MACAU UNIVERSITY OF SCIENCE AND TECHNOLOGY

School of Computer Science and Engineering

Faculty of Innovation Engineering

<<Software Project for Course Software Engineering>>

Homework ID : Task4-Object-Oriented Requirements Analysis

Report Title : A Development of Multifunctional Real-time Translation Software with AI-OCR and Natural language translation model

Student Name : Wang Haocheng , Wang Zi , Li Yuanqi , Ma Weiyan

Student No. : 1220023205,1220029080,1220029051,1220033292

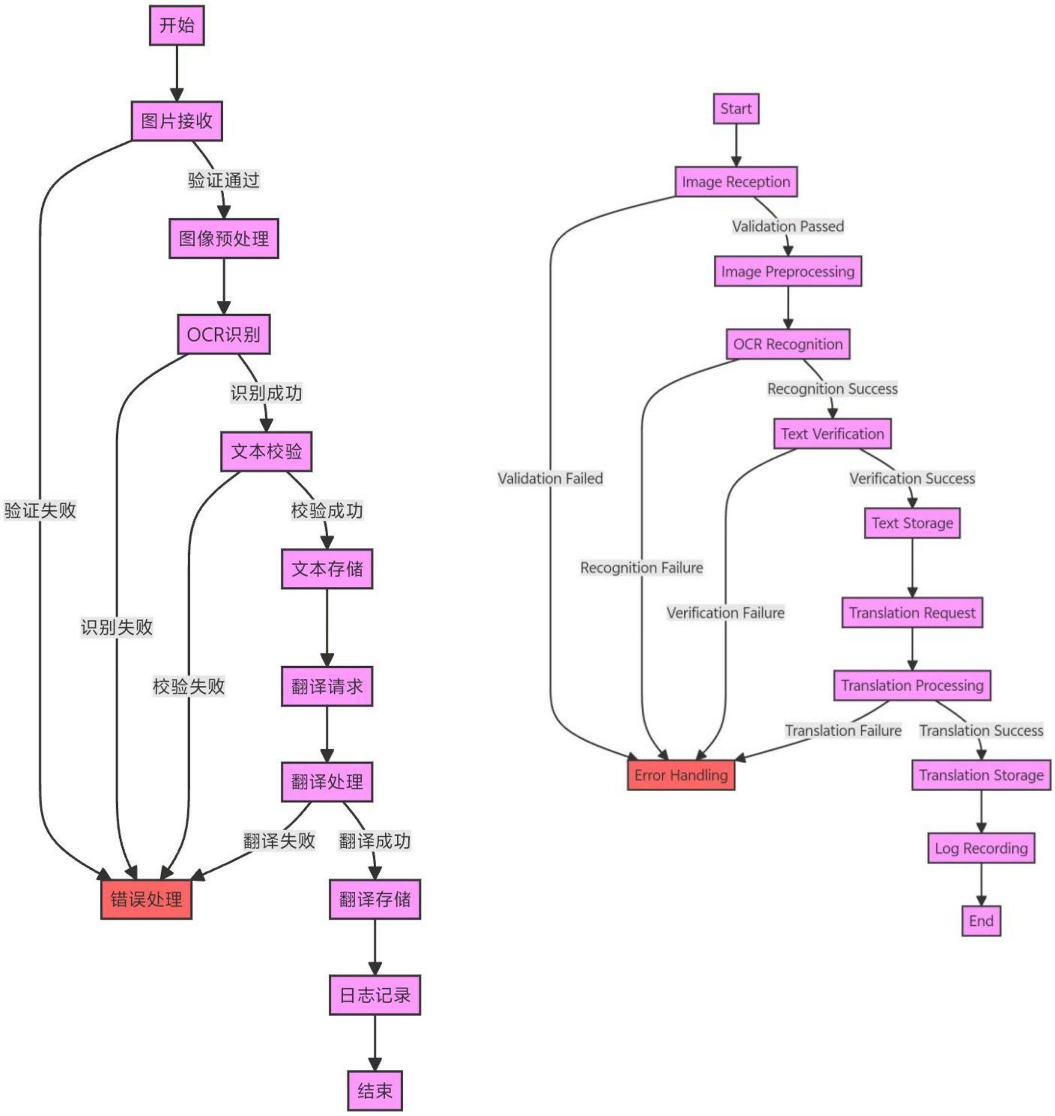
Date : 11.30

Abstract

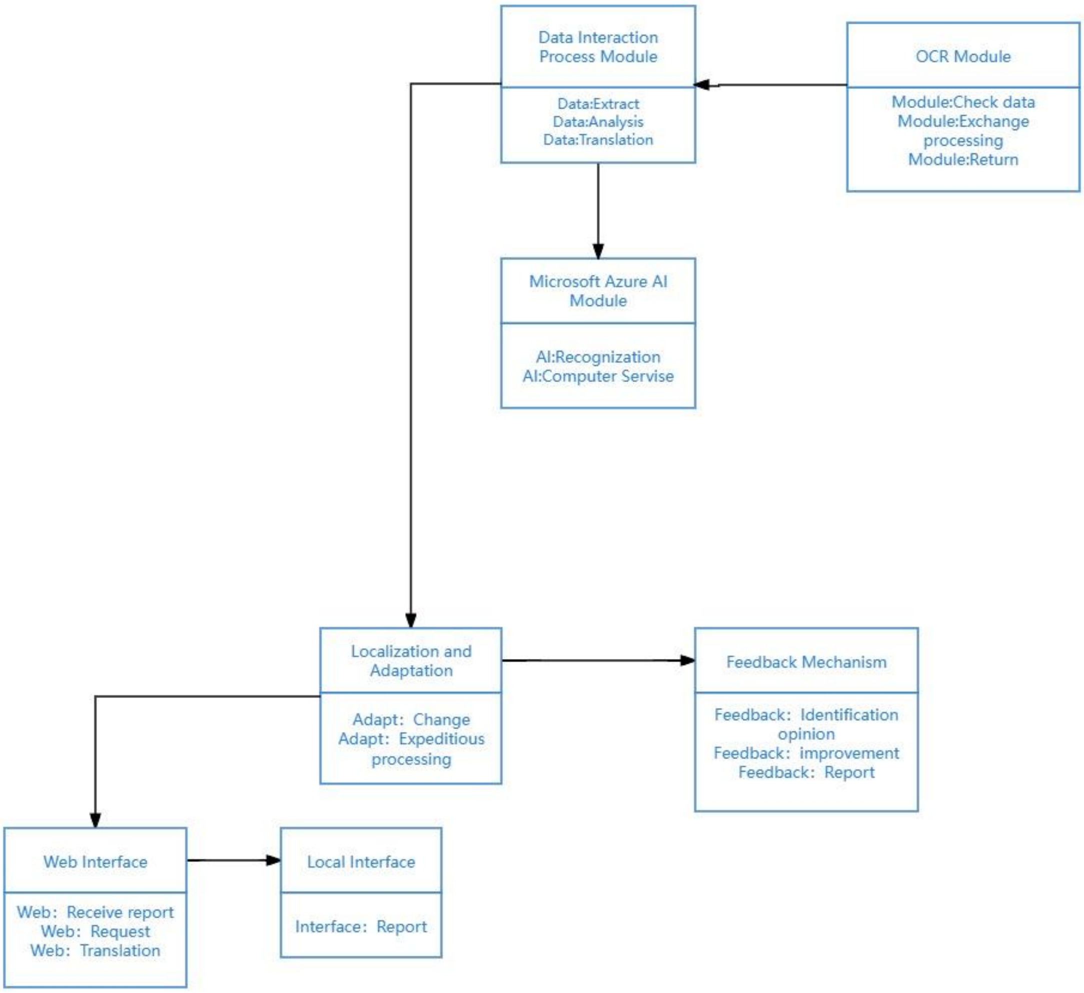
This software engineering project aims to use existing open source (or self-trained) deep learning artificial intelligence models to achieve the function of obtaining text from different scenes (images) and translating it into other languages in real time. This project relies on multiple existing open source projects and Microsoft's natural language model. Designed to provide support for anyone who is troubled by language barriers and needs localization work.

Chapter 1 Object-oriented requirements analysis Diagram

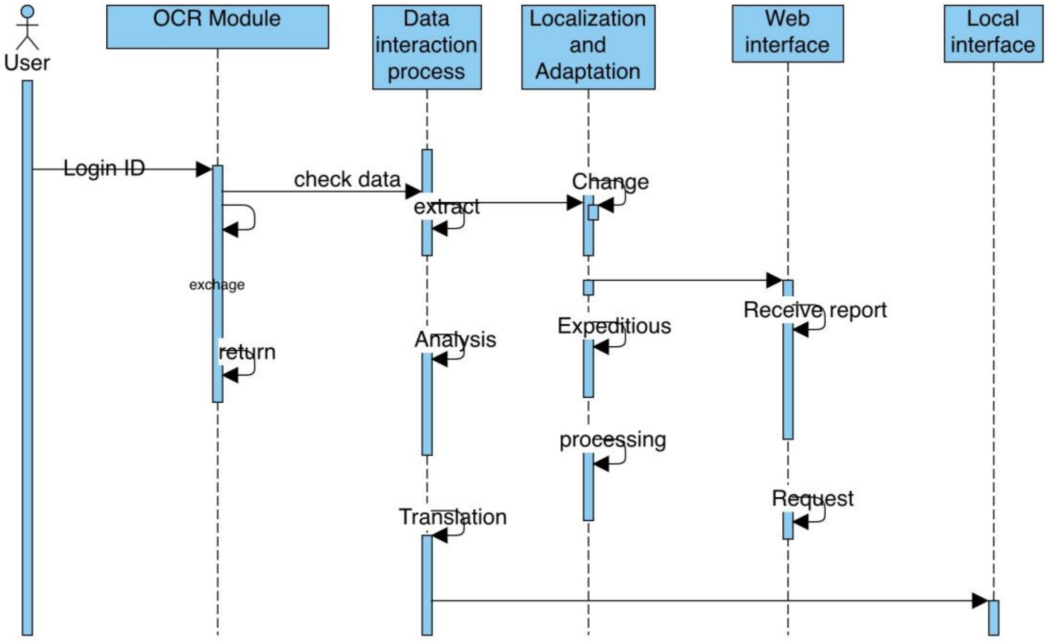
1.1 Functional modeling diagram



1.2 Static modeling (DFD)



1.3 Dynamic modeling



Chapter 2 Object-Oriented Requirements Analysis Specification

1. Introduction

1.1 Purpose

This document presents a detailed system requirements specification for an AI-driven multilingual text translation service, building upon the Business Requirements Specification (BRS) and necessitating a corresponding High-Level Design (HLD) document. Updates to the BRS or HLD will be incorporated as required during the development of this System Requirements Specification (SRS). A valid SRS, HLD, and Low-Level Design (LLD) form part of the essential operational support documentation for any application.

1.2 Notation

The Unified Modelling Language (UML) notation is used throughout this specification and its supporting documents and diagrams.

1.3 Scope

This document defines the scope of the AI-driven software system, which translates multilingual text embedded in images using Optical Character Recognition (OCR) and Natural Language Processing (NLP).

1.4 Context Diagram

A context diagram is provided to illustrate the integration of this system with related systems.

1.5 Definitions and Acronyms

- **OCR**: Optical Character Recognition.

1.6 References

The following references are integral to understanding and implementing the system:

- PaddleOCR GitHub Repository
- Azure AI Services Documentation

1.7 Overview

This section outlines the system requirements, and the referenced materials provide supporting documentation for the system’s functionalities.

2. General Description

2.1 System Functions

The primary function of the proposed system is to translate multilingual text in images using advanced AI technologies. Key functionalities include:

- **Text Detection and Extraction**: Accurate text detection and extraction from various image types using OCR, even under challenging conditions.
- **Translation**: Hybrid translation model with local and Azure AI services for extensive language support.
- **User Interface**: Intuitive web interface for easy image upload and translation.
- **Feedback Mechanism**: User feedback for translation accuracy improvement.
- **Localization and Adaptation**: Compatibility with various operating systems and deployment on high-performance GPUs or NPUs.

2.2 User Characteristics

The system caters to a diverse user base:

- **General Users**: Require quick translations via a simple interface.
- **Business Users**: Need bulk or specialized translations with integration capabilities.
- **Developers and Tech Enthusiasts**: Interested in technical details and potential API access.
- **Global Audience**: Requires broad language support and cultural sensitivity, accessible on various platforms and operating systems.

-

3. General Constraints

3.1 Software Constraints

The software adheres to the Apache License, Version 2.0:

- License Text
- **SPDX Identifier**: Apache-2.0
- OSI Approved License

3.2 Hardware Constraints

- **Web Version**: Compatible with any device using a browser (Chrome recommended).
- **Windows Version**: Requires Windows 10 & 11, at least 2GB RAM, and 200MB hard drive space.
- **Local Deployment OCR**: Requires an >8-core CPU, 12GB RAM, and optional NVIDIA 3070 GPU support.

-

4. Assumptions and Dependencies

Assumptions

1. **Availability of Datasets**: High-quality datasets for OCR and translation model training/testing.
2. **Technological Compatibility**: Compatibility with existing hardware/software and Azure AI services.
3. **User Adoption**: Users have the necessary devices and internet connectivity.
4. **Regulatory Compliance**: Adherence to data privacy and security regulations.
5. **Stable API Support**: Ongoing support from PaddleOCR and Azure AI APIs.

Dependencies

1. **PaddleOCR and Azure AI**: Dependence on their capabilities and reliability.
2. **Local Translation Models**: Availability for integration.
3. **User Feedback Mechanism**: Essential for gathering insights.
4. **Testing Environments**: Required for performance validation across platforms.

5.

5. Functional Requirements

REQ-1.1 Text Detection and Extraction

- **Description**: Accurately detect and extract text using OCR.
- **System Input**: Images in various formats (JPEG, PNG, etc.).
- **Display**: Show detected text and language translation options.
- **System Processing**: Use PaddleOCR for high-accuracy text extraction.
- **System Output**: Provide extracted text for translation.
- **Constraints**: Handle low-resolution and distorted text effectively.
- **Data Handling**: Use Unicode for text encoding.
- **Error Handling**: Implement fallback mechanisms for failed text detection.

6. External Interface Requirements

6.1 Data Interfaces

Integration with Azure AI translation services and local translation models.

6.2 User Interfaces

Design a web-based interface for image upload and interaction.

6.3 Other Interfaces

APIs for third-party application integration, if required.

Chapter 3　STAFF:

Wang Haocheng :

Chapter 2: Object-Oriented Requirements Analysis Specification

Li Yuanqi:

1.3　Dynamic modeling

Wang zi:

1.1　Functional modeling diagram

Ma Weiyan：

1.2　Static modeling (DFD)