A
Project Report
On
Question Answering System

(CE351 – Software Group Project III)

Prepared by

Anushka Sandesara (17CE097)
Pratyay Sapovadiya (17CE100)
Dhruvi Shah (17CE107)
Saloni Shah (17CE116)

Under the Supervision of

Prof Mrugendra Rahevar

Prof Sneha Padhiar

Prof Dipsi Dave

Submitted to

Charotar University of Science & Technology (CHARUSAT)
for the Partial Fulfillment of the Requirements for the
Degree of Bachelor of Technology (B.Tech.)
in Computer Engineering (CE)
for 6th semester B.Tech.

Submitted at

Accredited with Grade A by NAAC
Accredited with Grade A by KCG

U & P U. PATEL DEPARTMENT OF COMPUTER ENGINEERING
(NBA Accredited)
Chandubhai S. Patel Institute of Technology (CSPIT)
Faculty of Technology & Engineering (FTE), CHARUSAT
At: Changa, Dist: Anand, Pin: 388421.
April-2020

DECLARATION BY THE CANDIDATES

We hereby declare that the project report entitled Question Answering System submitted by us to Chandubhai S. Patel Institute of Technology, Changa in partial fulfilment of the requirement for the award of the degree of B.Tech in Computer Engineering, from U & P U. Patel Department of Computer Engineering, CSPIT/FTE, is a record of bonafide CE351 Software Group Project III (project work) carried out by us under the guidance of Prof Mrugendra Rahevar, Prof. Sneha Padhiar, and Prof. Dipsi Dave. We further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

Anushka Sandesara (17CE097)
Pratyay Sapovadiya (17CE100)
Dhruvi Shah (17CE107)
Saloni Shah (17CE116)

Prof Mrugendra Rahevar
Assistant Professor
U & P U. Patel Department of Computer Eng,
CSPIT/FTE, CHARUSAT-Changa.

Prof Sneha Padhiar
Assistant Professor U & P U. Patel
Department of Computer Eng,
CSPIT/FTE, CHARUSAT-Changa.

Dipsi

Prof Dipsi Dave
Assistant Professor U & P U. Patel
Department of Computer Eng,
CSPIT/FTE, CHARUSAT-Changa.

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

Accredited with Grade A by NAAC
Accredited with Grade A by KCG

CERTIFICATE

This is to certify that the report entitled **Question Answering** is a bonafied work carried out by **Anushka Sandesara (17CE097), Pratyay Sapovadiya (17CE100), Dhruvi Shah(17CE107), Saloni Shah (17CE116)** under the guidance and supervision of **Prof Mrugendra Rahevar, Prof. Sneha Padhiar, and Prof. Dipsi Dave** for the subject **Software Group Project III (CE351)** of 6th Semester of Bachelor of Technology in **Computer Engineering** at Chandubhai S. Patel Institute of Technology (CSPIT), Faculty of Technology & Engineering (FTE) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidates themself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred by the examiner(s).

Under the supervision of,

Prof Mrugendra Rahevar
Assistant Professor
U & P U. Patel Department of Computer Eng,
CSPIT/FTE, CHARUSAT-Changa.

Prof Sneha Padhiar
Assistant Professor
U & P U. Patel Department of Computer Eng,
CSPIT/FTE, CHARUSAT-Changa.

Dipsi
Prof Dipsi Dave
Assistant Professor
U & P U. Patel Department of Computer Eng,
CSPIT/FTE, CHARUSAT-Changa.

Dr. Ritesh Patel
Head - U & P U. Patel Department of Computer Engineering,
CHARUSAT, Changa, Gujarat.

Chandubhai S. Patel Institute of Technology (CSPIT)
Faculty of Technology & Engineering (FTE), CHARUSAT
At: Changa, Ta. Petlad, Dist. Anand, Pin:388421. Gujarat.

ABSTRACT

Question Answering (QA) system is an information retrieval system in which a direct answer is expected in response to a submitted query, rather than a set of references that may contain the answers. These QA systems are classified as Text based QA systems, Factoid QA systems, Web based QA systems, Information Retrieval or Information Extraction based QA systems, Restricted Domain QA systems and Rule based QA systems. Recently, QA has also been used to develop dialog systems and chatbots designed to simulate human conversation. Traditionally, most of the research in this domain used a pipeline of conventional linguistically-based NLP techniques, such as parsing, part-of-speech tagging and coreference resolution. QA systems aim to retrieve point-to-point answers rather than flooding with documents or even matching passages as most of the information retrieval systems do.

ACKNOWLEDGEMENT

With immense pleasure, we would like to present this project report on Machine Learning, "Question Answering System”. It has been an enriching experience for us to undergo our training under Prof. Mrugendra Rahevar, which would not have possible without the goodwill and support. The development of this project has given us a wide opportunity to think, implement and interact with various emerging technologies and fields.

We would like to express our gratitude to our Head of the Department and also our internal guide Prof. Mrugendra Rahevar, who guided us and supported throughout the project. We would also like to thank Prof. Sneha Padhiar and Prof. Dipsi Dave. Without them we would have not been able to complete this project successfully. We are very thankful to all the faculties of CE Department (CSPIT) that helped us at every step whenever needed.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

If we think about question answering as a human activity then what do we expect to happen when a person is asked a question to which they do not know the answer? In this situation it is likely that the person to whom the question was asked would consult some store of knowledge such as a book, library, the Internet etc. in order to find some answer so that they could determine the answer to that question. Then they could return to the person who had originally asked the question and tell the answer. They could also report on where they have found the answer which would allow the questioner to place some confidence in the answer. What they would not do would be simply to give a book or maybe copious of documents which might contain the answer.

Many people would like to use the Internet as a source of knowledge in which they could find answers to their questions. Although many search engines suggest that you can ask natural language questions, the results that they return are usually sections of documents which may or may not contain the answer but which do have many words in common with the question. Existing system information retrieval systems take set of keywords as input to search engine and outputs the list of ranked documents to the user which contain keywords of the given input. Even though it has lot of documents it does the best in ranking the documents but sometimes fails to provide the pertinent document to the user. Thus, the user has to study the documents retrieved and find answers which are apposite.

The Question Answering System can help the user in finding the precise answer than performing the tedious task of studying the entire documents and understanding them. It reduces a lot of time and effort of the user in searching the internet or any other sources. Also, it overcomes the disadvantages of information retrieval such as retrieval of documents that are not answerable to user, not handling natural language questions and wasting a lot of time on searching the entire list of documents. We used Stanford Question Answering Dataset (**SQuAD**) to implement this system.

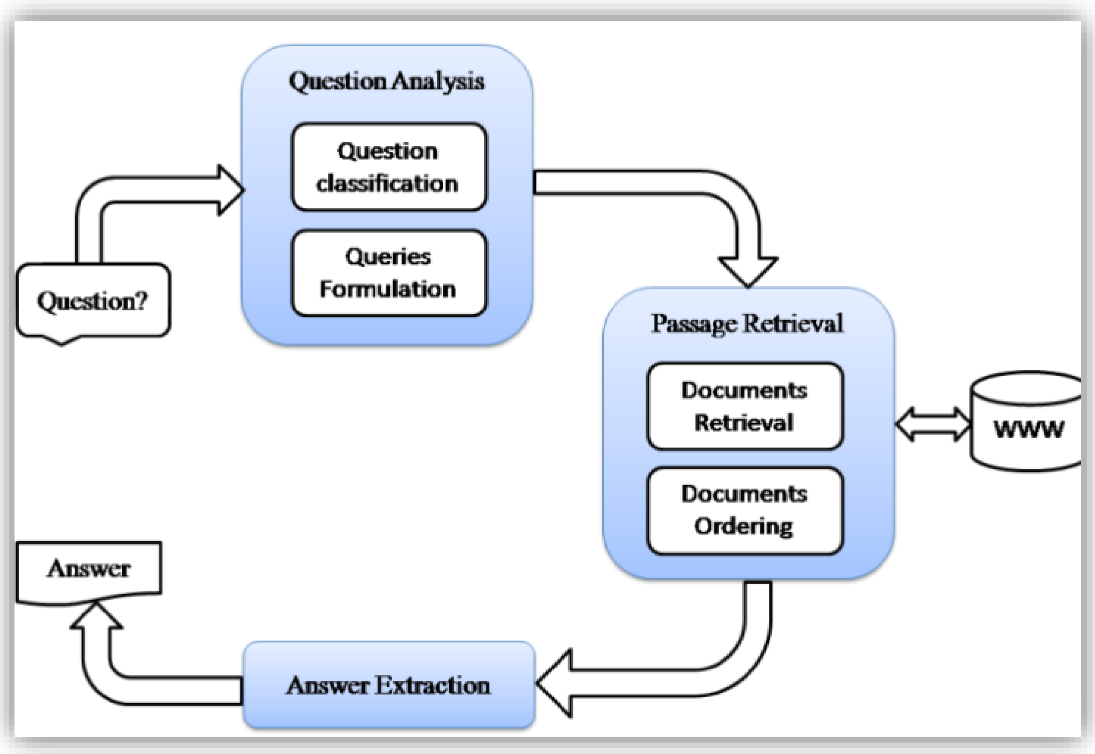


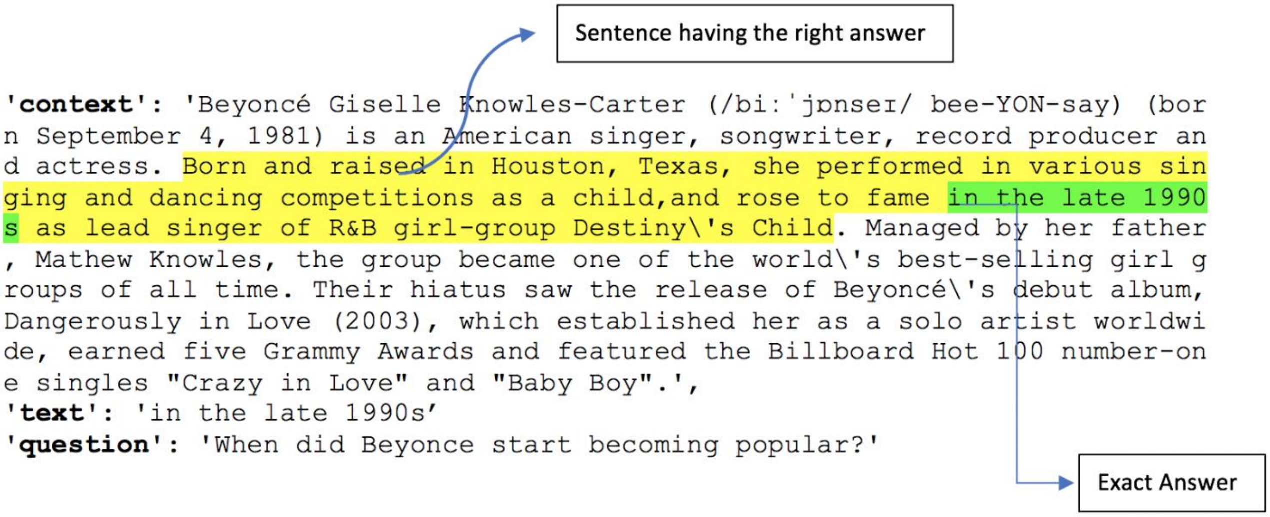
Figure 1.1 General Architecture

1.1 SQuAD DATASET

Stanford **Qu**estion **A**nswering **D**ataset (SQuAD) is a new reading comprehension dataset, consisting of questions posed by crowd workers on a set of Wikipedia articles, where the answer to every question is a segment of text, or span, from the corresponding reading passage. With 100,000+ question-answer pairs on 500+ articles, SQuAD is significantly larger than previous reading comprehension datasets. Suppose: -

Problem

For each observation in the training set, we have a context, question, and text. Example of one such observation.



```
'context': 'Beyoncé Giselle Knowles-Carter (/biːˈjɒnseɪ/ bee-YON-say) (bor
n September 4, 1981) is an American singer, songwriter, record producer an
d actress. Born and raised in Houston, Texas, she performed in various sin
ging and dancing competitions as a child,and rose to fame in the late 1990
s as lead singer of R&B girl-group Destiny\'s Child. Managed by her father
, Mathew Knowles, the group became one of the world\'s best-selling girl g
roups of all time. Their hiatus saw the release of Beyoncé\'s debut album,
Dangerously in Love (2003), which established her as a solo artist worldwi
de, earned five Grammy Awards and featured the Billboard Hot 100 number-on
e singles "Crazy in Love" and "Baby Boy".',
'text': 'in the late 1990s'
'question': 'When did Beyonce start becoming popular?'
```

Figure 1.2 Example of Question-Answering

The goal is to find the text for any new question and context provided. This is a closed dataset meaning that the answer to a question is always a part of the context and also a continuous span of context. I have broken this problem into two parts for now –

- Getting the sentence having the right answer (highlighted yellow)
- Once the sentence is finalized, getting the correct answer from the sentence (highlighted green)

CHAPTER 2

LITERATURE REVIEW

The following section represents the literature review on open and closed domain question-answering system. An open domain question answering system is dealt with questions on anything that depend on general ontologies and world knowledge. Closed domain question answering is engaged with questions within a specific domain like music, educational and medical etc. A closed domain question answering is more efficient than open domain for retrieval of better answers.

2.1 LITERATURE REVIEW ON OPEN DOMAIN QUESTION ANSWERING

The following section gives an overview of open domain question answering on semantic based technique, pattern matching, question type, linguistic approach and other approaches.

2.1.1 Semantic Based Technique

Hammond (1995) proposed FAQ Finder that helps users to navigate through the existing FAQ (Frequently Asked Question) collections. WordNet is one of the lexical dictionaries applied to extract semantic concepts.

Ask Jeeves is another automated FAQ answering system that extracts existing question-answer pairs from its knowledge base. Its knowledge base is derived from FAQ collections, and it applies shallow language understanding while matching a user question to FAQ entries in the knowledge base.

PowerAqua (2011) is a question answering system based on data combined from different and distributed sources. The probable downside of using different sources may result in incomplete, low quality or noisy data. Using heterogeneous data is significant in the future of Semantic Web and Question Answering Systems. It is not designed with a particular 70 ontology and also doesn’t make suppositions about the vocabulary of structure of datasets.

2.1.2 Pattern Matching Technique

Natural Language Interface to Data Bases (NLIDB), converts the user questions into a SQL query based on the semantics and database schema (2003). This method is based on the keyword identification or related to pattern matching for the queries. The drawback of this system is that it works on limited applications.

Querix (2009) is also a pattern matching natural language interface for answering that restricts only with regard to the beginning of the questions. It uses the syntactic structure of the questions to match with the knowledge bases. This method does not use ontology, rather asks user for clarification if any ambiguity occurs.

2.1.3 Question Type

LASSO constructs a question type hierarchy for analysis of TREC data on open domain. This approach automatically finds the type of questions and the type of answers. Some words of the questions like “What is not a Stack?” or “What is Last in First Out?” does not give exact answer (1999). Question types are associated with the concepts in the knowledge base. The answer type of the question helps to find the entity type based on the hierarchy.

2.1.4 Linguistic Procedure

The task of question answering automation has been considered from the initial days of computational linguistics. During 1960s various question answering systems are developed. The early methods had a fixed domain of expertise and hence named as restricted domain question answering system. BASEBALL (1961) is one such example of such system that uses shallow language parsing methodology. LUNAR (1977) is another example that was framed to answer queries related with moon rocks.

2.2 LITERATURE REVIEW ON CLOSED DOMAIN QUESTION ANSWERING

The following section gives an overview of closed domain question answering on semantic based technique, pattern matching, question type, linguistic approach and other approaches.

2.2.1 Semantic Based Technique

QASYO is a sentence level question answering system using natural language processing, ontologies and information retrieval techniques in a common context (Abdullah Moussa & Rehab Abdel-Kader 2011). It accepts queries expressed in natural language and YAGO ontology as input that present answers from the available semantic mark-up. It undergoes semantic analysis of questions to retrieve keywords and also detect the expected answer type for retrieving answers.

Semantic Web information Management with Automated Reasoning Tool (SMART) (Villanueva et al. 2007) is a web-based, ontologydriven, semantic web query answering application to life scientist that represents, integrate, manage and query heterogeneous and distributed biological knowledge.

Apart from structured databases, domain knowledge is also represented in semantic networks and ontologies by applying formats for knowledge representation namely RDF and OWL. Every pattern is linked to a database query method that extracts the required data. Along with the answer, the system also provides the user with added multimodal information namely routes, timetables, film trailers, city maps and images.

2.2.2 Rule Based Technique

AQUA is an ontology-based question answering system that uses classification rule for questions, answer selection method and answer extraction method (2004). It is capable of working on both closed domain and open domain question answering systems.

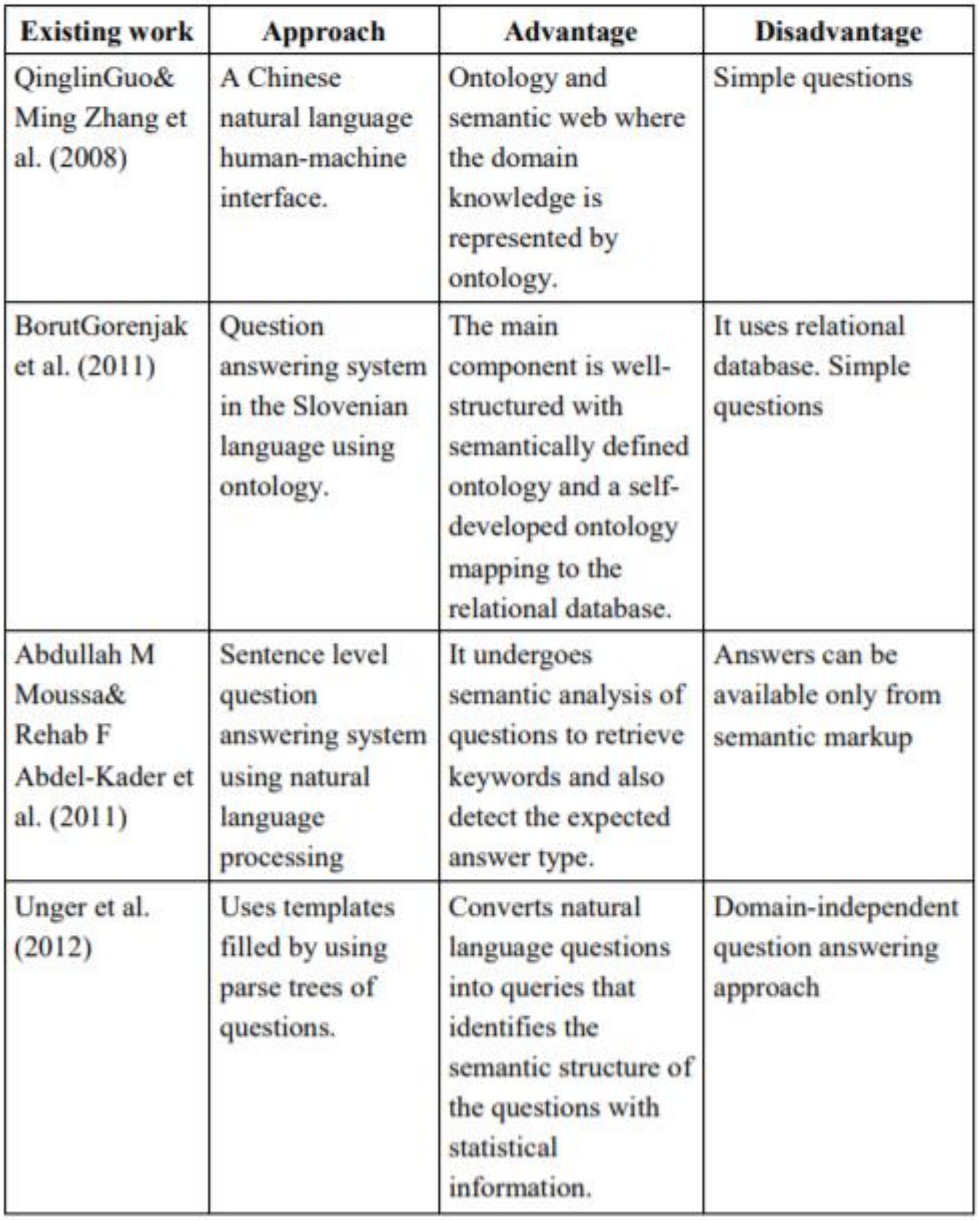
2.2.3 Linguistic Based Technique

Pythia et al. (2011) presented a system that depends on profound linguistic analysis of questions. They manually constructed meaning representations for the vocabulary of a given ontology. The system can process linguistically complex questions using the constructed lexicon that uses determiners. One of the drawbacks is that it requires a lexicon which needs to be created manually for extracting answer as it is not scalable for very large datasets.

2.3 Summary

The ontology construction and ontology mapping on closed domain is done for enhancement of answer retrieval. Less concentration on the ontology construction by the existing method leads to failure state. So, the proposed method is also tested on other closed domains to find the accuracy. The approaches on both open domain and closed domain are described briefly. The classifications on existing approaches are studied based on the features. The limitations of the existing system are analyzed.

Table 2.1 Summary of Literature Review



Existing work	Approach	Advantage	Disadvantage
QinglinGuo& Ming Zhang et al. (2008)	A Chinese natural language human-machine interface.	Ontology and semantic web where the domain knowledge is represented by ontology.	Simple questions
BorutGorenjak et al. (2011)	Question answering system in the Slovenian language using ontology.	The main component is well-structured with semantically defined ontology and a self-developed ontology mapping to the relational database.	It uses relational database. Simple questions
Abdullah M Moussa& Rehab F Abdel-Kader et al. (2011)	Sentence level question answering system using natural language processing	It undergoes semantic analysis of questions to retrieve keywords and also detect the expected answer type.	Answers can be available only from semantic markup
Unger et al. (2012)	Uses templates filled by using parse trees of questions.	Converts natural language questions into queries that identifies the semantic structure of the questions with statistical information.	Domain-independent question answering approach

CHAPTER 3

PROPOSED WORK/RESEARCH METHODOLOGY

In this chapter, we have described various approaches and methods for retrieval of relevant information, advantages and disadvantages of question-answering system. The solution initially proposed was to feed a document from the user itself which he/she want to find an answer, then take a query from the user in natural language English and give the exact answer to the user with in less time. The proposed system searches the document provided by the user itself, process that document by using computational linguistics techniques and finds the exact answer to the user. The Question Answering System reduces time of the user spending on understanding the documents. Sometimes the user may take days to know the required answer but the Question Answering System will give the exact result to the user within seconds.

3.1 FRAMEWORK OF A QA SYSTEM

Question Answering systems generally follow a pipeline structure with three major modules namely: Question Analysis, Passage Retrieval, and Answer Extraction. Figure 1 shows the flow of QAsystem's framework

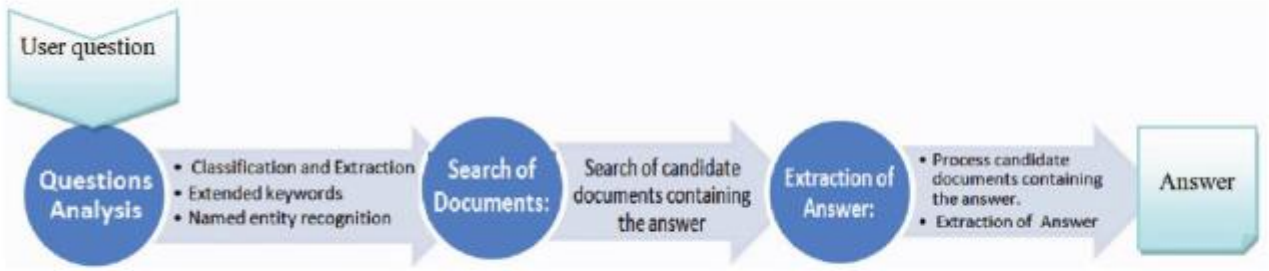

User question
Questions Analysis
• Classification and Extraction
• Extended keywords
• Named entity recognition
Search of Documents:
Search of candidate documents containing the answer
Extraction of Answer:
• Process candidate documents containing the answer.
• Extraction of Answer
Answer

Figure 3.1 The Flow of The Implementation

Typically, questions posed to QA systems need to be parsed and understood before answers can be found. Hence, all necessary question processing is carried out in the Question Analysis module. The input for this stage is the user query and the output are representations of the query; this is useful for analysis in other modules.

Different techniques exist for processes carried out in the Question Analysis stage. These include tokenization, disambiguation, internationalization, logical forms, semantic role labels, reformulation of questions, co-reference resolution, relation extraction and named entity recognition among others. Different QA systems have used and combined different methods for the question analysis phase to better fit the kind of data source they are working on or to solve a particular research problem.

Text retrieval structures split retrieval process in three stages: retrieval, processing and ranking. The processing step involves the use of query analyzers to identify texts in a database. Then, retrieval is done by matching documents with resemblance of the query patterns.

Answer extraction is a major part of a Question Answering system. It produces the exact answer from the passages that are generated. It does this by firstly producing a set of candidate answers from the generated passages and then ranking the answers using some scoring functions. Previous studies on answer extraction have discussed utilizing different techniques for answer extraction, including n-grams, patterns, named entities and syntactic structures.

3.2 DISCUSSION OF APPROACHES AND METHODS

Neural Networks are set of algorithms which closely resemble the human brain and are designed to recognize patterns. They interpret sensory data through a machine perception, labelling or clustering raw input. They can recognize numerical patterns, contained in vectors, into which all real-world data (images, sound, text or time series), must be translated.

3.2.1 Recurrent Neural Network

RNN is recurrent in nature as it performs the same function for every input of data while the output of the current input depends on the past one computation. After producing the output, it is copied and sent back into the recurrent network. For making a decision, it considers the current input and the output that it has learned from the previous input. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition. In other neural networks, all the inputs are independent of each other. But in RNN, all the inputs are related to each other.

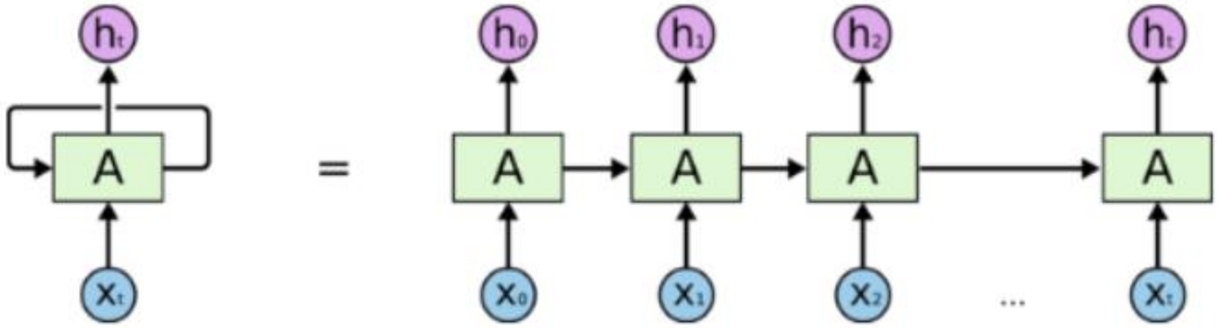


Figure 3.2 An Unrolled Neural Network

3.2.2 LSTM

Long Short-Term Memory (LSTM) networks are a modified version of recurrent neural networks, which makes it easier to remember past data in memory. The vanishing gradient problem of RNN is resolved here. LSTM is well-suited to classify, process and predict time series given time lags of unknown duration. It trains the model by using back-propagation. In an LSTM network, three gates are present:

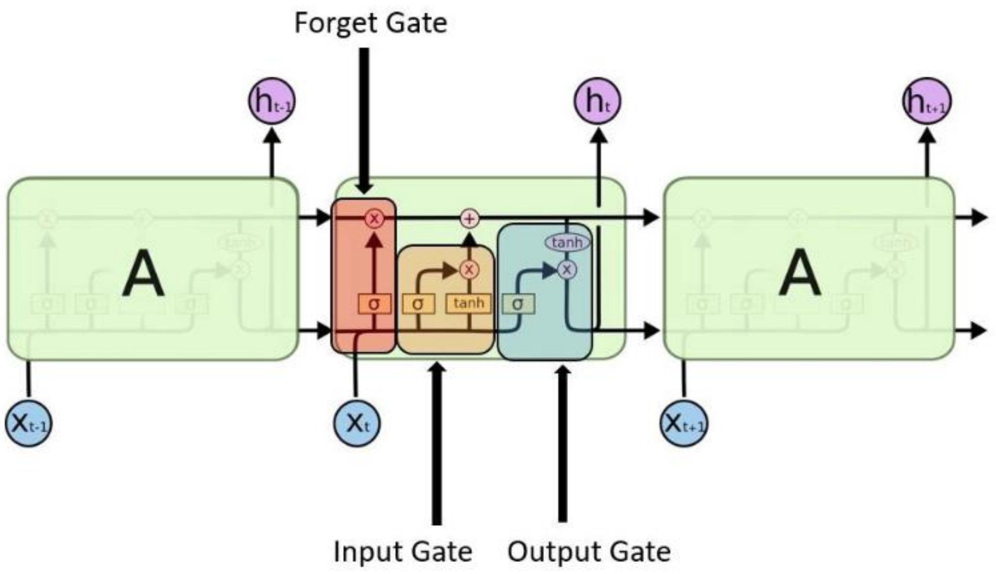


Figure 3.3 LSTM Gates

Input gate — discover which value from input should be used to modify the memory.

Forget gate — discover what details to be discarded from the block.

Output gate — the input and the memory of the block is used to decide the output.

3.2.3 BERT

BERT (Bidirectional Encoder Representations from Transformers) has caused a stir in the Machine Learning community by presenting state-of-the-art results in a wide variety of NLP tasks, including Question Answering (SQuAD v1.1), Natural Language Inference (MNLI), and others. BERT’s key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling. This is in contrast to previous efforts which looked at a text sequence either from left to right or combined left-to-right and right-to-left training.

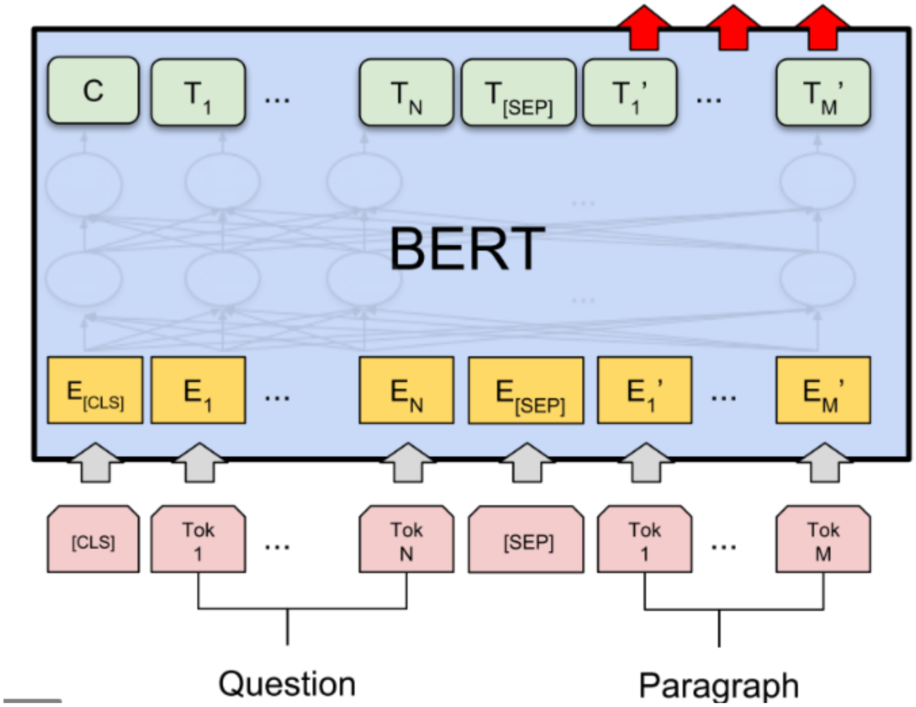


Figure 3.4 Implementation Using BERT

3.3 QUESTION-ANSWERING SYSTEM ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- Searches for information in natural language.
- Finds precise answers to custom questions.
- Greater relevance of found results.
- Document and knowledge management increase with QAS solution.

DISADVANTAGES

- Sometimes it gives wrong or approximate output not the accurate.
- Knowledge extraction is very difficult.
- Has to be enhanced further in future

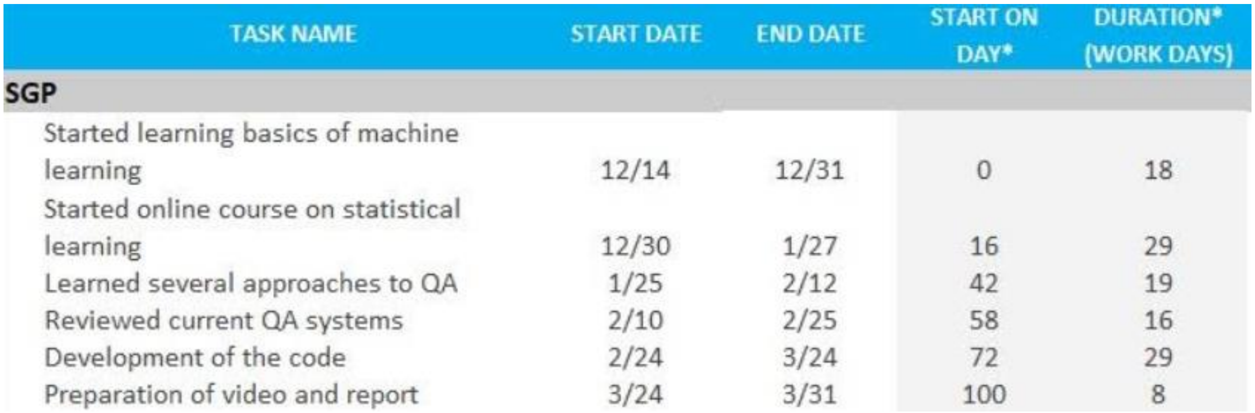
CHAPTER 4

IMPLEMENTATION/RESULTS

4.1 PROJECT SCHEDULING

We implemented our project as follows:

Table 4.1 Project Schedule



TASK NAME	START DATE	END DATE	START ON DAY*	DURATION* (WORK DAYS)
SGP				
Started learning basics of machine learning	12/14	12/31	0	18
Started online course on statistical learning	12/30	1/27	16	29
Learned several approaches to QA	1/25	2/12	42	19
Reviewed current QA systems	2/10	2/25	58	16
Development of the code	2/24	3/24	72	29
Preparation of video and report	3/24	3/31	100	8

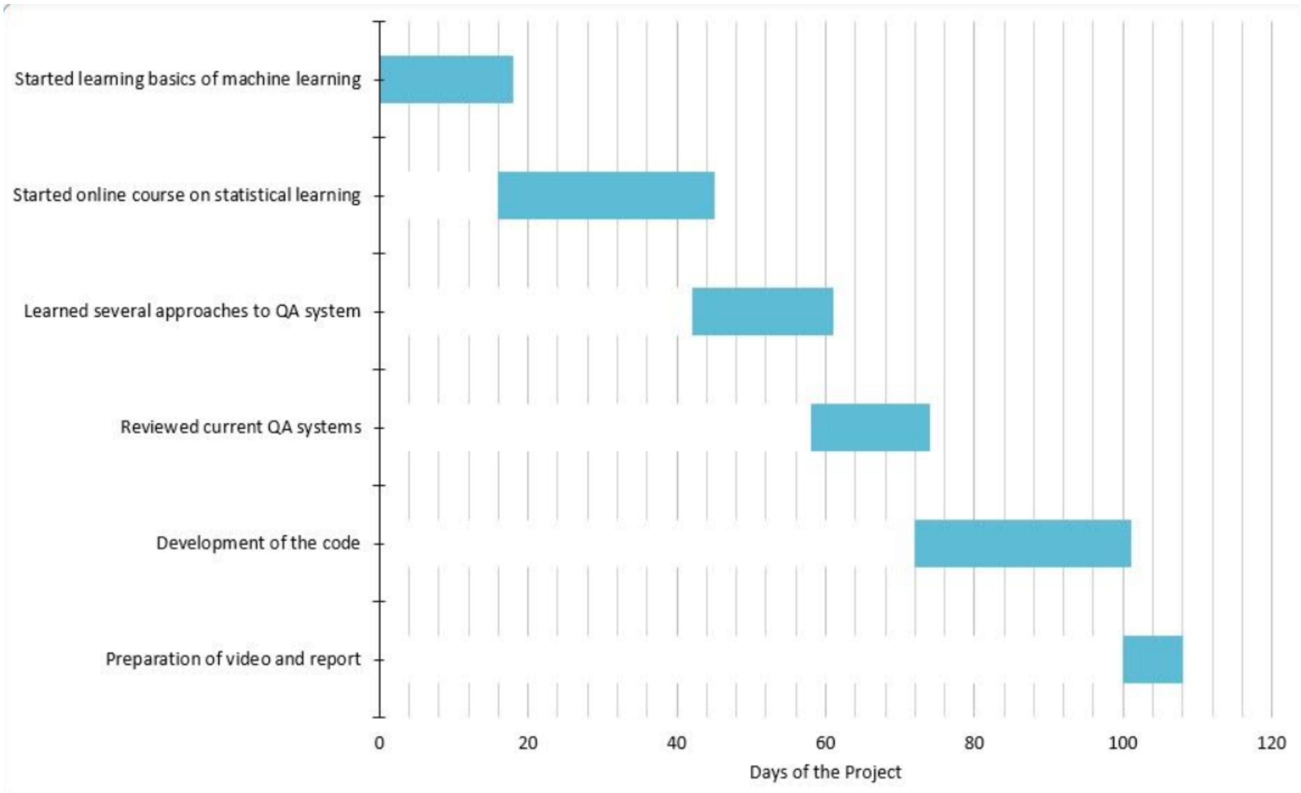


Figure 4.1 Gantt Chart

4.2 IMPLEMENTATION PROCESS

cdQA is considered as an end-to end closed domain Question Answering System. The cdQA architecture is based on two main components: The Retriever and the Reader. You can see below a schema of the system mechanism.

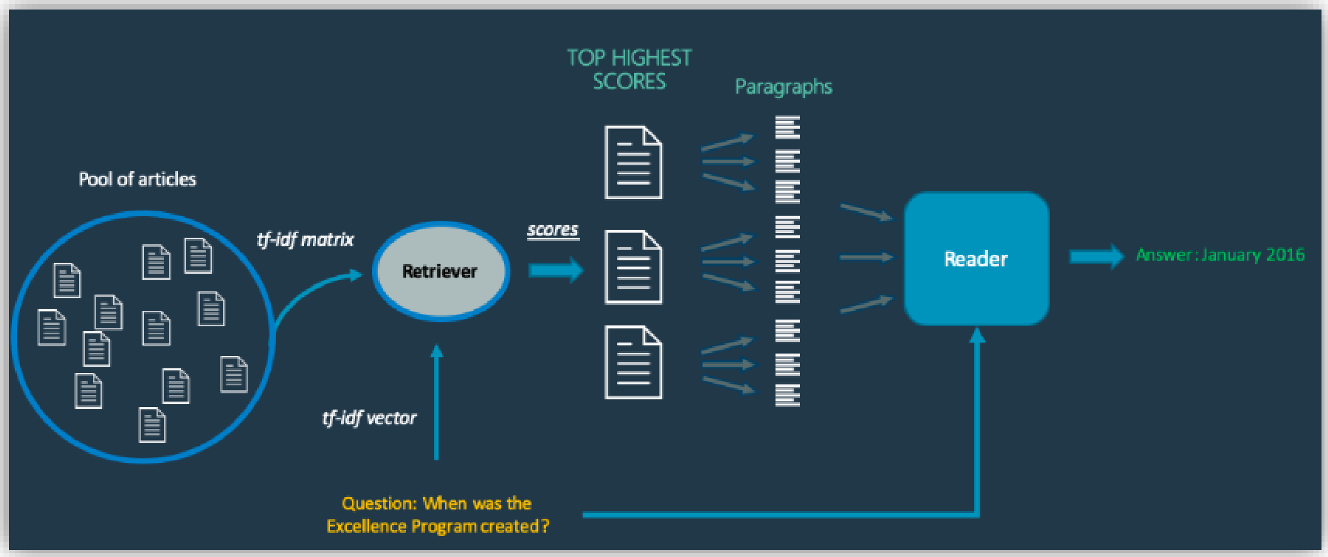


Figure 4.2 Mechanism of cdQA Pipeline

The cdQA-suite is comprised of three blocks:

cdQA: an easy-to-use python package to implement a QA pipeline

cdQA-annotator: a tool built to facilitate the annotation of question-answering datasets for model evaluation and fine-tuning

cdQA-ui: a user-interface that can be coupled to any website and can be connected to the back-end system

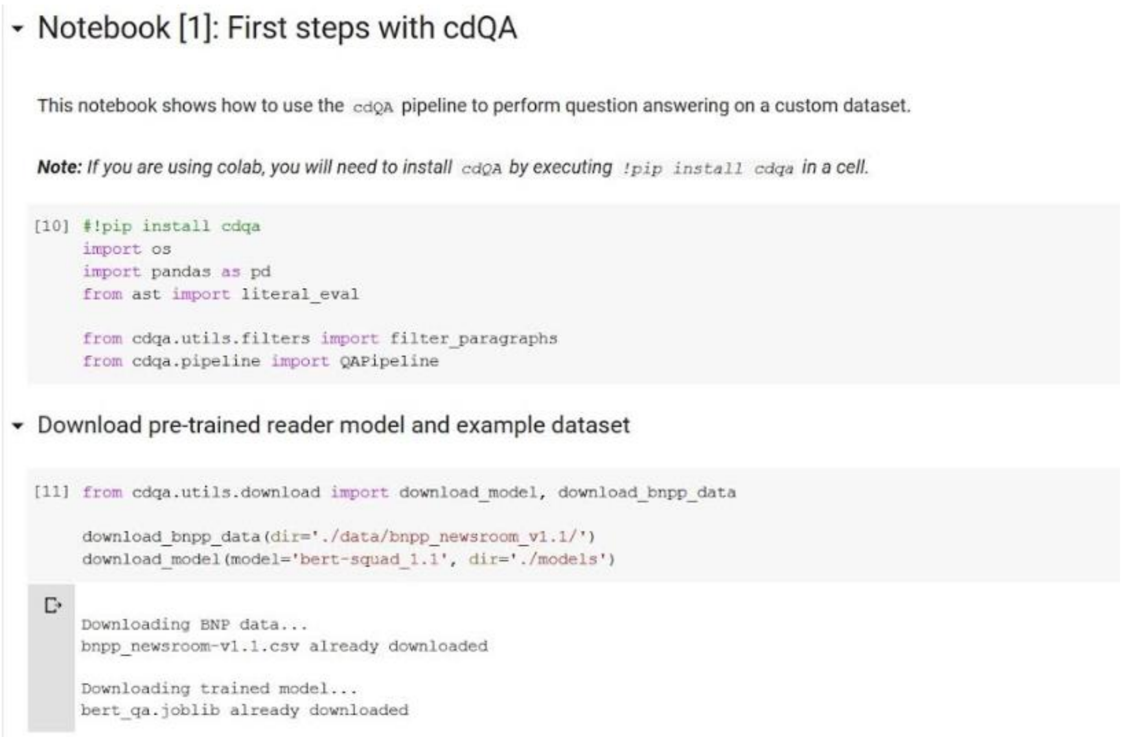
When a question is sent to the system, the Retriever selects a list of documents in the database that are the most likely to contain the answer. After selecting the most probable documents, the system divides each document into paragraphs and send them with the question to the Reader, which is basically a pre-trained Deep Learning model. The model used was the Pytorch version of the well-known NLP model BERT. After the Reader, there is a final layer in the system that compares the answers by using an internal score function and outputs the most likely one according to the scores

4.3 RESULTS

4.3.1 Using The cdQA Python Package

We downloaded the cdQA package using the following. After that we downloaded the pre-trained reader and then the dataset as shown below: **-**

Here download_model means to get a pre-trained model. The model we are going to use is a pre-trained BERT model, fine-tuned on SQuAD 1.1, a reading comprehension dataset that is often used to test and benchmark how well a question answering system is performing



Notebook [1]: First steps with cdQA

This notebook shows how to use the `cdQA` pipeline to perform question answering on a custom dataset.

Note: *If you are using colab, you will need to install* `cdQA` *by executing* `!pip install cdqa` *in a cell.*

```
[10] #!pip install cdqa
     import os
     import pandas as pd
     from ast import literal_eval

     from cdqa.utils.filters import filter_paragraphs
     from cdqa.pipeline import QAPipeline
```

Download pre-trained reader model and example dataset

```
[11] from cdqa.utils.download import download_model, download_bnpp_data

     download_bnpp_data(dir='./data/bnpp_newsroom_v1.1/')
     download_model(model='bert-squad_1.1', dir='./models')
```

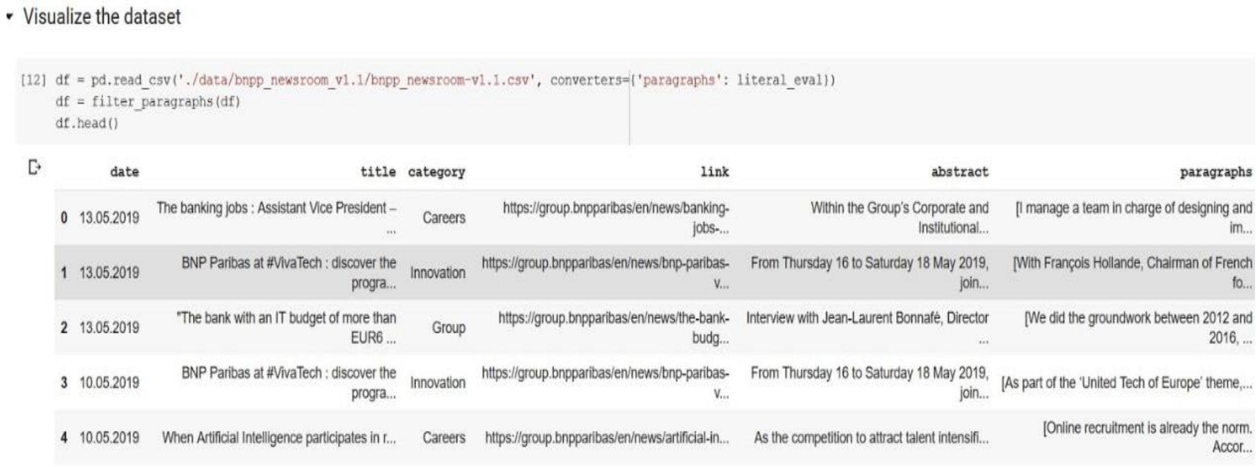
```
Downloading BNP data...
bnpp_newsroom-v1.1.csv already downloaded

Downloading trained model...
bert_qa.joblib already downloaded
```

Figure 4.3 Basic Downloading

4.3.2 Visualizing The Dataset

We visualized the dataset thoroughly and then used filter_paragraph to narrow down to just paragraphs found in our text (looks for a certain length or other parameters in the text essentially). To use, you give it the entire dataframe created.

After this, we instantiated the cdQA pipeline from a pre-trained model as shown below: -

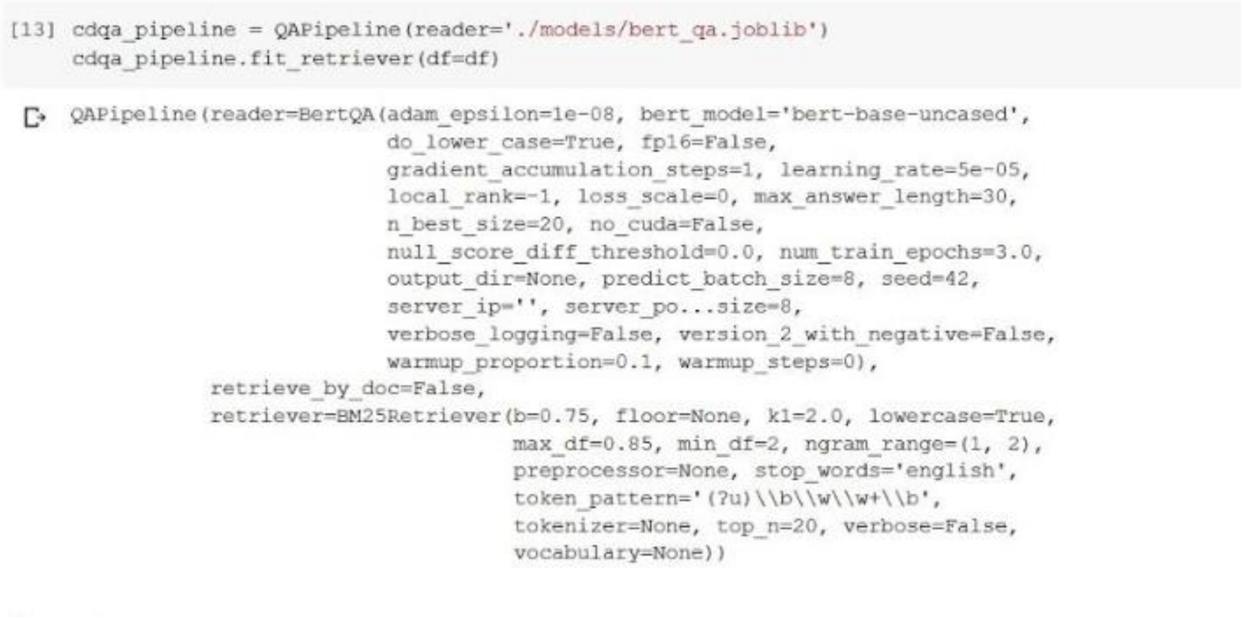


Visualize the dataset

```
[12] df = pd.read_csv('./data/bnpp_newsroom_v1.1/bnpp_newsroom-v1.1.csv', converters={'paragraphs': literal_eval})
     df = filter_paragraphs(df)
     df.head()
```

	date	title	category	link	abstract	paragraphs
0	13.05.2019	The banking jobs : Assistant Vice President – ...	Careers	https://group.bnpparibas/en/news/banking-jobs-...	Within the Group's Corporate and Institutional...	[I manage a team in charge of designing and im...
1	13.05.2019	BNP Paribas at #VivaTech : discover the progra...	Innovation	https://group.bnpparibas/en/news/bnp-paribas-v...	From Thursday 16 to Saturday 18 May 2019, join...	[With François Hollande, Chairman of French fo...
2	13.05.2019	"The bank with an IT budget of more than EUR6 ...	Group	https://group.bnpparibas/en/news/the-bank-budg...	Interview with Jean-Laurent Bonnafé, Director ...	[We did the groundwork between 2012 and 2016, ...
3	10.05.2019	BNP Paribas at #VivaTech : discover the progra...	Innovation	https://group.bnpparibas/en/news/bnp-paribas-v...	From Thursday 16 to Saturday 18 May 2019, join...	[As part of the 'United Tech of Europe' theme,...
4	10.05.2019	When Artificial Intelligence participates in r...	Careers	https://group.bnpparibas/en/news/artificial-in...	As the competition to attract talent intensifi...	[Online recruitment is already the norm. Accor...

Figure 4.4 Visualization



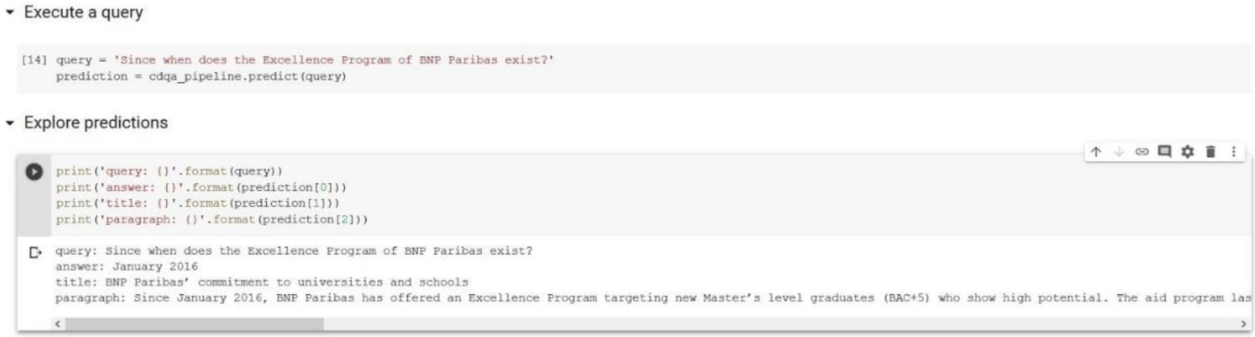
```
[13] cdqa_pipeline = QAPipeline(reader='./models/bert_qa.joblib')
     cdqa_pipeline.fit_retriever(df=df)
```

```
QAPipeline(reader=BertQA(adam_epsilon=1e-08, bert_model='bert-base-uncased',
                         do_lower_case=True, fp16=False,
                         gradient_accumulation_steps=1, learning_rate=5e-05,
                         local_rank=-1, loss_scale=0, max_answer_length=30,
                         n_best_size=20, no_cuda=False,
                         null_score_diff_threshold=0.0, num_train_epochs=3.0,
                         output_dir=None, predict_batch_size=8, seed=42,
                         server_ip='', server_po...size=8,
                         verbose_logging=False, version_2_with_negative=False,
                         warmup_proportion=0.1, warmup_steps=0),
           retrieve_by_doc=False,
           retriever=BM25Retriever(b=0.75, floor=None, k1=2.0, lowercase=True,
                                   max_df=0.85, min_df=2, ngram_range=(1, 2),
                                   preprocessor=None, stop_words='english',
                                   token_pattern='(?u)\\b\\w\\w+\\b',
                                   tokenizer=None, top_n=20, verbose=False,
                                   vocabulary=None))
```

Figure 4.5 Instantiation

4.3.3 Execution Of Query And Explore Predictions

We use our pipeline instance's `predict` method that returns a tuple with the answer, title of document found in, paragraph, and score. Here is an example of a search:



Execute a query

```
[14] query = 'Since when does the Excellence Program of BNP Paribas exist?'
     prediction = cdqa_pipeline.predict(query)
```

Explore predictions

```
print('query: {}'.format(query))
print('answer: {}'.format(prediction[0]))
print('title: {}'.format(prediction[1]))
print('paragraph: {}'.format(prediction[2]))
```

```
query: Since when does the Excellence Program of BNP Paribas exist?
answer: January 2016
title: BNP Paribas' commitment to universities and schools
paragraph: Since January 2016, BNP Paribas has offered an Excellence Program targeting new Master's level graduates (BAC+5) who show high potential. The aid program las
```

Figure 4.6 Concluding Step

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENT

The generic QA architecture can be modified by including some validation modules or removing some modules to fit the question environment under consideration. The performance of each module in the framework depends on the one before. That is, a proper query formulation from the Question Analysis phase increases the likelihood of retrieving important passages in the passage retrieval phase and finally enhancing the probability of extracting the right answers in the answer extraction module. This then calls for the implementation of efficient methods in each module in the QA framework.

The performance of QA systems is increased by combining different techniques together to strike out the inefficiency of individual implementations as seen in hybridized implementations, but this implementation becomes expensive for simple QA system or quite expensive and time consuming for complex ones, hence the need for better ways in merging techniques in QA systems. The output of a QA system is also reliant on a good knowledge base and a clear understanding of user question

For future enhancement we would try:

- To reach 100 % accuracy.
- To increase the speed of performance.
- To use artificial intelligence techniques to identify the pertinent answers.

All the code for this study is available on GitHub: https://github.com/pratyay12/Question-Answering-using-BERT

BIBLIOGRAPHY

"Natural Language Question Answering System: Technique of Semantic Headers" by Boris Galitsky.

- "Semantics for a Question-answering System"" by William Aaron Woods
- "The Process of Question Answering:A computer Simulation of Cognition" by Wendy Lenhert.
- "The Application of Theorem Proving to Question-answering Systems" by Cordell Green.
- "Foundations of Statistical Natural Language Processing" by Christopher Manning and Hinrich Schütze.
- Natural Language Processing with Python by Steven Bird, Ewan Klein and Edward Loper.
- http://www.cfilt.iitb.ac.in/resources/surveys/Question%20Answering%20Survey-biplab.pdf
- https://towardsdatascience.com/bert-based-cross-lingual-question-answering-with-deeppavlov-704242c2ac6f
- https://www.researchgate.net/publication/333627091_A_Review_of_Question_Answering_Systems