Automated Question Answering Assistant

Rutuja Kitukale
Department of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi-Mumbai, India

Nachiketh Pai
Department of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi-Mumbai, India

Prathamesh Nerkar
Department of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi-Mumbai, India

Archana Shirke
Department of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi-Mumbai, India

Jerin Jose
Risk Quotient Consultancy Private Limited,
India

2021 International Conference on Innovative Trends in Information Technology (ICITIIT) | 978-1-6654-0467-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICITIIT51526.2021.9399601

Abstract— IT firms have a number of clients from various sectors. These vendors have many questions or queries which are to be answered manually. Also the amount of questions asked are huge in numbers approximately in the club of 500 to 600. This includes reading the document thoroughly and extracting all the related information regarding the question and then representing the data in appropriate format required. Answering all such questions manually in a limited period of time is quite a tedious task. This leads to more time consumption and increases the human labour behind it. The project aims at developing an automated system which would create a deep learning model that will input the questions present in any format and answer them automatically with the help of the algorithm and give the output in the required format, thus simplifying the work of searching the answers in a given extract and finding the least error prone answer to the question thus increasing the accuracy. This will also reduce time required behind studying any document and framing the answers from them. This system will be able to answer all types of questions. The system can be used for generating answer keys for online exams. The system will encourage the research that returns answers directly instead of keyword extraction from the documents with ample number of queries. Even it can be used for open domain searching of information over the internet.

I. INTRODUCTION

Question-Answering systems (QA) were developed in the early 1960s which uses artificial intelligence. Early Question answering system was designed to answer questions about the US baseball league over a period of one year, easily *answered* which was closed domain i.e related to a specific topic. Automated answering Assistant is an information retrieval system which provides appropriate answers for the specified queries from a database which contains the answers of the specified queries. The system will check for the answers that the company has provided to the database that was created and send that answers to the client

A successful growing company is the one that satisfies the clients with their work. Many of these clients have queries regarding the policies of the company, their working, queries about their product. Answering such queries which are about hundreds in number from a single client manually is quite a tedious and time-consuming job to do. Hence, to make it less time consuming and easier for the customers to get their queries answered quickly we have tried to develop a question answering system that would automatically answer the queries of list of questions in less amount of time hence saving time and labor.

IT firms have a number of clients from various sectors. These vendors have many questions or queries which are to be answered manually. This includes reading the document thoroughly and extracting all the related information regarding the question and then representing the data in appropriate format required. Answering all such questions manually in a limited period of time is quite a tedious task. This leads to more time consumption and increases the human labor behind it.
Our project aims at reducing the time required for the process and also reduces the labor behind thus giving most relevant answers for the questions. Clients in the IT field have many queries that need to be answered. Each query is different and unique, but most of the time, the clients ask many queries that are frequently repeated. The number of people requesting a response for every query is also very vast. The Company cannot answer every client individually for their given query. This makes things very time consuming and a lot of technical assistants would be needed to answer every query provided by the clients (individually).To overcome the difficulties mentioned above, we have proposed an automated answering assistant. Our Project aims at developing a system that would fetch the context from the pdf given as input from the user thus finding all the answers for the list of questions given as input with more accuracy.
The objectives of our system are as follows:

1. User Interface
   A website for the user to input a para or a pdf/word document for the context, to input a particular question or list of questions that need to be answered.
2. Q/A Model

That would fetch context for the question answer system from the pdf process the question and answer it automatically.

This automated answering assistant has a wide range of applications and scope. It could be used as closed domain as well as open domain. It can be used to generate online answer keys for exams hence creating a teaching learning platform for

students which will be able to answer queries asked by students regarding a particular topic. This can be used in corporate offices, banks where queries by clients, customers can be answered quickly and accurately without human intervention. It also has great scope over open domain searching over the internet, it can be used in the medical field, as a doctor assistant .The System gives answers for all the questions in spite of the question being invalid. One need to manually answer the question that has been answered wrong.

II. LITERATURE SURVEY

A. Related Work

S. Jayalakshmi, Dr.Ananthi Sheshasaayee have presented a paper on Automated Question Answering System Using Ontology and Semantic Role this system uses semantic similarity. This paper fundamentally centers around comparability measure dependent on the posted question, and finding the fitting significance between the words. The proposed approach is utilized to break down and estimating the comparability between the words. It presents the Web And semantic information Driven automatic question answering system.

WAD Algorithm

Input:User's question (Qi)
Output: Corresponding answer for a given question (Aj)
Step1: For each Qi do
Step2.Preprocessing- Convert user query into NL Question format for input.
Step 3. Relevant Snippets –> Discard the irrelevant contents Step 4. Conditional Probability - -> to avoid ambiguity information retrieval.
Step 5. Ranking - - > use Head Word, NP, New Word
Step 6. Ranking factors - -> Path link distance and threshold limit
Step 7. Answer Type identification- -> use head word and WH
Step 8. Answer Validation - -> Determine Question Type (Qt) and Information Selection (Ai)

This study has presented a WAD approach for managing the ambiguity of answers and answer selection complexity in QA system. The main goal of WAD approach is to increase the accuracy of the document matching.

Merve Unlu, Ebru Arisoy, Murat Saraclar have introduced a paper containing an inquiry replying (QA) system created for spoken talk preparing. The inquiries are contribution to the system in composed arrangement and the appropriate responses are gotten back from address recordings. The principle thought of the system here is to consequently locate the applicable responses to text based or spoken inquiries from a book or spoken record. Particularly, the machine cognizance task, understanding an entry of text and addressing questions identified with that section has been examined by numerous specialists and huge upgrades have been gotten on the SQuAD datasets.

The two modules of the system are:

1.Automatic speech recognition
They have constructed an ASR framework utilizing the acoustic and the content information. The acoustic models were prepared utilizing the Kaldi toolbox.

The language model is a 4-gramlanguage model trained using the SRILM toolkit. The word error rate (WER) results for 3 different acoustic models (GMM-si: speaker independent GMM, GMM-sa: speaker adaptive GMM and DNN: deep neural network model) were obtained as 13.0%, 10.5% and 6.7% respectively.

2. Question Answering
The QA system was executed utilizing PyTorch 0.4.1. The QA syatem is a more testing task than the perusing understanding style QA task either on content or spoken setting because of long sections coming from the ASR records and inquiries with long replies. To underscore these difficulties and show the adequacy of the proposed entry question importance approach, they have arranged a few train-test situations. The QA model for every situation was prepared for 30 ages with 32 examples in a single clump utilizing 150 dimensional concealed vectors. All models were tried both on the reference records and the incorrect ASR records of the test information. The inquiry answer sets for the ASR records were gotten by adjusting the ASR records of the test information with the comparing reference records.

The system is based on competitive neural network-based reading comprehension models. They proposed a passage- question matching stage to handle a realistic scenario where the answer for each question is searched in a chapter of the course lectures.

Udhaya Surya A, Nandhni K, Ishwarya M have proposed a QAS which is an information retrieval technique
It Provides answers for given inquiries instead of web joins. It can do numerous things like recovering data from the site, online assessment, instruction, medical care, sports, geology, and so forth The web crawlers like Google, Yahoo give a rundown of web joins as the outcomes for the given question.

The systems which are accessible for giving the response to the given question from web records:

A. START
START (Syntactic Analysis utilizing Reversible Transformations) QAS responds to the inquiries regarding urban communities, lakes, workmanship, culture, climate, guides and nations and so forth It parses the inquiry and matches the questions with the information base and present the most suitable data to the client.

B. AskMSR
AskMSR changes the inquiry into substrings. For instance, for the given inquiry "who is the Prime priest of India?" was perceived as the accompanying substrings "PM of India", and it searches the web. From the removed outcome, AskMSR

separates a couple of substrings, and channels them dependent on the inquiry terms and accumulates the appropriate response. AskMSR extracts a few substrings, and filters them based on the question terms and gathers the answer.
The methodology is intended for improving the question Answering by utilizing web bits for productive recovery of answers to the inquiry. We have removed website pages and pieces from web index to give outcomes to the given inquiry.

B. Existing System

Intelligent Question Answering System based on Artificial Neural Network

The proposed question answering system (QAS) uses deep cases along with Artificial neural network to understand the contents present in the documents. It divides the sentences into small knowledge units and assign deep case to each word to improve the quality of knowledge extraction.
Here, they have proposed a method to create a deep neural network from the documents provided by the user and storing them for future. They process the questions asked by the user and then comprehend the questions and understand what answer is required and then try to find the answer from the deep neural network created previously from the documents which is provided by the user.

III Proposed Work

Modules

The system is divided into three modules.

1. Paragraph extraction.

Scanning the entire pdf for the question is a tedious job and takes time so In paragraph extraction the system will successfully extract a particular chapter mentioned on the excel sheet and accept it as the context.

2. Answering system.

For the answering system we have used bert() pretrained model trained on Squad dataset containing 100k rows. The model used is bert-large-uncased-whole-word-masking-fine tuned-model. After paragraph extraction it fetches the question, processes it and returns the most relevant answer, for a list of questions the model runs on a loop and provides answers for all the questions.

3. The User Interface.

The user Interface is developed on Django framework which integrates the model with the inputs given by the users.

Working

1. User have two options to input one is a paragraph and other is pdf. If a user enters a paragraph it directly takes it as text wrap and answers the query asked by the user.

2.If the user enters a pdf then while giving a list of input as question it fetches the particular chapter of the pdf mentioned in the excel sheet.
3.Then the model processes the question running in a loop and provides answers for the following questions running in a loop.

Policy No.	Policies	Questions
2	Information Security Policy	What is the Frequency of review of ISMS policy?
		What is the Frequency of MRM?
3	Organisation of Information Security	Is there a defined team to implement and maintain ISMS activities
		Do you maintain contact with relevant authorities and special interest groups?
4	Project Management Policy	At which stage Information security impact analysis is carried out during the project management?
5	Asset Management Policy	Do you have updated Asset inventory for all of your assets?
		What is the frequency of review of asset inventory?
		What is the classification scheme followed by the organization?
6	Data Retention and Disposal Policy	How long do you retain data?
		How do you dispose electronic media?
		How do you dispose paper media?
7	HR Security Policy	Do you perform reference checks for all employees?
		Does the agreement with the employee contain a confidentiality clause?
		Does your employee undergo information security training during induction?
		What is the frequency of Refresher trainings?
		What is the duration of unused user account getting blocked?
		What will happen if an employee breaches any of the organization's information security policies and practices.
9	Change Management Policy	Do you follow a Change Management process?
		Who will approve all the changes?
		What are the categories of changes?
10	Network control	Who maintains the Network diagram?
		How can an employee gets an access for a blocked website if required?
		What is the inactivity session time out on servers and network devices?
		Is sysyem time synced with NTP Server?

Fig 2.1 Format of excel sheet containing questions

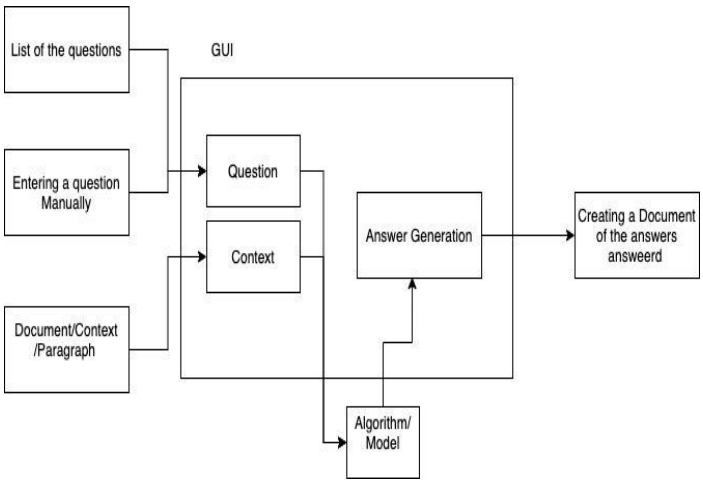

Fig 2.1: Architectural Block Diagram

1.List of Questions/questions : User have two options. It can ask a question or upload a excel file containg list of questions.

2.Document context/Paragraph : User then have to upload paragraph or a pdf/word file. for a word file the System implements paragraph extraction and feed the following data in the model.

3.Model: The data fetched from the GUI is then provided to the model i.e our System which then processes the question and forms the answer.

4.Document generation: The model is then made to run on loop which processes every question on loop and generates a document with answers for every question besides it.

IV Experimental Results

A. Preprocessing of dataset

Stanford Question Answering Dataset (SQuAD) is a perusing perception dataset, comprising of inquiries presented by swarm laborers on a bunch of Wikipedia articles, where the response to each address is a portion of text, or range, from the comparing understanding section, or the inquiry may be unanswerable. **SQuAD2.0** has updated 100,000 from SQuAD1,0 with over 50,000 unanswerable questions written by crowd workers to look familiar with present ones. The preprocessing of the dataset was done with the help of GloVe. GloVe is an unsupervised learning algorithm to get the vector representation of the words. The dataset was available in the following format after the preprocessing. After preprocessing the data, the data is visible in the below format

que_idxs/context_idxs:Question_char_idxs/Context_char_idxs

Y1 : index where answer begins
Y2 : Index where answer ends
Id : general id for the tuple

B. Training the model on LSTM networks

The system will initially check the type of question from the list of questions present. Depending upon the type of question, the answer will be searched from the document. For factual or descriptive types of questions, we will be using LSTM (Long short term memory) approach and will be able to find the answer for the particular question. We will be using the standard RNN architecture but will be replacing the RNN units by LSTM units.

For using the LSTM algorithm, we need to convert the text document into vectors with the help of any converting method of which will convert the document into embedded form. Then we will feed the model into the algorithm repeatedly and will achieve the answer for the related document. We will be using NLP for formating the final answer of the particular document.

Training of the data in the 5 layers.
1. Embedding Layer : Embedding word indices to get word vectors
2.Encdoing Layer : Encode the embedded sequence
3. Attention Layer : Apply an attention mechanism (here it means to couple the question and context/paragraph)

4. Model Encoder Layer : encoding the coupled sequence again

5. Output Layer : applying simple RNN Bidirectional LSTM in this layer and providing the output.

```
[11.01.20 00:58:09] Saved checkpoint: ./save/train/model1-12/step_100000.pth.tar
[11.01.20 00:58:10] New best checkpoint at step 100000...
[11.01.20 00:58:10] Dev NLL: 07.96, F1: 14.20, EM: 10.38, AvNA: 48.12
```

```
[11.02.20 04:32:58] Evaluating on dev split...
100% 5951/5951 [01:22<00:00, 72.25it/s, NLL=17.5]
[11.02.20 04:34:22] Dev NLL: 17.52, F1: 15.00, EM: 10.45, AvNA: 47.74
[11.02.20 04:34:23] Writing submission file to ./save/test/test1-06/dev_submission.csv...
```

C.Using the BERT Model

Bert uses transformer encoding blocks. The transformer encoder uses attention (multi Headed self attention) mechanism that learns contextual relations between words or subwords in text. For the task of obtaining questions, BERT takes the context and questions padded into a sequence. The input embeddings are the sum of the token embeddings and the segment embeddings. The input then gets processed where the tokens are added into the sequence.

Token embeddings: A [CLS] is a token which is at initial part of the sequence and a [SEP] token is inserted between the context and the question of the sequence.

Segment embeddings: An indicator between 2 sentences A and B. This allows the model to distinguish between sentences.In the BERT Model for fine tuning it initialises the start vector and an end vector. The probability for the start word of answer is calculated by taking a dot product between the final embedding of the word and the start vector, followed by a softmax over all the worlds. The word with the highest probability value is considered. Similarly the end word of the answer is also calculated.

[CLS]	101
what	2,054
is	2,003
the	1,996
frequency	6,075
of	1,997
review	3,319
of	1,997
is	2,003
##ms	5,244
policy	3,343
?	1,029
[SEP]	102
the	1,996
im	10,047
##s	2,015
document	6,254
are	2,024
revived	10,570
once	2,320
nu	16,371
##ally	3,973
.	1,012
[SEP]	102

Fig 3.1 Tokenization

D. Requirement analysis

Requirements	Hardware	Software
1.	Minimum 4gb RAM	Py Torch
2.	Intel i3 processor(miminum)	Transformers
3.	400 gb memory space	Pandas
4.	Operating system(windows/ubuntu)	Django 3.1.4
5.		Openpyxl 3.0.5
6.		Textract 1.6.3

Table 3.1 Requirement analysis

V. RESULT ANALYSIS

A. Paragraph extraction

The below figure is the screenshot of paragraph extraction where the system is able to fetch a particular chapter from a pdf given by the user.

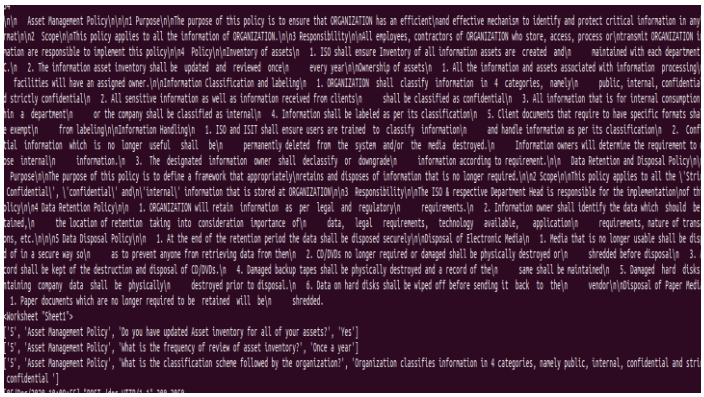

Fig 4.1 paragraph extraction

B. GUI

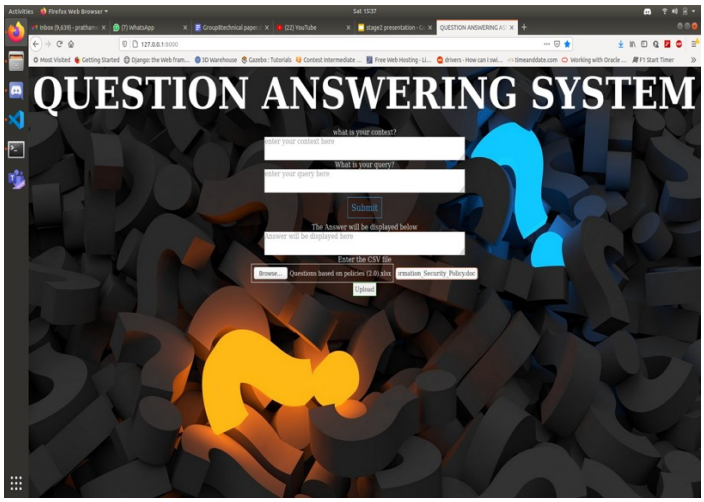

Fig 4.2 GUI

The Graphical user interface is built on Django that contains an input for context or a pdf or word document. It has an input for Query and its output or list of questions and answers beside it.

C.Query output

```
[ ] question = "what is the purpose of the policy"
    answer_question(question, bert_abstract)

    Query has 89 tokens.

    Answer: "to set the direction for information security for organization"
```

Fig 4.3 Query output

The Figure 4.3 shows how the query is executed within the model showing tokens present in that answer.

VI. CONCLUSION

We have successfully implemented the Question answering system using two approaches i.e. Using LSTM and BERT. For LSTM Trained 5000 rows for 30 epochs. It took around 24 hours for around 28 epochs with an accuracy of 48.12and the pre-trained model gave us accuracy of 87.07 due to the limitation of context we have used paragraph extraction where the model will search for particular chapter from the user entered pdf or word file and successfully return the most relevant answer. We could also be able to use pre pre-trained continual model for improving the performance of the model on a given domain specific data.

	EM	FI	Accuracy
LSTM	10.38	14.20	48.12
BERT	86.7	92.8	87.07

Table 5.1 Comparison of approaches

REFERENCES

[1] Ahlam Ansari, Moonish Maknojia, Altamash Shaikh, “Intelligent question answering system based on artificial neural network” , 2nd IEEE International Conference on Engineering and Technology (ICETECH), 17th& 18thMarch 2016, Coimbatore, TN, India.

[2] Merve Unlu, Ebru Arisoy, Murau Saraclar, “Question answering for spoken lecture processing”, in ICASSP 2019.

[3] Menaha R, Udhaya Surya , Ishwarya M, “Question answering system using web snippets”, in International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2017)

[4] S. Jayalakshmi , Dr. Anand Sheshasaayee , “Automated question answering system using ontology and semantic role”, in International Conference on Innovative Mechanisms for Industry Applications(ICIMIA 2017)

[5] Wang, X., Xu, B., & Zhuge, H. (2016). Automatic Question Answering Based on Single document. URL:https://ieeexplore.ieee.org/document/7815082

[6] Andrenucci, A., & Sneiders, E. (n.d.). Automated Question Answering:review of main approaches URL:https://ieeexplore.ieee.org/document/1488857