Rasa has 2 components

Rasa nlu and rasa core

Rasa nlu performs the nlu operations and rasa core returns the response

Nlu : we give intent in here ,we give name and examples

version: "3.1"

nlu:

- intent: greet

  examples: |

    - hey

    - hello

    - hi

    - hello there

    - good morning

    - good evening

    - moin

    - hey there

    - let's go

    - hey dude

    - goodmorning

    - goodevening

    - good afternoon

And now we define the intent and also the response in the domain file

version: "3.1"

intents:

  - greet

  - goodbye

  - affirm

  - deny

  - tell\_name

responses:

  utter\_greet:

  - text: "Hey! How are you?"

  utter\_cheer\_up:

  - text: "Here is something to cheer you up:"

    image: "https://i.imgur.com/nGF1K8f.jpg"

  utter\_did\_that\_help:

  - text: "Did that help you?"

Now we have to give the intended flow in the stories file

version: "3.1"

stories:

- story: happy path

  steps:

  - intent: greet

  - action: utter\_greet

  - intent: tell\_name

  - action: utter\_tell\_name

Define entities:

We define entities in the domain file

intents:

  - greet

  - goodbye

  - affirm

  - deny

  - my\_name:

     use\_entities:

      - name

  - my\_number:

     use\_entities:

      - number

  - tell\_name

entities:

  - name

  - number

Here use entities is to define that we need to use this , here we can also add ignore entities

- intent: my\_name

  examples: |

    - my name is [varun](name)

    - my name is [Varun](name)

    - my name is [Alice](name)

    - my name is [John](name)

    - I'm [Sara](name)

    - You can call me [Alex](name)

- intent: my\_number

  examples: |

    - my number is [7736577008](number)

    - my number is [7736577008](number)

    - my phone number is [1234567890](number)

    - you can reach me at [555-123-4567](number)

    - here's my number: [9876543210](number)

We need to give examples in the intenst as [value](entity)

slots:

 name:

   type: text

   mappings:

     - type: from\_entity

       entity: name

 number:

   type: text

   mappings:

     - type: from\_entity

       entity: number

slots are used to store the entity values

in domain

- story: happy path

  steps:

  - intent: greet

  - action: utter\_greet

  - intent: tell\_name

  - action: utter\_tell\_name

  - action: utter\_my\_name

  - intent: my\_name

    entities:

    - name: "rahul"

  - slot\_was\_set:

    - name: "rahul"

  - action: utter\_my\_number

  - intent: my\_number

    entities:

    - number: "7736577005"

  - slot\_was\_set:

    - number: "7736577008"

  - action: utter\_thanks

This done to set slot and use custom actions to get and use the entities

FORMS

Forms are used to handle multiple slots

Here we can uncomment the story above for using slots

This is action for form

# Custom Actions

class ValidationRestaurantForm(Action):

    def name(self) -> Text:

        return "user\_details\_form"

    def run(self, dispatcher: CollectingDispatcher, tracker: Tracker, domain: Dict) -> List[EventType]:

        required\_slots = ['name', 'number']

        for slot\_name in required\_slots:

            if tracker.slots.get(slot\_name) is None:

                # Ask for the first slot that is not filled

                return [SlotSet("requested\_slot", slot\_name)]

        # All slots are filled

        return [SlotSet("requested\_slot", None)]

class ActionSubmit(Action):

    def name(self) -> Text:

        return "action\_submit"

    def run(self, dispatcher, tracker: Tracker, domain: "DomainDict") -> List[Dict[Text, Any]]:

        dispatcher.utter\_message(response="utter\_details\_thanks", name=tracker.get\_slot('name'), number=tracker.get\_slot('number'))

now we need to set the path In the rules

- rule: Activate form

  steps:

  - intent: tell\_name

  - action: utter\_name

  - action: user\_details\_form

  - active\_loop: user\_details\_form

This done to activate the form ,the form will run in loop as long as we get all the value

Now we need to deactivate the forms

- rule: Submit form

  condition:

  - active\_loop: user\_details\_form

  steps:

  - action: user\_details\_form

  - active\_loop: null

  - slot\_was\_set:

    - requested\_slot: null

  - action: action\_submit

Here we check whether the userdetails form is active(condition) and calls the action : action\_submit

actions:

 - action\_submit

forms:

  user\_details\_form:

    required\_slots:

     - name

     - number

This defined in the domain name

Chit chat

We need to add intent as chitchat/intentname

- intent: chitchat/ask\_name

  examples: |

    - what is your name

    - may i know your name

    - what do people call you

    - do you have a name for yourselff

- intent: chitchat/ask\_weather

  examples: |

    - whats the weather like

    - does it look sunny outside

    - oh,do you mind checking the weather for me

    - i ,like sunny days in berlin

And now int the domain file we add the response same way as chitchat/intentname

 utter\_chitchat/ask\_name:

  - text: "my name is retrieval"

  - text: "my name is varun "

  - text: "i am batman"

  utter\_chitchat/ask\_weather:

  - text: "oh the weather is good"

  - text: "o weather is not bad"

  - text: "the weather is best"