

医疗 电子 健康 记录

日期: 2026-02-10 | 版本: 1.1.0

1. 医疗记录

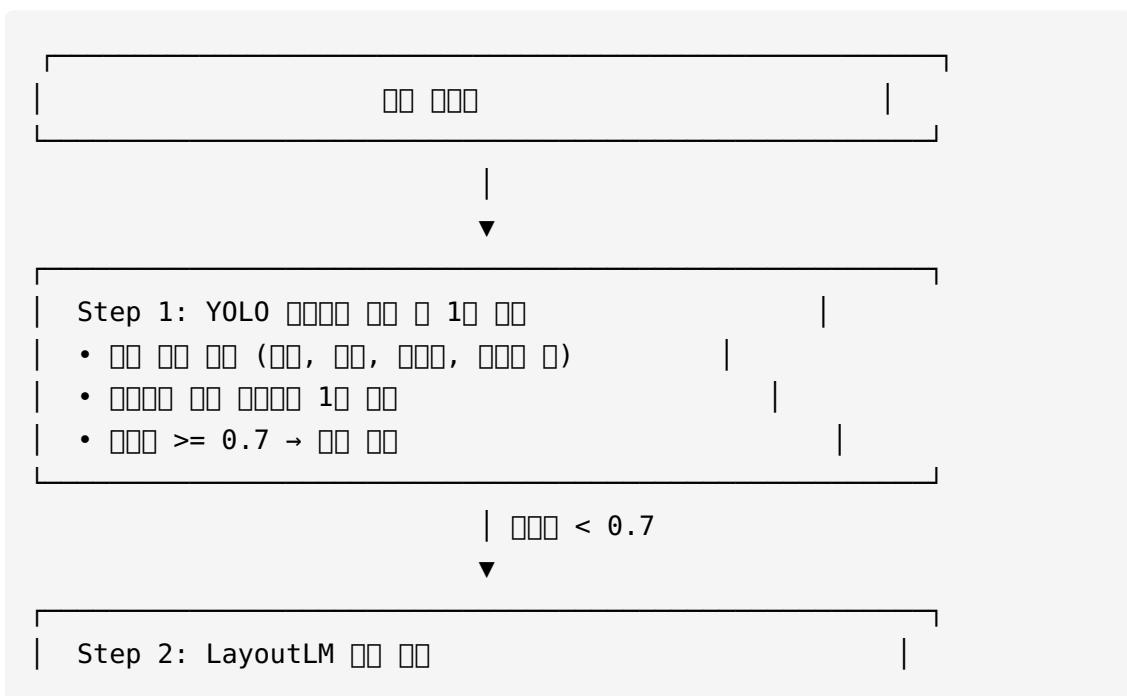
1.1 介绍

此医疗健康记录包含 3 页，每页包含 6 项医疗记录。

1.2 医疗记录 (6 项)

序号	记录类型	描述	序号	记录类型	描述
1	诊断	Diagnosis	4	住院/出院	Admission/Discharge
2	医疗意见	Medical Opinion	5	医疗收据	Medical Receipt
3	保险索赔	Insurance Claim	6	处方	Prescription

2. 医疗记录生成



- EasyOCR 用于识别文本
- 图像 + 文本框 用于识别文本
- 信心 >= 0.7 → 识别文本

| 信心 < 0.7



- Step 3: VLM 用于识别（图像）
- GPT-4V 或 Claude Vision 用于
 - 图像 用于 识别 图像

3. 识别结果

2.1 识别结果

识别结果是通过将图像输入到模型中并输出识别结果。

识别结果

- Python PIL 库可以实现识别
- 可以使用 TensorFlow、PyTorch 等框架：
- 图像，文本，表格，二维码，QR 码等

识别结果

参数	值	单位
训练集 (train)	60 × 500	张
验证集 (val)	60 × 100	张
总样本数	-	3600

2.2 识别结果

识别类型	识别方法	识别结果	识别工具
文本	EasyOCR	文本	EasyOCR
二维码	图像，文本框	二维码	图像，文本框，QR 码
表格	图像，文本框	表格	图像，文本框，QR 码
其他	图像，文本框	其他	QR 码

YOLO	YOLO	YOLO	YOLO
□□□□□	□, □□□, □□□	-	□□□, QR□□
□□□□□	□□□, □□□	□□, □□□□	QR□□
□□□	□□□, QR□□	□□, □□□□, □□	□□□

2.3 YOLO □□ □□□

- □□: YOLO □□ (□□□ + □□ □□□ □□)
- □□□ □: 6 □ 0: stamp (□□) 1: signature (□□) 2: table (□□□) 3: barcode (□□□) 4: qrcode (QR□□) 5: hospital_logo (□□ □□)

2.4 LayoutLM □□ □□□

- □□: □□□ + OCR □□□ + □□□ □□
- **OCR □□□:** □□ □□□ □□□ □□ Mock OCR □□
- □□□: "□□□", "□□□", "□□□", "□□□□", "□□□□" □
- □□□: "□□□", "□□□□", "□□□□", "□□□□" □
- □□□□□: "□□□□□", "□□□□", "□□□□", "□□□□" □
- □□□□□: "□□□□□", "□□□□", "□□", "□□□" □
- □□□□□: "□□□□□", "□□□□", "□□", "□□□" □
- □□□: "□□□", "□□□□□", "□□□", "□□□□" □

4. Step 1: YOLO □□

3.1 □□ □□

- □□□ □□: YOLOv8n (ultralytics)
- □□ □□□: 50
- □□□ □□: 640×640
- □□ □□: 16

3.2 □□ □□

```
Epoch 50/50 □□
- Box Loss: 0.5765
- Class Loss: 0.3342
- DFL Loss: 0.8823
```

3.3 ດັບ ດົກ (Validation)

ໂທ ດົກ

ໂທ	□
mAP50	99.5%
mAP50-95	96.0%
Precision	98.7%
Recall	98.8%

ໂທ AP50

ໂທ	AP50
stamp	99.5%
signature	99.5%
table	99.5%
barcode	99.5%
qrcode	99.5%
hospital_logo	99.5%

3.4 ເພີ້ມ ດົກ ດົກ

ໂທ ດົກ ດົກ

ໂທ	ໂທ	ໂທ
□□	10/10 (100%)	-
□□	0/10 (0%)	□□□□ □□
□□□□	10/10 (100%)	-
□□□□	10/10 (100%)	-
□□□□	10/10 (100%)	-
□□	10/10 (100%)	-
□□	50/60 (83.3%)	-

□□

- □□/□□ □□: □ □ □ □ □ □ □ □(□+□□□, □□ □□) □ □ YOLO □ □ □ □ □ □ □
 - □ □ □□: Step 2 (LayoutLM) □ □ □ □ □ □ □ □
-

5. Step 2: LayoutLM □□

4.1 □□ □□

- □□□ □□: microsoft/layoutlmv3-base
- □□ □□□: 5
- □□ □□: 2
- □□ □□□ □□: 512
- □□□: 5e-5 (warmup steps: 30)

4.2 □□ □□

□□□□ □□

Epoch	Train Loss	Eval Loss	Eval Accuracy
1	0.0030	0.0011	100%
2	0.0020	0.0011	100%
3	0.0014	0.0007	100%
4	0.0012	0.0006	100%
5	0.0010	0.0006	100%

4.3 □□ □□

□□ □□ □□

□□ □□	□□□
□□□	10/10 (100%)
□□□	10/10 (100%)
□□□□□	10/10 (100%)
□□□□□	10/10 (100%)

□□ □□	□□□
□□□□□	10/10 (100%)
□□□	10/10 (100%)
□□	60/60 (100%)

4.4 OCR

⠀	Mock OCR	Real OCR (EasyOCR)
⠀	10/10 (100%)	10/10 (100%)
⠀	10/10 (100%)	10/10 (100%)
⠀	10/10 (100%)	10/10 (100%)
⠀	10/10 (100%)	10/10 (100%)
⠀	10/10 (100%)	10/10 (100%)
⠀	10/10 (100%)	10/10 (100%)
⠀	10/10 (100%)	10/10 (100%)
⠀	60/60 (100%)	60/60 (100%)

- Mock OCR 100% 100% 100% EasyOCR 100% 100% 100% 100%
 - EasyOCR 100% 100% 100% 100% 100% 100% 100%

4.5 □□ □□

- $\text{Step 1}/\text{Total}$ \approx 0.99: Step 1 \approx 99% of the total 100% \approx 100%
 - $\text{Step 2}/\text{Total}$: \approx 0.99 \approx 100%
 - $\text{Mock OCR}/\text{Total}$: Mock OCR \approx 99% \approx 100% OCR \approx 100%

6. Step 1 vs Step 2

Step 1 (YOLO)	Step 2 (LayoutLM)
95.0%	EasyOCR 95% + LayoutLM 95%
95.0%	100%
95.0% / 95.0%	95%

步驟	Step 1 (YOLO)	Step 2 (LayoutLM)
時間 (毫秒)	~0.1s	~2s
效能	高精度 高耗能	低精度 低耗能

7. API 結構

6.1 API 應用

- 框架: FastAPI
- 伺服器: Uvicorn
- 介面: Swagger UI (API 文檔)

6.2 API 路由

路徑	方法	說明
/	GET	API 文檔
/health	GET	狀態監控
/classes	GET	類別清單
/classify	POST	單張圖片分類
/classify/batch	POST	多張圖片分類
/docs	GET	Swagger UI

6.3 API 版本 (v1.1)

- 是否 OCR: `use_mock_ocr` 值為 `False` 表示使用 EasyOCR
- 模型選擇: 可選 YOLO, LayoutLM, EasyOCR 三種模型
- OCR 數據: EasyOCR 提供的字串
- OCR 狀態: OCR 狀態碼為 422 表示錯誤

6.4 API 回應範例

```
{
  "predicted_class": "書",
  "confidence": 0.9972,
  "final_step": 2,
```

```

"processing_time": 4.11,
"step1_result": {
    "predicted_class": "બાળ",
    "confidence": 1.0,
    "requires_step2": false
},
"step2_result": {
    "predicted_class": "બાળ",
    "confidence": 0.9972,
    "all_probabilities": {
        "બાળ": 0.9972,
        "લોક": 0.0008,
        "બાળનીય": 0.0005,
        "બાળનીય": 0.0006,
        "બાળનીય": 0.0005,
        "બાળનીય": 0.0004
    },
    "ocr_words_count": 21,
    "mock_ocr_used": false
}
}

```

8. કોડ રિવ્યુ

```

કોડાર્થ/
├── api/
│   ├── __init__.py
│   └── app.py          # FastAPI રૂપ
├── config/
│   └── config.yaml     # કોડ રિવ્યુ
└── src/
    ├── pipeline.py      # એટ પાઠી
    ├── preprocessor/    # પ્રોસેસર (દોષ કરી શકતું)
    ├── step1_yolo/       # YOLO રૂપ રૂપ
    ├── step2_layoutlm/   # LayoutLM રૂપ રૂપ OCR
    ├── step3_vlm/        # VLM રૂપ રૂપ (placeholder)
    └── utils/            # ઉદ્દેશ્ય
└── scripts/
    ├── train_yolo.py     # YOLO રૂપ
    ├── train_layoutlm_simple.py # LayoutLM રૂપ
    ├── test_step1_classifier.py # Step 1 રૂપ
    └── test_step2_layoutlm.py  # Step 2 રૂપ

```

```
|   └── test_real_ocr.py      #  OCR vs Mock OCR の 比較
|   └── test_api.py          # API の 検証
└── data/
    ├── models/              # モデルの フォルダ
    ├── yolo_dataset/         # YOLO の データセット
    └── sample/               # サンプルの フォルダ
└── run_api.py              # API の 実行スクリプト
└── requirements.txt        # リクエスト ファイル
```

9. まとめ

8.1 ディレクトリ

```
pip install -r requirements.txt
```

8.2 API の 実装

```
python run_api.py
```

8.3 実験

```
# Step 1 の 検証
python scripts/test_step1_classifier.py

# Step 2 の 検証
python scripts/test_step2_layoutlm.py

#  OCR vs Mock OCR の 比較
python scripts/test_real_ocr.py

# API の 検証
python scripts/test_api.py
```

10.

9.1 //

- [] Step 3 (VLM) ☐: GPT-4V ☐ Claude Vision ☐
 - [x] ~~☐ OCR ☐~~: EasyOCR ☐ ☐ (v1.1), ☐ 100% ☐ ☐ ☐
 - [] ☐☐☐☐☐☐☐☐☐: ☐☐☐, ☐☐☐☐☐

9.2 |||

- []       
 - []       
 - [] Docker   

9.3

- []    
 - []  
 - []     (AWS/GCP)

11

3 .

1. **YOLO**: mAP50 99.5%
2. **LayoutLM**: 100%
3. **beit**: Step 2
4. **OCR**: EasyOCR
5. **REST API**: FastAPI
6. **UI**: 100%

□□□: Claude Opus 4.6 (Co-authored with BongwooChoi)