Question And Answer Generating System In The Form Of Fill Ups Using RNN

Akshaya A
Master of Computer Applications
Hindustan Institute of Technology and Science
Padur, India
aksh153026@gmail.com

P. S. Sreeja
Department of ComputerApplication
Hindustan Institute of Technology and Science
Padur, India
srj_ps@yahoo.com

Abstract—**The questions and answers generation system (QAG) is a computer science discipline in the areas of information retrieval and natural language (NLP), involving building systems that respond automatically to questions asked in natural language. In this paper, we proposed a research work to generate questions and answers for the given input text. This work is modeled by using RNN and NER for technical terms. QAG system has been generated the questions and answers with 95 % of accuracy. This will reduce the time consumption and workload of the question setters. The questions and answer system is designed for educational institutions (schools, universities, training centres etc).**

Keywords— ***QAG, RNN, Deep Learning, NER, Artificial Intelligence***

I. INTRODUCTION

Asking closely connected and appropriate questions are always a part of human learning techniques. It helps human in understanding of a various ideas and models (an essay, an article, etc.). However, creating queries manually can be difficult at times. Automated Question Answer Generating (QAG) systems help to solve large number of problems in a small amount of time. Such a system includes applications such as Filling in the Blanks generation, intellectual training systems and virtual assistants. The QAG system produce worthful, correct, and natural questions from the text. In addition, further automate the assessment of human users, it is highly desirable that the questions are relevant to the text and have sub-answers in the text.

We are going to propose an efficient question generation method for Fill in the Blanks types of questions. teachers and students can benefit from in their teaching and learning environment. The system takes one paragraph of input and generates questions with important sentences extracted from the paragraph. The Filling in the Blanks can be made up of complex and simple sentences. To overcome this, we use the concept of deep learning. Our system generates questions and answers from a text and the questions will be Fill in the Blanks. In this system, we implemented three phases. The first phase is the data preprocessing. The second phase is to identify technical Named Entities. The third phase is to generate questions and answers from the given text using RNN. We have implemented this system by training Core Java learning materials.

II. RELATED WORD

In a survey of questionnaire creation [1], the user will be given lessons to create a query on the computer, which will then go to the database, and then automatically generate queries. The issue in this work is the output as a PDF format or MS word. In Automatic Question Generation from Paragraph system [2], this is an improvement given the input test to the system, which is rational.

In Unanswerable Questions for SQuAD system [3], it identifies the unanswered questions from the SQuAD. In Machine Comprehension by Text-to-Text Neural Question Generation system [4], it uses the neural encoder-decoder model to generate meaningful and distinct queries from natural language sentences. The encoder reads the input message and the response status by creating a display of the input data that will be provided to the decoder. We conducted a preliminary study on how to create neurological questions using text from the SQuAD data set, and test results show that our methods are able to create dynamic and diverse search queries.

In Automating Reading Comprehension by Generating Question and Answer Pairs system [5], for the first step, it will answer the sentence using the pointer networks from the database. For the second stage, it will perform a series of query models for query creation, which will be enhanced with rich linguistic features. Finally, this will create a more relevant question to answer.

Question generation[6] system will identify keywords from the text and use them as answers to the question using the Naive Bay Algorithm. After that, this application is to replace the answer from the sentence with the blank space and use it as the basis for the question.

In Reinforced Mnemonic Reader for Machine Reading Comprehension[7], first, in order to avoid the problems of attention redundancy and attention deficit disorder, a reassessment mechanism is proposed to refine the present attention by directly accessing the temporal minds temporarily memorized in a multi-circuit alignment framework. Second, a new, positive way Called dynamic critical learning it is recommended to extend the standard methods of guidance. It is always encouraging To predict the most acceptable response to address There was a problem of merger suppression Traditional Reinforcement Learning Instructions. SQuAD show that our model is Achieving sophisticated results.

In Automatic Question-Answer Pairs Generation From Text[8], for the first step, pre-processing is done. For the second stage, it will perform a Sentence Selection. For the third stage, this will perform Question Formation.

Using Natural Language Processing for Smart Question Generation[9] selects the best possible set of sentences from the given text entry, from which queries are created (sentence selection).

From the literature survey, it is clear that there was no system to generate Fill-up questions and answer. So we are proposing a system for generating questions and answers.

III. Proposed System

The proposed system QAG is used to create questions in the form of filling in the blanks from the given input textual content.

When the input is given this proposed system will identify with the technical Named Entities in the given input text. Deep learning RNN will generate Filling in the Blank from the given text. Later, it leads to generate the question and answer in the form of filling in the blanks.

Figure 1 indicates the Questions and Answers Generation(QAG) System is to generate the question and answer for Filling in the Blank.

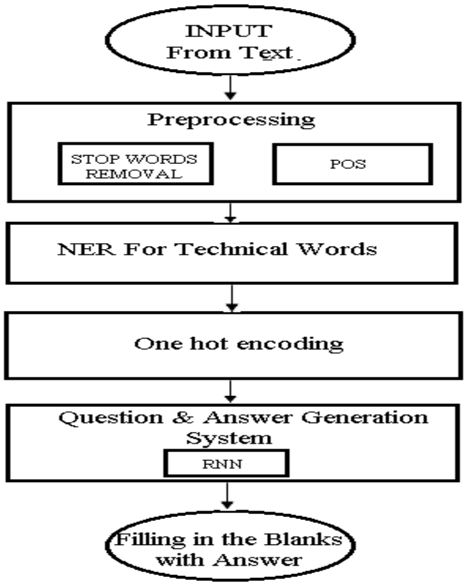
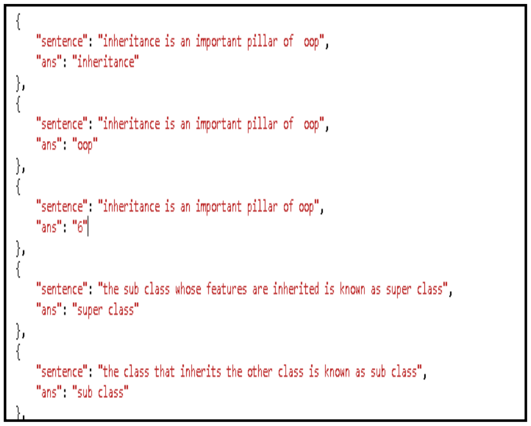


Fig. 1. Architecture Of Proposed System

A. Prepocessing

Figure 2:depitcts the dataset consists of Core Java technical material for the QAG system. This dataset consists of two units of attributes as sentences and answers.



```
{
   "sentence": "inheritance is an important pillar of  oop",
   "ans": "inheritance"
},
{
   "sentence": "inheritance is an important pillar of  oop",
   "ans": "oop"
},
{
   "sentence": "inheritance is an important pillar of oop",
   "ans": "6"
},
{
   "sentence": "the sub class whose features are inherited is known as super class",
   "ans": "super class"
},
{
   "sentence": "the class that inherits the other class is known as sub class",
   "ans": "sub class"
},
```

Fig. 2. Core Java Technical Dataset

In the preprocessing, the QAG application will create the complete dataset consists of some functions like Java Named Entities R, Part of speech (POS) tags and Word count.

Stop words are the most common words in any natural language. These punctuation words do not add much value to the meaning of the document when it comes to analyzing textual data and building NLP models. Example In general, the most not unusual words utilized in a speech are "where", "in", "for", "at", "when", "to".

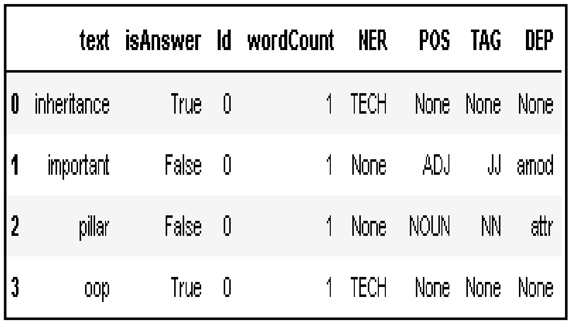
The toughest and quickest rule within the NLP is not to delete phrases. It depends on what we are working on. It relies upon on what we are operating on.

For tasks which include textual content category, the text need to be labeled into one of a kind categories, with terminated phrases being eliminated or excluded from the given text so that more attention can be paid to the words that outline the that means of the textual content..

In the preprocessing, this application tokenized each non-stop word from the sentence in the Core Java technical dataset.

B. Techinical NER

The Java Named Entities are recognized from the tokenized phrases. A Part of speech (POS tagger) recognized the phrase categories inclusive of noun, verb, adjective, etc. Figure 3 stated the list of technical Named Entities recognized from the given input text.



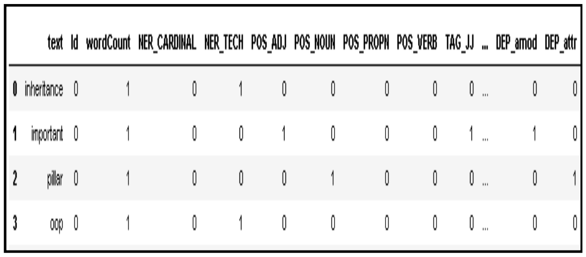
	text	isAnswer	Id	wordCount	NER	POS	TAG	DEP
0	inheritance	True	0	1	TECH	None	None	None
1	important	False	0	1	None	ADJ	JJ	amod
2	pillar	False	0	1	None	NOUN	NN	attr
3	oop	True	0	1	TECH	None	None	None

Fig. 3. Feature Engineering Table

C. Answer Identification

we carried out binary classification every phrase from the sentence to perceive answers. The spaCy is applied for the word tagging on every phrase from the sentence.

Figure 4 depicts the output of one-hot encoding which is a binary classification of Features



	text	Id	wordCount	NER_CARDINAL	NER_TECH	POS_ADJ	POS_NOUN	POS_PROPN	POS_VERB	TAG_JJ	...	DEP_amod	DEP_attr
0	inheritance	0	1	0	1	0	0	0	0	0	...	0	0
1	important	0	1	0	0	1	0	0	0	1	...	1	0
2	pillar	0	1	0	0	0	1	0	0	0	...	0	1
3	oop	0	1	0	1	0	0	0	0	0	...	0	0

Fig. 4. Binary Classification Table

D. Question and Answer Generation

Figure 5 depicts the Question and Answer Generation System, LSTM Recurrent Neural Networks(RNN) is used. Unlike standard feedforward neural networks, LSTM has concept links.

.

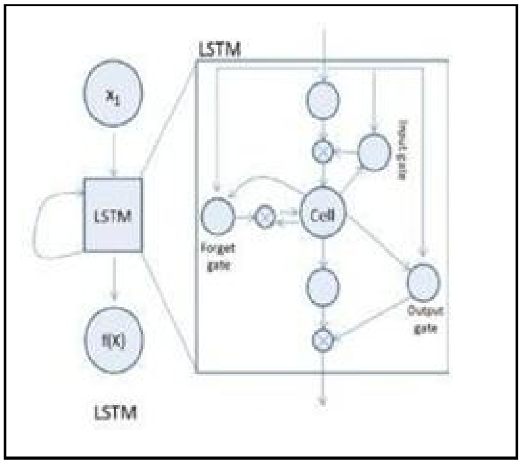


Fig. 5. LSTM Recurrent Neural Networks

1. *Input gate layer:* Equation (1) determines which values to store in the cellular state:

   $i_t= \sigma(w_i x_t+u_i h_{t-1}+b_i)$ (1)

2. *Forget gate layer:* As the name suggests determines which statistics to throw cellular state. It is calculated via Equation (2)

   $f_t= \sigma(W_f x_t+U_f h_{t-1}+ b_f)$ (2)

3. *Output gate layer:* A vector of values will be delivered to the cell state via Equation (3)

   $O_t= \sigma(W_o x_t+u_i h_{t-1}+b_o)$ (3)

4. *Memory cell:* The cell recollects values at arbitrary time durations and the 3 gates control the go with the flow of data in and out of the cellular with the aid of Equation(4)

   $c_t=f_t o c_{t-1}+i_t o*\tanh(W_c x_t+u_c h_{t-1}+b_c)$ (4)

The keras models are known as organization of layers. A fully connected network architecture with three layers will be used. These layers are expressed in two classes, namely LSTM classes and dense class. First, LSTM layer is having 50 memory units and input shape arguments. Second, LSTM layers is having 128 memory units. Third, dense output layer is having as single neuron and a sigmoid activation function to make a zero or 1 prediction (True or False for both the classes in the problem. Let y be output determined by Equation (6).

$$y = f(x) = \frac{1}{(1 + \exp(-x))} \quad (6)$$

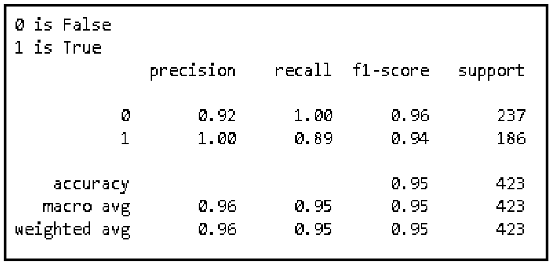
Presently the model is resolved, we can collect it. We have to determine the misfortune capacity to be utilized to assess the arrangement of weights, the enhancer used to scan for various weights for the network. For this situation, we will utilize cross entropy as a misfortune contention. This misfortune is for double order issues and is characterized as "binary_crossentropy" in Keras.

The optimizer is defined as an efficient algorithm "Adam" because it is a popular version as it adjust itself and provides efficient results for many problems. As it fixes the problem faster and efficiently, it is applied to 15 epochs.

A batch size of 30 reviews has come out weight updates. In the output layer , a function is used sigmoid activation function. so the predictions will be a probability in the range between 0 and 1. After that identify the words using TRUE and FALSE values and generate the question using identified words.

IV. Result And Analysis

Figure 6 shows the QAG system has been predicted 95% of accuracy in the testing dataset. Figure 7 shows the QAG system has been visualizing in result predicted and accuracy value in the graph. The predicted value will become in 0 (True) and 1(False) Redline is a real answer and the Blue line is predicted answer. Figure 8 depicts the confusion matrix of the QAG System.

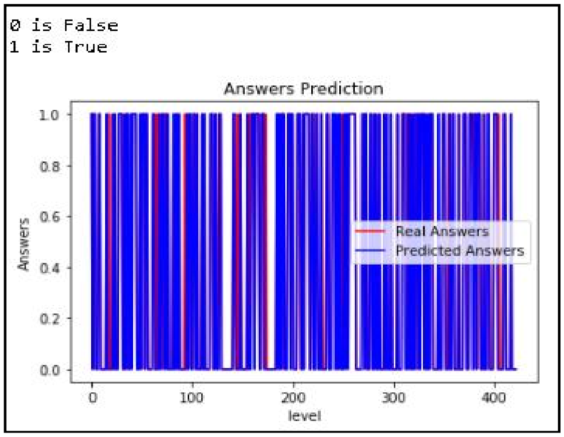


```
0 is False
1 is True
              precision    recall  f1-score   support

           0       0.92      1.00      0.96       237
           1       1.00      0.89      0.94       186

    accuracy                           0.95       423
   macro avg       0.96      0.95      0.95       423
weighted avg       0.96      0.95      0.95       423
```

Fig. 6. Visualizing in result predicted and accuracy value



```
0 is False
1 is True
```

Fig. 7. Predicted accuracy % testing dataset

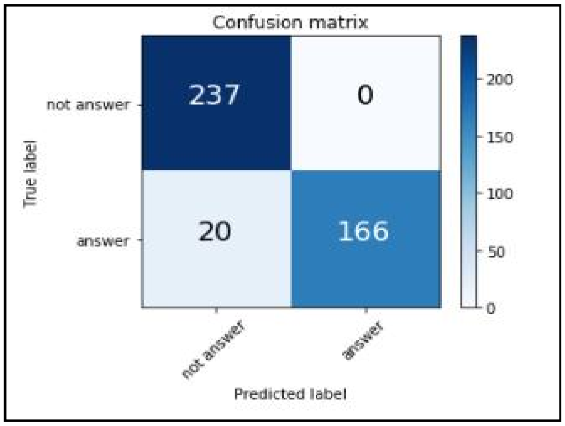
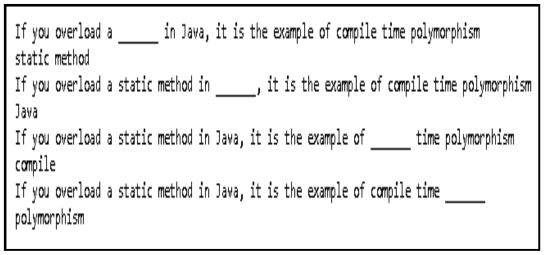


Fig. 8. Visualizing in confusion matrix as Predicted and True Label

Figure 9 shows sample question-answer generated, as well as its context sentence. This system also generate the answers for generated question.



If you overload a _____ in Java, it is the example of compile time polymorphism
static method
If you overload a static method in _____, it is the example of compile time polymorphism
Java
If you overload a static method in Java, it is the example of _____ time polymorphism
compile
If you overload a static method in Java, it is the example of compile time _____
polymorphism

Fig. 9. Sample question-answer generation

V. Conclusion and Future Work

The proposed QAG system secured 95% of accuracy. Our system is predicting answer using the technical NER. The trained dataset consists the core Java learning material. As a future work, the training dataset has to be extended for all the subjects. In this proposed work we generated the questions at Blooms Taxonomy first level. This has to be extended to the other levels of Bloom's taxonomy.

References

[1] Aniruddha Joshi, Mayur Joshi, Prathamesh Kudnekar, Siddhesh Doiphode, “A Survey on Question Paper Generation System”, IJCA Proceedings on National Conference on Role of Engineers in National Building NCRENB, July 2016.

[2] Dhaval Swali, Jay Palan, Ishita Shah, “Automatic Question Generation from Paragraph “, International Journal of Advance Engineering and Research Development, Scientific Journal of Impact Factor (SJIF), Dec 2016.

[3] Pranav Rajpurkar, Robin Jia, Percy Liang, “Unanswerable Questions for SQuAD”, Computer Science Department,Stanford University, July 2018.

[4] Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman1, Sandeep, Saizheng Zhangand, Adam Trischler, “Machine Comprehension by Text-to-Text Neural Question Generation”, July 2015.

[5] Vishwajeet Kumary Kireeti Boorlay, Yogesh Meenay,Ganesh Ramakrishnany, and Yuan-Fang Lix, “Automating Reading Comprehension by Generating Question and Answer Pairs“, Indian Institute of Technology Bombay, IITB-MonashResearch Academy in Mumbai ,Monash University in Australia, Mar 2018.

[6] Kristiyan Vachev, “Question Generation“, studied Bachelor Degree Informatics and Mathematics & Software Engineering at Sofia University.

[7] Reinforced Mnemonic Reader for Machine Reading Comprehension Minghao Hu, Yuxing Peng , Zhen Huang , Xipeng Qiu , Furu Wei, Ming Zhou, College of Computer, National University of Defense Technology, Changsha, China School of Computer Science, Fudan University, Shanghai, China Microsoft Research, Beijing, China, Jun 2018.

[8] Automatic Question-Answer Pairs Generation From Text, Holy Lovenia, Felix Limanta, Agus Gunawan, Department of Informatics, Bandung Institute of Technology, Indonesia.

[9] Using Natural Language Processing for Smart Question Generation,By aditya s., published on July 4, 2018.