Implementing a Question Answering system using neural networks

Vanshika Sangtani, Chris Mary Benson, and Dr. Elakkiya Rajesh

Department of Computer Science,
Birla Institute of Technology and Science, Pilani,
Dubai International Academic City,
Dubai, United Arab Emirates

ABSTRACT

The task of Question Answering (QA) in information retrieval is to answer questions intelligently using either pre-structured databases or natural language documents to provide correct answers to questions asked by humans. As opposed to a search engine which provides a list of documents as the result, a QA system presents only the requested information in a summarized and concise format. A major purpose of the Question Answering system (QA) is to encourage research into systems that return answers since many users prefer direct answers, and to gain experience in large-scale evaluation of QA procedures. By using natural language to answer a specific question, QA strives to satisfy users. One of the research gaps in the question answering field is that most traditional models struggle to understand contexts consisting of long sequences and the underlying relationships and dependencies present in it. In reality, systems contain large amounts of information which should be understood by question answering models to generate accurate answers for a particular query. This paper proposes a system to solve this by using a deep learning approach consisting of a memory network of comprising of two different models:- a unidirectional long short term memory (LSTM) and a bidirectional long short term memory (BiLSTM) a for intelligent learning from a corpus consisting of question-answer pairs along with a context. Results from the experiments done on these models show that the BiLSTM based memory network model was able to understand the context better and capture relevant pieces of information when compared to the LSTM based memory network model.

1. INTRODUCTION

Question-Answering (QA) models, in simple terms, is an information retrieval (IR) system that retrieves an answer in a natural language from a corpus of data in response to a query posed by a human. The concept of these question answering models arose as a solution to the problem of the ever-increasing growth of information which has led to the overloading of data. Search engines usually retrieve documents as a response to a query, but due to the aforementioned overloading of data, users would find it challenging and overwhelming to find a satisfactory answer from all the documents presented to them. Thus, question-answering models have proved to be efficient as

they are able to provide users with accurate answers while saving the users' time as well. Now, with the success of QA systems, they have been used as the basis for many applications such as chatbots in customer service, etc with the most notable application being that of the well-known chatbot, ChatGPT.

Question Answer Systems (QAS) mainly combine the fields of Information Retrieval(IR), Information Extraction (IE) and Natural Language Processing (NLP). Through the literature survey conducted, it is observed that there have been many approaches that have been implemented to develop a QA model, from the traditional methods of rule-based approaches to the modern methods of transformers and neural networks. While all of these approaches have different methodologies, they have an information retrieval system, of which the basic idea is common to all of them.
Information Retrieval (IR)-Question Answering (QA) systems typically incorporate a question processing module that can identify the type of query and the type of desired response. This system applies a number of modules that use increasingly complex natural language processing (NLP) techniques on progressively less material after the question analysis. A document retrieval module in this system makes use of search engines to locate the documents or parts within the document set that are most likely to contain the solution. Next, a filter preselects small text fragments in the retrieved documents that include strings that match the expected response. For example, if the query is "Who invented the telephone?", the filter will return texts that have names of people, given that "person" is the expected answer type. Finally, an answer extraction module looks for more hints in the text to determine if the response contender can truly answer the question.

QAs are mainly of two types:- open-domain and closed-domain QA systems.
Closed-domain QA systems refer to those models that are restricted to a specific domain. In other words, this kind of system addresses queries inside a particular domain such as medical information, movies, etc. It can be perceived as a simpler task since natural language processing (NLP) systems are able to utilize domain-specific knowledge that is often codified in ontologies. The term "closed-domain" may describe an environment in which a specific category of questions is allowed, like inquiries concerning descriptive data as opposed to procedural data. On the other hand, open-domain QA models rely on general ontologies and common knowledge, and it handles queries about almost anything. These systems typically have access to a lot more data, making it easier to extract the solution. For example, a model trained on the benchmark dataset SQuAD, which consists of data from Wikipedia articles, is an open domain QA model, as it is trained to generate responses for queries regardless of what domain the context of that query belongs to.

While QA models have improved the way search engines work to obtain brief and accurate results, researchers still face some challenges in building effective question-answering models. These challenges include ambiguity, language diversity, knowledge acquisition, memory capacity, computational resources, and so on. QA systems have problems with polysemy and ambiguity, such as lexical and syntactic ambiguity, which makes context determination challenging. Lack of common sense thinking and difficulty interpreting allusions in larger contexts are obstacles to semantic understanding. Handling idiomatic phrases, colloquialisms, and accurately comprehending word order and grammar are made more difficult by language complexity. System accuracy is impacted by knowledge base constraints, gaps in knowledge, and challenges adjusting to new information. Challenges related to question variability include answering a variety of question types, dealing with negation, and answering intricate queries that call for deduction and reasoning. Managing massive data volumes and attaining real-time performance are also some challenges that affect the scalability and efficiency of QA models as a model needs to be trained on a huge corpus of data to get optimal results.

Enhancing QA systems for more robustness and versatility is the goal of research, with a focus on the critical role that ambiguity understanding plays in natural language processing. In this paper, we strive to implement a deep learning model, consisting of a memory network that utilizes a bidirectional LSTM to develop a QA system. We also experiment this model with a unidirectional LSTM to observe which model is able to perform better. This QA system will be open-domain as it is trained on the SQuAD dataset. In the following sections, we discuss the literature review of various question answering systems that have developed in recent years as well as the proposed methodology to be developed, the results and an analysis of these results as well.

2. LITERATURE REVIEW

Information seekers frequently use question answering systems on the web to find information. The system allows information seekers to present their questions in natural language in order to receive a concise response to their inquiry.

The concept of Question Answering Systems(QAS) has gained a lot of traction in the past few years; the focus of QAS on information retrieval rather than document retrieval (executed by current search engines) has proved to be useful as it provides users with direct answers in response to the queries rather than leaving the task to the users, which is the case in document retrieval. The aim of paper [1] was to develop an intelligent learning system that can acquire knowledge from the given input in the form of text files. With this knowledge, the system will then be able to answer the queries proposed by the user. The questions that these systems usually encounter are of two

categories:- factual and expert. Questions that contain words such as what, where and when are considered as factual questions while those with words such as why, how, etc are considered as expert questions, those which require reasoning in the answers. Based on the type, the questions are classified and accordingly, the answers are extracted and formulated. The QA system mainly consists of three processes:- document and passage retrieval, question processing module(where classification is done) and finally, the answering module which refers to the passages retrieved from the document in the first stage.

Paper [2] gives an overview on the various approaches that have been implemented to create a question answering system as well as their advantages and disadvantages. Initially, QA systems adopted the rule-based approach where each question type has a set of rules that were formulated and represented in the form of decision trees meant to replicate the grammatical rules understood by humans. The path followed in the tree according to the rule chosen would determine the answer extracted. While this was a good approach, it was a rather cumbersome process as increase in data would require the formulation of new rules. The second approach was the statistical approach which was a much better method as it could deal with large and diverse datasets. Different statistical techniques, such as Support Vector Machine (SVM) classifiers and Bayesian Classifiers, are applied to make a prediction on what the users' expected answer type for the given question. In the Machine Learning Approach, a knowledge base or a taxonomy was built from the corpus using Named Entity Recognition. Due to the learnability that comes with this approach, the high scalability as a result is a huge advantage. Nowadays, to obtain an optimal QA model, the machine learning approach is often combined with the statistical approach. This has resulted in powerful models that are used in the present day.

In paper [3], the main goal was to achieve a QA system through a deep learning approach with the help of neural networks. Neural networks such as Long Short Term Memory models (LSTMs) have been a promising solution for the development of optimal QA models as compared to the traditional methods such as parsing, part-of-speech tagging, etc. The dataset used in this deep learning model is the bAbI dataset; it is a closed dataset consisting of twenty QA tasks. Regularization, encoding and embedding of the questions and its corresponding contexts was done as a part of pre-processing. An end-to-end memory network was implemented using Jaccard similarity as a metric for selecting sentences from the dataset. From the results obtained from this model, it was observed that high accuracy was achieved for the tasks and the results were akin to those of the state-of-the-art models.

The objective of paper [4] was to integrate linguistic and semantic resources that would help solve problems that arise from the natural language structure such as semantic ambiguity. It also aims to simplify the task of e-learning for users who want answers from documents containing large amounts of text. The model generation consisted of two steps: - dataset creation and implementation of the QA system. For the implementation, the first step was to tokenize the questions in the dataset and obtain the keywords by the process of lemmatization and removing stopwords. Then using wordnet, the synonyms of these keywords were found. Finding the synonyms is useful if in the case the user types out a query, the keywords in the query can be related to its synonyms present in the dataset to find the answer. Then, cosine similarity is used to obtain the similarity between the keyword and the words in the dataset for the answer retrieval for the query. This model was able to generate an F-measure of 0.8689 for 200 questions tested on the QA system and its performance was high when compared to other QA models.

Paper [5] aims to build a QA model using Support Vector Machines (SVM) which is a machine learning algorithm. This model is used as a classification method for classifying the different types of factual questions (what, why, how, where, etc). The input to the model are the vector features of the questions consisting of a common noun, main verb, Wh- words, etc. Wordnet is also used to find synonyms for the keywords extracted from the question. For the retrieval of the answer for the question, firstly, the documents are retrieved from the World Wide Web which then undergo passage retrieval. In passage retrieval, the passages are ordered based on their term frequency-inverse document frequency (tf-idf) score. To obtain accurate answers, the similarity between the question and each passage is calculated and the passages with highest similarity are chosen. An average F-score of 80% was obtained for the SVM classifier and a score of 0.523 was recorded for the Mean Reciprocal Rank (MRR) metric of the answer extraction task.

Building a model using neural networks for document-based question answering was the goal of paper [6]. The method proposed is akin to the logic taken by a student when reading scientific articles. The model would first relate the title of the document to the keywords extracted from the question. A recurrent neural network (RNN) is implemented to encode the question and the title. The output of this model is then fed to a bi-directional RNN where we obtain the sentence vector from the document This model obtained a Mean Average Precision (MAP) score of 84.43% showing that this model is highly efficient and is considered as a state-of-the-art model.

Using Question Answering System, paper [7] reviews some of the methods and implementation techniques. Question answering systems using NLP techniques are more complex compared to other retrieval of information by different types of systems. It

is possible to develop QA systems for web resources, semi-structured and structured knowledge bases. In contrast to open domain QA systems, closed domain QA systems provide more accurate answers, but they are restricted to one domain. In order to provide accurate and suitable more correct responses to users' queries on closed domain documents related to education acts, a QA system is proposed using NLP and information retrieval techniques.

As previously mentioned, the three primary components of QA systems are question classification, information retrieval, and answer extraction. Thus, the QA researchers were interested in all three of these elements. Paper [8] highlights the classification of questions in the following table:

Question class	Question	Answer type
WHAT	basic-what/ what-who /whatwhen/ what where	Money/ No./ Definition/ Title/ NNP/ Undefined
WHO		Person
HOW	basic-how how-many howlong how-much how much how-far how-tall how-rich how-large	Manner Number Time/Distance Money / Price howmuch Undefined Distance Number Undefined
WHERE		Location
WHEN		Date
WHICH	which-who which-where which-when which-what	Person Location Date NNP
NAME	name-who name-where name-what	Person/ORG. Location Title / NNP
WHY		Reason
WHOM		Person

Table 1: Classification of questions

The task of mechanically answering a human-posed question in natural language is known as Question Answering (QA) in Information Retrieval (IR) and Natural Language Processing (NLP). Question analysis, document retrieval, and answer extraction are the three primary, discrete subtasks that make up the task of quality assurance as

described in paper [9] (see Fig. 1). These three subtasks are followed by the majority of QAS. But how they carry out each smaller duty could vary.

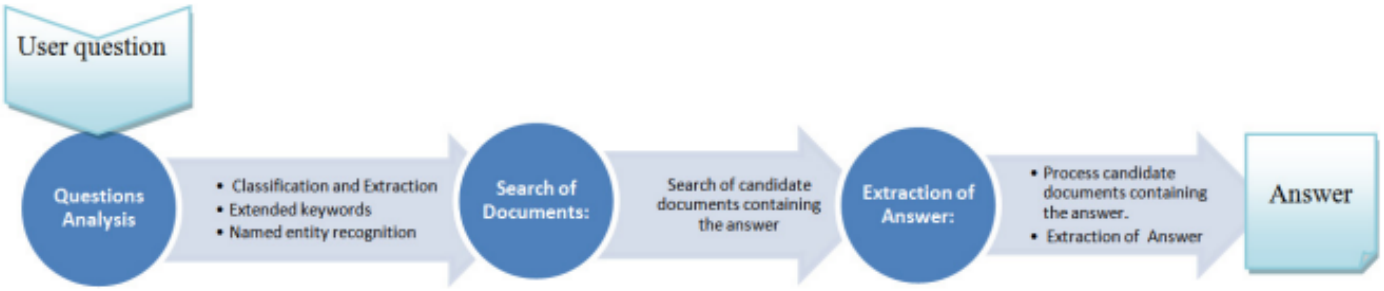

Fig 1: Subtasks of QAs dedicated to Web of documents

According to Paper [10], the QA systems can be categorized using four interconnected dimensions (see Table 2):

1. The input or query types (facts, dialogs, etc.) that it can handle
2. How it handles the classic fundamental issues (such ambiguity and adaptation) that the search environment places on every non-trivial search system.
3. The sources (structured vs. unstructured data) from which it can obtain the answers
4. The scope (domain independent vs. domain specific).

Dimensions			
Input Types	**Search Environment** (Traditional Intrinsic Problems)	**Sources**	**Scope**
▪ Keywords/definitions ▪ Factoids (wh-, affirm / negate) ▪ Understanding and causality reasoning (why, how) ▪ Temporal and spatial reasoning ▪ Facts from different sources ▪ Common sense reasoning ▪ Interactive dialogs	▪ Large scale (scalability) ▪ Heterogeneity (mapping, disambiguation) ▪ Openness (fusion, ranking) ▪ Cross-lingual (multilingual) ▪ Trust	▪ Structured (NLIDB) ▪ Semi-structured (documents) ▪ Textual (TREC, Web) ▪ Semantic (ontologies)	▪ Domain dependent (closed-domain) ▪ Domain independent (open domain) ▪ Proprietary KBs (private)

Table 2: The dimensions of QA amd query and search interfaces

Information retrieval (IR), natural language processing (NLP), human computer interaction (HCI), and, more recently, artificial intelligence (AI), machine learning (ML), and knowledge management (KM) are some of the multidisciplinary fields that are combined in the paper [11] on IQASs. The AI community has shown a significant and increasing interest in QASs as a result of recent advancements in the NLP and ML domains, particularly with regard to their use in digital assistants (e.g., Apple Siri, Google Assistant, and Amazon Alexa). Because of their seemingly comparable functions, QASs are typically categorized as a more advanced type of information retrieval systems (IRSs) and fall under the larger category of IRSs.
These days, QASs have developed into a distinct field of study with ever-expanding goals that include new research subjects such as semantic entailment and knowledge representation.

Paper [12] discusses the use of a transformer-based model to obtain an optimal answer selection model (AS2); AS2 is a model that picks out the best answer among a set of candidate answers. The main objective of this work was to train a sequence to sequence transformer with the candidate answers, the transformer model would then generate an answer from the candidate answer set. This was proposed from observations of an unsatisfactory answer being selected when the candidate sets are of poor quality. The final layer of the traditional seq2seq model is extended to generate an answer from the candidate answer set for a query. The WikiQA and three other answer selection datasets are used to fine tune the seq2seq model, the GenQA model. From the results obtained, it is observed that the GenQA model is able to outperform state of the art models such as TANDA. The GenQA model obtained an accuracy of 99.2 while the accuracy of TANDA was 59.5 on the WikiQA dataset.

The goal of paper [13] was to develop a question answering system that could fetch a list of answers of expert level in response to queries about the COVID-19 disease. Moreover, the relation between an information retrieval system and the answer extraction model was examined as well. The first step was to retrieve the relevant documents. Next, the relevant passages from these documents are extracted, after which these passages are ranked using evaluation metrics to achieve the most relevant and accurate answer. It was discussed whether doing document retrieval in the first step or if doing passage retrieval instead would be the best strategy. Through experiments, it was found that document retrieval using a recall based strategy outperformed the passage retrieval strategy. As for the answer extraction model, a SciBERT model is used; SciBERT is a pre-trained model trained on a corpus of scientific text. In this work, the model was fine tuned on the SQuAD2.0 and the QuAC dataset. The SciBERT model extracts answers based on the question and its context. These answers were

then ranked using a context retrieval score and a score calculated for each answer obtained from the model.

Paper [14] aims to survey the latest open domain question answering systems (OQDA), in order to determine the efficiency of these systems. OQDAs deal with a large corpus of data, hence it often faces some limitations in obtaining high accuracy due to the requirement of high memory consumption and inference latency. The architecture of OQDAs can be classified into three different frameworks:- Retriever Reader, Retriever -only and Generator-Only. Retriever Reader consists of a retriever module and a reader module using methods such as tf-idf, etc for retrieving documents. This framework faces the limitation of slow processing. Retriever-only consists of only a retriever module, this removes the limitation of slow processing due to the absence of the reader module but faces a disadvantage of low performance as compared to the retriever reader framework as less information is taken into consideration. The generative framework does not have a retriever or a reader module, it instead tries to generate an answer using the prompt. But this type of framework still requires a lot of improvement to obtain a good accuracy. This work proposes some solutions in order to reduce processing time such as reducing the indexing size and the model size, model pruning, etc. The existing challenges faced by OQDAs are also discussed, mainly the fact that it is computationally heavy, leaving researchers in a dilemma as this poses a disadvantage for low power devices such as mobile phones.

The objective of paper [15] was to build a QA system that could provide an answer in response to questions regarding scientific articles with good reasoning. That is, answers that are not shallow in nature and provide accurate reasoning for “how” or”why” questions. This work proposes the QASA system trained on the QASA dataset which consists of question answer pairs that require full-stack reasoning. The QASA system aims to deal with surface, testing and deep questions by implementing a large language model (LLM) through three tasks:- associative selection, evidential rationale generation and systematic composition. Associative selection includes selecting information from the articles that is relevant to the question. The next task is to extract the evidential rationale such as the main content, background information, etc from the selected paragraphs. The final task is to systematically compose the evidence rationales to generate a proper answer. Experiment results show that QASA was able to perform much better than the state of the art InstructGPT.

A domain-specific QA system using knowledge graphs and transfer learning with a pre-trained language model BERT was proposed in paper [16]. Domain-specific tasks usually are time-consuming and require a lot of training data. In this work, a pre-trained model BERT, which has open-domain knowledge, is fine tuned to construct a

domain-specific BioMed system that requires less training data. A BioMed QA dataset was constructed from scratch for implementation in this paper. Various subtasks are also included in this QA model such as subject identification (from the query), relation classification (finding the relevant relations from the subject), subject-relation linking, and answer generation. From the results, this model proved to outperform state of the art models with an accuracy of 95%.

A system for answering frequently asked questions (FAQs) was outlined in paper [17]. A detailed description of how the corpus was preprocessed and the experiments were performed on it can be found here. A combinatorial approach to searching was also described. In spite of the poor evaluation results, a search engine was developed for FAQs that is remarkably accurate.

Paper [18] shows how these technologies can be adapted so that source code can be learned from large datasets and used to answer questions. QA systems cannot be built in a single paper to answer all questions about programming. In peer-reviewed literature, context-based QA systems have not even been demonstrated to interpret and answer natural language questions based on source code. It is important to realize, however, that quality assurance systems are rarely used independently.

The QANet model for machine reading comprehension was constructed from scratch in Paper [19]. The SQuAD 1.1 dataset was used to assess the model, and the outcomes were examined. QANet is a multi-layered, intricate model. It was a big challenge to properly develop the architecture and the data pre-processing logic in a short amount of time based on the paper's sparse explanations. The primary reason for selecting this project was to thoroughly examine the fundamental concepts (such as highway networks, attention mechanisms, etc.) from the most recent advancements in neural architectures, upon which the QANet is based.

Based on the overall architecture of the question-answering framework, a thorough survey of question-answering systems was published in paper [20]. We found commonalities and provided a generic description of a question-answering system that allowed us to explain its formal structure in an abstract manner, notwithstanding the heterogeneity of QA tasks and design principles. Additionally, task-specific benchmark datasets and assessment scores were provided in this research in order to better understand the fundamental design concepts and assumptions beneath QA systems. To address this, a taxonomy has been developed, with the primary branches being the separation of machine-trained evaluation scores (MTES) from human-centric evaluation scores (HCES) and automatic evaluation scores (AES) from untrained automatic evaluation scores (UAES).

From the papers reviewed in this section, it is apparent that there are many research gaps which are yet to be solved in the question answering field. One of these is with respect to the context. Most traditional models struggle to understand contexts consisting of long sequences and the underlying relationships and dependencies present in it. In reality, systems contain large amounts of information which should be understood by question answering models to generate accurate answers for a particular query.

3. METHODOLOGY

3.1 Dataset

A system's results are determined by two criteria: first, the algorithms and methods it uses for natural language processing, and second, the domain in which it is operating. For implementing the proposed model in this paper, we will be using The Stanford Question Answering Dataset (SQuAD2.0). SQuAD2.0 is a benchmark reading comprehension dataset consisting of three main attributes:- question, context, and the answer for the corresponding question. The context was built from various passages belonging to the Wikipedia articles. This dataset consists of 100,000+ question-answer pairs; some questions do not have an answer, this is to ensure that the model is able to understand the context to check whether the answer is present or not.

3.2 Proposed Methodology

The model proposed in this paper focuses on an information retrieval system using the concept of a neural network consisting of an end-to-end memory network implemented using two models:- a unidirectional LSTM and a bidirectional LSTM. There are past works which have developed question answering systems using a memory network. In this study, we wanted to experiment with a hybrid, combining the advantages of a memory network and a unidirectional/bidirectional LSTM to develop a question answering system. LSTMs are known for their ability to capture long-term dependencies and relationships among the input sequences. They consist of a gating mechanism, an input and a forget gate, as well as a memory cell which is beneficial in dealing with the vanishing gradient problem in deep learning models. We will discuss each component and its implementation in depth in the sections below.

Fig 3.2.1 Architecture Diagram of the Q&A system

3.2.1 Memory Networks

Memory networks are a type of neural network that is used for utilizing the memory for various tasks, in our model it is used in relation to the context in the

data. This concept is beneficial in question answering systems as these systems require a model to understand the context related to the query for generating an answer. Contexts containing many sequences contain dependencies and relationships among them that need to be properly understood by the model in order to articulate an accurate answer for the corresponding query. For example, in the figure below where the context is given, the answer to the query requires the system to understand that the 'this' in the second line of the context refers to 'tathagatagarbha' which appears in the previous line.

```
Context: the tathāgatagarbha sutras are a collection of mahayana sutras
that present a unique model of buddha-nature . even though this collection
was generally ignored in india , east asian buddhism provides some
significance to these texts .

Question:  what type of sutras were generally ignored in india ?
```

Fig 3.2.1.1 Context and corresponding query

Memory networks are able to understand the relationships among the sequences in the context by utilizing an external memory matrix that acts as a storage unit. In simple terms, memory networks are neural networks with an external memory. This external memory is used by the model to retain and to retrieve information where it can be accessed multiple times to keep track of the information in the context that is relevant to the query. This ability to access the memory multiple times is called multi-hop reasoning, allowing the model to perform several iterations in relation to a specific context. Memory networks also have a soft attention mechanism that enables the model to keep track of the relevant information in the context, such that it can retrieve the accurate information. In other words, this attention mechanism allows the model to selectively focus on the important pieces of information with respect to the corresponding query. Thus, memory networks have the ability to recognize the dependencies and the relationships that exist in contextual information, which is further essential in question answering.

This concept offers an advantage over traditional methods such as that of RNNs which have a disadvantage of vanishing gradients; that is they are not able to retain information consisting of long sequences and cannot identify the existing dependencies between these sequences.

3.2.2 Unidirectional LSTM

In a unidirectional LSTM (Long Short-Term Memory) architecture, input sequences are processed only one way, from the past to the future. In traditional RNNs, the vanishing gradient can pose a problem, which is overcome by the LSTM (Long Short-Term Memory). With their complex structure, LSTMs can store and retrieve information over long sequences of inputs, forget gates, and output gates.

An LSTM employs unidirectional processing, which means that it processes the input sequence either forward or backwards, but not simultaneously.

- From the beginning to the end of the input sequence, Forward LSTMs are processed.
- An LSTM that works backwards is one that works from the end to the beginning of the input sequence.

3.2.3 Bidirectional LSTM

Traditional LSTMs (Long Short-Term Memories) can be redesigned in a bidirectional manner using LSTMs. An LSTM that operates in both directions processes input sequences: backwards from the present to the future, and forwards from the present to the future. At each time step, the network captures both past and future information, allowing it to provide a more comprehensive context for understanding the input sequence. Like a unidirectional LSTM, forward LSTMs process input sequences from start to finish. In the reverse LSTMs, the input sequence is processed by this part starting at the end and working its way back.

In concatenation, the final representation for a given time step is usually formed by concatenating or combining the outputs of the forward and backward LSTMs at that time step. The following layers or prediction-making processes then employ this composite representation.

3.3 Implementation Details

3.3.1 Preprocessing

Data collection and analysis are included in the first module of the proposed work. The necessary data set is gathered and examined for the SQuAD dataset by importing the dataset using the datasets python library. A total of 8000 samples are collected from the SQuAD dataset for our model. The text belonging to the question, answer and context attribute of the SQuAD dataset are preprocessed by first converting all the text from uppercase to lowercase. This is done through the process of tokenization using the nltk package. After the preprocessing is done, the data is split into a train test split using the sklearn python library with a test size of 0.2. Following this, using a tokenizer, the

context and the query for each element is tokenized into words after which it is then appended to a list. Each element in this list consists of a context, query and its corresponding answer.

A vocabulary of the unique words present in the data is generated. This is followed by the creation of two dictionaries:- one to map a unique index as the key to each word in the vocabulary and the other to map the words in the vocabulary as the key to a corresponding index. The next part in this section consists of vectorizing the context, query and the answer from the data. This is done by creating a list consisting of the index for each word in the context and the query, using the word to index vocabulary. The answers are vectorized into one hot encodings by converting it into an array of the vocabulary size. The element in the array which has the corresponding answer is labeled as one while the rest of the elements are labeled as zero. These vectors are then padded to obtain a fixed length such that they can be used as inputs in the neural network.

3.3.2 Model

After preprocessing and vectorizing the data, we obtain the padded vectors of the context, query and the answer for the train and test data. The model consists of the following:- input layers, sequential encoders (where the memory matrix is generated) and LSTM/BiLSTM layers. We will explain the implementation of these layers below.

In the implementation of the model, the first step was to define the input layers of the model for the context and the query and setting the size of each layer to the respective sizes of the context and query. Next, three sequential encoders were defined:- the sentences from the context are embedded into two memory vectors where the first sequential encoder has an output dimension of 64 and the second encoder has an output dimension of the maximum length of a query. The aforementioned implementation is that of the memory network for the storage of the context. The embedding is done to convert the padded vectors, which was obtained during preprocessing, into dense vectors of a fixed size. Dense vectors assign specific values to the elements present in the vector, usually to capture the contextual information present in the input. In addition to the embedding layers, a dropout layer of 0.3 was added to each encoder for better generalization of data and to prevent overfitting. The third sequential encoder is to embed the queries into dense vectors with an output dimension of 64, followed by the addition of a dropout layer of 0.3.

Next, a dot product is calculated between the first encoded context (obtained from the first sequential encoder) and the encoded query which is done to help the model recognize which parts of the context are relevant or important with respect to the query.

The dot product can also be seen as a basis for assigning the attention weights to the pieces of information in the context that correspond with the query. Proceeding this, a softmax activation function is applied to the dot product to transform the attention weights obtained into a probability distribution. This helps the model in generating the answer by selecting the most relevant information. These attention weights are then added to the second context embeddings and this result is then concatenated with the question encodings to output a tensor that is eventually inputted to the LSTM and BiLSTM layers.

Following the generation of the tensor containing the attention weights, it is inputted into two LSTM layers where sequential processing of the vectors present in the tensor takes place. The purpose of the LSTM here is to recognize the dependencies between the attention weighted information present in the vectors. The same process described above is done for the bidirectional LSTM as well. For both of these layers in each model, a dropout layer of 0.5 is added.

Finally, a dense layer is applied to the answer generated to get a probability distribution over all the elements present in the vocabulary. Both the models were trained for a total of 200 epochs each with a batch size of 32.

3.3.3 Mathematical Notations

Suppose there are a set of input sentences :- $x_{1,}\, x_{2,} \ldots, x_{n}$. The two memory vectors in the memory network is represented by m_i and c_i. The embedding matrices generated for the input context which is stored in the memory matrices are represented as A and C and the embedded question is represented as q. As explained in the section above, a dot product of the memory matrix m_i and q is calculated. This can be mathematically represented as:-

$$y_1 = \sum m_i . q$$

y_1 contains the attention weights for the current input context and this tensor is then added with the c_i memory matrix to generate y_2.

$$y_2 = c_i + y_1$$

y_2 is concatenated with the embedded question to obtain:-

$$o = \sum y_2 q$$

o represents the output of the memory network. Coming to the bidirectional LSTM in the model, there is a forward pass and a backward pass which is represented as shown below : -

$$ht = \sigma(Wfxt+Ufht-1+bf) \quad \text{(Forward Pass)}$$

$$ht = \sigma(Wbxt+Ubht+1+bb) \quad \text{(Backward Pass)}$$

Wf, Uf, bf are the parameters for the forward pass while *Wb, Ub, bb* are the parameters for the backward pass. *xt* is the current input at time step t. The hidden states obtained from the forward and backward pass are concatenated and then used for the update of the cell state.

$$ct = ft \odot ct-1+it \odot \tanh(Wcxt+Ucht+bc)$$

Finally, the model generates an output tensor consisting of a probability distribution over all the elements in the vocabulary.

3.3.4 Evaluation Metrics

1. **Exact Match (EM Score)**
   Exact Match is an evaluation metric commonly used in question answering systems to compare how well the model generated to the answer matches with the actual answer for a specific query. This metric generates binary output, i.e ‘0’ for a candidate answer that does not match. Even the slightest deviation from the reference answer can cause the EM result to be a zero. It generates a ‘1’ for candidate answers that are exactly the same as the reference answer. For example, if the predicted answer is ‘New York’ and the actual answer is ‘New York City’, the EM score would be a zero. Exact match is basically a strict measure for comparing candidate and reference answers.

2. **F1-score**
   The F1-score is another metric usually used in question answering systems. In this system, it is used to measure the accuracy of how well the system is able to predict an answer with respect to the query. It even considers partial accuracy, i.e if the model is able to recognize the type of query (what, how many, where, etc) and generate an answer that is accurate to the type of query asked. F1-score takes into account precision and recall which is used to measure the relevance of the answers from the context for a particular question.

$$F1 = \frac{2 * precision * recall}{precision + recall}$$

4. RESULTS AND DISCUSSION

After training the LSTM and the BiLSTM model for a total of 200 epochs each, the LSTM model was able to obtain a training accuracy of 84.27% while the BiLSTM model was able to obtain a training accuracy of 92.31%, showing that the BiLSTM model was able to obtain a higher performance than the LSTM model.

Answers generated by LSTM based memory network model on the test set:

```
1/1 [==============================] - 0s 23ms/step
Context:  at the age of 21 he settled in paris . thereafter , during the
last 18 years of his life , he gave only some 30 public performances ,
preferring the more intimate atmosphere of the salon . he supported
himself by selling his compositions and teaching piano , for which he was
in high demand . chopin formed a friendship with franz liszt and was
admired by many of his musical contemporaries , including robert schumann
. in 1835 he obtained french citizenship . after a failed engagement to
maria wodzińska , from 1837 to 1847 he maintained an often troubled
relationship with the french writer george sand . brief and unhappy visit
to majorca with sand in 1838-39 was one of his most productive periods of
composition . in his last years , he was financially supported by his
admirer jane stirling , who also arranged for him to visit scotland in
1848 . through most of his life , chopin suffered from poor health . he
died in paris in 1849 , probably of tuberculosis .
Question:  what was the likely cause of death for chopin ?
Prediction: tuberculosis | Actual: tuberculosis
Exact Match (EM) score:  1
Precision:  1.0
Recall:  1.0
f1 score:  1.0
```

Fig 4.1 Correct answer prediction

```
1/1 [==============================] - 0s 23ms/step
Context:  Beyoncé has worked with Tommy Hilfiger for the fragrances true
star singing a cover version of wishing on a star and true star gold she
also promoted emporio armani 's diamonds fragrance in 2007 . beyoncé
launched her first official fragrance , heat in 2010 . The commercial ,
which featured the 1956 song fever , was shown after the watershed in the
United kingdom as it begins with an image of beyoncé appearing to lie
naked in a room . In February 2011 , beyoncé launched her second fragrance
, heat rush . beyoncé 's third fragrance , pulse , was launched in
september 2011 . in 2013 , the mrs. carter show limited edition version of
heat was released . the six editions of heat are the world 's best-selling
celebrity fragrance line , with sales of over 400 million .
Question:  beyonce 's first fragrance had what name ?
Prediction: pulse | Actual: heat
Exact Match (EM) score:  0
Precision:  0.2
Recall:  0.25
f1 score:  0.22222222222222224
```

Fig 4.2 Partial answer prediction

```
1/1 [==============================] - 0s 25ms/step
Context:  st. barthélemy has about 25 hotels , most of them with 15 rooms
or fewer . the largest has 58 rooms . hotels are classified in the
traditional french manner 3 star , 4 star and 4 star luxe . of particular
note are eden rock and cheval blanc . hotel le toiny , the most expensive
hotel on the island , has 12 rooms . most places of accommodation are in
the form of private villas , of which there are some 400 available to rent
on the island . the island 's tourism industry , though expensive ,
attracts 70,000 visitors every year to its luxury hotels and villas and
another 130,000 people arrive by luxury boats . it also attracts a labour
force from brazil and portugal to meet the industry needs .
Question:  about how many villas are available for rent in st. barts ?
Prediction: 58 | Actual: 400
Exact Match (EM) score:  0
Precision:  0.0
Recall:  0.0
f1 score:  0
```

Fig 4.3 Wrong answer prediction

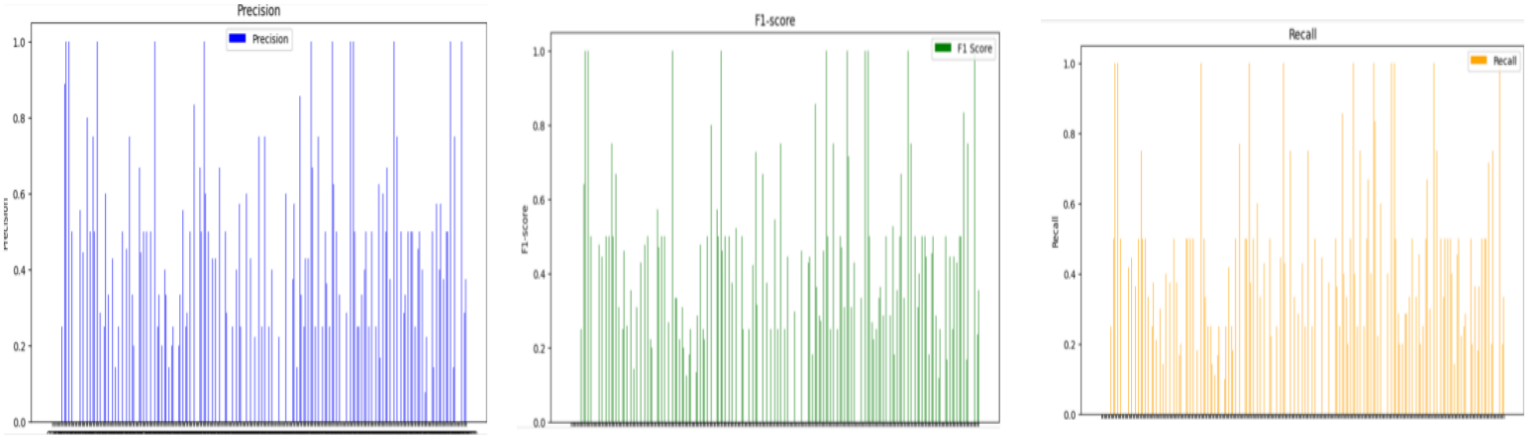

Fig 4.4 Graphs for the evaluation metrics(Precision, F1-score, Recall)

In fig 4.4, the graphs depict the precision, F1-score and recall calculated for each answer predicted by the model for the test data. It is observed from the values of 1.0 obtained for the F1-score, precision and recall, that the model was able to accurately predict the answer for a given query. However, the values for a large number of queries remain in the range of 0.4 - 0.6, depicting that the model was predicting partially accurate answers signifying that the model was able to understand the type of query and the type of answer that was expected. It can be concluded that the LSTM based model was able to understand the query and was even able to predict the correct answers for the unseen data.

Answers generated by BiLSTM based memory network model on the test set:

```
1/1 [==============================] - 0s 29ms/step
Context:  traveller in the early middle ages could obtain overnight
accommodation in monasteries , but later a demand for hostelries grew with
the popularity of pilgrimages and travel . the hostellers of london were
granted guild status in 1446 and in 1514 the guild became the worshipful
company of innholders .
Question:  when did the hostellers of london become a guild ?
Prediction: 1446 | Actual: 1446
Exact Match (EM) score:  1
Precision:  1.0
Recall:  1.0
f1 score:  1.0
```

Fig 4.5 Correct answer prediction

```
1/1 [==============================] - 0s 23ms/step
Context:  global city , boston is placed among the top 30 most economically powerful cities
in the world , encompassing 363 billion , the greater boston metropolitan area has the
sixth-largest economy in the country and 12th-largest in the world .
Question:  what ranking in the country does greater boston metro hold as far as economy ?
Prediction: 12th-largest | Actual: sixth-largest
Exact Match (EM) score:  0
Precision:  0.8333333333333334
Recall:  0.7692307692307693
f1 score:  0.8
```

Fig 4.6 Partial answer prediction

```
1/1 [==============================] - 0s 23ms/step
Context:  the university of kansas school of business is a public business
school located on the main campus of the university of kansas in lawrence ,
kansas . the school of business was founded in 1924 and currently has more
than 80 faculty members and approximately 1500 students .
Question:  when was the university of kansas school of business established
?
Prediction: half | Actual: 1924
Exact Match (EM) score:  0
Precision:  0.0
Recall:  0.0
f1 score:  0
```

Fig 4.7 Wrong answer prediction

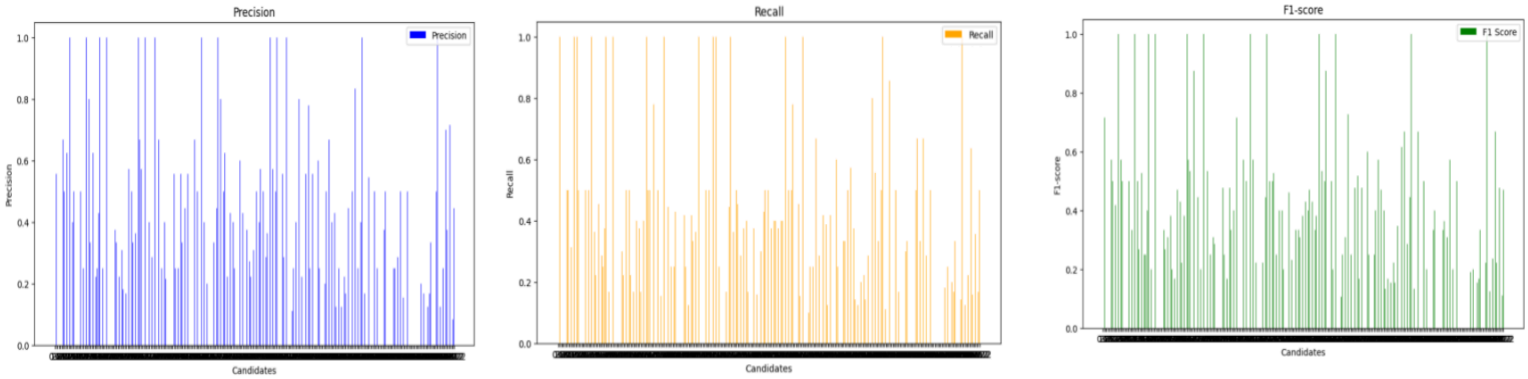

Fig 4.8 Graphs for the evaluation metrics(Precision, F1-score, Recall)

In fig 4.8, the graphs depict the precision, F1-score and recall calculated for each answer predicted by the biLSTM based model for the test data. It is observed that this model was able to predict a larger number of accurate answers than the LSTM based model. However, similar to the LSTM based model, the values for a large number of queries remain in the range of 0.4 - 0.6, depicting that the model was predicting partially accurate answers signifying that the model was able to understand the type of query and the type of answer that was expected. The biLSTM based memory network model was able to capture the dependencies present in the context better than the LSTM based memory network model.

Context and Query	**Predicted answer (LSTM based Memory network model)**	**Predicted answer(BiLSTM based Memory network model)**
Context: after the Estonian war of independence in 1919 , the **Estonian** language became the state language of the newly independent country . In 1945 , 97.3% of Estonia considered itself ethnic Estonian and spoke the language . **Question:** after Estonia achieved independence what was made their state language ? **Actual:** Estonian	Tibet	**Estonian**

Context: the above are further subdivided into 31 planes of existence . web 4 rebirths in some of the higher heavens , known as the śuddhāvāsa worlds or pure abodes , can be attained only by skilled buddhist practitioners known as anāgāmis non-returners . rebirths in the ārūpyadhātu formless realms can be attained by only those who can meditate on the **arūpajhānas** , the highest object of meditation . **Question:** what is the highest object of meditation called ? **Actual**: arūpajhānas	field	**arūpajhānas**

Table 4.1 Comparing the predicted answers of the two models

Table 4.1 depicts the answers predicted by the LSTM and biLSTM based memory network models for the same query. We can see that the biLSTM based memory network model is able to accurately capture the answer from the given context, while the LSTM model was not able to predict the correct answer.

Model	Precision	Recall	F1 score	Exact Match
LSTM Memory network	0.353	0.326	0.33	0.05
BiLSTM Memory Network	0.351	**0.34**	**0.334**	0.07

Table 4.2 Evaluation metric scores for each model

From table 4.2, it is evident that the BiLSTM based memory network is able to perform better than the LSTM memory network as it has obtained higher Recall and F1-scores. This is further supported with Table 4.1 where the BiLSTM model is able to capture the relevant information from the context while the LSTM model could not do so.

From the figures provided with the answer prediction, both models are able to predict accurate answers with respect to the context and the query. The partial accurate answers refer to the ability of the model to understand the type of question in the query i.e what, why, how many, etc. While the answer is not accurate to the actual answer, it shows that the model has understood what type of answer is expected for a particular type of question.

CONCLUSION

Question-Answering (QA) models, in simple terms, is an information retrieval (IR) system that retrieves an answer in a natural language from a corpus of data in response to a query posed by a human. The challenges of responding to questions using the standard traditional methods are examined, and a methodology is proposed in this paper. The BiLSTM based memory network model was found to outperform the LSTM model in terms of performance in understanding the relationships between the sequences in the data. In short, the model was able to understand the context for each question and provide accurate answers. The bidirectional LSTM is used in the aforementioned analysis to create a neural network model for Q&A by integrating it with a memory network. This BiLSTM-based Q&A system is used to assess the SQuAD dataset in an all-encompassing manner.

The selection of BiLSTM points to a desire to record context in both directions in order to enhance question-answering skills. However, the model can be improved in performance by training it with a larger number of epochs and with a wider variety of diverse training examples.

REFERENCES

[1] Singh, S. et al. (2016). The Question Answering System using NLP and AI. *International Journal of Scientific & Engineering Research*, 7(12), 55-60. https://www.ijser.org/researchpaper/The-QuestionAnswering-System-Using-NLP-and-AI.pdf

[2] Ishwari, K. S. D. (2019). Advances in Natural Language Question Answering: A Review, *Computation and Language*. doi: https://doi.org/10.48550/arXiv.1904.05276

[3] Stroh, E. et al. Question Answering using Deep Learning. https://api.\semanticscholar.org/CorpusID:39221661

[4] Almotairi, M. et al. (2022). Developing a Semantic Question Answering System for E-Learning Environments using Linguistic Resources, *Journal of Education and e-Learning Research*, 9(4), 224-232. doi: 10.20448/jeelr.v9i4.4201

[5] Ahmed, W. et al. (2017). An Automatic Web-Based Question Answering System for E-Learning. Information Technologies and Learning Tools, 58(2). http://hdl.handle.net/20 500.12424/1621313

[6] Li, W. et al. (2018). A Unified Model for Document-Based Question Answering Based on Human-Like Reading Strategy, *AAAI Conference on Artificial Intelligence.* doi: 10.1609/a aai.v32i1.11316

[7] Lende, S. P., & Raghuwanshi, M. M. (2016). Question answering system on education acts using NLP techniques, *2016 world conference on futuristic trends in research and innovation for social welfare (Startup Conclave)*, 1-6.

[8] Singh, S., Das, N., Michael, R., & Tanwar, P. (2016). The Question Answering System Using NLP and AI, *International Journal of Scientific & Engineering Research*, *7*(12), 2229-5518.

[9] Bouziane, A., Bouchiha, D., Doumi, N., & Malki, M. (2015). Question answering systems: survey and trends. *Procedia Computer Science*, 73, 366-375.

[10] Biancofiore, G. M., Deldjoo, Y., Di Noia, T., Di Sciascio, E., & Narducci, F. (2022). Interactive Question Answering Systems: Literature Review. *arXiv preprint arXiv:2209.01621*.

[11] Elworthy, D. (2000). Question Answering Using a Large NLP System. In *TREC*.

[12] Hsu, C. C. et al. (2021). Answer Generation for Retrieval-Based Question Answering Systems, *Findings of the Association for Computational Linguistics*, 4276-4282. doi: https://arxiv.org/abs/2106.00955

[13] Otegi, A. et al. (2022). Information Retrieval and Question Answering: A Case Study on COVID-19 Scientific Literature, *Knowledge-Based Systems*, 240. doi: 10.1016/j.knosys.2021.108072

[14] Zhang, Q. et al. (2023). A Survey for Efficient Open Domain Question Answering, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 1, 14447-14465. doi: https://doi.org/10.48550/arXiv.2211.07886

[15] Lee, Y. et al. (2023). QASA: Advanced Question Answering on Scientific Articles, *Proceedings of Machine Learning Research.* doi: https://proceedings.mlr.press/v202/lee23n/lee23n.pdf

[16] Vegupatti, M., Nickles, M., & Chakravarthi, B. (2020). Simple Question Answering Over a Domain-Specific Knowledge Graph using BERT by Transfer Learning, *Irish Conference on Artificial Intelligence and Cognitive Science*. doi: https://ceur-ws.org/Vol-2771/AICS2020_paper_42.pdf

[17] Bhardwaj, D., Pakray, P., Bentham, J., Saha, S., Mizoram, N. I. T., Gelbukh, A., & Mexico, I. P. N. (2016, December). Question answering system for frequently asked questions. In *Proceedings of the final workshop*, 7, 129.

[18] Bansal, A., Eberhart, Z., Wu, L., & McMillan, C. (2021, March). A neural question answering system for basic questions about subroutines, *2021 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 60-71. IEEE.

[19] Chafik Taiebennefs, Department of Computer Science Stanford University. Question Answering System Implementation Using QANet Architecture. In Stanford CS224N, Stanford University.

[20] Farea, A., Yang, Z., Duong, K., Perera, N., & Emmert-Streib, F. (2022). Evaluation of Question Answering Systems: Complexity of judging a natural language. *arXiv preprint arXiv:2209.12617*.