Open Domain Question Answering for Telugu

Internship project

Makka Venu-B161087
Sk Shareef-B161109
Aishwarya Vuppala-B161256

OSLO METROPOLITAN UNIVERSITY
STORBYUNIVERSITETET

Photo: Ronny Østnes / OsloMet

Table of contents

INTRODUCTION

Open-Domain question answering is the task of identifying answers to natural questions from large corpus of documents. In QA, some advanced information retrieval technologies are being used to extract text from all documents and some models only concentrate on extracting the essential information.

Objective

Our main objective of the project is getting an open domain question answering model for **Telugu** language which is highly challenging task as proper Question classifier is not present for this language.

CORPUS DESCRIPTION

Before implementing the model, we need a labeled data set. We **trained** our model with a data set which consist **1455** Telugu queries which have categorized labels based on the answer type namely PERSON, ORGANIZATION, DATE, LOCATION, NUMBER, CURRENCY, TIME and PERCENTAGE.

And then we **tested** our model on the test set which has **300** labeled Telugu queries with answers.

REQUIREMENTS

- Technologies : Python 2.6 0r above
- Domain : Machine learning, Natural language processing

  Libraries:
  - nltk
  - sklearn
  - Google translate API
  - Beautifulsoup9
  - Spacy10

METHODOLOGY

Implementation of this model starts with the given input user query, which is in Telugu. To get the answer type for given telugu user query. we need a Question Classifier (QC). The user input needs to be translated into English to extract the relevant information in a search engine.To obtain that,Google translate API is used.So we can implement this project in 3 modules, those are

- Question Classification
- Information Retrieval
- Answer Extraction

Question Classification

To get the answer type for a user query, initially we have to classify the query into predefined **Named Entity Recognizer** (NERs). Then finding the classifier which works better on tagged data for Telugu would be the major task in this module.

Each query in the data set is represented in its vector form using **Term Frequency Inverse Document Frequency (TF-IDF) vectorizer**.

Pre-tagged data then divided into train and test data like 1091, 364 respectively and classifiers (**MLP, LR, SVM**) accuracies are 73.9, 73.9, 76.6 percentages respectively.

Information Retrieval

After translating the user query into english, to get the relevant information, **web scraping** technique is used to extract the unstructured data on the web into structured form using the google, Bing search engines.
To achieve this we used python library BeautifulSoup9 which is used to extract data from web pages.

After extraction, to find out most relevant sentences with respect to our query we use cosine similarity and Sequence matching methods. By using cosine similarity matrix method, we ranked the top K-sentences(better accuracy obtained for K=15) useful for the query to give the accurate answer.

Answer Extraction

NER tagging is done on the retrieved k ranked sentences using **Spacy10**. If more than one possible answers for any particular query was obtained then the first answer will be considered as final answer as because first answer containing sentence having more cosine similarity than others. This would be the final predicted answer for the user query.

At the final step, the answer will be converted to Telugu with the help of Google Translate.

RESULTS

GOOGLE SNIPPET EXTRACTION

we have used 3 classifiers SVM,LR,MLP. For each classifier we have calculated Exact match, partial match for cosine similarity greater than 0.7 and partial match for cosine similarity greater than 0.5. All the results are tablulated below.

SVM Classifier				
NER Tags	Exact Match	PM($\geq$0.7)	PM($\geq$0.5)	QC ACCURACY
PERSON	34	52	52	96
LOCATION	18	22	22	90
ORGANIZATION	12	24	32	84
DATE	16	32	48	84
CARDINAL	20	24	28	92
CURRENCY	26	42	44	90

LOGISTIC REGRESSION				
NER Tags	Exact Match	PM($\geq$0.7)	PM($\geq$0.5)	QC ACCURACY
PERSON	36	52	52	100
LOCATION	14	18	18	86
ORGANIZATION	8	14	22	68
DATE	14	22	40	70
CARDINAL	22	26	34	96
CURRENCY	26	40	44	70

MLP CLASSIFIER				
NER Tags	Exact Match	PM($\geq$0.7)	PM($\geq$0.5)	QC ACCURACY
PERSON	36	52	52	100
LOCATION	14	18	18	85
ORGANIZATION	8	24	40	82
DATE	14	20	40	72
CARDINAL	22	26	34	94
CURRENCY	26	44	48	78
For 300 queries	20	30.666	38.666	85.166

PARAGRAPH EXTRACTION

SVM CLASSIFIER						
	BING			Google		
NER Tags	EM	PM(>=0.7)	PM(>=0.5)	EM	PM(>=0.7)	PM(>=0.5)
PERSON	14	30	38	2	24	40
LOCATION	12	16	16	6	8	14
ORGANIZATION	2	10	18	6	16	18
DATE	10	16	36	6	20	32
CARDINAL	12	16	18	10	14	20
CURRENCY	28	52	60	24	46	52
For 300 queries	13	23.33	31	9	21.333	29.3333

LOGISTIC REGRESSION						
	BING			Google		
NER Tags	EM	PM(>=0.7)	PM(>=0.5)	EM	PM(>=0.7)	PM(>=0.5)
PERSON	14	26	38	2	22	36
LOCATION	12	14	14	6	8	14
ORGANIZATION	4	10	18	4	14	18
DATE	6	16	32	6	18	30
CARDINAL	4	10	10	10	16	20
CURRENCY	22	52	54	24	46	46
For 300 queries	9	21.33	27.66	8.666	20.66	22.3333

MLP CLASSIFIER						
	BING			Google		
NER Tags	EM	PM(>=0.7)	PM(>=0.5)	EM	PM(>=0.7)	PM(>=0.5)
PERSON	10	28	36	2	24	40
LOCATION	12	16	16	6	8	14
ORGANIZATION	4	10	18	6	16	18
DATE	6	10	30	6	20	32
CARDINAL	4	10	10	10	14	20
CURRENCY	22	52	58	24	46	52
For 300 queries	9.66	21	28	9	21.333	26

Accuracies of Different Classifier

NER Tags	SVM(linear SVC)	LOGISTIC REGRESSION	MLP
PERSON	98	100	100
LOCATION	90	86	85
ORGANIZATION	84	66	82
DATE	80	70	72
CARDINAL	92	96	94
CURRENCY	90	70	78
For 300 queries	89	81.33	85.166

Observations on Exact match cases

Considering EXAMPLE 1:
Extracted answer:Edgar snow
Actual answer: Edger snow
for above case cosine similarity is 0 but sequence matching is 0.9
so we used sequence matcher
EXAMPLE 2:
Extracted answer:William Shakespeare
Actual answer: Shakespeare
These cases making exact match accuracy less.

Observations on Partial Match cases

Partial matches are the cases for which cosine similarity between extracted answer and actual answer is greater than or equal to 0.7// In addition to this we have taken another partial match for which cosine similarity is greater than or equal to 0.5. Example below shows why we have used this: Question: In which year india got independence? Exracted answer is: August 15, 1947 Actual answer is: 1947 For these answer the cosine similarity is 0.58 which is less than 0.7 but answer is not completely wrong

CONCLUSION

In this project we have developed an ODQA system for Telugu language in which we trained our data set with different models.

After going through the Accuracy scores out and out we conclude our best model as

SVM Model for all 300 question gave 89 percent accuracy

Regarding Answer extraction Google snippet extraction gave faster and accurate results than paragraph extraction.

FUTURE WORK

In this project we have implemented QA system by two ways Paragraph Extraction and Snippet Extraction. We have got accurate and faster results with Google Snippet Extraction. But for some queries snippets are not available. so for those queries for which snippets not available if we perform paragraph extraction we can increase the overall accuracy. In future we combine both snippet extraction and paragraph extraction to increase the accuracy of the system.

REFERENCES

[1] Priyanka Ravva, Ashok Urlana, Manish Shrivasthava.2020. AVADHAN:System for Open-Domain Telugu Question Answering. International Institute Of Information Technology,Hyderabad.LTRC lab.

[2] Zhiping Zheng. 2002. AnswerBus question answering system. In Proceedings of the second international conference on Human Language Technology Research. Morgan Kaufmann Publishers Inc., 399–404.

[3] Dell Zhang and Wee Sun Lee. 2003. A web-based question answering system.(2003).

[4] Spacy model tutorial from website https://towardsdatascience.com/named-entity-recognition-with-nltk-and-spacy-8c4a7d88e7da