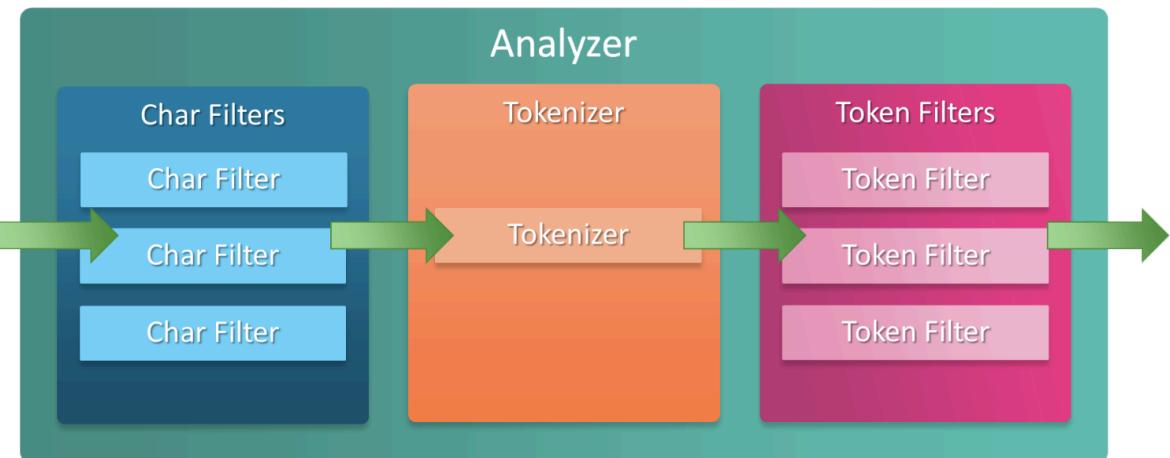
모델명	EleutherAI/ polyglot-ko-1.3b	meta-llama/ Llama-3.2-3B-Instruct
파라미터 수	1.3B	3B
학습 언어	한국어 특화(데이터 90% 이상)	다국어(영어 중심, 한국어 포함 8개국어)
컨텍스트 윈도우	2,048(2k)	128,000(128k)
아키텍처	GPT-NeoX기반	Transformer

- **Fine-tuning (LoRA):**

- **튜닝 목적:** 낚시 도메인 특화 용어(물색, 애기 등) 이해도 제고 및 논리적 추천 답변 생성 능력 배양
- **학습 데이터:** 1,000 건의 Instruction Tuning 데이터셋
  - **구성:** [물색 - 애기 색상 - 전문가 Q&A] Pair
- **학습 전략:** LoRA (Low-Rank Adaptation) 기법을 통한 전체 파라미터 중 소수(Adapter)만 미세 조정하여 학습 효율 및 이식성 확보
- **주요 하이퍼파라미터 (Technical Specs):**
  - **Rank (r):** 16 (어댑터 가중치 행렬 크기)
  - **LoRA Alpha:** 32 (학습 영향도 조절 계수)
  - **Dropout:** 0.05 (과적합 방지를 위한 정규화 비율)
  - **Epochs:** 512

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05



- **Retriever (정보 검색 모듈):**

- **엔진:** Elasticsearch (Sparse Retriever + BM25)

- **선정 이유:**

- 낚시 영역에서 자주 사용되는 키워드의 정밀한 매칭을 위해 ElasticSearch를 사용함.
    - 자주 사용되는 키워드는 전처리 과정을 통해 자주 사용되는 단어들을 분석하여 Sparse Retriever의 단점을 보완함.
    - Analyzer(분석기)구조의 핵심 기능인 형태소 분석기는 **OKT**와 **Kiwi** 를 통해 낚시 도메인 용어를 보존함으로써 정보 검색 시 발생할 수 있는 왜곡을 최소화 함.
    - **쿼리 파이프라인:** [CNN 결과(물색/에기색)] → [텍스트 쿼리 변환] → [스크립트 검색]
    - **Context Injection:** 관련 점수가 높은 3개의 스크립트를 프롬프트에 주입

&lt; 모델 정의서 &gt;

작성일자 : 2026-01-05

---

## 5. 향후 계획 (통합)

- yolo 성능 향상을 위한 추가 **데이터 확보 및 학습**
- 역광에 의한 반사, 파도 거품, 안개 등 바닷물을 탐지할 때 방해되는 환경 변수에 대응하기 위해, 기상 시뮬레이션 기반 증강 기법을 도입해 **탐지 강건성 확보**
- GPU 리소스 보강을 통해 차세대 모델인 Llama-3.2-3B-Instruct-Korean-Blossom 모델로 전환하여 **지시어 이행 능력과 자연어 생성 품질을 제고할 수 있음**
- 웹 크롤링을 활용해 최신 낚시 트렌드 및 전문가들의 실전 스크립트를 실시간으로 자동 수집하여 Elasticsearch 기반 지식 검색을 고도화 할 수 있음
- 추가적인 LoRA 튜닝을 통해 두족류 낚시 뿐만 아니라 **돔, 우럭, 농어 등 바다낚시로 영역 확장 가능**