Project report

Query answering system (in Hindi)

Saransh Rajput - 2018114016
E Nikhil - 2018114019
30 November 2019

Introduction

With the spread of internet, the number of multilingual people on the internet have grown significantly. The number of natural languages along with their dialects is estimated to be close to 4000. Out of these, Hindi takes the 5th position as the most used language with more than 200 million active users. The popularity of Hindi presses for a system which takes in queries in Hindi language and outputs the results.

A question(or query) answering system answers queries posed to it in natural language.This system provides a more lucid interface for users to get the pertinent information.
This is done by querying a structured database encompassing one or many necessary domains.
This involves retrieving information from domains like railway data.

Approach

Step 1 - User input

The input is a simple question in natural language:

गोआ से **पुने** तक ट्रैन कब जाती है?

The question does not have to be grammatically correct or complete. Much like in real life the system supports incomplete queries like

गोआ से **दिल्ली** तक

STEP 2 - Transliteration (optional)

It's highly inconvenient to use Hindi as a medium of input so transliteration option is provided to the user where the user is allowed in input in English and the system takes care of it.Although not very accurate, indic_transliteration tools work fairly well.

So the user input can be "goaa se tren kaba jaatii hai?" and it is converted to "गोआ से ट्रेन कब जाती है?"

This option is provided simply for the sake of convenience. The indic_transliteration python library has been used for this task. Alternatively, this can be done by making api calls to google input tools which is much more accurate.

STEP 3 - Query formation

The data in the database is of the form:
['12267', 'अहमदाबाद', 'कालका', '10:15', '17:20']

Where the items are:

- Train number
- Departure station
- Destination
- Time of departure
- Time of arrival

From the input query, the system extracts as much information as readily available like source station or arrival time by simple regex . Now we check for the Wh-words in the sentence which decides what type of question it is:

1) कितनी
2) कौनसी
3) कब
4) कहाँ

Let's say, we have the word **'कितनी'.** This means that the given question is querying for the count of trains for certain parameters.
The term for which the query is made is replaced by 'X' and the unimportant terms are made empty.
A question like "गोआ से पुने तक ट्रैन कब जाती है"? Takes the logic form

[` `, `गोआ` , `पुने `,`X`,` `]

STEP 4 - Filtering

Once the query has been converted to a list, we go term by term and filter the data at each step. Empty strings and 'X' are ignored.

Let's take the example "गोआ से पुने तक ट्रैन कब जाती है".

This takes the form [` `, `गोआ` , `पुने `,`X`,` `] .

We iterate over this list and filter the data available to us by `गोआ` and then by `पुने` In that order.

The lines which do not match with our terms are discarded.

STEP 5 - Results

```
1. भाषा: हिन्दी
2. भाषा: अंग्रेज़ी
3. समाप्त
आपकी पसंद>  2
अपना प्रश्न दर्ज करें : goaa se pune taka
['गाड़ी संख्या', 'स्रोत स्टेशन', 'लक्ष्य स्रोत', 'प्रस्थान समय', 'पोहोचने का समय']
['12032', 'गोआ', 'पुने', '15:45', '22:50\n']
['12031', 'पुने', 'गोआ', '10:20', '17:40\n']
['12036', 'गोआ', 'पुने', '12:10', '23:40\n']
['12037', 'पुने', 'गोआ', '11:30', '15:00\n']
['12021', 'पुने', 'गोआ', '11:00', '15:45\n']
```

Intermediate results provided for comparison

```
['गाड़ी संख्या', 'स्रोत स्टेशन', 'लक्ष्य स्रोत', 'प्रस्थान समय', 'पोहोचने का समय']
12032 गोआ पुने 15:45 22:50

12036 गोआ पुने 12:10 23:40
```

Actual results

Challenges

1) First and foremost would be the lack of resources. Most of the research papers we came across employed a machine learning based approach.

2) We based our project on Baseball model but in order to make it more robust and flexible we used the keyword approach rather than using strict templates for which lot of cases needed to be covered.

3) The transliteration library that python provides is not very accurate and does not give satisfactory results.

4) The parsers which are available for Hindi are not very accurate and are usually strict on input but our model requires flexibility.

References

1) https://www.researchgate.net/publication/267988637_Prashnottar_A_Hindi_Question_Answering_System

2) http://web2py.iiit.ac.in/research_centres/publications/download/mastersthesis.pdf.836494dbf8e87f8f.507572706f73654e6574204f6e746f6c6f6779206261736564205175657374696f6e20416e73776572696e672e2e2e28526973686162682053726976617373746176612c204d532c203230313230373631292e706466.pdf

3) https://www.researchgate.net/publication/235923461_QArabPro_A_Rule_Based_Question_Answering_System_for_Reading_Comprehension_Tests_in_Arabic

4) https://web.stanford.edu/class/linguist289/p219-green.pdf

5) https://www.sciencedirect.com/science/article/pii/S1877050916311590