RASA

Bots de conversación

Rasa

- Es un framework para la creación de asistentes y chatbots, escrito en Python y de código abierto.

- Un bot de charla (chatbot) es un programa que simula y procesa conversaciones humanas.

- NLP (Procesamiento de Lenguaje Natural)
  - NLU: Entendimiento de Lenguaje Natural
  - NLG: Generación de Lenguaje Natural

Componentes principales

- **NLU** es la parte encargada de tomar el texto, analizarlo y descomponerlo de tal manera que el bot comprenda el contenido del mensaje.

- **Core** es la parte encargada de tomar decisiones y dar seguimiento a la conversación según lo que NLU provee.

- Ambos elementos utilizan técnicas de aprendizaje de máquina.

Componentes principales

Rasa NLU

- Enseña al bot a comprender las entradas del usuario.

Conversation_1:
U: Hello
B: Hello, how are you doing?
U: I am doing great!
B: Great. Carry on!

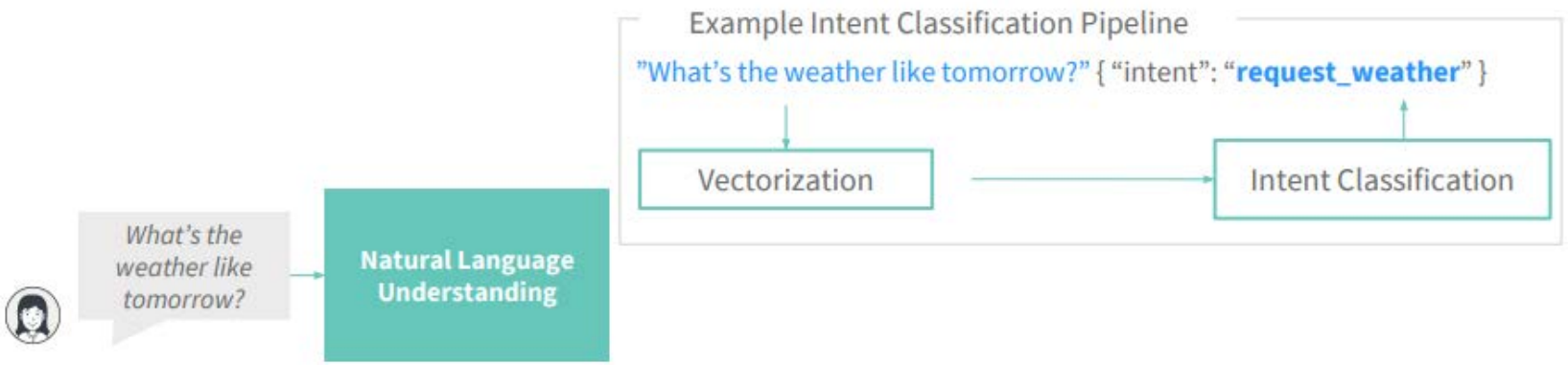
Conversation_2:
U: Hey
B: Hello, how are you doing?
U: I am very sad
B: To cheer you up, I can show you a cute picture of a cat, a dog or a bird. Choose one :)
U: A kitten
B: Here is something to cheer you up. Did that help?
U: Yes
B: Goodbye

Modelo Rasa NLU

- Los datos de entrenamiento consisten en una lista de mensajes que uno espera recibir del bot. Estos datos se anotan con la intención y las entidades que Rasa NLU debe aprender a extraer.

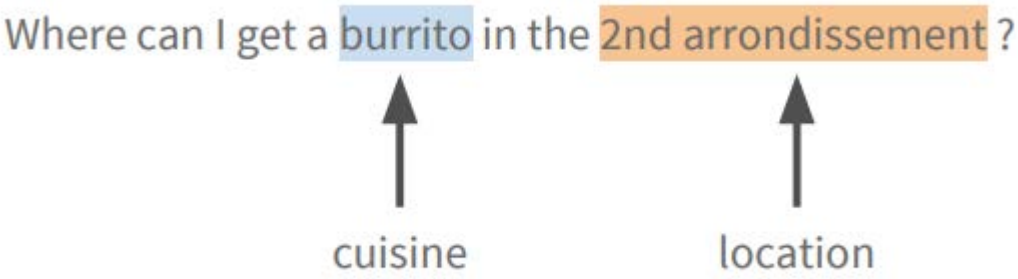
  - **Intención:** La intención describe lo que piden los mensajes. Por ejemplo, para un robot de predicción del tiempo, la oración: What's the weather like tomorrow?" tiene una request_weatherintención.

Modelo Rasa NLU



Modelo Rasa NLU

- **Entidad:** piezas de información que ayudan a un chatbot a comprender qué es lo que específicamente pregunta un usuario *al reconocer los datos estructurados en la oración* .



- Aquí la cocina y la ubicación son dos entidades extraídas.

Modelo Rasa NLU

- **nlu.md -** En este primer archivo describiremos la manera en la que el bot comprenderá el texto.

Modelo Rasa NLU

- config.yml - Por último tenemos el archivo de configuración. Aquí se definirá el pipeline, que son los procesos que se le aplicarán al texto de entrada para que el bot pueda usarlo. Cada parte del pipeline es un componente diferente que ayuda en las tareas de Lenguaje Natural, por ejemplo:

  - Tokenizers: la tokenización consiste en separar el texto en entidades llamadas tokens, para facilitar el procesamiento de este.
  - Featurizers: estos componentes convierten texto en features, que son representaciones numéricas del texto, para su uso en otros algoritmos.
  - Classifiers: los clasificadores se encargarán de decidir a qué intent se refieren los mensajes del usuario.

Modelo Rasa NLU

```yaml
language: es

pipeline:
  - name: "SpacyNLP"
  - name: "SpacyTokenizer"
  - name: "SpacyFfeaturizer"
  - name: "DIETClassifier"
    entity_recognition: False
    epochs: 50

policies:
  - name: TEDPolicy
  - name: MemoizationPolicy
  - name: MappingPolicy
```

Modelo Rasa NLU

- De igual manera, en este archivo se listan las políticas (policies) bajo las cuales se regirá la conversación. Cada política tiene un principio de funcionamiento diferente, y la combinación de ellas hace que el bot sea más robusto y pueda tomar mejores decisiones.

  - TEDPolicy: usa una red neuronal recurrente para predecir la mejor acción a tomar, basándose en las historias de entrenamiento.
  - MemoizationPolicy: decide la acción siguiente si las interacciones previas ocurrieron en las historias de entrenamiento.
  - MappingPolicy: ejecuta acciones basadas en “mapeos”; se pueden programar acciones específicas cuando se encuentren ciertos intents.

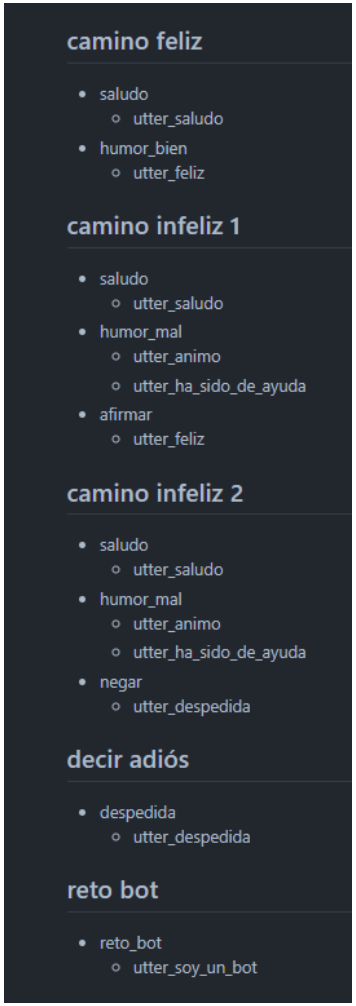
Rasa Core.

- Bot, ahora es capaz de entender lo que el usuario está diciendo, es decir, si nuestro estado de ánimo es feliz o triste. Ahora la siguiente tarea sería hacer que el Bot responda a los mensajes. Enseñaremos a Robo a hacer respuestas formando un modelo de gestión de diálogo utilizando Rasa Core.

Rasa Core

- stories.md - En este archivo se escriben las interacciones entre el bot y el usuario con las que se entrenará el asistente. Se deben incluir múltiples ejemplos de “historias”, con el objetivo de que el bot pueda decidir cuál es la mejor acción a tomar después de una interacción con el usuario.
- Los intents del usuario se colocan con asteriscos (*), mientras que las acciones o mensajes del bot se escriben con guiones (-). Las historias pueden tener títulos, sin embargo estos no afectarán el entrenamiento.

Rasa Core




```
## camino feliz
* saludo
  - utter_saludo
* humor_bien
  - utter_feliz

## camino infeliz 1
* saludo
  - utter_saludo
* humor_mal
  - utter_animo
  - utter_ha_sido_de_ayuda
* afirmar
  - utter_feliz

## camino infeliz 2
* saludo
  - utter_saludo
* humor_mal
  - utter_animo
  - utter_ha_sido_de_ayuda
* negar
  - utter_despedida

## decir adiós
* despedida
  - utter_despedida

## reto bot
* reto_bot
  - utter_soy_un_bot
```

Modelo Rasa NLU

- **domain.yml -** En este archivo debemos registrar todos los *intents* que buscaremos captar del usuario, así como los mensajes (*utterances*) y acciones (*actions*) que el bot realizará. Aquí es donde definiremos el texto que contendrán los mensajes que incluimos en las interacciones de *stories.md*.

Modelo Rasa NLU

```yaml
intents:
  - saludo
  - despedida
  - afirmar
  - negar
  - humor_bien
  - humor_mal
  - reto_bot

responses:
  utter_saludo:
  - text: "¡Hola! ¿Cómo estás?"
    buttons:
    - title: "bien"
      payload: "/humor_bien"
    - title: "mal"
      payload: "/humor_mal"

  utter_animo:
  - text: "Aquí tienes algo para animarte:"
    image: "https://i.imgur.com/nGF1K8f.jpg"

  utter_ha_sido_de_ayuda:
  - text: "¿Ha sido de ayuda?"

  utter_feliz:
  - text: "¡Genial! ¡Sigue adelante!"

  utter_despedida:
  - text: "Adiós."

  utter_soy_un_bot:
  - text: "Soy un bot, Rasa es mi motor."

session_config:
  session_expiration_time: 60
  carry_over_slots_to_new_session: true
```

Referencias

- *Rasa Learning Center*. (n.d.). Rasa.com. Retrieved November 23, 2023, from https://learning.rasa.com/
- Tenorio, J. (2020, July 15). Construyendo un Chatbot en español usando RASA — Moodbot - Planeta Chatbot. *Planeta Chatbot - Comunidad de expertos en IA Conversacional*. https://planetachatbot.com/construyendo-un-chatbot-en-espanol-usando-rasa-moodbot/