International Conference on Recent Trends in Data Science and its Applications
DOI: rp-9788770040723.086

Agribot - Using Rasa Framework

Narayana Darapaneni
Director - AIML
Great Learning
Bangalore, India
darapaneni@gmail.com

Anwesh Reddy Paduri
Senior Data Scientist - AIML
Great Learning
Bangalore, India
anwesh@greatlearning.in

Divya Abraham
Student – MTecch
PES University
Bangalore, India
dvyjodd@gmail.com

Swagatika Samal
Student – MTecch
PES University
Bangalore, India
swagatikasamal16@gmail.com

Aslam Pasha
Student – MTecch
PES University
Bangalore, India
aslampasha199510@gmail.com

Vishnu Bansal
Student – MTecch
PES University
Bangalore, India
vishnukbansal@yahoo.com

***Abstract*—India's economy and society are largely dependent on agriculture, and the nation currently has one of the highest percentages of farm producers in the world. Because they are ignorant of new technologies and criteria that could help them increase productivity, farmers lose production. Farmers require technological assistance and support. From January 2015 to September 2017, all call records from the Kisan Call Center (KCC) were made public by the Indian government. Farmers and agriculture experts provided the researchers with similar KCC datasets. We identified four key areas in need of information support based on both sources: plant protection, pests (diseases), weather, and best practices.Natural language and in-depth learning The use of chatbots has increased significantly in recent years. They are used in many different fields, such as personal assistance, reservation systems, and customer service. To overcome challenges in their fields, farmers continue to rely significantly on the advice of their colleagues. Inadequate usage of these technologies has prevented farmers from receiving the crucial information on time. This project aims to develop "AgriBot," a closed-domain ChatBot for the agricultural industry. Farmers can interact with closed-domain and get expert advice about their industry. The basis for "AgriBot" is the RASA Open-Source Framework. The "AgriBot" deciphers the user's words to determine the issue and entity, then retrieves and distributes the solution from the database.We tested the Bot with existing data and found it to be promising.**

Keywords—Rasa, Deep Learning, NLP, KCC, Agribot, Personal Assistant

I. INTRODUCTION

By leveraging natural language technology to respond to questions on agriculture, horticulture, and animal husbandry, our Agri Chatbot can have a significant influence on marginalized communities. The farmer will be able to access localized information and agricultural information, including weather forecasts and the current market prices of different crops in his or her region. On the Kissan website, a farmer can send a direct message to our AI-enabled Chatbot and receive a response in their native tongue. Our approach would make it possible for the farmer to ask as many questions as they like, whenever they like, which would help current farming technology spread more quickly and to more farmers.

Below are a few examples of chatbot developments in the field of agriculture:

This research uses a virtual conversational assistant to ask a question about agriculture and receive a text response. Further improvement is possible by responding in the respondents' native tongue, and production and rainfall predictions are also possible. In its path to efficiency, sustainability, and meeting the world's food demands, agriculture is predicted to experience exciting times in the future thanks to the use of cognitive technologies. Natural Language Processing techniques are used by this conversational assistant to comprehend user inquiries in their native tongue. This will enable the machine tounderstand input queries that are grammatically incorrect [1]. The user queries go through a pre-processing stage where they are first tokenized into words, stop words like a, is, and the like are eliminated to reduce the likelihood that the queries will be classified according to their respective classes, and finally the steaming process is carried out where the words are converted to their root words. For the classification method to process the words effectively, they are first transformed into a bag of words and then into a vector form. The training dataset is then used to train the bot. The gradient descent approach is used to form a neural network from the training set of data and optimize error [2].

The same pre-processing steps, classification steps, and neural network formation are applied to the test data set. To obtain accurate findings, the class with the highest probability is iterated.

We aim to create a chatbot that can respond to farmers' basic questions and possibly offer some agricultural information. We therefore have total visibility throughout the agricultural value chain thanks to the traceability software we are utilising from Source Trace. It affects how farmers live, aids in the adoption of data-driven agriculture by an organisation, and promotes improved stakeholder relationships and trust[6]. Half of India's workforce was involved in agriculture, which contributed 17–18% of the nation's GDP. In 2014, the employment rate in agriculture and closely associated businesses including forestry, animal husbandry, and fisheries was over 31%. In 2016, these sectors contributed 15.4% of GDP.

In order to enable remote contact between users/farmers and the agricultural environment, this project aims to develop a chatbot that makes use of natural language processing. Our goal is to develop a chatbot that can respond to common inquiries from farmers and provide knowledge and solutions related to agriculture. This chatbot can learn on its own and improvise responses because it has received training in natural language processing [3].Farmers or agriculturists are the study's intended audience. Their efforts will be based on the developed model. The results of the study will therefore be useful to agriculturalists.

DOI: rp-9788770040723.086

II. MATERIALS AND METHODS

We suggest an Agribot as a solution, which is interactive for farmers and can quickly and effectively respond to their questions[4]. The workflow begins with the bot responding with a "hello" message, after which farmers will ask questions and the bot will provide potential answers based on the trained dataset we provided for training. We use RASA as a tool to create personalised AI chatbots using Python and NLU (NLU). The user can train the model and add custom actions as well.

A. Data Overview

On the Rajasthan KCC dataset, we are working. The researchers received inquiries from farmers and agri-experts that were comparable to those they discovered in the KCC dataset.

They identified two key areas in need of information support based on both sources:

- Plant Protection: In the KCC dataset, 30.59 percent of the farming calls—or nearly half of the dataset—were about questions relating to plant protection.
- Weather: In the KCC dataset, 46.67 percent of the farming calls—or nearly half of the dataset—were connected to weather-related inquiries. Farmers eagerly awaited weather updates because rain can wash away expensive insecticides that have been sprayed and because the ideal time to harvest crops depends on the weather.
- Market Information: 13.5% of farming-related questions concerned market data. It primarily contains inquiries about the mandi rate and market rate of a select few crops, including guar, groundnut, and others.
- Fertilizer Usage and Availability: In the KCC dataset, 20% of the farming calls—or nearly half of the dataset—were connected to inquiries about plant protection. It informs the farmer on the type of fertiliser needed for the crops, among other things.
- Soil testing: In the KCC dataset, 10% of the farming calls—or nearly half of the dataset—were connected to inquiries about plant protection.
- Nutrient Management: In the KCC dataset, 28 percent of the farming calls were related to plant protection related questions which is almost nearly half of the dataset.It gives insight to farmers the nutrients that need to be given to the pulses and crops. The rest of the data set includes minor factors like varieties, Government schemes, etc.

Data from the Kisan Call Center (KCC) is comprised of questions posed by farmers and the KCC's replies. The complete corpus is accessible on the official website of the Indian government, "data.gov.in," under the name Kisan Call Centre[10]. Data in CSV format is available from the years 2006 through 2020. Every month, a different catalogue is kept for each district in each state. There are 11 fields total on each call log, including the KCC agri-response. expert's

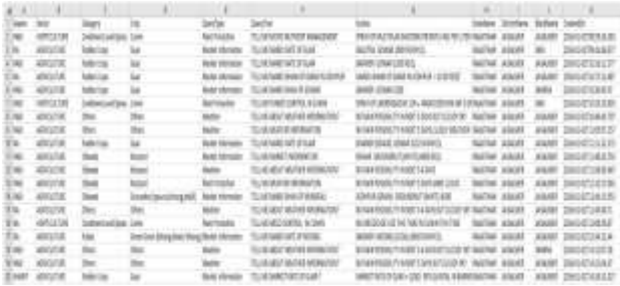

Fig.1 Different Variables in the data

- Season: Details about the season during which the query was posed.
- Sector: What industry was the question about? Examples include agriculture, horticulture, animal husbandry, and fishing.
- Category: Which category, for instance—vegetables, animals, cereals, drugs and narcotics, etc.—does the query concern within a given sector?
- Crop: Which crop is in question? Examples include potatoes, paddy, wheat, bananas, goslings, and others.
- Query Type: such as Plant Protection, Weather, Nutrient, Management, etc., fall within the general category of specific question posed.
- Query Test: The user's actual inquiry.
- KCC Answer: Response given to caller Kisan by the KCC agent.
- State Name: State in which the question has been asked.
- District: The state where the query was posed.
- Block Name: Block in the neighborhood where the question was posed.
- Created On: Date that the question was posed.

B. Exploratory Data Analysis

For our investigation and demonstration, we examined the Rajasthan data set from 2015 to 2020. There were 1,16,163 searches. More than 50% of inquiries are on the season Rabi, which is followed by Kharif and NA. According to industry-specific analysis, agriculture and horticulture account for about 85% of all enquiries.

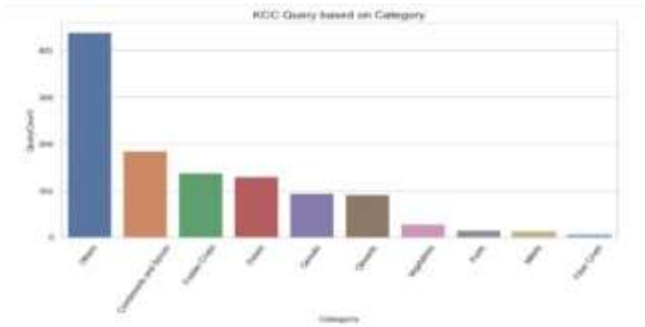

Fig. 2. Describes different Crop Types In each season

DOI: rp-9788770040723.086

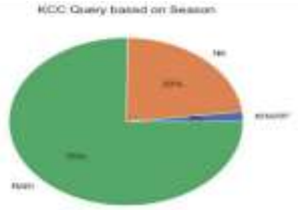

Fig 3 Shows the different Queries by the farmers across various sectors

As we look more closely at the questions, we discover that the two most important agricultural seasons in India, Kharif and Rabi, come up frequently. This data can be used to scale up or down the number of servers required and to calculate the infrastructure needed to host the solution [11]. Farmers ask questions in a number of subject areas, with agriculture and horticulture being the most popular ones. This is also in line with the fact that the majority of Rajasthan's agricultural production consists of cereals, vegetables, and pulses. This allocation could help the both of them become more effective.

Farmers have a wide range of queries, with season and crop-related inquiries being the most common.

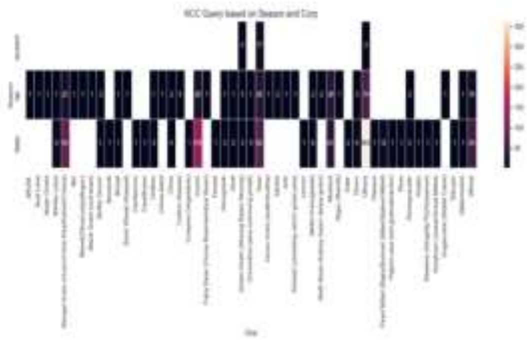

Fig 4 Shows different Queries from farmers on various crops and season of the region.

C. RASA Framework

Rasa is a tool that builds customized AI chatbots using Python and NLU (NLU). Rasa offers a platform for building AI chatbots that make use of natural language comprehension (NLU). Moreover, the user can train the model and add unique actions [12]. Slack, Microsoft Bot, and Facebook Messenger are just a few of the platforms where chatbots created by Rasa have been implemented. Rasa is composed of two primary parts:

- **Rasa NLU** (Natural Language Understanding: The open-source natural language processing programme Rasa NLU ascertains the user's request, extracts the entity from the chatbot in the form of structured data, and helps the chatbot understand what the user is saying.
- **Rasa Core :** Instead of utilising an if/else statement to determine the best course of action, a chatbot system with machine learning-based dialogue management forecasts the best path of action using a probabilistic model like an LSTM neural network. Under the hood, reinforcement learning is also used to improve the prediction of the best course of action.

D. ChatBot Implementation

Before beginning programming, Rasa must be installed because it is an open-source framework. Using a virtual environment to prevent version inconsistency across various components is strongly advised.

Environment Preparation

- Download and Install the latest python library using below commands for windows machine.

pip install python

- To create a virtual environment, decide upon a directory where you want to place it, and run the venv module as a scriptwith the directory path.

python3 -m venvchatbotenv

- Once you've created a virtual environment, you may activate it.

  \chatbotenv\Scripts\activate.bat

- Install the rasa framework on the same directory.

  pip install rasa

- initialize rasa project.

  rasainit

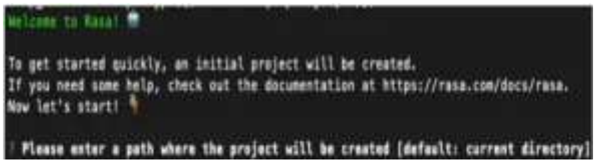

Fig 5. Shows the default screen of the Environment preparation.

E. Model Pipeline Design

A pipeline-based general framework is Rasa NLU. Rasa has a lot of versatility as a result.

Each component's data processing order is specified by a pipeline. Some components are dependent on one another. The pipeline will fail if even one of these dependence conditions fails. Rasa NLU examines each component's dependence requirements individually. Rasa will halt the program and issue the appropriate errors and warnings if any of the dependency requirements fail [13].

- **Language model component:** This loads the language model files to support the following components. There are multiple inbuilt language models available and we have used **LanguageModelFeaturizer in Agribot**.

Components	Model	Notes
SpacyNLP	spaCy	Pre-trained spaCy models need to be downloaded in advance. For more details, please check the spaCy official website.
MitieNLP	MITIE	MITIE needs a pre-trained model. You need to pre-train or download a model from the internet.
HFTransformersNLP	Transformer	To use the HFTransformersNLP component, install Rasa Open Source with pip install rasa[transformers]. HFTransformersNLP is deprecated in the newer version of Rasa 2.x, and LanguageModelFeaturizer now implements its behavior.

Fig 6 – Shows different language model files that supports Different components.

DOI: rp-9788770040723.086

- **Tokenizer component:** Languages and tokenizer components have a close relationship. All languages cannot be supported by a tokenizer. You should select the right tokenizer based on the language that will be your target. Space-splitable languages can employ "Whitespace Tokenizer." In other words, if English is your target language, you should utilise "Whitespace Tokenizer." You should utilise "JiebaTokenizer" for Chinese. In languages (like Japanese, Chinese, or Korean) that don't employ space to break words, "Mitie Tokenizer" (which makes use of a pre-trained model from MitieNLP) is typically used for word segmentation. The tokenizer you choose should be compatible with the language that your pre-trained model uses. [14] [17] Currently, "Spacy Tokenizer" supports 63 different languages. We have supported English language hence "**Whitespace Tokenizer" is used for our Agribot.**

Components	Requirement	Model	Notes
WhitespaceTokenizer			Tokenizer using whitespaces as a separator
JiebaTokenizer	Jieba	Conditional Random Field	For Chinese
MitieTokenizer	MITIE	Structured SVM	
SpacyTokenizer	spaCy	Multiple models	

Fig 7 – Shows DifferenetTokenizer for differentrequired languages.

- **Featurizer component:** Features supplied by upstream components are necessary for entity extraction as well as intent classification. To perform feature extraction, developers can employ a variety of components [18]. Developers can freely select and mix those components, as they have feature union functionality implemented. We have used **RegexFeaturizer in Agribot.**

Component	Requirements	Notes
MitieFeaturizer	MitieNLP	
SpacyFeaturizer	SpacyNLP	
ConveRTFeaturizer	Tokenization	Based on ConveRT from Poly AI.
LanguageModelFeaturizer	Tokenization	Based on Transformers from HuggingFace.
RegexFeaturizer	Tokenization	This component reads regular expression configurations from training data.
CountVectorsFeaturizer	Tokenization	Based on Bag-of-words model. Usually used in toy projects.
LexicalSyntacticFeaturizer	Tokenization	Gives linguistic features, for example, whether it is the head or tail of a sentence or whether it is a number.

Fig 8. Shows different type of feature Expressions used according to the Requirements.

- **Entity extractor component:** Rasa is compatible with several entity extraction components. With a few exceptions that can be utilized together under specific circumstances, the majority of those components shouldn't be used together. A few components can only create predefined entities; they cannot be educated on the entities that the developer creates. Rasa suggests DIETClassifier [21] because it typically performs better; as a result, we have followed her advice.

Components	Requirement	Model	Notes
CRFEntityExtractor	sklearn-crfsuite	**Conditional Random Field (CRF)**	
SpacyEntityExtractor	spaCy	Averaged perceptron	Pretrained entities.
MitieEntityExtractor	MITIE	Structured SVM	
EntitySynonymMapper	Existing entities		Standardization of synonyms.
DIETClassifier	Tensorflow	CRF on top of a transformer	
RegexEntityExtractor			Use lookups and regular expressions in the training data.
DucklingEntityExtractor			We need to run a Duckling server for it.

Fig 9. Different Entities that RASA supports.

- **Intent classifier component:** This element is used to categories each query's intent[13]. DIETClassifier is suggested by Rasa for improved performance.

Components	Requirement	Model	Notes
MitieIntentClassifier	MITIE	Structured SWM	
SklearnIntentClassifier	Scikit-learn		
KeywordIntentClassifier			Based on the keywords matching. Usually used in toy projects.
DIETClassifier	TensorFlow	CRF on top of a transformer	
FallbackClassifier			Set intent value to nlu_fallback if the other component gives too low a confidence score on intent.

Fig 10.Shows different Classifier to be used for better performance.

- **Structure output**: This generates structured data from the prediction results that have been organized. This portion of the pipeline is a built-in function rather than a component. It cannot be directly accessed by developers as a component.

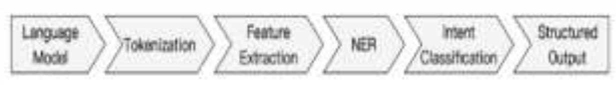

Fig 11. - Shows the Pipeline of the Data Processing.

The following pipeline is designed for Agribot

```
pipeline:
 No configuration for the NLU pipeline was provided. The following default pipeline was
 If you'd like to customize it, uncomment and adjust the pipeline.
 See https://rasa.com/docs/rasa/tuning-your-model for more information.
   - name: WhitespaceTokenizer
   - name: RegexFeaturizer
   - name: LexicalSyntacticFeaturizer
   - name: CountVectorsFeaturizer
   - name: CountVectorsFeaturizer
     analyzer: char_wb
     min_ngram: 1
     max_ngram: 4
   - name: DIETClassifier
     epochs: 100
     constrain_similarities: true
   - name: EntitySynonymMapper
   - name: ResponseSelector
     epochs: 100
     constrain_similarities: true
   - name: FallbackClassifier
     threshold: 0.3
     ambiguity_threshold: 0.1
```

Fig 12. Shows the snippet of the Pipeline that was followed.

F. Session Configuration

The session is a conversation between the user and the bot. One session can persist for multiple dialogue turns. Currently,[22].

Rasa supports two types of session configurations

- **session_expiration_time:** specifies how many minutes after the user receives the most recent message the

DOI: rp-9788770040723.086

message will expire. If it is set to 0, there is no expiration time.

- **carry_over_slots_to_new_session:** decides if the system should carry over the slots from the previous session to the new session. If set to false, the previous session's slot values will not be sent to the next session.

G. Training Data Preparation

- **nlu.yml:** We have training data for several intentions in the NLU training data file. The name of the intent and the name of the entity make it simple to determine its meaning. All of the training data for our project is kept in the data/nlu.yml file. We'll show you a portion of the training file right here.

```
- intent: bot_challenge
  examples: |
    - are you a bot?
    - are you a human?
    - am I talking to a bot?
    - am I talking to a human?

- intent: guar_rate
  examples: |
    - TELL ME MANDI RATE OF GUAR
    - TELL ME MANDI BHAV OF GOWAR
    - TELL ME MANDI BHAV OF GWAR

- intent: weather
  examples: |
    - Asking about weather forecast?
    - TELL ME WEATHER FORECASTING
    - TELL ME ABOUT WEATHER INFO
    - TELL ME WEATHER REPORT IN JAISALMER
    - TELL ME WEATHER INFORMATION
    - TELL ME WEATHER INFORMATION?

- intent: fertilizer_in_wheat
  examples: |
    - TELL ME FERTILIZER IN WHEAT
    - what will be the good fertilizer in wheat
```

Fig 13. Shows the stored training data in yml file format.

- **stories.yml:** Rasa gains knowledge from conversations and organises it through story-based instruction. A high-level semantic method of capturing talks is the tale. Together with user expressions, it also documents when the system's state changes correctly.

```
- story: happy path
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_great
  - action: utter_happy

- story: sad path 2
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: deny
  - action: utter_goodbye

- story: path3
  steps:
  - intent: guar_rate
  - action: utter_guar_rate
  - intent: goodbye
  - action: utter_goodbye

- story: path 4
  steps:
  - intent: weather
  - action: utter_weather

- story: path 5
  steps:
  - intent: pest_mus
  - action: utter_pest_mus
```

Fig 14.Shows the stored user expressions from user's.

- **domain.yml:** All the knowledge a chatbot requires is defined by a domain, including intents, entities, slots, actions, forms, and responses. All of this information provides precise definitions of a model's inputs and outputs. This is the domain file that has been prepared for Agri Chatbot.

```
intents:
  - greet
  - goodbye
  - guar_rate
  - weather
  - fertilizer_in_wheat
  - fertilizer_in_kalonji
  - disease_control_of_onion
  - disease_control_of_cumin
  - fungal_disease_control_of_gram
  - pest_mus
  - pest_must
  - control_of_blight

responses:
  utter_greet:
  - text: "Hey! How are you?"

  utter_goodbye:
  - text: "Bye"

  utter_guar_rate:
  - text: "BALOTRA  GOWAR 2900 RUPAY/Q."

  utter_weather:
  - text: "Partly cloudy,No Possibility of Rain."
  - text: "NO RAINFALL NEXT FIVE DAYS"

  utter_fertilizer_in_wheat:
  - text: "SPRAY ZINK SALPHET 33% 5 GM+BUJHA CHUNA 2.5 GM/LITER WATER"

  utter_fertilizer_in_kalonji:
  - text: "SPRAY SOLUBLE NPK  18:18:18 GRADE 10 GM/LTR WATER"
```

Fig 15. Shows the clear definitions of the inputs and outputs of a model

H. Train the Model

We can start the model to modify the weights based on training data once we have prepared all of the training data and placed the file or appropriate folder structure. In order to verify the accuracy of the module, we may also prepare the testing data. The interactive shell creation and execution commands are listed below.

- **rasa train**

This command will train the model using the NLU train data, story data, and configuration. By doing this, the model zip file will be created and saved in the default model folder.

- **rasa shell**

With this command, you can interact with the bot by running the Rasa server and launching the interactive command line interface.

III. Result And Discussions

The user was able to communicate with AGRIBot and it was able to recognise the entity and intent for which it was taught.The Bot shared the user with the proper response it had found in the database. Even for questions that were beyond the capabilities of the Bot, it performed effectively. The AGRI ChatBot is intended to seamlessly respond to beneficiary questions, and it is an effort to give mass farmers access to a channel of communication through which they can ask questions and receive answers at any time without having to be concerned about call centre operating hours or network congestion problems. We decided to add the chatbot to this platform because the Kissan website has one of the most popular chat platforms in India. The user can then continue with their questions when the bot responds to their greeting with an appropriate welcoming message about itself. The bot will check to see if the user is satisfied with the response after providing the information and will then welcome any additional questions. If the response is unsatisfactory, the bot will give the farmer the phone number to call the call centre during business hours.

The NLP algorithms can be integrated in a variety of ways to build up a solution on the RASA platform, which is the foundation around which the bot has been constructed.

DOI: rp-9788770040723.086

We can use the platform's pre-built models or feed it any custom models to create the final model utilising transfer learning with the given data.

Fig 16. The above figures show the snippet of the response from the bot for all scenarios where the confidence rate was high enough from our bot.

Possible Queries from the "FARMER"	Output From "THE AGRIBOT"
TELL ME MICRO NUTRIENT MANAGEMENT	SPRAY OF MULTIPLEX (MICRONUTRIENT) 4 ML PER LITER WATER
TELL ME MANDI RATE OF GUAR	BALOTRA GUWAR 2900 RUPAYE/Q
TELL ME MANDI BHAV OF GWAR IN JODHPUR	MANDI BHAV OF GWAR IN JODHPUR = 5150 RS/QT
TELL ME MANDI BHAV OF GEWAR	BARMER GUWAR 5200
TELL ME FUNGUS CONTROL IN CUMIN	SPRAY OF CARBENDAZIM 12% + MANCOZEB 63% WP 2 GM PER LITER WATER
TELL ME ABOUT WEATHER INFORMATION ?	NO RAIN POSSIBILITY IN NEXT 5 DAYS BUT CLOUDY SKY
TELL ME WEED CONTROL IN CUMIN	NO WEEDICIDE USE THIS TIME IN CUMIN THIS TIME
TELL ME MANDI RATE OF MOONG	BARMER MOONG (LOCAL) 6450 RUPAYE/Q
TELL ME ABOUT WEATHER INFORMATION ?	NO RAIN POSSIBILITY IN NEXT 3-4 DAYS BUT CLOUDY SKY
TELL ME ABOUT MARKET INFORMATION OF TIL	JODHPUR (GRAIN) TIL (BAJAR) 5500 RS/QT
TELL ME PEST DISEASE CONTROL IN MUSTARD	DRENCHING OF HEXACONAZOLE 2 ML PER LITER WATER IN ROOT ZONE
TELL ME CONTROL OF APHID IN MUSTARD ?	SPRAY OF DIMETHOATE 30% EC 2 ML PER LITER WATER
TELL ME DOSE OF UREA IN MUSTARD FOR TOP DRESSING	DOSE OF UREA IN MUSTARD FOR TOP DRESSING= 65 KG PER HECTER
TELL ME ROOT ROT CONTROL IN CUMIN	DRENCHING OF CARBENDAZIM 1 KG PER HA IN ROOT ZONE WITH IRRIGATION
TELL ME PEST CONTROL IN JAWAR	SPRAY OF IMIDACLOPRID 17.8% SL 0.5 ML PER LITER WATER

Fig 17 – Shows all the Queries from the Farmer tested by us and the Replies captured from the Agribot developed in Tabular form.

A beneficiary may ask some questions that are not relevant to the FAQ in response, which the bot might not be able to address. When this occurs, the bot will choose for a fallback response because it has less than a 70% confidence in its ability to respond to the inquiry.

We have tested the built model and found it to be satisfactory in providing a relevant response.

A. Discussion on the Result

Almost half of the population in India depends on agriculture for a living, making it a significant source of employment. The general public requires information on a variety of topics, including weather, seeds, crop protection, schemes, etc. The KCC was created with this objective in mind, and it has been assisting farmers for a number of years. The weak network connectivity and scarcity of contact centers, however, impede the flow of information that is so important to farmers. In an effort to close this gap, AGRIBOT chooses the advantages from both ends.

We strive to use a chat channel to provide farmers with the information they require, while at the same time using the massive KCC dataset.

In an effort to test the bot, we have built out a FAQ-like chatbot that can respond based on previously asked questions using a tiny portion of Rajasthan's KCC data set for the last five years. Our goal was to have the bot react to the beneficiary's general inquiries. The bot responds to the queries fairly well in our minified dataset.

IV. Conclusion

Agribot is the first attempt to combine two previously distinct economic sectors for the laudable goal of providing Indian farmers with information promptly and easily. In order to provide the end users (farmers) with a smooth user experience via the Kissan Website and assist them in finding answers to their questions at any time, big data technology and mobile technology have been combined.

On the one side, this will save call centre expenses, and on the other, it will allow the beneficiaries to get the information whenever they need it without having to wait for KCC to open or worry about network congestion problems. The system also introduces a balancing act in that it either responds to inquiries that it can understand from the current corpus or else offers the end users a seamless way to contact a call center for support by supplying the call center number.

Although the solution in this paper is only a proof-of-concept, it can absolutely be developed further and made into an enterprise solution because the sample testing results were encouraging. The following stage allows us to include additional solutions to expand the corpus depending on any new queries submitted by users and their resolution using the KCC representative's suggestions.

A. Future Scope of AgriBot

Virtual Conversational Assistant Chatbot

Here, a text response to a question about agriculture is provided. Further improvements can be made by responding in their own native tongues, and productivity and rainfall predictions can be added. In its path to efficiency, sustainability, and meeting the world's food demands, agriculture is predicted to experience exciting times in the future thanks to the use of cognitive technologies. Natural Language Processing techniques are used by this conversational assistant to comprehend user inquiries in their native tongue. This will enable the machine to interpret input queries that are grammatically incorrect. The user queries go through a pre-processing stage where they are first tokenized into words, stop words like a, is, and the like are removed to reduce the likelihood that the queries will be classified according to their respective classes, and finally the stemming process is carried out where the words are converted to their root words. For the classification method to process the words effectively, they are first transformed into a bag of words and then into a vector form. The training

DOI: rp-9788770040723.086

data set is then used to train the bot. The gradient descent approach is used to form a neural network from the training set of data and optimize error.

The same pre-processing steps, classification steps, and neural network formation are applied to the test data set. To obtain accurate findings, the class with the highest probability is iterated.

- TEXTUAL ASSISTANT CHATBOT: Agricultural chatbots are essential to the agriculture sector since they help all farmers and those with an interest in agricultural operations by assessing their questions and giving them the right answers. Most questions may be answered by this Question-Answer system accurately and without the need for human intervention. Greater human resource utilization and the avoidance of unnecessary costs related to the opening of additional contact centers would result from this. Natural Language Processing techniques are used by this conversational assistant to comprehend user inquiries in their native tongue. This will enable the machine to interpret input queries that are grammatically incorrect. The user queries go through a pre-processing stage where they are first tokenized into words, stop words like a, is, and the like are removed to reduce the likelihood that the queries will be classified according to their respective classes, and finally the stemming process is carried out where the words are converted to their root words.
- GPT BASED CHAT BOT: Generative Pre-Trained Transformer is referred to as GPT. The term "generative language model" refers to Chat GPT. In reality, it is regarded as an AI chat that has been programmed and created to carry on normal conversations. OpenAI, a research firm, owns Chat GPT. By creating innovations and attempting to raise their levels by assessing the quantitative and qualitative data in various ways, GPT-based chat bots have provided solutions. Upgrading to these tiers will definitely assure two-way communication. The worst error any chatbot can do is to make the user wait for a response; in this case, the farmer. This will let the user down and make them look for other options. Based simply on online data, this new machine-learning model was able to produce any kind of writing. And it was the most accurate representation of its kind ever made! The GPT-based chatbots are so advanced, according to Open AI, the company that created the technique, that they can pass the Turing test.
- INTEGRATING BERT MODEL WITH RASA: BERT is an acronym for "Bidirectional Encoder Representations from Transformers." It is a machine learning method based on transformers that Google created for pretraining natural language processing. Jacob Devlin and his Google colleagues developed and released BERT in 2018.In order to understand the context of the statement, BERT uses bidirectional training, which reads the sentence from both directions. Just keep in mind that BERT is an encoder. It is lacking a decoder.

REFERENCES

[1] Inc Future Markets. The global market for quantum dots. Technical report, 2013.

[2] Chen-Chung Liu, Mo-Gang Liao, Chia-Hui Chang, and Hung-Ming Lin,“An analysis of children’interaction with an ai chatbot and its impact on their interest in reading,” Computers & Education, vol. 189, p. 104576, 2022.

[3] Aliv Faizal Muhammad, Dwi Susanto, Akhmad Alimudin, Farah Adila, Moh Hasbi Assidiqi, and Salim Nabhan, “Developing english conversation chatbot using dialogflow,” In 2020 International Electronics Symposium (IES), IEEE, pp. 468–475, 2020.

[4] Zoroayka V Sandoval, “Design and implementation of a chatbot in online higher education settings,” Issues in Information Systems, vol. 19, no. 4, 2018.

[5] Eleni Adamopoulou and Lefteris Moussiades, “An overview of chatbot technology,” In Ilias Maglogiannis, Lazaros Iliadis, and Elias Pimenidis, editors, Artificial Intelligence Applications and Innovations, Cham, Springer International Publishing, pp. 373–383,2020.

[6] Sarthak Doshi, Suprabha Pawar, Akshay Shelar, and Shraddha Kulkarni, “Artificial intelligence chatbot in android system using open source program-o,” IJARCCE, vol. 6, pp. :816–821, 04, 2017.

[7] Pazhani. A, A. J., Gunasekaran, P., Shanmuganathan, V., Lim, S., Madasamy, K., Manoharan, R., & Verma, A. (2022). Peer–Peer Communication Using Novel Slice Handover Algorithm for 5G Wireless Networks. Journal of Sensor and Actuator Networks, 11(4), 82.

[8] Sangeeta Kumari, Zaid Naikwadi, Akshay Akole, and Purushottam Darshankar, “Enhancing college chat bot assistant with the help of richer human computer interaction and speech recognition,” In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 427–433, 2020.

[9] Andrii Yarovii, Dmytro Kudriavtsev, and Prozor Olena, “Improving the accuracy of text message recognition with an intelligent chatbot information system,” In 2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT), vol. 1, pp. 76–79, 2020.

[10] Agnese Augello, Gaetano Saccone, Salvatore Gaglio, and Giovanni Pilato, “Humorist bot: Bringing computational humour in a chat-bot system,” In 2008 International Conference on Complex, Intelligent and Software Intensive Systems, pp. 703–708, 2008.

[11] Jen-Hui Chen, Obinna Agbodike, Wen-Ling Kuo, Lei Wang, Chiao-Hua Huang, Yu-Shian Shen, and Bing-Hong Chen, “Online textual symptomatic assessment chatbot based on q and a weighted scoring for female breast cancer prescreening,” Applied Sciences, vol. 11,no. 11, 2021.

[12] Dhanabalan, S. S., Sitharthan, R., Madurakavi, K., Thirumurugan, A., Rajesh, M., Avaninathan, S. R., & Carrasco, M. F. (2022). Flexible compact system for wearable health monitoring applications. Computers and Electrical Engineering, 102, 108130.

[13] Hemkesh Vijaya Kumar, Jeevan Nagaraj, Mohammed Irfan, Nilisha Maheshwari, Pooja Balusani, Pratik Chatterjee, and Gowri Srinivasa. “Pesubot: An empathetic goal oriented chatbot,” In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1083–1089, 2018.

[14] B. S. Prasad and Chandra Mouli Venkata Srinivas Akana,“Polarity sentiment-based intelligent chat bot for judicious customer service escalation,” In 2021 2nd Global Conference for Advancement in Technology (GCAT), pp. 1–6, 2021.

[15] Vinayak Mathur, Yannis Stavrakas, and Sanjay Singh, “Intelligence analysis of tay twitter bot,” In 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), pp 231–236, 2016.

[16] Eko Handoyo, M. Arfan, Yosua Alvin Adi Soetrisno, Maman Somantri, Aghus Sofwan, and Enda Wista Sinuraya, “Ticketing chatbot service using serverless nlp technology,” In 2018 5th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), pp 325–330, 2018.