**RASA Chatbot for Shoe Recommendation**

**Subrata Ku. Sahu, 202110040**

**Soumya Kanta Moharana, 202110043**

**Abstract:**

This paper presents a Rasa chatbot designed to streamline shoe shopping by offering personalized recommendations based on user preferences. The chatbot leverages natural language understanding to interpret user queries about shoe styles, colors, and brands. It integrates a CSV database to provide accurate, real-time recommendations. By validating user inputs and filtering products according to specified criteria, the chatbot enhances the shopping experience. Future enhancements include a visual try-on feature and multilingual support. This chatbot demonstrates how AI can simplify the shopping process, making it more efficient and user-friendly.

**Introduction:**

Chatbots have revolutionized the way users interact with online services, enhancing user experiences by providing instant, personalized assistance. In the realm of e-commerce, particularly shoe shopping, chatbots can significantly simplify the process. Traditional online shopping can be overwhelming due to the vast array of products available. A Rasa chatbot tailored for shoe recommendations addresses this challenge by offering users personalized suggestions based on their individual preferences. By leveraging natural language understanding, this chatbot interprets user queries about shoe styles, colors, and brands, providing accurate recommendations from a pre-loaded CSV database. This ensures users receive relevant options quickly, making their shopping experience more efficient and enjoyable. The need for such a chatbot arises from the growing demand for seamless, user-friendly shopping experiences that save time and reduce the complexity of product selection. This Rasa chatbot exemplifies how AI can transform the online shopping landscape, offering a smarter, more intuitive way to find the perfect pair of shoes.

**Overview of Rasa Framework**

**Rasa Framework Overview:**

Rasa is an open-source framework for building sophisticated chatbots and conversational AI. Its architecture consists of two main components: Rasa NLU (Natural Language Understanding) and Rasa Core.

**Rasa NLU:** Rasa NLU is responsible for understanding the user input. It processes natural language, extracting structured data such as intents and entities. Intents represent the user's goal (e.g., searching for shoes), while entities are specific data points (e.g., shoe color, size). By training the NLU model on labeled examples, the chatbot can accurately interpret diverse user inputs.

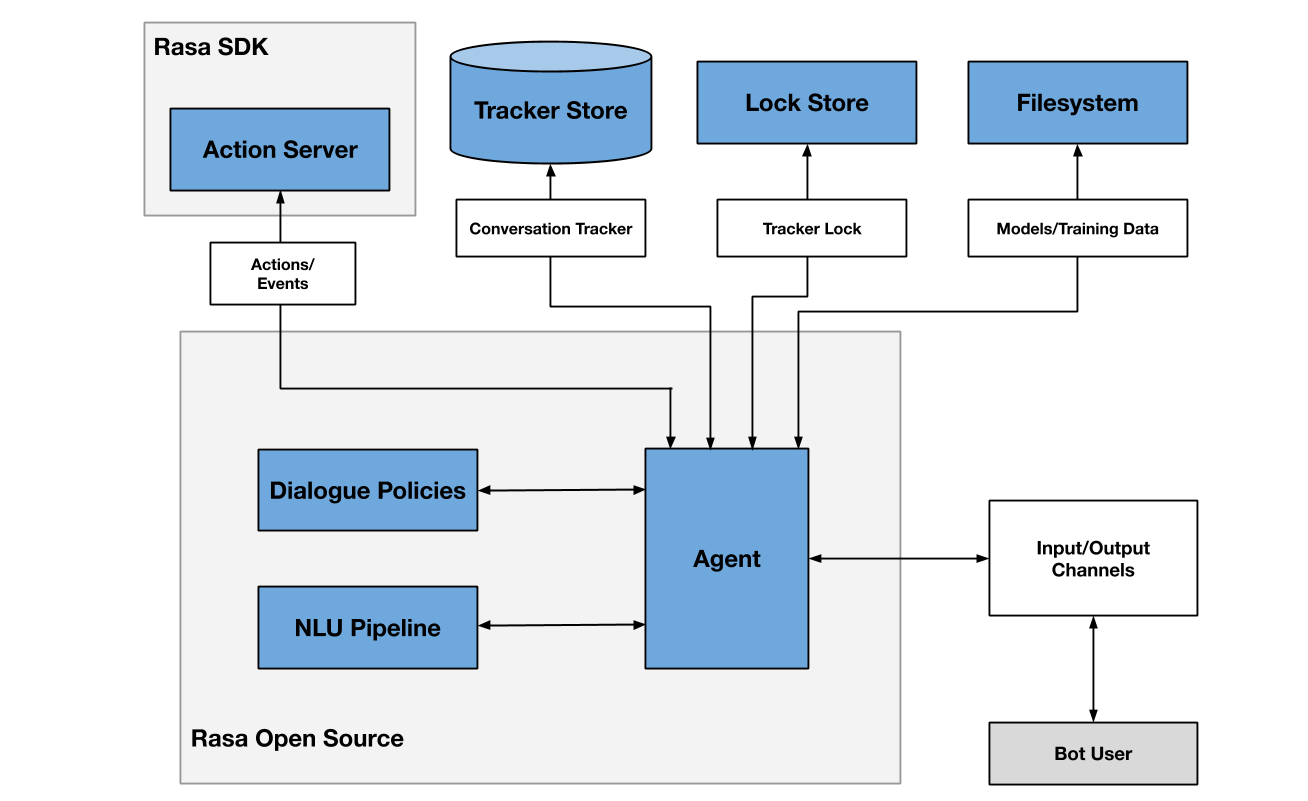
**Rasa Core:** Rasa Core manages the dialogue flow, determining the chatbot's responses based on user input and context. It uses machine learning policies to predict the next action, allowing for flexible and dynamic conversations. Core's ability to handle complex dialogues and maintain context makes it suitable for intricate tasks like shoe recommendations.

**Suitability for Building Conversational AI Applications:**

Rasa's modularity and flexibility make it ideal for developing customized conversational AI applications. Its open-source nature allows developers to tailor the chatbot's behavior and integrate with various platforms. For a shoe recommendation bot, Rasa's robust NLU ensures accurate understanding of user preferences, while Core's dialogue management facilitates smooth, personalized interactions. The framework supports seamless integration with databases and external APIs, enabling real-time product recommendations. This combination of features makes Rasa a powerful tool for creating effective, user-friendly chatbots.

### Architecture of the RASA

The RASA bot utilizes a well-structured architecture to deliver personalized and accurate recommendations. Here’s a breakdown of its functionality:



**Tracker Store and Lock Store:** The Tracker Store maintains the conversation history, enabling the bot to retrieve past interactions and ensure continuity in user conversations. The Lock Store guarantees data consistency when multiple users interact with the bot simultaneously, preventing conflicts.

**Action Server:** The Action Server executes actions determined by the dialogue policies. In the context of the shoe recommendation bot, actions include querying the product database, filtering results based on user preferences, and returning relevant shoe options.

**NLU Pipeline:** The Natural Language Understanding (NLU) pipeline processes user messages, extracting structured data such as intents (e.g., searching for shoes) and entities (e.g., color, size). This structured data allows the bot to comprehend user needs accurately.

**Dialogue Policies:** Dialogue policies decide the bot’s next action based on the conversation history and user input. For example, if a user asks for red running shoes, the policies ensure the bot queries the database for products matching these criteria.

**Agent:** The Agent coordinates all components of the Rasa framework, providing a unified interface for managing interactions. It ensures seamless communication between the NLU, dialogue policies, and action server.

**Input/Output Channels:** Users interact with the bot through various channels, such as web chats, social media, or messaging platforms. These channels facilitate user input and deliver the bot’s responses.

**Models/Training Data:** The models trained on curated datasets power the NLU pipeline and dialogue policies. These models enable the bot to understand user intents and manage conversations effectively.

In essence, the Rasa shoe recommendation bot leverages machine learning to interpret user requests and provide tailored shoe recommendations, enhancing the online shopping experience.

**Components:**

**domain.yml** defines the capabilities and behavior of the Rasa shoe recommendation bot. It lists intents (e.g., searching for shoes), entities (e.g., shoe attributes like color and size), actions (e.g., retrieving product information), and responses. This file ensures the bot understands user intents and responds appropriately, facilitating effective communication.

intents:

- greet

- inquire\_product

- order\_status

entities:

- product

- email

- color

- subcategory

responses:

utter\_greet:

- text: "Hello! How can I assist you today?"

utter\_inquire\_product:

- text: "We have a variety of {product}. What specifically are you looking for?"

utter\_order\_status:

- text: "Can you provide your order ID, please?"

actions:

- action\_insertNewFeedback

- validate\_searchedProducts\_form

- action\_selectProductInformation

- action\_filter\_products\_by\_color\_and\_subcategory

**nlu.yml** contains training data for the Rasa NLU pipeline in the shoe recommendation bot. It includes examples of user messages categorized by intents (e.g., asking about shoe styles or sizes) and annotated with entities (e.g., color, brand). This data trains the bot to accurately understand and classify user inputs

nlu:

- intent: affirm

examples: |

- yes

- y

- yep

- yepp

- okep

- okeps

- intent: greet

examples: |

- hey

- hello

- hi

- hello there

- good morning

- good evening

- moin

- hey there

- let's go

- hey dude

**stories.yml** outlines sample dialogues and user interactions for the Rasa shoe recommendation bot. It defines sequences of intents and corresponding actions (e.g., user asks for red shoes, bot retrieves matching products). These stories train the bot on how to navigate conversations based on different user inputs, ensuring coherent dialogue flows.

stories:

- story: greet and inquire product

steps:

- intent: greet

- action: utter\_greet

- intent: inquire\_product

- action: utter\_inquire\_product

- story: order status

steps:

- intent: order\_status

- action: utter\_order\_status

- action: action\_check\_order\_status

**config.yml** configures the training pipeline for the Rasa shoe recommendation bot. It specifies components like the NLU pipeline and dialogue management policies (e.g., Memoization, TEDPolicy). This file defines hyperparameters and settings for training machine learning models that enable the bot to understand user intents, manage dialogue flows, and provide tailored recommendations based on shoe preferences.

language: en

pipeline:

- name: WhitespaceTokenizer

- name: RegexFeaturizer

- name: LexicalSyntacticFeaturizer

- name: CountVectorsFeaturizer

- name: DIETClassifier

- name: EntitySynonymMapper

- name: ResponseSelector

- name: FallbackClassifier

policies:

- name: MemoizationPolicy

- name: RulePolicy

- name: TEDPolicy

- name: FallbackPolicy

**action.py** defines custom actions for the Rasa shoe recommendation bot. It includes classes like ValidateSearchedProductsForm for validating product search inputs, and SelectProductInformation for retrieving and displaying shoe recommendations based on user queries and ActionFilterProductsByColorAndSubcategory for retrieveing and displaying shoes according to the color and subcategory. These actions interact with the bot's dialogue flow, ensuring it responds accurately and dynamically to user requests, enhancing the overall user experience.

**Dataflow of Rasa System**

In the Rasa shoe recommendation bot workflow, user inputs, such as queries about shoe preferences, are first processed by the nlu.yml file. This file's training data helps the NLU pipeline understand and classify intents and extract entities like shoe color or brand. The config.yml then configures the training pipeline, specifying how these intents and entities are processed and used in dialogue management. domain.yml defines the bot's capabilities and responses, guiding how it interacts based on user intents. stories.yml outlines possible dialogue paths, guiding the bot's actions in action.py, where custom logic retrieves and presents personalized shoe recommendations based on user inputs.

**Future Enhancements**

Moving forward, our vision for enhancing the Rasa shoe recommendation bot includes integrating advanced features such as a visual try-on capability, allowing users to virtually try shoes before purchasing. Additionally, we plan to implement multilingual support to cater to a broader audience, enhancing accessibility and user engagement.

**Conclusion**

In conclusion, the Rasa chatbot for shoe recommendations streamlines the shopping experience by providing personalized suggestions based on user preferences. Its ability to understand natural language through nlu.yml, manage dialogue flows via stories.yml, and execute custom actions in action.py ensures accurate and responsive interactions. This bot not only simplifies product discovery but also enhances user satisfaction, illustrating its potential to significantly improve the online shopping journey for shoe enthusiasts.