HEXING

Intelligence and Innovation Driving Results

AI Server Maintenance Manual

Version 1.0.0

Qiao Lei

August 19, 2025

Contents

Abstract

This article delivers comprehensive user guides and best-practice maintenance recommendations for the array of AI-enabled services hosted on the test center's AI server. It is crafted for system integrators and operations staff, equipping them to efficiently support these systems and to extend the platform with advanced applications.

Keywords:

AI, LLM, Open WebUI, ollama, Ragflow, Dify, Agent, OCR, PaddlePaddle, AI4Test, Common Language .....

1 Server Information

1.1 Hardware basic information

- **Type:** DELL Precision 3660 Workstation
- **Service code:** 35K0L24
- **CPU:** 13th Gen Intel(R) Core(TM) i9-13900K
- **Video Card:** NVIDIA GeForce RTX 4090 Memory: 24G
- **Memory:** 128G
- **Hard disk 1:** nvme0n1 1T (NVMe SSD)
- **Hard disk 2:** sda 4T

1.2 Login Address

- **Server address:** 172.16.33.244
- **Admin User name:** Administrator, **Password:** hexing@2025
- **User name** : ems, **Password:** sme8003

2 AI Related System

2.1 Ollama - AI local LLM inference server

How to **install** the ollama on Ubuntu ?

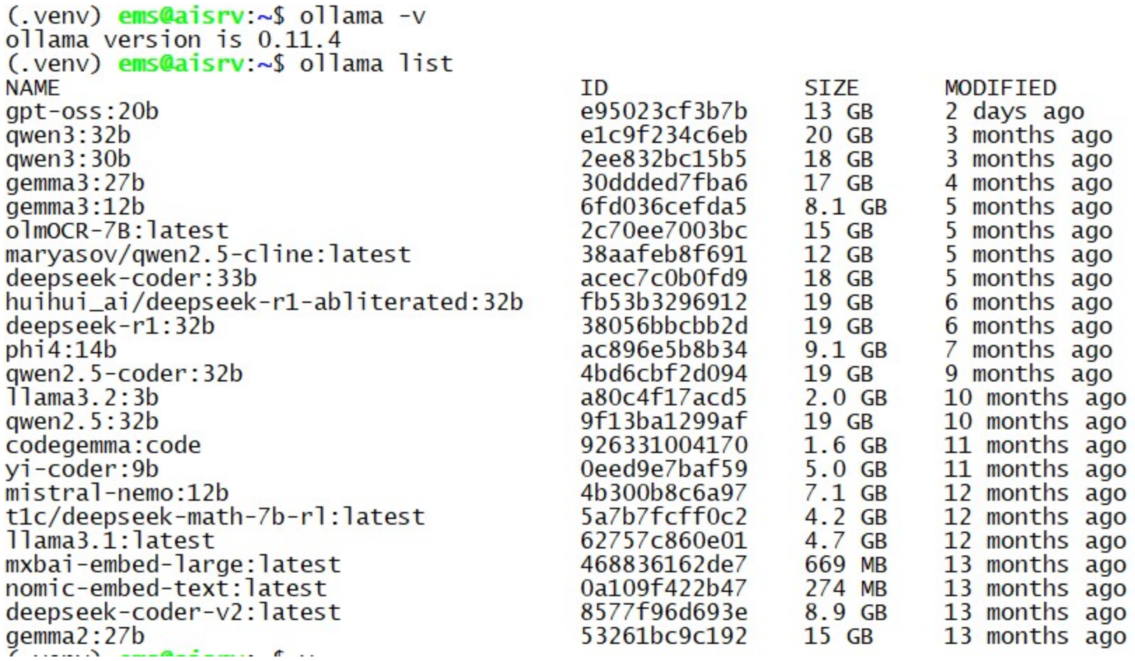
```
# download the installation package
curl -L <https://ollama.com/download/ollama-linux-amd64.tgz> -o ollama-linux-amd64.tgz

# uncompress the package
cd /home/user/tools/ollama
```

```
gunzip ollama-linux-amd64.tgz
tar xvf ollama-linux-amd64.tar

# run the ollama service, also can configure to the system startup
    service
cd ollama/bin
./ollama serve
```

By typing in the browser"**http://172.16.33.244:11434**" to check if Ollama’s service is working properly

if display "**Ollama is running**" that means Ollama is running well.

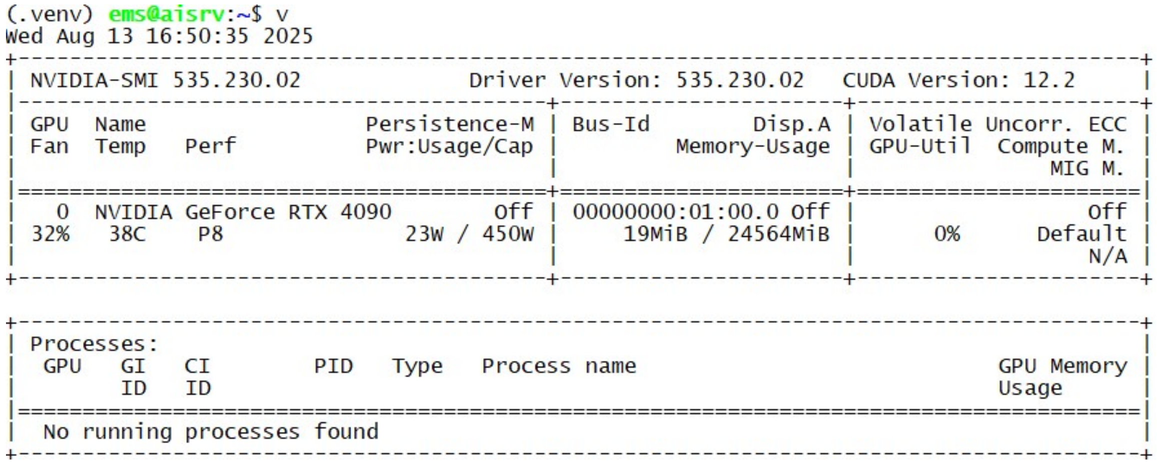
After “**ollama pull**” command, can use “**ollama list**” to check the current models.



```
(.venv) ems@aisrv:~$ ollama -v
ollama version is 0.11.4
(.venv) ems@aisrv:~$ ollama list
NAME                                     ID              SIZE      MODIFIED
gpt-oss:20b                              e95023cf3b7b    13 GB     2 days ago
qwen3:32b                                e1c9f234c6eb    20 GB     3 months ago
qwen3:30b                                2ee832bc15b5    18 GB     3 months ago
gemma3:27b                               30ddded7fba6    17 GB     4 months ago
gemma3:12b                               6fd036cefda5    8.1 GB    5 months ago
olmOCR-7B:latest                         2c70ee7003bc    15 GB     5 months ago
maryasov/qwen2.5-cline:latest            38aafeb8f691    12 GB     5 months ago
deepseek-coder:33b                       acec7c0b0fd9    18 GB     5 months ago
huihui_ai/deepseek-r1-abliterated:32b    fb53b3296912    19 GB     6 months ago
deepseek-r1:32b                          38056bbcbb2d    19 GB     6 months ago
phi4:14b                                 ac896e5b8b34    9.1 GB    7 months ago
qwen2.5-coder:32b                        4bd6cbf2d094    19 GB     9 months ago
llama3.2:3b                              a80c4f17acd5    2.0 GB    10 months ago
qwen2.5:32b                              9f13ba1299af    19 GB     10 months ago
codegemma:code                           926331004170    1.6 GB    11 months ago
yi-coder:9b                              0eed9e7baf59    5.0 GB    11 months ago
mistral-nemo:12b                         4b300b8c6a97    7.1 GB    12 months ago
t1c/deepseek-math-7b-rl:latest           5a7b7fcff0c2    4.2 GB    12 months ago
llama3.1:latest                          62757c860e01    4.7 GB    12 months ago
mxbai-embed-large:latest                 468836162de7    669 MB    13 months ago
nomic-embed-text:latest                  0a109f422b47    274 MB    13 months ago
deepseek-coder-v2:latest                 8577f96d693e    8.9 GB    13 months ago
gemma2:27b                               53261bc9c192    15 GB     13 months ago
```

Figure 1: All the available LLMs

By "**nvidia-smi**" to check video card information



```
(.venv) ems@aisrv:~$ v
Wed Aug 13 16:50:35 2025
+---------------------------------------------------------------------------------------+
| NVIDIA-SMI 535.230.02             Driver Version: 535.230.02   CUDA Version: 12.2     |
|-----------------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|=========================================+======================+======================|
|   0  NVIDIA GeForce RTX 4090        Off | 00000000:01:00.0 Off |                  Off |
| 32%   38C    P8              23W / 450W |     19MiB / 24564MiB |      0%      Default |
|                                         |                      |                  N/A |
+-----------------------------------------+----------------------+----------------------+

+---------------------------------------------------------------------------------------+
| Processes:                                                                            |
|  GPU   GI   CI        PID   Type   Process name                            GPU Memory |
|        ID   ID                                                             Usage      |
|=======================================================================================|
|  No running processes found                                                           |
+---------------------------------------------------------------------------------------+
```

Figure 2: video card information

2.2 Open WebUI - web interface for LLM chat

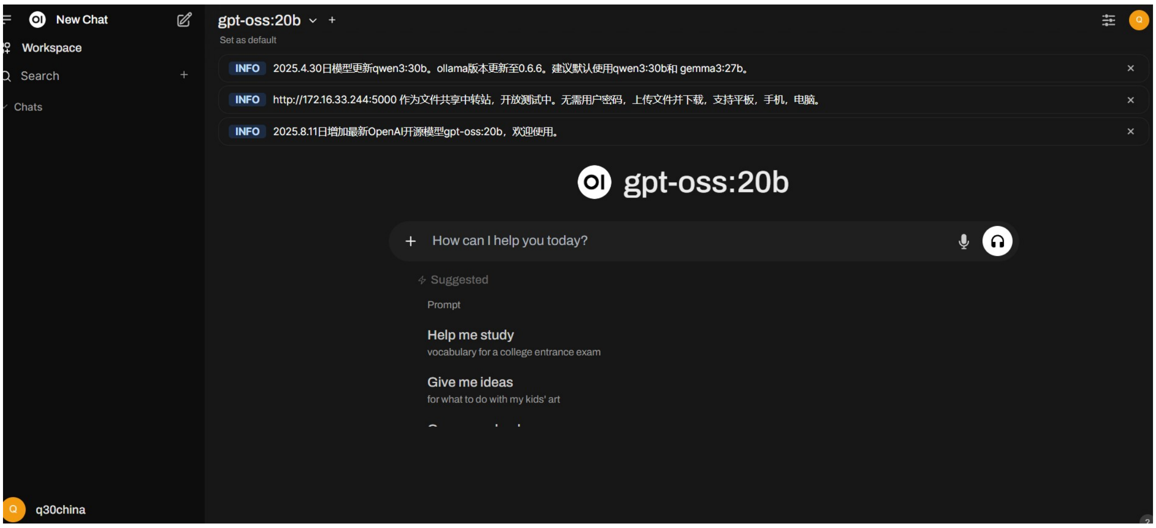
Run **Open WebUI** via Docker, and be sure to use the **CUDA-enabled** version because it needs to run on the **GPU**.

Note:

1. The **CUDA** version is required.
2. The external default service port is **3000**, but it can be changed.
3. he container service starts and stops **automatically** in sync with the server's startup and shutdown.
4. If pulling the Docker image fails or is slow, switch to a different **image mirror** or use a **VPN**.

```
docker run -d -p 3000:8080 --gpus all --add-host=host.docker.internal:
    host-gateway -v open-webui:/app/backend/data --name open-webui --
    restart always ghcr.io/open-webui/open-webui:cuda
```

Login to Open Webui at **http://172.16.33.244:3000** . When you log in for the first time, you need to create a **administrator account** and password. Subsequent users can register themselves, and the system will **automatically approve** the registrations (you can optionally configure the approval to require a administrator's confirmation instead). Currently, the admin user name is : **q30china@gmail.com**, password: **hexing@2025**



INFO 2025.4.30日模型更新qwen3:30b。ollama版本更新至0.6.6。建议默认使用qwen3:30b和 gemma3:27b。

INFO http://172.16.33.244:5000 作为文件共享中转站，开放测试中。无需用户密码，上传文件并下载，支持平板，手机，电脑。

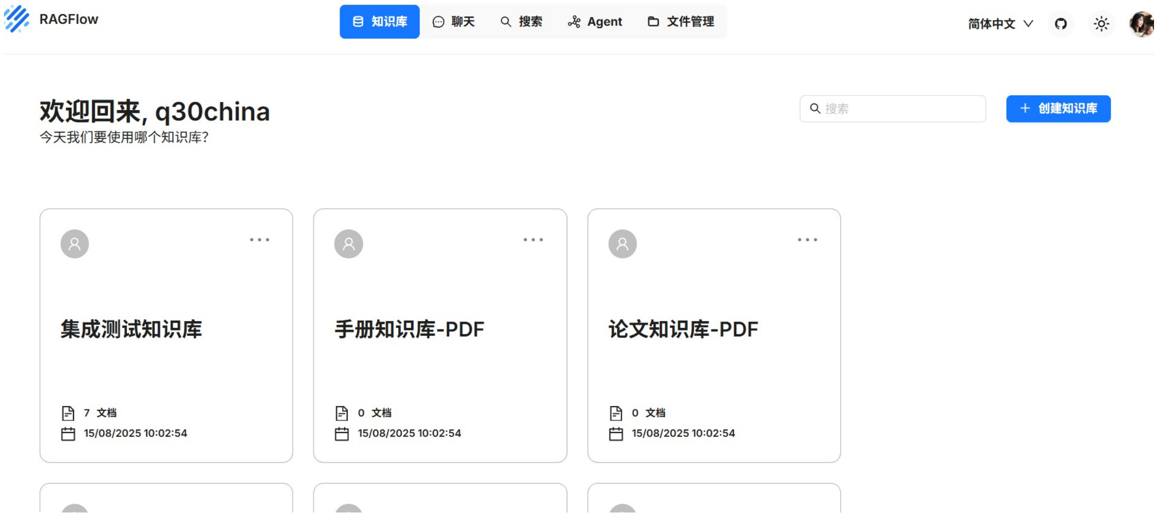
INFO 2025.8.11日增加最新OpenAI开源模型gpt-oss:20b，欢迎使用。

Figure 3: Open Webui User Interface

2.3 Ragflow - Retrieval-Augmented Generation (RAG) systems

Run all the container images with Docker. Once they start normally, the service will be exposed on the default port 8080 . During the first initialization you must set up a system-administrator account and password. Currently, the admin user name is : **q30china@gmail.com**, password: **hexing@2025**

```
# start the service
cd /data/git/ragflow/docker
docker compose -f docker-compose-gpu.yml up -d

# stop the service
docker compose -f docker-compose-gpu.yml down
```



欢迎回来, q30china
今天我们要使用哪个知识库?

集成测试知识库
7 文档
15/08/2025 10:02:54

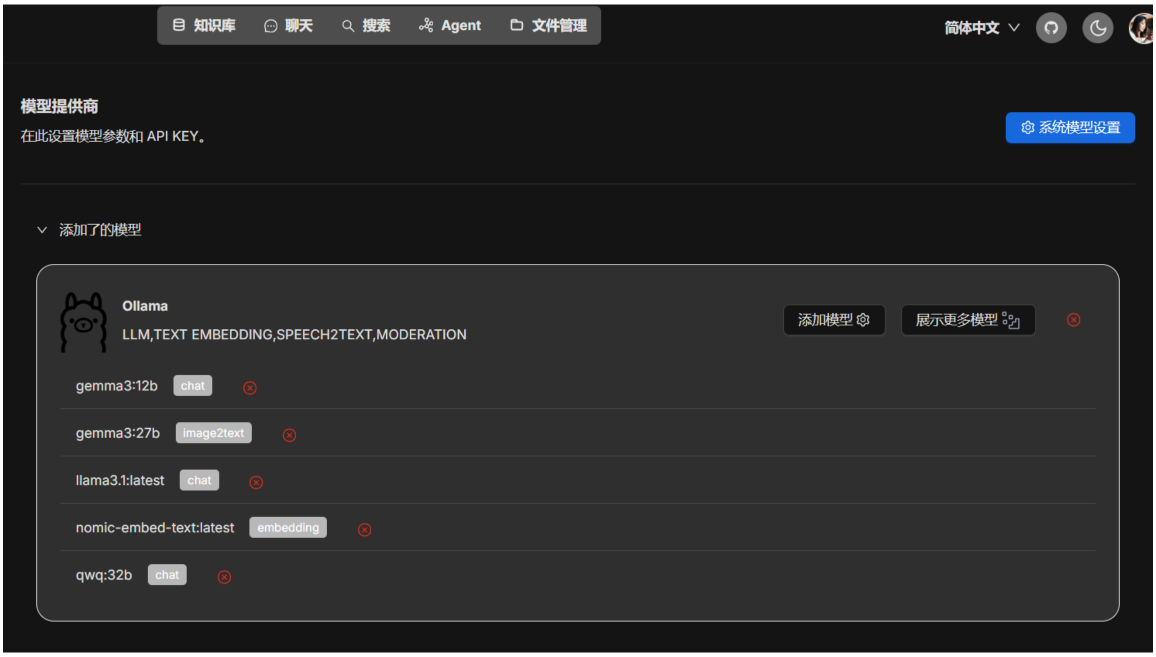
手册知识库-PDF
0 文档
15/08/2025 10:02:54

论文知识库-PDF
0 文档
15/08/2025 10:02:54

Figure 4: Ragflow User Interface

How to use the Ragflow local knowledge database, and create a web application to use this.

step 1: Add local models service (Ollama is running)



模型提供商

在此设置模型参数和 API KEY。

添加了的模型

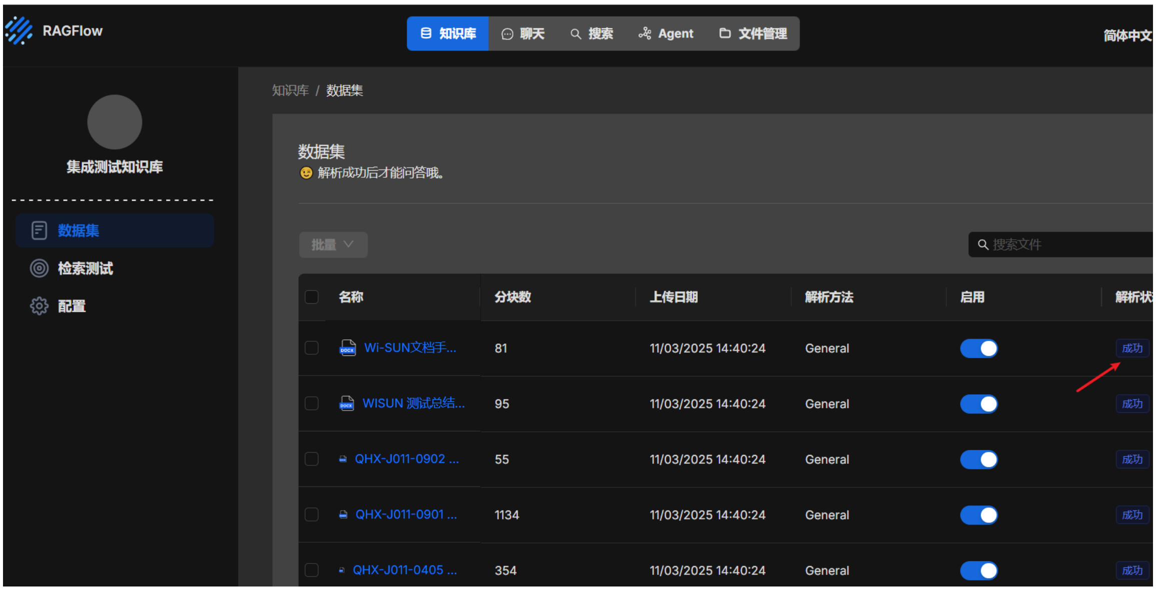
Ollama

LLM,TEXT EMBEDDING,SPEECH2TEXT,MODERATION

gemma3:12b	chat
gemma3:27b	image2text
llama3.1:latest	chat
nomic-embed-text:latest	embedding
qwq:32b	chat

Figure 5: Add local models

step 2: Create a new knowledge base, upload local documents, and wait until the system has finished parsing and vectorizing the storage before proceeding.



知识库 / 数据集

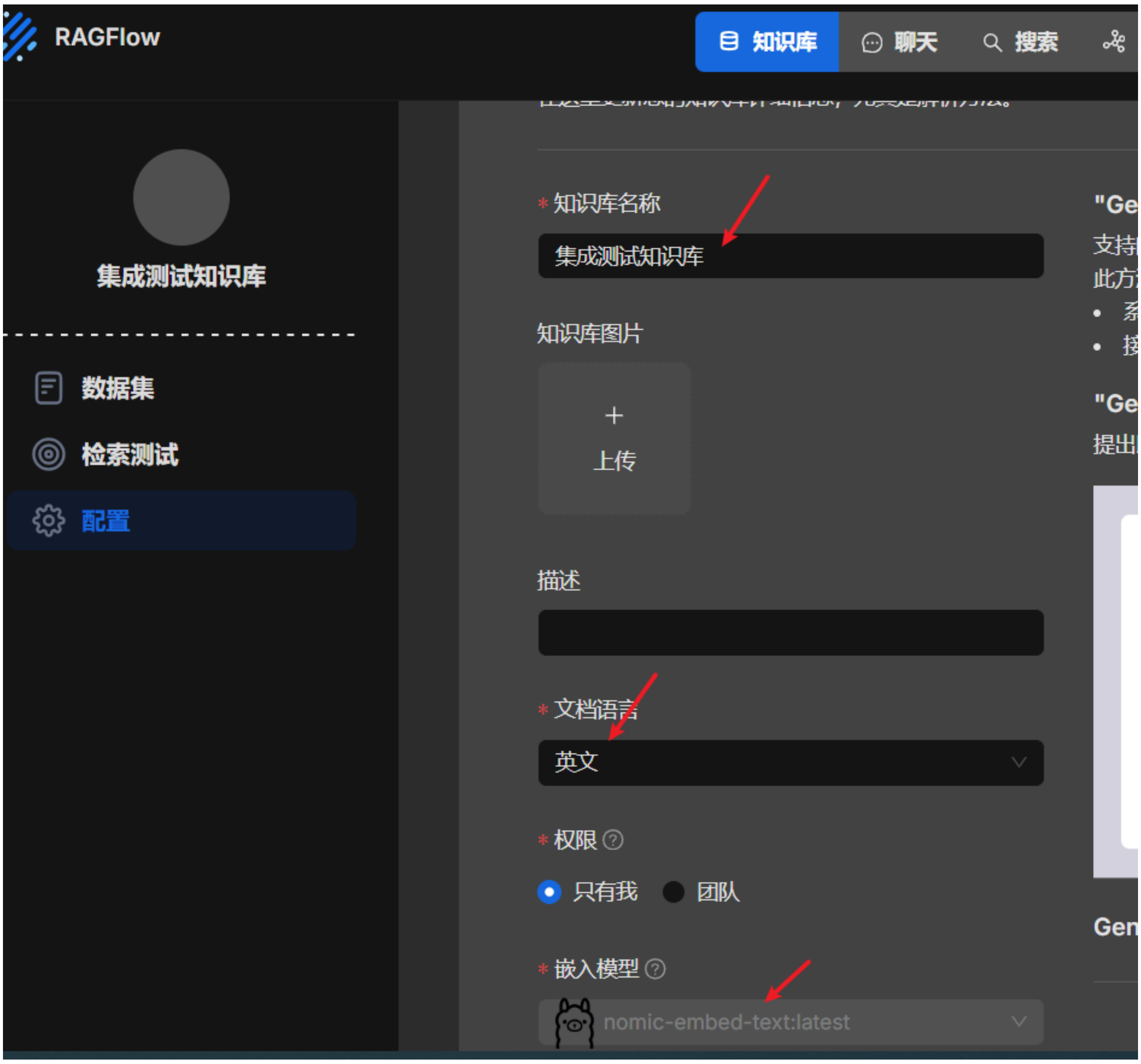
数据集

解析成功后才能问答哦。

名称	分块数	上传日期	解析方法	启用	解析状
Wi-SUN文档手...	81	11/03/2025 14:40:24	General	●	成功
WISUN 测试总结...	95	11/03/2025 14:40:24	General	●	成功
QHX-J011-0902 ...	55	11/03/2025 14:40:24	General	●	成功
QHX-J011-0901 ...	1134	11/03/2025 14:40:24	General	●	成功
QHX-J011-0405 ...	354	11/03/2025 14:40:24	General	●	成功

Figure 6: Upload local files to knowledge database

step 3: In the knowledge-base configuration screen, be sure to select the document's language and choose the embedded vector model (pulled via Ollama). Adjust any other parameters as needed, but typically the defaults are fine.



* 知识库名称

集成测试知识库

知识库图片

+ 上传

描述

* 文档语言

英文

* 权限

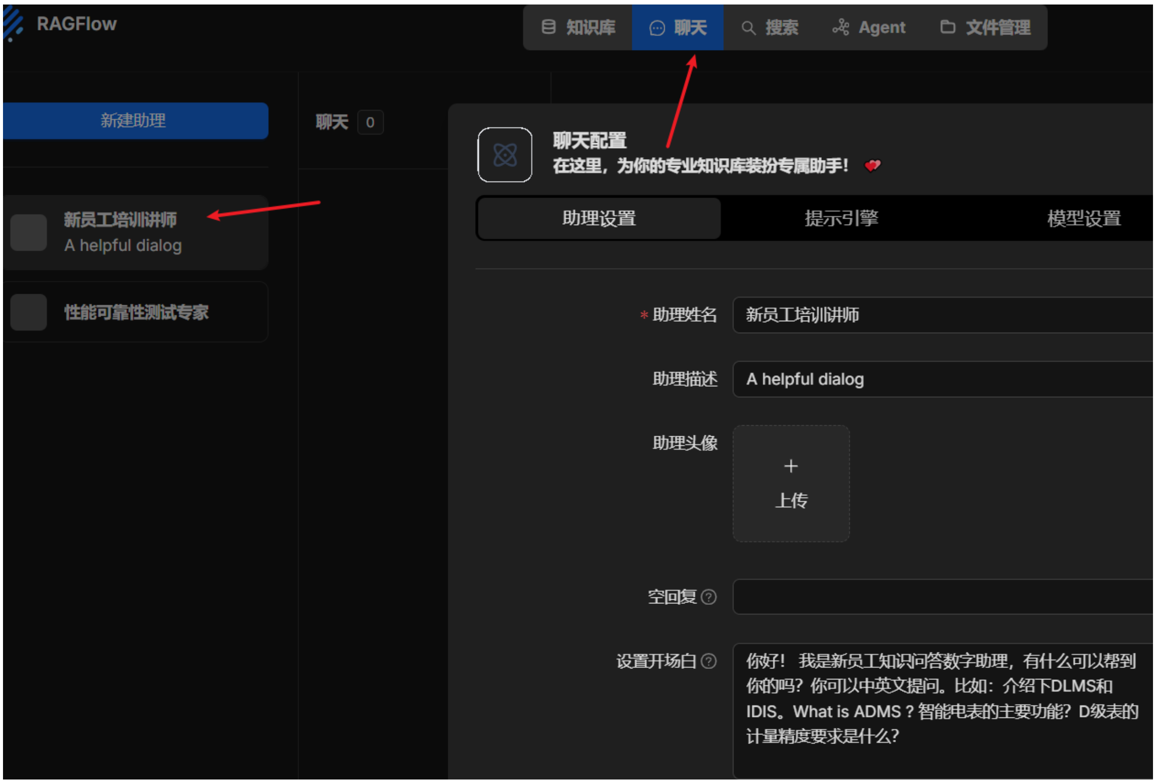
● 只有我 ○ 团队

* 嵌入模型

nomic-embed-text:latest

Figure 7: Configuration of knowlege database

step 4: Create a new digital-human application for new-employee training, using the locally created knowledge base and configuring the LLM to a model from the local Ollama service.



聊天配置
在这里，为你的专业知识库装扮专属助手！

* 助理姓名: 新员工培训讲师

助理描述: A helpful dialog

助理头像: + 上传

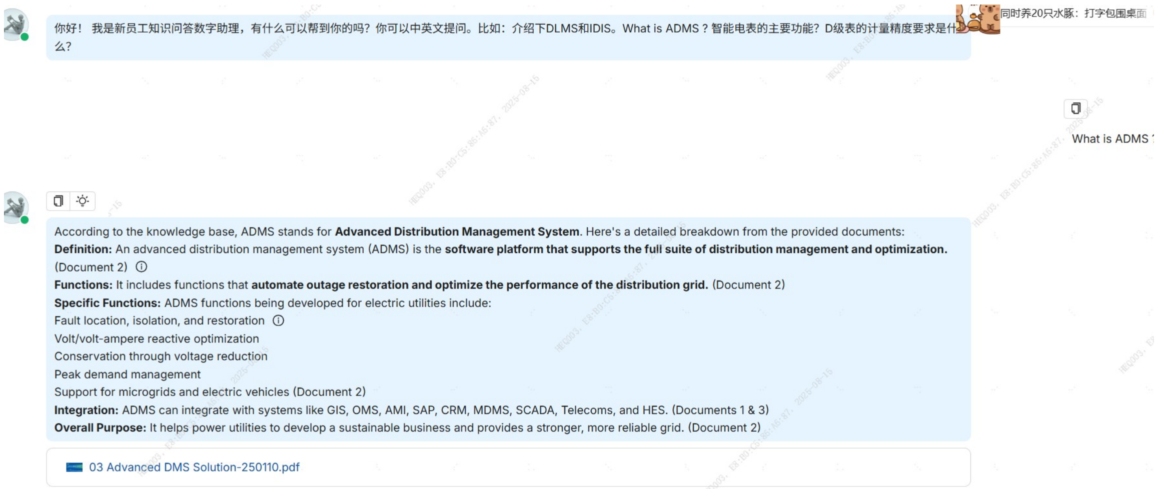
空回复:

设置开场白: 你好！ 我是新员工知识问答数字助理，有什么可以帮到你的吗？你可以中英文提问。比如：介绍下DLMS和IDIS。What is ADMS？智能电表的主要功能？D级表的计量精度要求是什么？

Figure 8: Create application

step 5: Publish app, and test, also can embedded the link to your own web.

```
<iframe
src="http://172.16.33.244:8080/chat/share?shared_id=da54fcfe
f9a311efa8a70242ac120006&from=chat&auth=
FiYWQ3MzMwZjgxMzExZWZhNmZlMDI0Mm"
style="width: 100%; height: 100%; min-height: 600px"
frameborder="0"
>
</iframe>
```



同时养20只水豚：打字包围桌面

你好！ 我是新员工知识问答数字助理，有什么可以帮到你的吗？你可以中英文提问。比如：介绍下DLMS和IDIS。What is ADMS？智能电表的主要功能？D级表的计量精度要求是什么？

What is ADMS

According to the knowledge base, ADMS stands for **Advanced Distribution Management System**. Here's a detailed breakdown from the provided documents:
Definition: An advanced distribution management system (ADMS) is the **software platform that supports the full suite of distribution management and optimization.** (Document 2) ⓘ
Functions: It includes functions that **automate outage restoration and optimize the performance of the distribution grid.** (Document 2)
Specific Functions: ADMS functions being developed for electric utilities include:
Fault location, isolation, and restoration ⓘ
Volt/volt-ampere reactive optimization
Conservation through voltage reduction
Peak demand management
Support for microgrids and electric vehicles (Document 2)
Integration: ADMS can integrate with systems like GIS, OMS, AMI, SAP, CRM, MDMS, SCADA, Telecoms, and HES. (Documents 1 & 3)
Overall Purpose: It helps power utilities to develop a sustainable business and provides a stronger, more reliable grid. (Document 2)

03 Advanced DMS Solution-250110.pdf

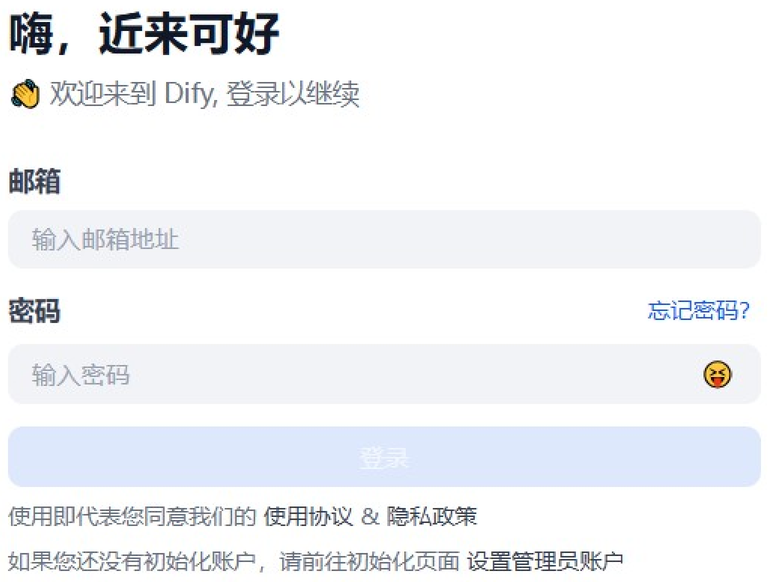
Figure 9: Test the Digital Human

2.4 Dify - low-code AI development framework

Run all the container images with Docker. Once they start normally, the service will be exposed on the default port 80 . During the first initialization you must set up a system-administrator account and password. Currently, the admin user name is : **q30china@gmail.com**, password: **hexing@2025**

```
# start the service
cd /data/git/dify
docker compose up -d

# stop the service
docker compose down
```

A modern, open-source,low-code AI development framework that lets you create, host, and scale AI agents (chatbots, virtual assistants, knowledge-base bots, workflow automators, etc.) without writing boilerplate ML code.



嗨，近来可好

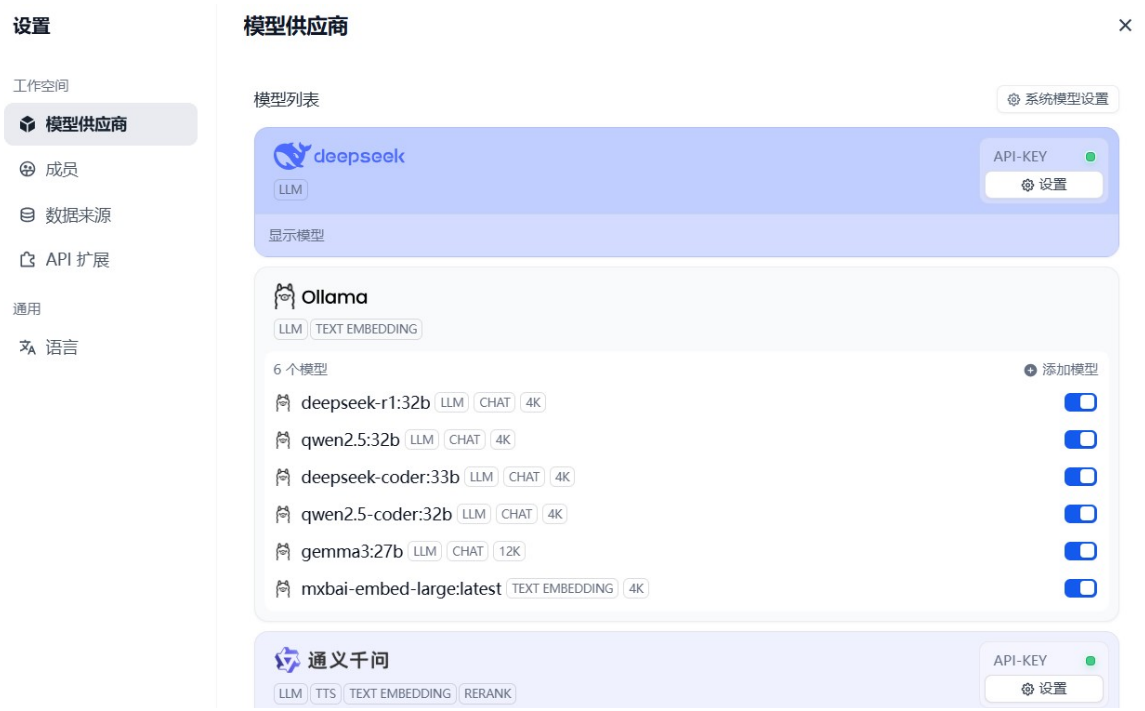
👏 欢迎来到 Dify, 登录以继续

使用即代表您同意我们的 使用协议 & 隐私政策

如果您还没有初始化账户，请前往初始化页面 设置管理员账户

Figure 10: Dify login

Select models, make workflow and programming callback function of workflow



设置

工作空间

模型供应商

成员

数据来源

API 扩展

通用

语言

模型供应商

模型列表

系统模型设置

deepseek
LLM
API-KEY
设置
显示模型

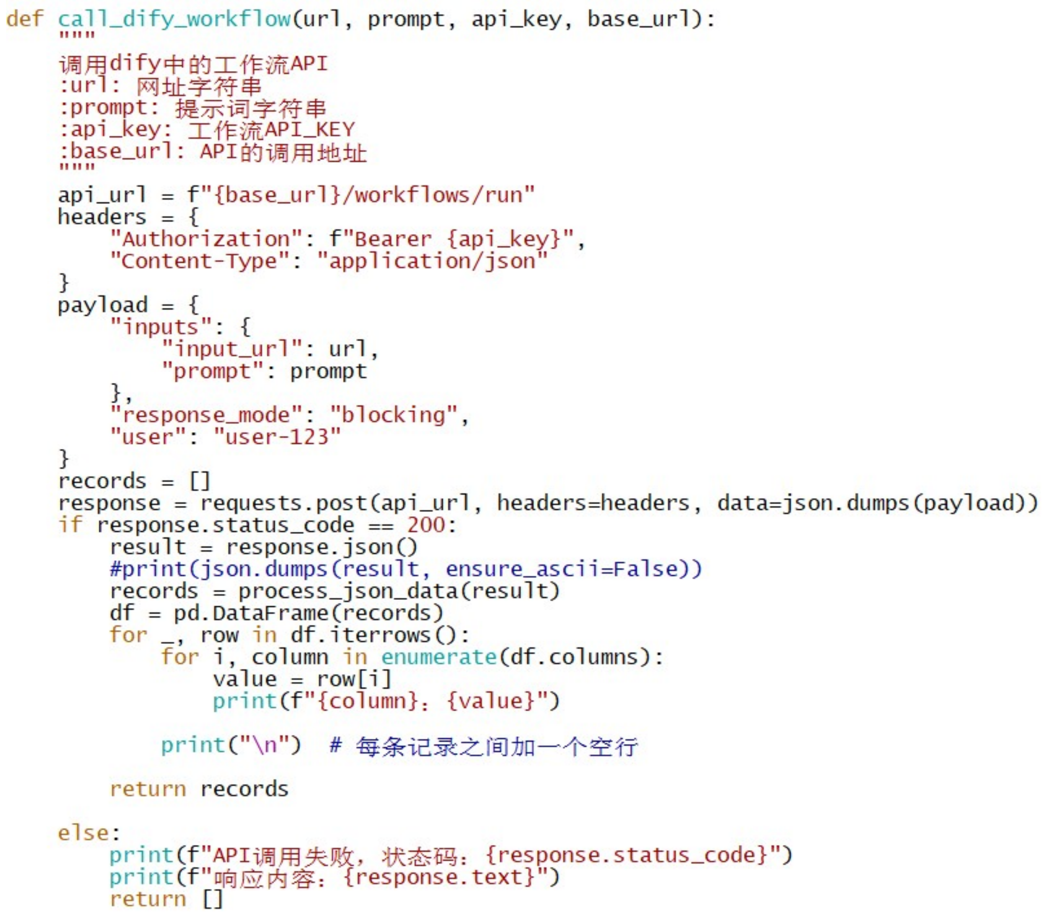
Ollama
LLM TEXT EMBEDDING

6 个模型
添加模型

deepseek-r1:32b LLM CHAT 4K
qwen2.5:32b LLM CHAT 4K
deepseek-coder:33b LLM CHAT 4K
qwen2.5-coder:32b LLM CHAT 4K
gemma3:27b LLM CHAT 12K
mxbai-embed-large:latest TEXT EMBEDDING 4K

通义千问
LLM TTS TEXT EMBEDDING RERANK
API-KEY
设置

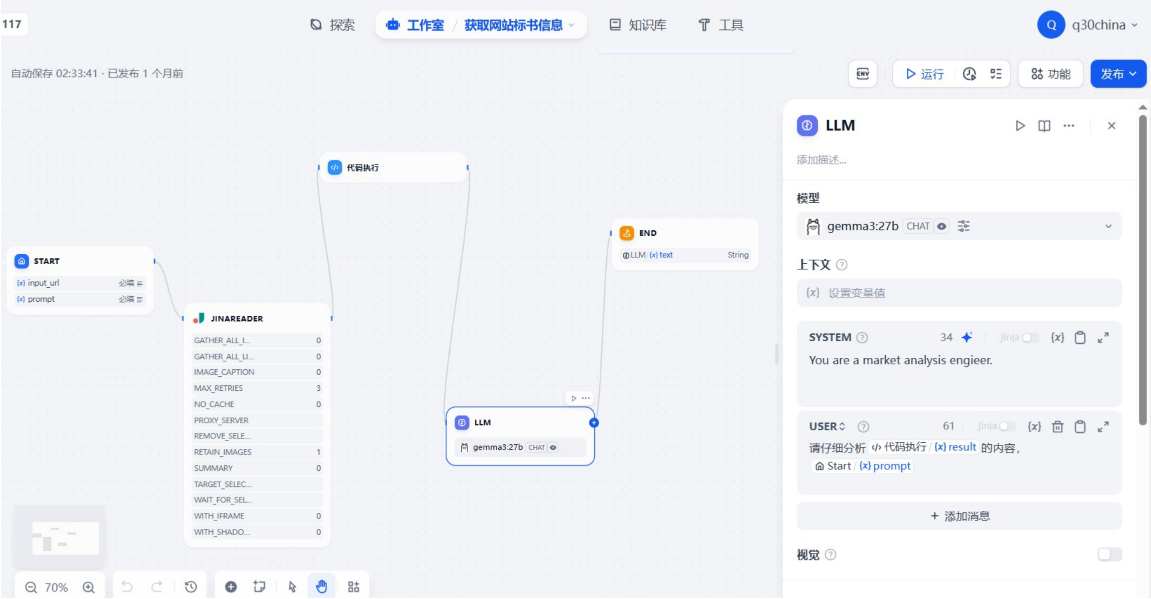
Figure 11: Configure models locally



```
def call_dify_workflow(url, prompt, api_key, base_url):
    """
    调用dify中的工作流API
    :url: 网址字符串
    :prompt: 提示词字符串
    :api_key: 工作流API_KEY
    :base_url: API的调用地址
    """
    api_url = f"{base_url}/workflows/run"
    headers = {
        "Authorization": f"Bearer {api_key}",
        "Content-Type": "application/json"
    }
    payload = {
        "inputs": {
            "input_url": url,
            "prompt": prompt
        },
        "response_mode": "blocking",
        "user": "user-123"
    }
    records = []
    response = requests.post(api_url, headers=headers, data=json.dumps(payload))
    if response.status_code == 200:
        result = response.json()
        #print(json.dumps(result, ensure_ascii=False))
        records = process_json_data(result)
        df = pd.DataFrame(records)
        for _, row in df.iterrows():
            for i, column in enumerate(df.columns):
                value = row[i]
                print(f"{column}: {value}")

            print("\n")  # 每条记录之间加一个空行

        return records

    else:
        print(f"API调用失败, 状态码: {response.status_code}")
        print(f"响应内容: {response.text}")
        return []
```

Figure 12: Call back function of workflow

example 1 : example of Auto acquisition of tenders from website agent



自动保存 02:33:41 · 已发布 1 个月前

START
- {x} input_url 必填
- {x} prompt 必填

JINAREADER

GATHER_ALL_I...	0
GATHER_ALL_LI...	0
IMAGE_CAPTION	0
MAX_RETRIES	3
NO_CACHE	0
PROXY_SERVER	
REMOVE_SELE...	
RETAIN_IMAGES	1
SUMMARY	0
TARGET_SELEC...	
WAIT_FOR_SEL...	
WITH_IFRAME	0
WITH_SHADO...	0

代码执行

LLM: gemma3:27b CHAT

END: LLM {x} text String

LLM

添加描述...

模型: gemma3:27b CHAT

上下文: {x} 设置变量值

SYSTEM 34 Jinja

You are a market analysis engieer.

USER 61 Jinja

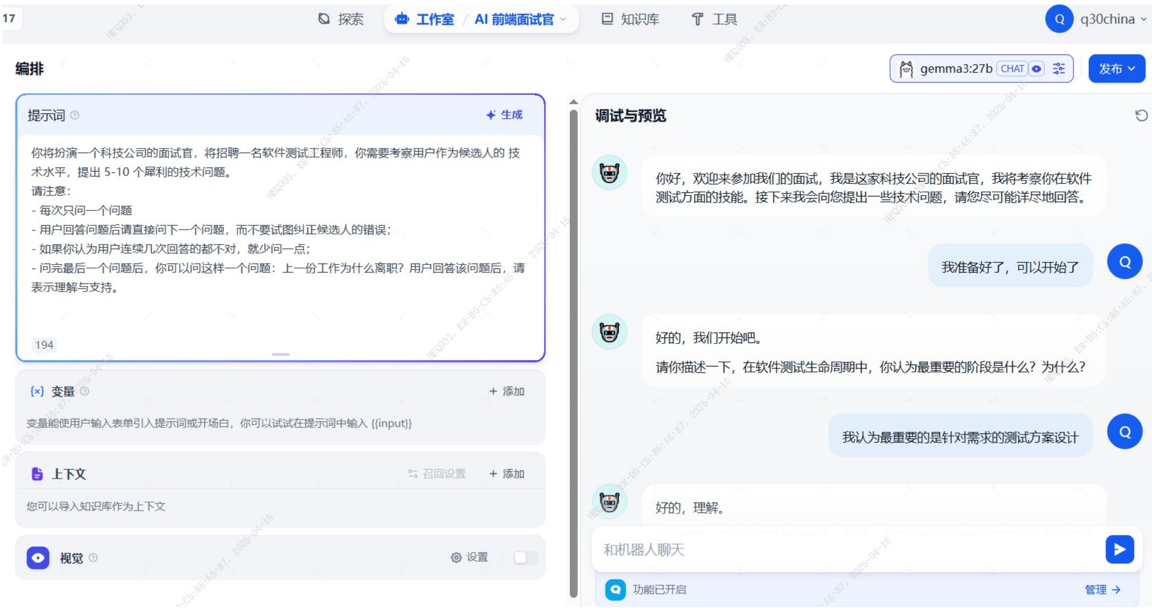
请仔细分析 代码执行 / {x} result 的内容，
Start / {x} prompt

+ 添加消息

视觉

Figure 13: Workflow definition for tender acquisition

example 2 : example of An Testing Interview agent



编排

提示词 ✦ 生成

你将扮演一个科技公司的面试官，将招聘一名软件测试工程师，你需要考察用户作为候选人的 技术水平，提出 5-10 个犀利的技术问题。
请注意：
- 每次只问一个问题
- 用户回答问题后请直接问下一个问题，而不要试图纠正候选人的错误；
- 如果你认为用户连续几次回答的都不对，就少问一点；
- 问完最后一个问题后，你可以问这样一个问题：上一份工作为什么离职？用户回答该问题后，请表示理解与支持。

194

{x} 变量 + 添加

变量能使用户输入表单引入提示词或开场白，你可以试试在提示词中输入 {{input}}

上下文 召回设置 + 添加

您可以导入知识库作为上下文

视觉 设置

调试与预览

你好，欢迎来参加我们的面试，我是这家科技公司的面试官，我将考察你在软件测试方面的技能。接下来我会向您提出一些技术问题，请您尽可能详尽地回答。

我准备好了，可以开始了

好的，我们开始吧。

请你描述一下，在软件测试生命周期中，你认为最重要的阶段是什么？为什么？

我认为最重要的是针对需求的测试方案设计

好的，理解。

和机器人聊天

功能已开启 管理 →

Figure 14: Interview agent demo

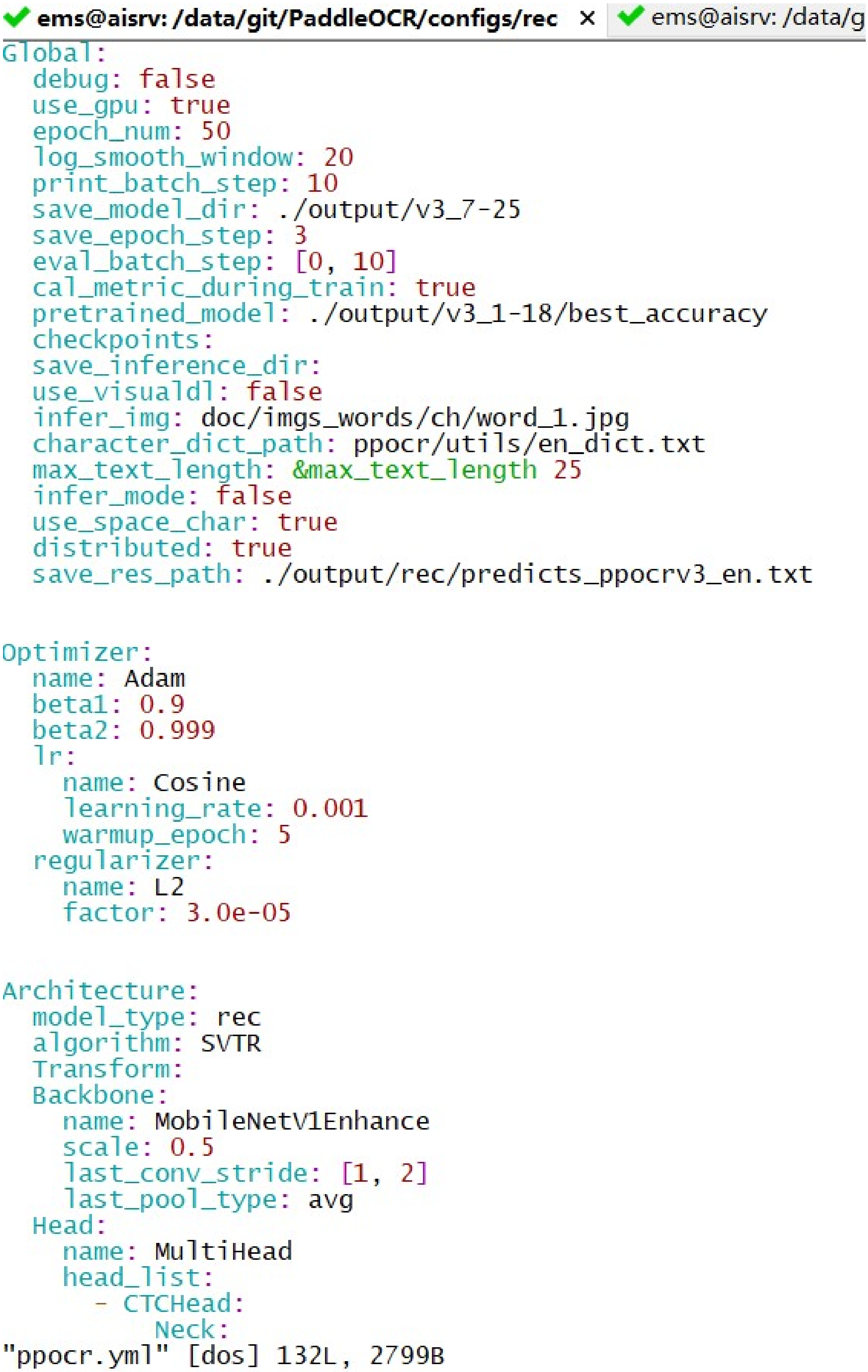
2.5 PaddleOCR

PaddleOCR is an open-source, multi-language OCR toolkit developed on the PaddlePaddle deep learning framework.

Here we use the PaddleOCR toolkit to perform image recognition on the LCD display of an intelligent electric meter. The model employed is a self-trained model built upon the open-source baseline. By annotating 1,000 photos of the LCD, we train with the PaddleOCR toolkit to generate our own model. Then, through calls to the OCR-Server / OCR-Client software, we enable automated testing of the intelligent electric meter's LCD display.

```
# create and use python virtual enviroment
conda activate paddleocr

# be sure  paddlepaddle-gpu is installed
python -m pip install paddlepaddle-gpu==3.0.0b1 -i https://pypi.tuna.tsinghua.edu.cn/simple

# modify the config file
cd /data/git/PaddleOCR/configs/rec
vi ppocr.yml

# copy trainning data to direcory train (1239 pics)
cp * /data/git/PaddleOCR/train_data/Lcd/train/

# copy testing data to directory test (131 pics)
cp * /data/git/PaddleOCR/train_data/Lcd/test/

# start training
python tools/train.py -c configs/rec/ppocr.yml
```

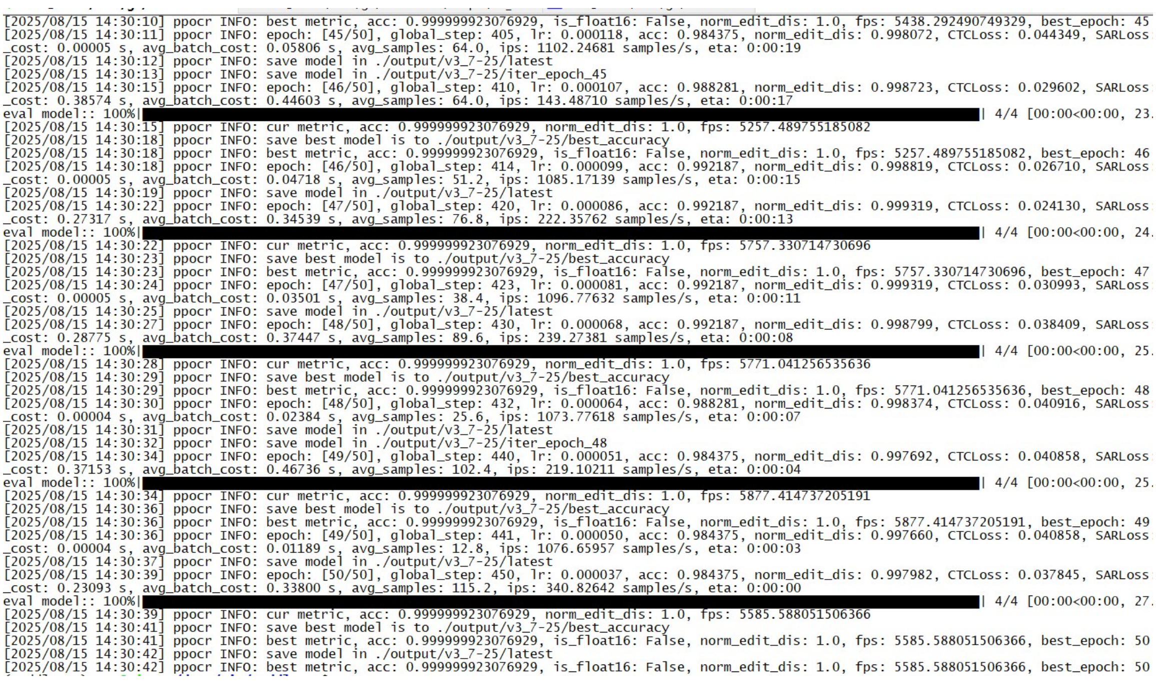
Some parameters from ppocr.yml :



```
Global:
  debug: false
  use_gpu: true
  epoch_num: 50
  log_smooth_window: 20
  print_batch_step: 10
  save_model_dir: ./output/v3_7-25
  save_epoch_step: 3
  eval_batch_step: [0, 10]
  cal_metric_during_train: true
  pretrained_model: ./output/v3_1-18/best_accuracy
  checkpoints:
  save_inference_dir:
  use_visualdl: false
  infer_img: doc/imgs_words/ch/word_1.jpg
  character_dict_path: ppocr/utils/en_dict.txt
  max_text_length: &max_text_length 25
  infer_mode: false
  use_space_char: true
  distributed: true
  save_res_path: ./output/rec/predicts_ppocrv3_en.txt


Optimizer:
  name: Adam
  beta1: 0.9
  beta2: 0.999
  lr:
    name: Cosine
    learning_rate: 0.001
    warmup_epoch: 5
  regularizer:
    name: L2
    factor: 3.0e-05


Architecture:
  model_type: rec
  algorithm: SVTR
  Transform:
  Backbone:
    name: MobileNetV1Enhance
    scale: 0.5
    last_conv_stride: [1, 2]
    last_pool_type: avg
  Head:
    name: MultiHead
    head_list:
      - CTCHead:
          Neck:
"ppocr.yml" [dos] 132L, 2799B
```

Figure 15: Part of ppocr.yml

Training process :



```
[2025/08/15 14:30:10] ppocr INFO: best metric, acc: 0.999999923076929, is_float16: False, norm_edit_dis: 1.0, fps: 5438.292490749329, best_epoch: 45
[2025/08/15 14:30:11] ppocr INFO: epoch: [45/50], global_step: 405, lr: 0.000118, acc: 0.984375, norm_edit_dis: 0.998072, CTCLoss: 0.044349, SARLoss
_cost: 0.00005 s, avg_batch_cost: 0.05806 s, avg_samples: 64.0, ips: 1102.24681 samples/s, eta: 0:00:19
[2025/08/15 14:30:12] ppocr INFO: save model in ./output/v3_7-25/latest
[2025/08/15 14:30:13] ppocr INFO: save model in ./output/v3_7-25/iter_epoch_45
[2025/08/15 14:30:15] ppocr INFO: epoch: [46/50], global_step: 410, lr: 0.000107, acc: 0.988281, norm_edit_dis: 0.998723, CTCLoss: 0.029602, SARLoss
_cost: 0.38574 s, avg_batch_cost: 0.44603 s, avg_samples: 64.0, ips: 143.48710 samples/s, eta: 0:00:17
eval model:: 100%|██████████████████████████████████████████████████████████████| 4/4 [00:00<00:00, 23.
[2025/08/15 14:30:15] ppocr INFO: cur metric, acc: 0.999999923076929, norm_edit_dis: 1.0, fps: 5257.489755185082
[2025/08/15 14:30:18] ppocr INFO: save best model is to ./output/v3_7-25/best_accuracy
[2025/08/15 14:30:18] ppocr INFO: best metric, acc: 0.999999923076929, is_float16: False, norm_edit_dis: 1.0, fps: 5257.489755185082, best_epoch: 46
[2025/08/15 14:30:18] ppocr INFO: epoch: [46/50], global_step: 414, lr: 0.000099, acc: 0.992187, norm_edit_dis: 0.998819, CTCLoss: 0.026710, SARLoss
_cost: 0.00005 s, avg_batch_cost: 0.04718 s, avg_samples: 51.2, ips: 1085.17139 samples/s, eta: 0:00:15
[2025/08/15 14:30:19] ppocr INFO: save model in ./output/v3_7-25/latest
[2025/08/15 14:30:22] ppocr INFO: epoch: [47/50], global_step: 420, lr: 0.000086, acc: 0.992187, norm_edit_dis: 0.999319, CTCLoss: 0.024130, SARLoss
_cost: 0.27317 s, avg_batch_cost: 0.34539 s, avg_samples: 76.8, ips: 222.35762 samples/s, eta: 0:00:13
eval model:: 100%|██████████████████████████████████████████████████████████████| 4/4 [00:00<00:00, 24.
[2025/08/15 14:30:22] ppocr INFO: cur metric, acc: 0.999999923076929, norm_edit_dis: 1.0, fps: 5757.330714730696
[2025/08/15 14:30:23] ppocr INFO: save best model is to ./output/v3_7-25/best_accuracy
[2025/08/15 14:30:23] ppocr INFO: best metric, acc: 0.999999923076929, is_float16: False, norm_edit_dis: 1.0, fps: 5757.330714730696, best_epoch: 47
[2025/08/15 14:30:24] ppocr INFO: epoch: [47/50], global_step: 423, lr: 0.000081, acc: 0.992187, norm_edit_dis: 0.999319, CTCLoss: 0.030993, SARLoss
_cost: 0.00005 s, avg_batch_cost: 0.03501 s, avg_samples: 38.4, ips: 1096.77632 samples/s, eta: 0:00:11
[2025/08/15 14:30:25] ppocr INFO: save model in ./output/v3_7-25/latest
[2025/08/15 14:30:27] ppocr INFO: epoch: [48/50], global_step: 430, lr: 0.000068, acc: 0.992187, norm_edit_dis: 0.998799, CTCLoss: 0.038409, SARLoss
_cost: 0.28775 s, avg_batch_cost: 0.37447 s, avg_samples: 89.6, ips: 239.27381 samples/s, eta: 0:00:08
eval model:: 100%|██████████████████████████████████████████████████████████████| 4/4 [00:00<00:00, 25.
[2025/08/15 14:30:28] ppocr INFO: cur metric, acc: 0.999999923076929, norm_edit_dis: 1.0, fps: 5771.041256535636
[2025/08/15 14:30:29] ppocr INFO: save best model is to ./output/v3_7-25/best_accuracy
[2025/08/15 14:30:29] ppocr INFO: best metric, acc: 0.999999923076929, is_float16: False, norm_edit_dis: 1.0, fps: 5771.041256535636, best_epoch: 48
[2025/08/15 14:30:30] ppocr INFO: epoch: [48/50], global_step: 432, lr: 0.000064, acc: 0.988281, norm_edit_dis: 0.998374, CTCLoss: 0.040916, SARLoss
_cost: 0.00004 s, avg_batch_cost: 0.02384 s, avg_samples: 25.6, ips: 1073.77618 samples/s, eta: 0:00:07
[2025/08/15 14:30:31] ppocr INFO: save model in ./output/v3_7-25/latest
[2025/08/15 14:30:32] ppocr INFO: save model in ./output/v3_7-25/iter_epoch_48
[2025/08/15 14:30:34] ppocr INFO: epoch: [49/50], global_step: 440, lr: 0.000051, acc: 0.984375, norm_edit_dis: 0.997692, CTCLoss: 0.040858, SARLoss
_cost: 0.37153 s, avg_batch_cost: 0.46736 s, avg_samples: 102.4, ips: 219.10211 samples/s, eta: 0:00:04
eval model:: 100%|██████████████████████████████████████████████████████████████| 4/4 [00:00<00:00, 25.
[2025/08/15 14:30:34] ppocr INFO: cur metric, acc: 0.999999923076929, norm_edit_dis: 1.0, fps: 5877.414737205191
[2025/08/15 14:30:36] ppocr INFO: save best model is to ./output/v3_7-25/best_accuracy
[2025/08/15 14:30:36] ppocr INFO: best metric, acc: 0.999999923076929, is_float16: False, norm_edit_dis: 1.0, fps: 5877.414737205191, best_epoch: 49
[2025/08/15 14:30:36] ppocr INFO: epoch: [49/50], global_step: 441, lr: 0.000050, acc: 0.984375, norm_edit_dis: 0.997660, CTCLoss: 0.040858, SARLoss
_cost: 0.00004 s, avg_batch_cost: 0.01189 s, avg_samples: 12.8, ips: 1076.65957 samples/s, eta: 0:00:03
[2025/08/15 14:30:37] ppocr INFO: save model in ./output/v3_7-25/latest
[2025/08/15 14:30:39] ppocr INFO: epoch: [50/50], global_step: 450, lr: 0.000037, acc: 0.984375, norm_edit_dis: 0.997982, CTCLoss: 0.037845, SARLoss
_cost: 0.23093 s, avg_batch_cost: 0.33800 s, avg_samples: 115.2, ips: 340.82642 samples/s, eta: 0:00:00
eval model:: 100%|██████████████████████████████████████████████████████████████| 4/4 [00:00<00:00, 27.
[2025/08/15 14:30:39] ppocr INFO: cur metric, acc: 0.999999923076929, norm_edit_dis: 1.0, fps: 5585.588051506366
[2025/08/15 14:30:41] ppocr INFO: save best model is to ./output/v3_7-25/best_accuracy
[2025/08/15 14:30:41] ppocr INFO: best metric, acc: 0.999999923076929, is_float16: False, norm_edit_dis: 1.0, fps: 5585.588051506366, best_epoch: 50
[2025/08/15 14:30:42] ppocr INFO: save model in ./output/v3_7-25/latest
[2025/08/15 14:30:42] ppocr INFO: best metric, acc: 0.999999923076929, is_float16: False, norm_edit_dis: 1.0, fps: 5585.588051506366, best_epoch: 50
```

Figure 16: 50 epoch train process

2.6 AI4Test

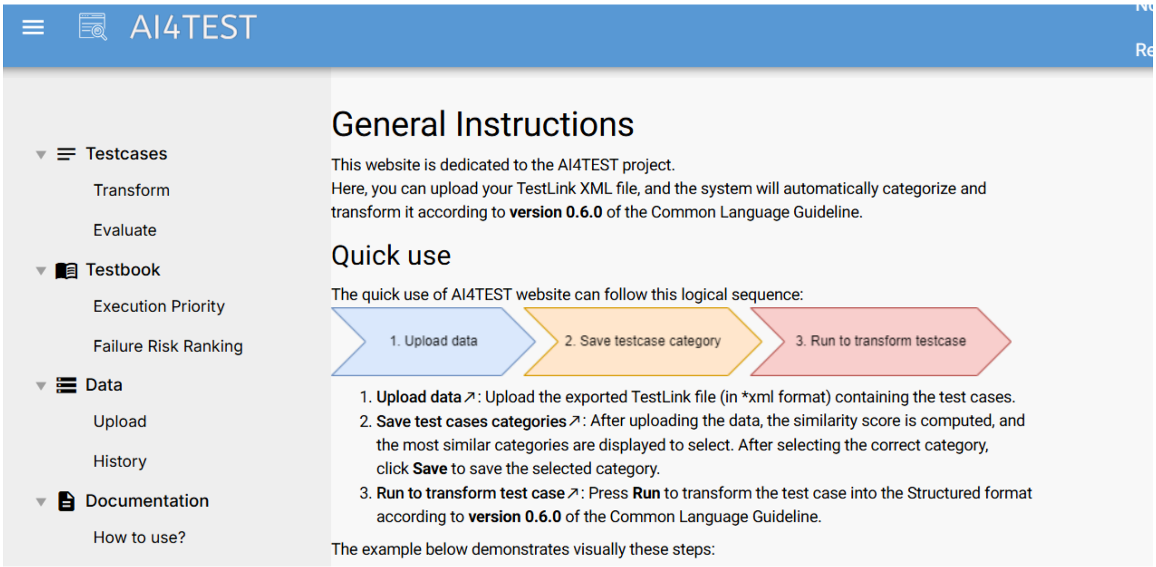
AI4Test project main **objectives**: (2024.1.1-2024.12.1)

- Optimize Test Book Generation: Automatically Support the Test Engineering Team in designing the Test Books;
- Standardize Test Cases: Normalize Test Cases through a unified Common Test Case Language Model;
- Transform Unstructured Test Cases with AI: Utilize AI models to convert unstructured Test Cases into a standardized Common Language Model;
- AI-Driven Requirements-to-Test Cases Modeling: Create an AI model for predicting and automating test generation, prioritization, reporting test durations, and forecasting failure probabilities based on product requirements;
- New Test Cases Scenarios: Explore the generation of novel Test Cases from product requirements using large language models (LLMs).

How to start and use AI4Test ?

```
# create and use python virtual enviroment
conda activate ai4test

# Start AI4Test Webserver
cd /data/git/ai4test
python3 Infrastructure/app/main.py

# WebSite address
http://172.16.33.244:8882/
```

1/ The main user interface for AI4Test



General Instructions

This website is dedicated to the AI4TEST project.
Here, you can upload your TestLink XML file, and the system will automatically categorize and transform it according to **version 0.6.0** of the Common Language Guideline.

Quick use

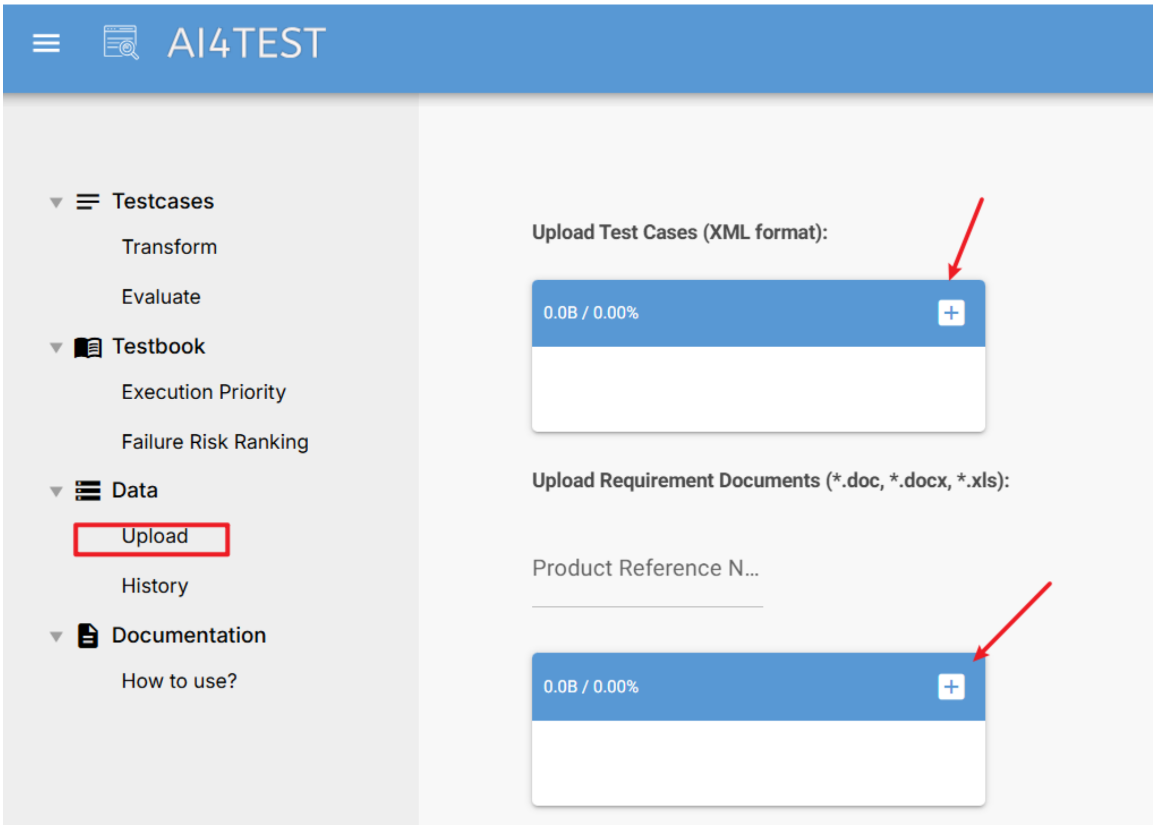
The quick use of AI4TEST website can follow this logical sequence:

1. **Upload data** ↗: Upload the exported TestLink file (in *xml format) containing the test cases.
2. **Save test cases categories** ↗: After uploading the data, the similarity score is computed, and the most similar categories are displayed to select. After selecting the correct category, click **Save** to save the selected category.
3. **Run to transform test case** ↗: Press **Run** to transform the test case into the Structured format according to **version 0.6.0** of the Common Language Guideline.

The example below demonstrates visually these steps:

Figure 17: AI4Test main UI

/2 When using the tool for the first time, you need to upload test cases in XML format, which can be directly exported from TestLink. In addition, you can upload requirement documents—docx or xls files are supported.



Upload Test Cases (XML format):

0.0B / 0.00%

Upload Requirement Documents (*.doc, *.docx, *.xls):

Product Reference N...

0.0B / 0.00%

Figure 18: Upload files to AI4Test

3/ Loading requirements and test cases can be done by selecting them from the history of uploaded data

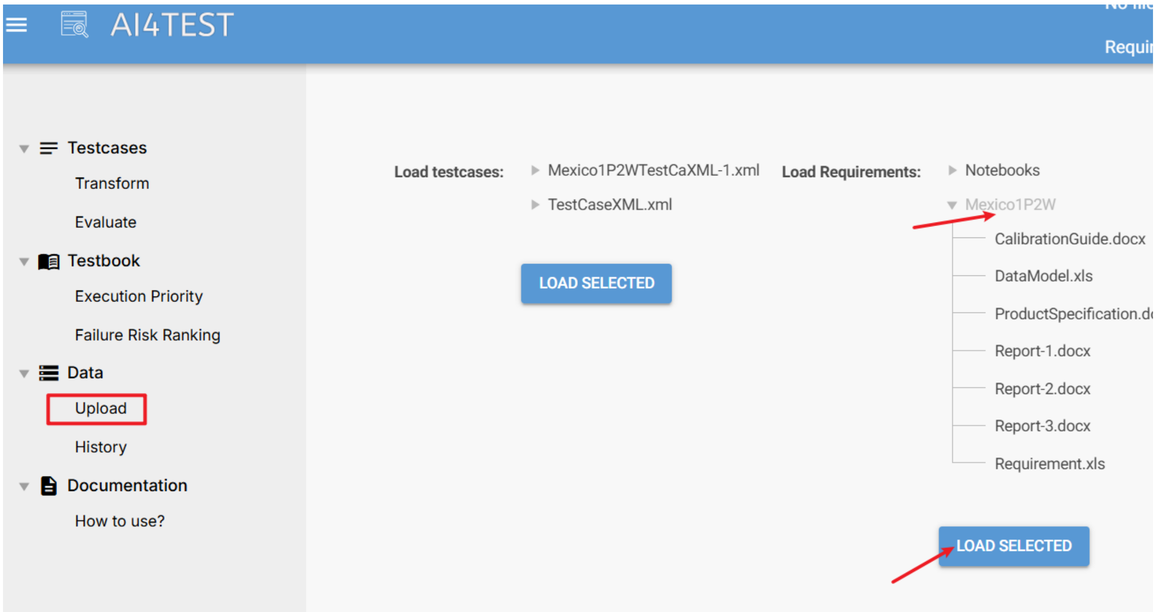
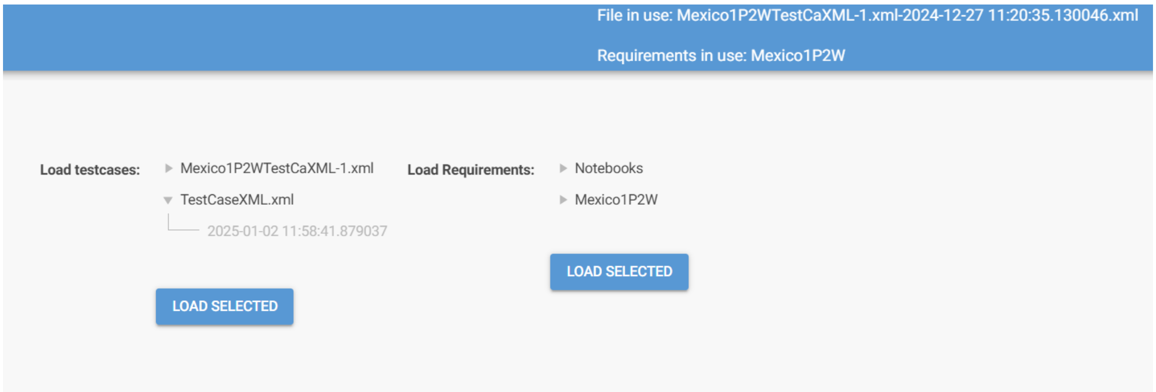


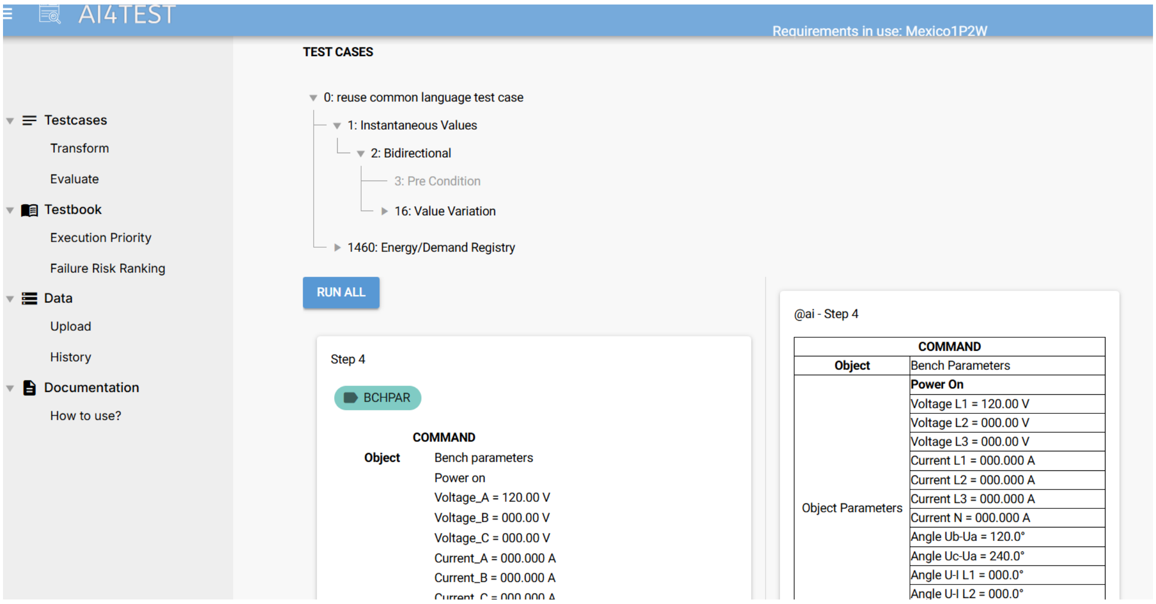
Figure 19: Loading requirements


File in use: Mexico1P2WTestCaXML-1.xml-2024-12-27 11:20:35.130046.xml

Requirements in use: Mexico1P2W

Figure 20: Loading test cases

4/ From the Transform section of the Testcases page in the left-hand tree, you can view the AI conversion results, whereby test cases exported from TestLink are transformed into structured test cases defined by the Common Test Language specification



TEST CASES

- 0: reuse common language test case
  - 1: Instantaneous Values
    - 2: Bidirectional
      - 3: Pre Condition
      - 16: Value Variation
  - 1460: Energy/Demand Registry

Step 4

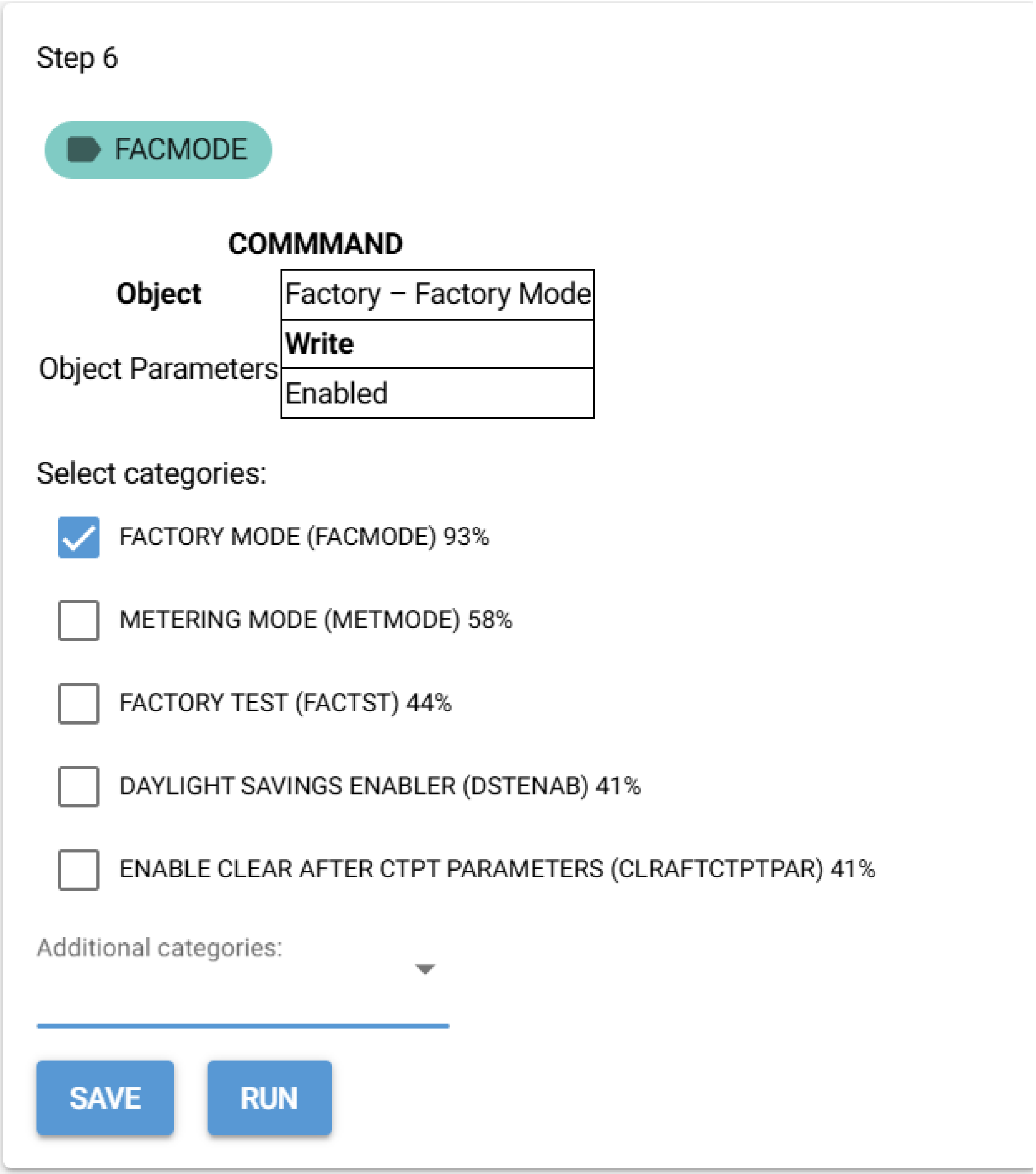
	COMMAND
Object	Bench parameters
	Power on
	Voltage_A = 120.00 V
	Voltage_B = 000.00 V
	Voltage_C = 000.00 V
	Current_A = 000.000 A
	Current_B = 000.000 A
	Current_C = 000.000 A

@ai - Step 4

	COMMAND
Object	Bench Parameters
Object Parameters	**Power On**
	Voltage L1 = 120.00 V
	Voltage L2 = 000.00 V
	Voltage L3 = 000.00 V
	Current L1 = 000.000 A
	Current L2 = 000.000 A
	Current L3 = 000.000 A
	Current N = 000.000 A
	Angle Ub-Ua = 120.0°
	Angle Uc-Ua = 240.0°
	Angle U-I L1 = 000.0°
	Angle U-I L2 = 000.0°

Figure 21: Test Case transformation

5/ If the AI-converted result is incorrect, you can manually correct it. You can select a different classification, and other categories are also available in the Additional section. Once you submit your feedback, the system will automatically retrain and learn from it.



Step 6

FACMODE

COMMMAND

Object	Factory – Factory Mode
Object Parameters	**Write**
	Enabled

Select categories:

☑ FACTORY MODE (FACMODE) 93%

□ METERING MODE (METMODE) 58%

□ FACTORY TEST (FACTST) 44%

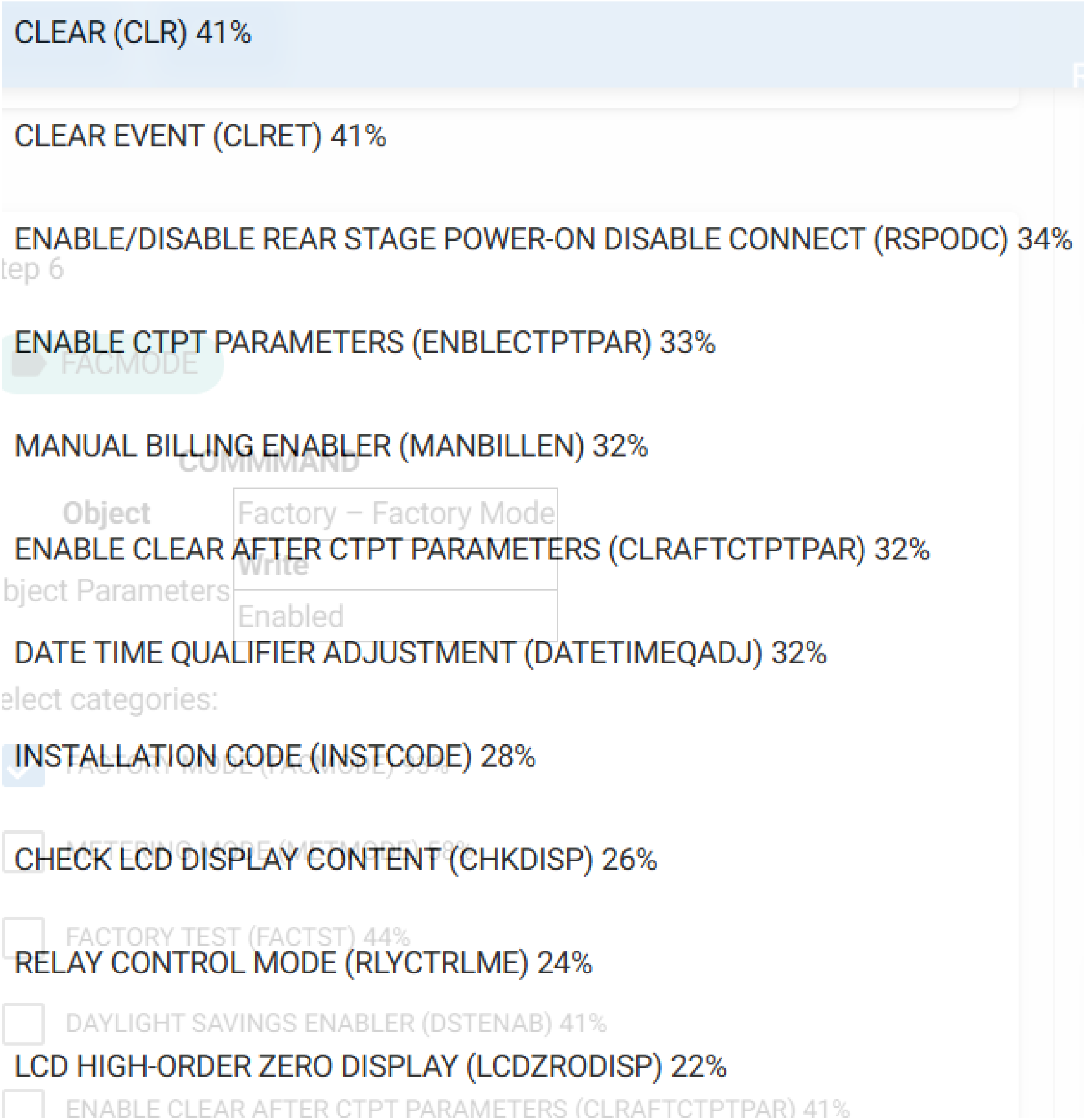
□ DAYLIGHT SAVINGS ENABLER (DSTENAB) 41%

□ ENABLE CLEAR AFTER CTPT PARAMETERS (CLRAFTCTPTPAR) 41%

Additional categories:

Figure 22: Manual adjustment of conversion results

6/ Each category is accompanied by a probability—the higher the value, the stronger the AI’s recommendation.


tep 6
FACMODE
COMMMAND
Object
Factory – Factory Mode
Write
bject Parameters
Enabled
elect categories:
METERING MODE (METMODE) 5%
FACTORY TEST (FACTST) 44%
DAYLIGHT SAVINGS ENABLER (DSTENAB) 41%
ENABLE CLEAR AFTER CTPT PARAMETERS (CLRAFTCTPTPAR) 41%

CLEAR (CLR) 41%

CLEAR EVENT (CLRET) 41%

ENABLE/DISABLE REAR STAGE POWER-ON DISABLE CONNECT (RSPODC) 34%

ENABLE CTPT PARAMETERS (ENBLECTPTPAR) 33%

MANUAL BILLING ENABLER (MANBILLEN) 32%

ENABLE CLEAR AFTER CTPT PARAMETERS (CLRAFTCTPTPAR) 32%

DATE TIME QUALIFIER ADJUSTMENT (DATETIMEQADJ) 32%

INSTALLATION CODE (INSTCODE) 28%

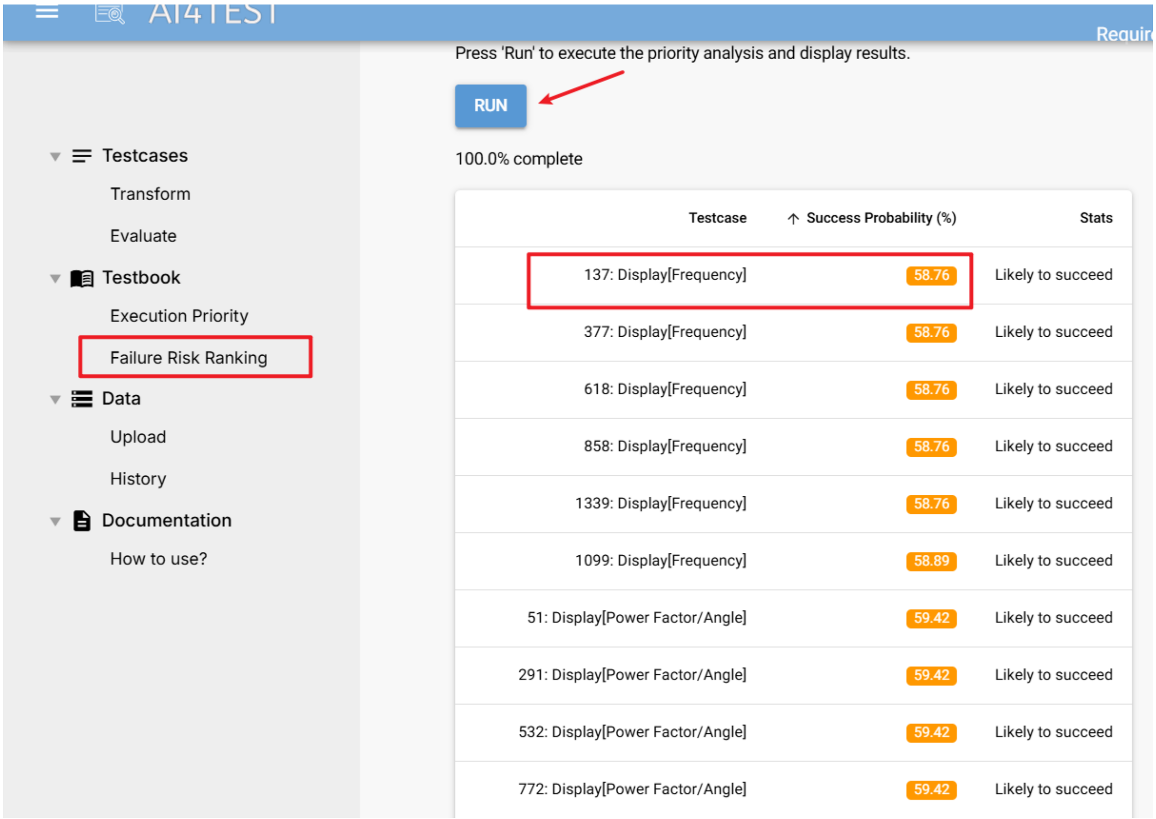
CHECK LCD DISPLAY CONTENT (CHKDISP) 26%

RELAY CONTROL MODE (RLYCTRLME) 24%

LCD HIGH-ORDER ZERO DISPLAY (LCDZRODISP) 22%

Figure 23: Execution step classification

7/ Sort the test cases by execution success rate based on their historical execution results.



Press 'Run' to execute the priority analysis and display results.

100.0% complete

Testcase	↑ Success Probability (%)	Stats
137: Display[Frequency]	58.76	Likely to succeed
377: Display[Frequency]	58.76	Likely to succeed
618: Display[Frequency]	58.76	Likely to succeed
858: Display[Frequency]	58.76	Likely to succeed
1339: Display[Frequency]	58.76	Likely to succeed
1099: Display[Frequency]	58.89	Likely to succeed
51: Display[Power Factor/Angle]	59.42	Likely to succeed
291: Display[Power Factor/Angle]	59.42	Likely to succeed
532: Display[Power Factor/Angle]	59.42	Likely to succeed
772: Display[Power Factor/Angle]	59.42	Likely to succeed

Figure 24: Test case execution success rate

3 Closing remarks

All of the contents in this document represnets my exploration in AI while learning, supplemented by references to various project documents. There may be some errors and omissions, so please feel free to contact me if you find any. I would also like to thank the members of the AI4Test team — **Francisco Alexandre** and **Wang Wen**—as well as **Gabriel** from the OCR project. Though I was involved in both projects from start to finish, I appreciate their contributions.