

Proceeding Paper

FQRS: Farmer Query Redressal System Using Open-Source Framework [†]

Praveen Gujjar Jagannath ^{1,*}, Guru Prasad Marigundanahalli Siddabasappa ² ,
Prasanna Kumar Huthinagadde RamakrishnaBhat ³, Amith Keerthi Jain ⁴ and Prabhdeep Singh ²

¹ Faculty of Management Studies, Jain (Deemed-to-be University), Bengaluru 560009, India

² Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun 248002, India; gurusasad.cse@geu.ac.in (G.P.M.S.); prabhdeep Singh.cse@geu.ac.in (P.S.)

³ Department of Information Science and Engineering, PES Institute of Technology and Management, Shivamogga 577204, India; hrpbhat@gmail.com

⁴ Department of Electronics and Communication Engineering, SDM Institute of Technology, Ujire 574240, India; amithkjain@sdmit.in

* Correspondence: dr.praveengujjar@cms.ac.in

[†] Presented at the International Conference on Innovative Research in Renewable Energy Technologies, West Bengal, India, 16–17 March 2022.



Citation: Jagannath, P.G.; Marigundanahalli Siddabasappa, G.P.; Huthinagadde RamakrishnaBhat, P.K.; Jain, A.K.; Singh, P. FQRS: Farmer Query Redressal System Using Open-Source Framework. *Mater. Proc.* **2022**, *10*, 9. <https://doi.org/10.3390/materproc2022010009>

Academic Editors: Sudipta Das, K. Vasu Babu, Samrat Paul and Kunal Chakraborty

Published: 29 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: The Farmer query redressal system (FQRS) is an application designed as a chatbot. The FQRS is developed with the help of machine learning and deep learning techniques. The FQRS interacts with the farmers and this system can understand and respond to text or speech. The FQRS is developed using the nlu-based open-source framework called Rasa. The Rasa framework uses building blocks such as the nlu and nlu core. The Rasa nlu and Rasa core are the building blocks for the development of a conversational chatbot. The FQRS will use natural language processing and mimic conversations with the customers or users in a manner identical to a human being. The FQRS is implemented in Rasa 3.0. In this paper, to address farmer-related query, a chatbot has been created and the results show that the Rasa framework is more versatile than other chatbot development platforms for creating chatbots. The results show that farmers' queries can be effectively addressed with the proper training of the model.

Keywords: artificial intelligence; deep learning; chatbot; NLU; Rasa

1. Introduction

A chatbot is a conversational artificial intelligence assistant. Chatbot make use of neural networks to enhance the effectiveness replying to a user query. All natural language processing happens by understanding every word and character and interpreting the semantic meaning of the sentence. Chatbot make use of natural language processing by breaking paragraphs into sentences, sentences into words, and words into characters. A farmers' query is provided using natural language; hence, a robust natural language processing technique is required to create a robust redressal system. The open-source framework Rasa has two major components: the Rasa nlu and Rasa core. These two are responsible for understanding and identifying the farmer's intent and entities from their query. For the purpose of understanding the semantics of the farmers' queries, stemming, tokenization, and lemmatization techniques are applied. A machine learning-based predictive chatbot can give the proper guidance for a user's query [1] or it can provide a proper response for a user's query. There are two different broad classifications of chatbot: rule-based chatbots and machine learning-based chatbots. Machine learning chatbot learn from experience and provides more accurate results.

2. Related Work

Natural language processing (NLP) plays a vital role in chatbot development. An Nlu.yml file in Rasa is responsible for the classification of the intent and entities. A limited technical knowledge is enough to create a robust chatbot [2]. The Rasa framework has major two components that act similar to a black box: the Rasa nlu and Rasa core [3]. There are two different board classifications of chatbot, the rule-based chatbot and the machine learning-based conversational chatbot. Machine learning chatbots learn from experience and give more accurate results [3]. Non-technical users can also create a more effective chatbot by understanding the basic syntax of the Rasa framework. Rasa has two main components: the Rasanlu and Rasa core. The Rasa nlu helps to understand the user's intent and entities [4]. Necessary decisions can be made in the Rasa core. Different chatbot development frameworks have been compared and it was concluded that machine learning-based chatbot provide more accurate results [5]. The Rasa communication procedure is outlined in Figure 1. The Rasa stack is used in Rasa communication [6]. Chatbots can be used for various social network platforms [7] to solve the user's queries. A very well-known chatbot is deployed in social networks to answer the users' queries in one of the Spanish football leagues [8].

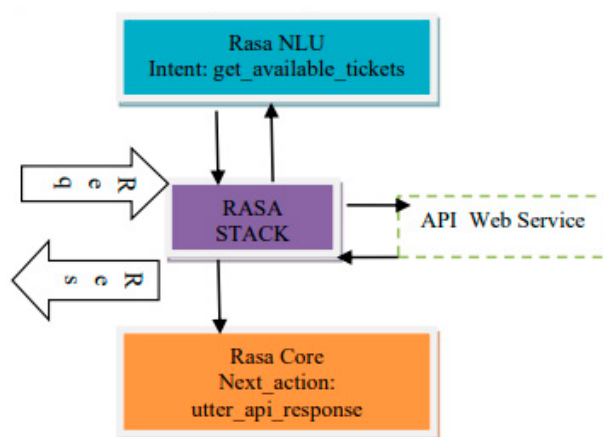


Figure 1. API communication procedure.

Conversational AI helps the developer to develop more efficient chatbots by understanding the intent and entities of the user's query. The sentiment of the user query can be identified with the help of machine learning and deep learning techniques [9–11]. The neural network model uses the Tensorflow technique for efficient accuracy [12]. Tensorflow uses dataflow graphs and operates well in heterogeneous environments at a large scale. Scikit learning can be used for entity extractors and intent classifiers [13].

3. Methods

The latest version of Rasa is Rasa 3.0. Rasa 3.0 has a slight advantage compared to Rasa 2.0. Rasa Open Source 3.0 separated the model architecture from the framework architecture, which enables us to run arbitrary model architectures. It also comes with several enhancements focused on improving the developer's experience when building conversational AI assistants with Rasa. The revamped computational backend enables us to experiment with architectures, reduce maintenance costs, and enables a collaborative development at scale. The Rasa architecture is shown in Figure 2. In this architecture, a farmer's query is provided as an input and the interpreter understands the message using natural language processing. The interpreter understands the intent and entities from the farmers' queries. Once the intent and entities are identified they will pass to the tracker, which tracks the farmer's query to provide a suitable response. A policy identifies the suitable action for the farmer's query and the action renders the suitable message to the query.

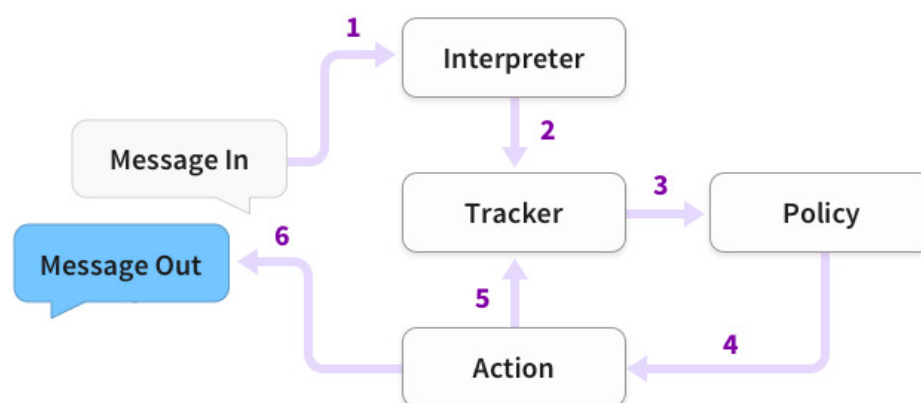


Figure 2. Rasa architecture.

The Rasa framework has various files such as `nlu.yml`, `domain.yml`, `stories.yml`, `action.py`, etc., and `nlu.yml` deals with the intent and entities. `Nlu.yml` helps in intent classification. `stories.yml` enables the flow of a potential interaction with the chatbot. Domain file in the Rasa is considered as a universe of Rasa framework which helps to utter the reply to the farmers' queries. `Action.py` file helps in customizing the chatbot's response, and further customization is possible through python code.

4. Results and Discussion

A bot created using Rasa can yield better results. The developer can customize the output of the bot according to the desired specifications. In this paper, the chatbot is prepared to address the queries related to farmers and the corresponding files are outlined in this section. The `Nlu.yml` sample data contains intent followed by examples as shown below.

- intent: my_name

examples: |

- my name is [farmer_first_name1]{"entity": "name", "value": "farmer_first_name1"}

- my name is [farmer_first_name2]{"entity": "name", "value": "farmer_first_name2"}

- my name is [farmer_first_name3]{"entity": "name", "value": "farmer_first_name3"}

The `stories.yml` displays the intent followed by the corresponding action.

- story: search fertilizers

steps:

- intent: search_fertilizers

- action: action_search_fertilizers

Similarly, the `domain.yml` file is also modified according to the agricultural and farmer's requirements.

actions:

- action_search_fertilizers

responses:

utter_greet:

- text: "welcome to farmers world"

In Rasa, a chatbot has been designed in such a way that it can effectively utter the message according to the user's query. Agriculturally based farmer-friendly chatbots can be created using different platforms such as Google dialog flow, the Rasa framework, Chatbot, Recurrent neural network, etc. The Recurrent Neural Network's (RNN) and Rasa nlu's performances are compared in Table 1. RASA NLU's accuracy is 0.95 percent whereas RNN's accuracy is 90 percent. Rasa is an open source framework for chatbot development designed by customizing python code in the `action.py` file, whereas dialogflow has a web interface and is a closed source product.

Table 1. Comparison of RNN and RASA NLU.

Sl. No.	Types	Accuracy	Integrity of Entity
1	RASA NLU	0.95	0.90
2	RNN	0.90	0.94

The recurrent neural network can also be used for text mining and chatbot development, but the results show that the accuracy is greater for Rasa whereas the integrity of the entity is greater in the recurrent neural network.

5. Conclusions

In this paper, to address the farmer-related queries a chatbot has been created and the results show that the Rasa framework is more versatile for creating chatbots compared to other chatbot development platforms. The results show that farmers' queries can be effectively addressed with the model's proper training. RASA is an nlu-based machine learning chatbot and developers can develop the chatbot by writing customized python code in an action.py file. Hence, this research asserts that Rasa is a better platform for developing chatbots in a more versatile manner. This paper focuses on the chatbot development using the Rasa open-source framework. Conversation AI-based chatbots mimic the conversations with the customers similar to human beings. The building blocks of the Rasa framework are the Rasa nlu and Rasa core. The results conclude that the Rasa platform can be used for creating a robust chatbot.

Author Contributions: P.G.J. and P.K.H.R. conceived the idea to study the farmers query redressal system. The implementation of the chatbot carried out jointly by P.G.J., P.K.H.R. and G.P.M.S. in Rasa framework. P.G.J. wrote the manuscript in consultation with P.K.H.R., G.P.M.S., A.K.J. and P.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest, or non-financial interest in the subject matter, or materials discussed in this manuscript.

References

1. Mondal, A.; Dey, M.; Das, D.; Nagpal, S.; Garda, K. Chatbot: An automated conversation system for the educational domain. In Proceedings of the 2018 International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAI-NLP), Pattaya, Thailand, 15–17 November 2018; pp. 1–5.
2. Bocklisch, T.; Faulkner, J.; Pawlowski, N.; Nichol, A. Rasa: Open source language understanding and dialogue management. *arXiv* **2017**, arXiv:1712.05181.
3. Lacerda, A.R.T.D. *Rasa-ptbr-Boilerplate: FLOSS Project that Enables Brazilian Portuguese Chatbot Development by Non-Experts*; Universidade de Brasília: Brasília, Brazil, 2019.
4. Jiao, A. An Intelligent Chatbot System Based on Entity Extraction Using RASA NLU and Neural Network. *J. Phys. Conf. Ser.* **2020**, *1487*, 012014. [[CrossRef](#)]
5. Singh, A.; Ramasubramanian, K.; Shivam, S. Introduction to Microsoft Bot, RASA, and Google Dialogflow. In *Building an Enterprise Chatbot*; Apress: Berkeley, CA, USA, 2019; pp. 281–302.
6. Sharma, R.K.; Joshi, M. An Analytical Study and Review of open Source Chatbot framework, RASA. *Int. J. Eng. Res. Technol. (IJERT)* **2020**, *9*, 1011–1014.
7. Frommert, C.; Häfner, A.; Friedrich, J.; Zinke, C. Using Chatbots to Assist Communication in Collaborative Networks. In *Working Conference on Virtual Enterprises*; Springer: Cham, Switzerland, 2018; pp. 257–265.
8. Segura, C.; Palau, A.; Luque, J.; Costa-Jussà, M.R.; Banchs, R.E. Chatbol, a chatbot for the Spanish “La Liga”. In *9th International Workshop on Spoken Dialogue System Technology*; Springer: Singapore, 2019; pp. 319–330.

9. Praveen, G.; Prasanna Kumar, H.R. Sentimental analysis for running text in Email Conversation. *Int. J. Comput. Sci. Eng. (IJCSE)* **2020**, *9*, 67–68.
10. Manjunatha, T.; Praveen, G. Performance Analysis of Indian Information Technology Companies using DuPont Model. *IUP J. Manag. Res.* **2018**, *17*, 7–14.
11. Hinton, G.; Srivastava, N.; Swersky, K. Neural Networks for Machine Learning. 2020. Available online: <http://www.cs.toronto.edu/~hinton/coursera/lecture6/lec6.pdf> (accessed on 1 January 2021).
12. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M. TensorFlow: A system for large-scale machine learning. In *12th USENIX symposium on Operating Systems Design and Implementation (OSDI 16)*; USENIX: Boston, MA, USA, 2016; pp. 265–283.
13. Gujjar, J.P.; Kumar, H.P.; Chiplunkar, N.N. Image classification and prediction using transfer learning in colab notebook. In *Proceedings of the International Conference on Computing System and its Applications (ICCSA-2021)*, Bengaluru, India, 16 July 2021; Volume 2, pp. 382–385.