

Agricultural Related Query Clarifier System

VIJAI KUMAR S
MCA(R)
2020178060

GUIDED BY:
Dr.D.NARASHIMAN

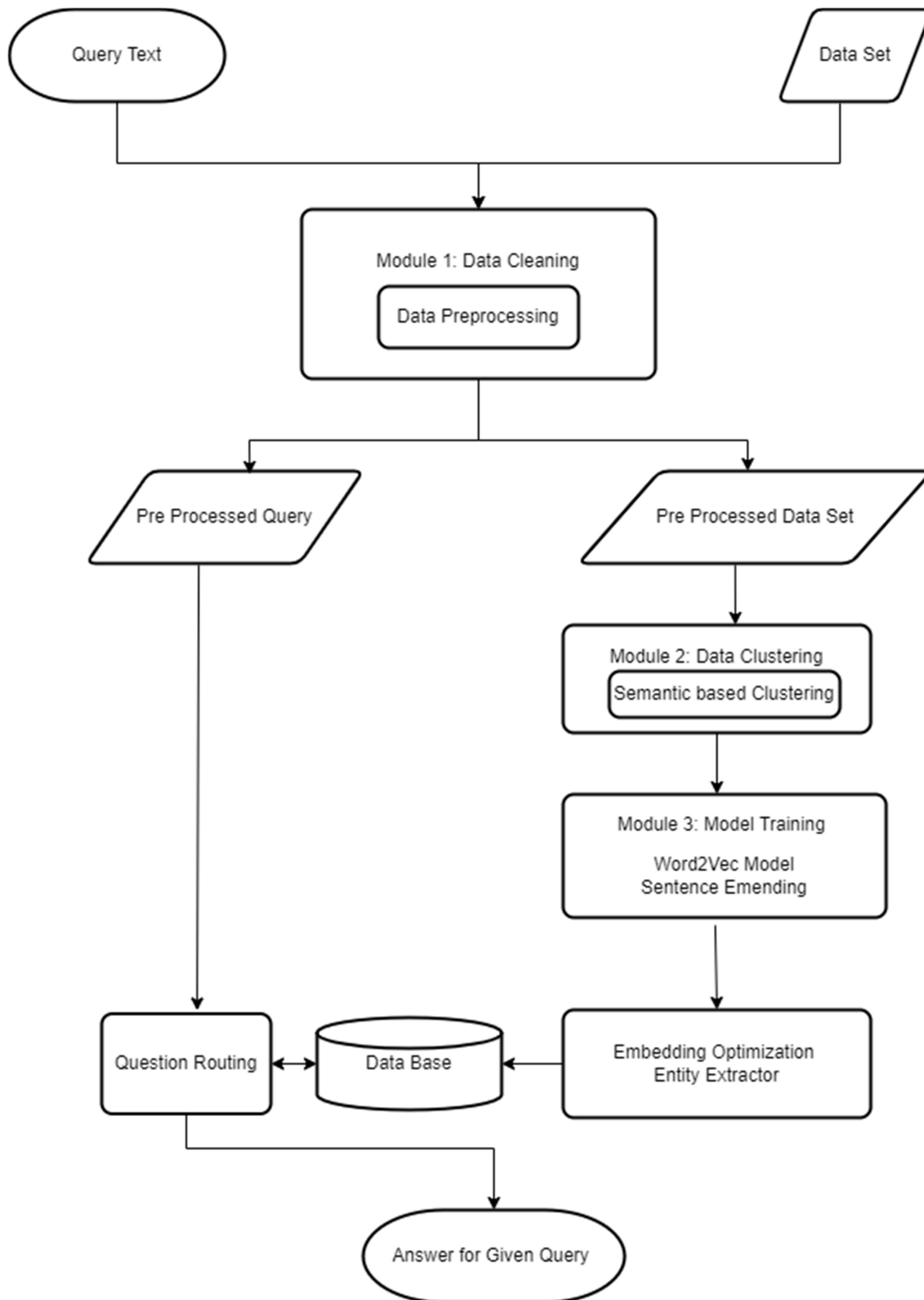
Problem Statement

To provide solution for agriculture related queries

Need for new system

- Traditionally Field officers visit the farmlands and provide training, advice, and support to the farmers
- Many of the rural villages misses this facility
- Wastage of time and money spent on obtaining information or contacting officials
- Nowadays many young generation people are come forward in Agricultural field
- By this they are struggled to clarify their any basic doubts about farming. Lots of people are calling Kissan Call Centre (Owned by Government) to clarify their basic doubts

Architecture Diagram:



Details of Data Set:

| | |
|---------------|--------------------------------------|
| Size | 300 MB |
| No of Records | 20 lakh |
| Source | Kissan Call Center(Government Owned) |

Features in Dataset:

1) **Season:** This feature contains different season about the query asked, ex: Rabi, Jayad, and Kharif.

| | |
|--------|-------------------|
| Rabi | March to June |
| Kharif | June to September |
| Jayad | October to March |

2) **Sector:** KCC classify the query based on sectors like Agriculture, Horticulture, Fisheries, and Animal Husbandry.

3) **Crop:** This field gives use the information about for which crop respective has been asked for example Apple, Banana, Rice, Wheat, Garlic, Cucumber, etc.

4) **Query Type:** Disease, Feed preparation, Training, poultry, Verities, Water management, Agricultural Mechanization, etc.

5) **Query Text:** Contains the Query statement asked by the farmer, this query has been entered by DEO from government.

6) **KCC Answer:** Contains Solution or suggestions for the Respective Query asked by farmer.

7) **State and District name:** Contains particular query asked from which State and District. This feature helps more to organize our dataset.

8) **Created on:** Contains Date and time of the particular query is asked.

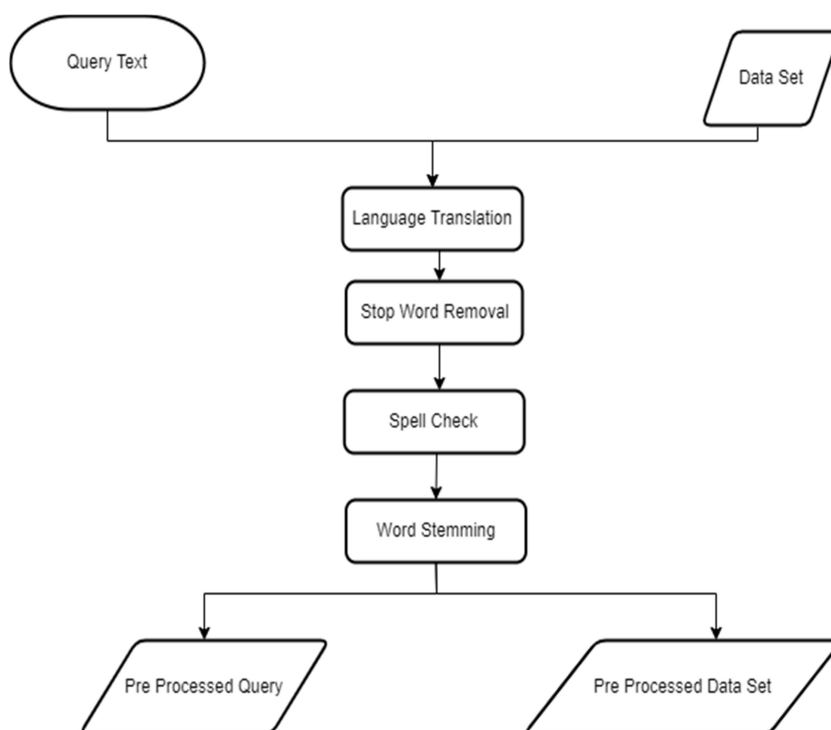
Sample Data:

| Season | Sector | Crop | QueryType | QueryText | KccAns | StateName | DistrictName | CreatedOn |
|--------|----------|------|-----------|---|--|----------------|--------------|---------------------|
| KHARIF | AGRICULT | 1037 | 3 | ASKED ABOUT THE CONTROL OF SUCKING PEST IN RED GRAM | RECOMMENDED TO SPRAY ROGOR@1.5GR/LIT | ANDHRA PRADESH | SRIKAKULAM | 2009-01-22T20:01:29 |
| KHARIF | HORTICUL | 1268 | 29 | ASKED ABOUT THE CONTROL OF MANGO HOPPERS | RECOMMENDED TO SPRAY PHASPHOMIDON 0.5ML/LT | ANDHRA PRADESH | VIZIANAGARM | 2009-01-23T16:51:08 |
| KHARIF | AGRICULT | 1075 | 3 | ASKED ABOUT THE CONTROL OF STEM BORER IN MAIZE | RECOMMENDED TO SPRAY MONOCROTOPHOS@2ML/LIT | ANDHRA PRADESH | VIZIANAGARM | 2009-01-26T06:31:30 |
| KHARIF | HORTICUL | 1279 | 29 | ASKED ABOUT THE CONTROL OF FRUIT AND SHOOT BORER IN BRINJAL | RECOMMENDED TO SPRAY PROFINOPHOS@2ML/LIT | ANDHRA PRADESH | VIZIANAGARM | 2009-01-31T15:09:02 |

Module Wise Details

1) Data Cleaning:

Data Flow Diagram:



Sample Input:

```
{ "Season": "RABI", "Sector": "AGRICULTURE", "Category":  
"Others", "Crop": "Others", "QueryType": "Government  
Schemes", "QueryText": "FARMER ASKED ABOUT PM  
KISAN?", "KccAns": "पीएम किसान साठी तुमच्या जवळील तहसील ऑफिसला जाऊन भेट  
द्यावी धन्यवाद", "StateName": "GOA", "DistrictName": "GOA  
SOUTH", "BlockName": "SANGUEM", "CreatedOn": "2022-04-  
01T10:47:15.43" }
```

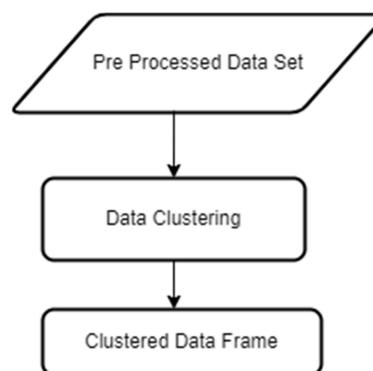
Sample Output:

| Season | Sector | Crop | QueryType | QueryText |
|--------|-------------|--------|--------------------|---------------------|
| Rabi | AGRICULTURE | Others | Government Schemes | FARMER ASK PM KISAN |

| QueryText | KccAns | StateName | DistrictName | BlockName | CreatedOn |
|---------------------|---|-----------|--------------|-----------|------------------------|
| FARMER ASK PM KISAN | Cucumber picavarial fruit control Quinolphos 30ml/Litre | GOA | GOA | SANGUEM | 2022-04-01T10:47:15.43 |

2) Text Clustering:

Data Flow Diagram:



Clustred data frame contains Query, Query Type, State and District Name, Time of Query, List of Answer for the query.

Algorithm:

```
total_dictionary = {}
with open('/content/drive/MyDrive/project/Data Set/Final_merged1.csv') as fin:
    reader = csv.DictReader(fin)
    for row in reader:
        values = [row['QueryType'], row['StateName'], row['DistrictName'], row['CreatedOn'], row['Answer']]
        query = row['Query']
        query = (" ".join(query.split()))
        # branches[row['River Name']].append(branch)
        total_dictionary.setdefault(query, [])
        total_dictionary[query].append(values)
with open('/content/drive/MyDrive/project/Data Set/Final_merged2.csv') as fin:
    reader = csv.DictReader(fin)
    for row in reader:
```

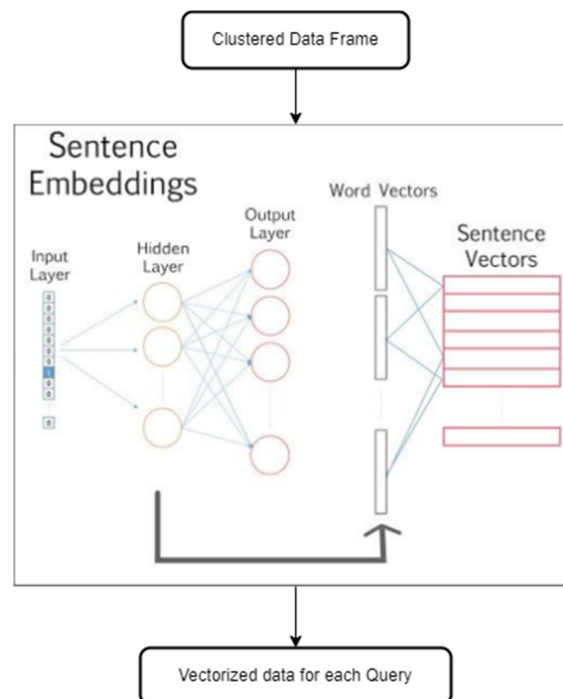
```

values = [row['QueryType'], row['StateName'], row['DistrictName'], row['CreatedOn'], row['Answer']]
]
query = row['Query']
query = (" ".join(query.split()))
# branches[row['River Name']].append(branch)
total_dictionary.setdefault(query, [])
total_dictionary[query].append(values)
with open('/content/drive/MyDrive/project/Data Set/Final_merged3.csv') as fin:
    reader = csv.DictReader(fin)
    for row in reader:
        values = [row['QueryType'], row['StateName'], row['DistrictName'], row['CreatedOn'], row['Answer']]
]
query = row['Query']
query = (" ".join(query.split()))
# branches[row['River Name']].append(branch)
total_dictionary.setdefault(query, [])
total_dictionary[query].append(values)

```

3) Training Model:

Data Flow Diagram:



Details of User Interface:

- **Login Page**

User has to provide necessary details to login like Name, Gmail ID.

The wireframe shows a web browser window titled "New Screen". The header contains a navigation bar with "Logo", "Home", "About", and "Contact" links, and "LOG IN" and "SIGN UP" buttons. The main heading is "Agricultural Related Query Clarifier System". Below this is a "Login / signup" section featuring a circular profile picture placeholder. The form includes input fields for "Name", "Email", "State", and "District", each with a "input field" label. At the bottom of the form is a "SIGN UP/REGISTER" button. The footer contains a navigation bar with "Logo", "Home", "About", and "Contact" links, and a "Shopping" button.

- **Query Clarifying page**

This page contains feature where user can enter their agricultural related query and will get response for the same.

The wireframe shows a web browser window with a header containing "Logo", "Home", "About", "Contact", "LOG IN", "SIGN UP", and "LOG OUT" buttons. The main content area is divided into two sections. The top section, titled "Product", has a "Description" label and a "Button". The bottom section is a large box containing a table with two columns: "Automated machine" and "user input". The table has four rows of sample prediction results, each with a "Query from user" label. The last row is labeled "Type your query here...".

Evaluation Metrics

In order to evaluate our metric for the prediction of the test data questions. Using these predictions and the ground truth we can define a threshold for both scores. The threshold tells the model which predictions are to be considered as good results. By using the metrics for ranking our answers, where the final predicted answer is given by

Output answer = $\text{argmax} [\text{score} (\text{question}, \text{answer}_i)]$

Reference:

- 1) Bikel, Daniel M, Richard Schwartz and Ralph M. Weischedel, " An algorithm that learns what's in a name". Machine Learning 1999 Issue No.34 pp. 211–2, 1999.
- 2) Sanjeev Arora, Yingyu Liang, Tengyu Ma, "A simple but tough-to-beat baseline for sentence embedding's", in the proceeding of International Conference on Learning Representations, 2017
- 3) Xiaoxue Shen , Adele Lu Jia, Siqi Shen and Yong Dou, "Helping the Ineloquent Farmers: Finding Experts for Questions with Limited Text" ----- 2020.
- 4) Rong and Xin, "Word2vec Parameter Learning Explained" , Coronell University , 2016.
- 5) SerhadSarica ,JianxiLuo "Stopwords in technical language processing". Journal PLOS ONE August 2021
- 6) Banerjee, S., Pedersen, T., An adapted Lesk algorithm for word sense disambiguation using WordNet (2002)
- 7) Ming Che Lee,¹ Jia Wei Chang,² and Tung Cheng Hsieh^{3A} "Grammar-Based Semantic Similarity Algorithm for Natural Language Sentences" Hindawi Publishing Corporation, The Scientific World Journal Volume 2014, Article ID 437162,17 pages.