

Automated Test Case Generation from Requirements

Introduction:

The project focuses on developing a tool that automatically generates test cases from software requirements written in natural language. The goal is to efficiently and accurately convert textual requirements into structured test cases.

Approach:

1.Preprocessing:

Data Cleaning: Text is cleaned by removing non-ascii characters.

2.Categorization:

Requirements are categorized based on prefixes (e.g., Functional, Non-Functional) for structured analysis.

3.Custom NER Model:

A custom Named Entity Recognition (NER) model is applied to extract specific entities such as actors, actions, conditions, and outcomes from the requirements. This model is crucial for accurately identifying the key components needed for test case generation.

4.Test Case Generation:

A rule-based approach is used to generate test cases using the entities extracted by the NER model.

Test cases include Title, Steps, Preconditions, and Expected Result.

5.Evaluation:

The tool's effectiveness is measured using Precision (0.67), Recall (0.75), F1 Score (0.71), and Accuracy (0.75).

Workflow:

Dataset (nrf.txt)

|

Preprocessing

|

nrf.txt(cleaned)

Categorization

|

requirements (default dict)

Preparing Custom entities file

|

custom_entites.json

Custom NER model

|

parsed_requirements.json

Rules based Testcase Generation

|

testcases(list)

Evaluation