Abstract—With the increased adoption of E-learning platforms, keeping online learners engaged throughout a lesson is challenging. One approach to tackle this challenge is to probe learners periodically by asking questions. The paper presents an approach to generate questions from a given video lecture automatically. The generated questions are aimed to evaluate learners’ lower-level cognitive abilities. The approach automatically extracts text from video lectures to generate wh-kinds of questions. When learners respond with an answer, the proposed approach further evaluates the response and provides feedback. Besides enhancing learner’s engagement, this approach’s main benefits are that it frees instructors from designing questions to check the comprehension of a topic. Thus, instructors can spend this time productively on other activities.

Keywords—E-learning, Engagement of online learners, Automatic question generation, Cognitive evaluation, Natural Language Processing.

I. INTRODUCTION

There has been a shift in recent times from traditional classroom learning towards remote learning. This implies that students can enrol in courses from different universities without needing to be physically present on the universities’ campuses. Massive Open Online Courses (MOOCs) platforms like MIT OpenCourseWare¹, Coursera², EdX³ have become popular because of their flexible nature and their ability to provide excellent video lectures. To ensure that students have grasped the concepts explained in a video lecture, there is a need to regularly assess students on the topics presented in a video lecture. Traditionally, a course instructor often manually prepares a quiz containing simple questions to assess if students have had a holistic understanding of the lecture. However, manually preparing questions is a time consuming and cumbersome task. An instructor could use the time spent designing questions to prepare lecture material and presentations, which will be more fruitful and beneficial for the students as well as the course instructor.

Instructors can automate the question designing task with advanced techniques from Natural Language Processing and Video Processing. Automatic question generation revolves around the generation of questions from a given text. In our case, the input provided would be a video lecture for which assessment needs to be done. This paper presents the design of a system that generates questions from real-time video lectures. The objective behind generating questions is to increase online learners’ engagement by asking questions to check the comprehension of the topic being presented in a video lecture. Some of the questions being generated would also have relevant screenshots from the video. These images could be directly linked to the question, could provide context to a question, or could serve to refresh a student’s concept when he is asked that question.

The paper contributes by designing an automated approach to generate question papers from the videos. The approach’s main highlights include (i) Generated questions tests lower-level cognitive abilities necessary during formative assessment. (ii) Linking questions with images to enhance the test-taking experience. (iii) Providing timely feedback on the learning progress.

The rest of the paper is organized as follows. Section 2 discusses the Related Work. The proposed methodology is elaborated upon in Section 3 along with implementation in section 4. Results and analysis of our approach has been elaborated upon in Section 5. Conclusion and future scope are presented in Section 6 with references at the end.

II. RELATED WORK

Automatic Question Generation has observed great scope in the education domain. The research done in question generation (QG) has expanded exponentially in the past few years. Heilman[1] has developed a state-of-the-art system that uses various Natural Language Processing (NLP) techniques and tools such as the Tregex expressions for T-Surgeon, BBN Identifier and Stanford Parser to generate questions from a given text. In recent years, there has been large improvements in the QG models. Sequence-to-Sequence models are becoming the most widely adopted model for Question Generation today. These models often use LSTMs, but the use of transformers is also quickly gaining popularity [2] [3]. These models have been improved to become state-of-the-art
by the use of extra linguistic features [4] or by the use of answer-awareness[5].

The majority of systems focus on automatic generation of questions from textual documents rather than the audio-visual medium. A few notable systems that work towards question generation from videos are specified as follows. The authors of [6] have proposed the first model that generates questions of varying types by taking images or captions as input. [7] presents an automatic question generation system from MOOCs’ videos at runtime. The transcripts of the videos use automatic discourse segmentation followed by content retrieval from Wikipedia documents to generate questions. The authors of [8] use a web scraper to crawl the transcripts of TED videos to generate two types of Multiple-Choice Questions (MCQs) that aim to assess listening comprehension, i.e., evaluating the listener’s comprehension of the lecture’s gist and the details described in it. The most related work to ours is [9] which produces questions from the subtitles of accompanying documentary videos. This is followed by referencing the timestamp of the subtitle and attaching a screenshot of the video along with the question.

Druschkov, P. N. and V. D. Kustikova [10] have conducted a survey on the deep learning methods for image classification as well as object detection. They have compared the different approaches used in image classification such as sparse coding, autoencoders, restricted Boltzmann machines and finally, convolutional neural networks (CNN). They have additionally discussed the loss functions and learning methods used in the aforementioned approaches. Albawi et al.[11] present an in-depth understanding of each element present in a CNN such as stride, padding, etc. They have also elucidated the underlying mathematics that is used in the convolution process, as well as given an in-depth explanation of different layers like Pooling and Fully-Connected layers. Altenberger and Lenz [12] have presented an extensive survey that enlists the famous CNN architectures such as LeNet-5 [13] AlexNet [14], ZFNet [15], VGGNet [16] and GoogLeNet [17], ResNet[18] and Inception-v4[19]. Zou et al. [20] have conducted an extensive survey on the methods of object detection. Some of the widely used methods for object detection include the Regions with CNN (R-CNN), Faster R-CNN, Spatial Pyramid Pooling Networks (SPPNet), and RetinaNet and You Only Look Once (YOLO) object detection algorithm.

This paper works to combine the techniques described in the following section in ways that enhance the test-taking experience and provide timely feedback to learners on their progress.

III. PROPOSED METHODOLOGY

The stages of the system are shown in Fig 1. Input is given in the form of videos. The output of each stage is provided as input to the next stage. Questions are outputted and feedback is provided based on the answers given by the learner.

A. Video processing

The objective of this step is to generate subtitles from the given video segment. The videos are split into 1-minute intervals in the case of real-time videos. First, the audio track is extracted from the video using the FFmpeg\(^4\) library. The input video format can be in any of the formats (MPEG-1, MPEG-2, MPEG-4, ACM, etc) supported by the FFmpeg library. This methodology allows for the extraction of audio from a variety of videos. Once the audio track has been extracted, the audio track is passed to the Google Speech-to-Text API\(^5\), converting the audio to text. The Google Speech-to-Text API can process audio streams of 1-minute duration synchronously. For audio streams that are greater than 1 minute in length, asynchronous processing of the stream can be done. Then the generated text from the Google Speech-to-Text API is processed by the srt API to create subtitles in ‘.srt’ format.

B. Dividing the transcript into subtopics

The next step involved is dividing the text into multiple subtopics. The subtopics are identified with the aid of the TextTiling Algorithm [21]. The TextTiling Algorithm does not detect the subtopics, but rather detects the shift in topics by identifying changes in vocabulary. To identify these shifts in topics, pseudo sentences are formed by grouping 20 consecutive stemmed words. The lexical cohesion between pseudo sentences is computed, which signifies the similarity between the sentences. These scores can then be plotted to show peaks and valleys. The depth score, which is the depth of the valley, is computed for each token-sequence pair. The depth scores are sorted and used to determine segment boundaries based on a threshold value. These computed boundaries are used for displaying changes in topics and are stored for later computation.

C. Question Generation

Once the video has been processed, subtitles have been generated, and subtopic boundaries have been identified, the next step is Question Generation. Since the proposed system is being used for real-time videos and lectures, the kind of question that shall be generated must be comprehensive enough to test the content of the videos. At the same time, they should not be very complicated or difficult because students are still in the learning stage and they are yet to master the topic. With this in mind, the focus is on shallow question generation rather than deep question Generation. In other words, these questions shall evaluate the lower-level cognitive abilities in terms of Bloom’s Taxonomy i.e., Remember, Understand and Analyze [22].

For the purpose of question generation from transcripts, we use Google's Text-To-Text Transfer Transformer (T5) base model[23]. The T5 model has been pretrained on the open-source pre-training dataset, called the Colossal Clean Crawled Corpus (C4)\(^6\). The T5 model achieves state-of-the-art results on many NLP benchmarks while being flexible

---

\(4\)  [https://www.ffmpeg.org](https://www.ffmpeg.org)

\(5\)  [https://cloud.google.com/speech-to-text/docs/aesnc-recognize](https://cloud.google.com/speech-to-text/docs/aesnc-recognize)

\(6\)  [https://www.tensorflow.org/datasets/catalog/c4](https://www.tensorflow.org/datasets/catalog/c4)
enough to be fine-tuned to a variety of important downstream tasks[24]. This T5 model has been further trained for Answer-Aware Question Generation Task7. In answer-aware models, the model is provided with the context and the answer as input, and it generates the question for that answer corresponding to context as output. In our case, the context would be the transcript of the video.

For the generation of questions, the first task is the generation of answers. To do this, first the sentences are tokenized using an appropriate tokenizer model (such as the Transformer's pretrained tokenizer model). These tokenized sentences are passed as inputs to the pretrained transformer model, the output of which is the answer encapsulated in the <sep> tag. Now that the answers have been generated, each sentence that has an answer is highlighted with <hl> tokens. The target text is generated by joining the answers in that sentence with <sep> tokens. This target text is then used to generate questions using the pretrained transformer model.

D. Image Classification

To enhance the learning and test-taking experience, the questions generated in the previous steps are linked with appropriate images from the video wherever possible. To identify appropriate images from the video, we have used the well-known LeNet model[13]. We have modified the model to have a single output neuron in the final Dense Layer, and used the “Binary Crossentropy” loss function, paired with the “sigmoid” activation function for the final layer to scale the output between 0 and 1.0. 0 is the class that represents “equations” and 1 is the class that represents “graphs”. If the value of the output is less than or equal to a threshold value (0.25 in our case), we infer the prediction of the class “equation”. For values above another threshold (0.75) we infer the prediction of the class “graph”. For any values between 0.25 and 0.75, we discard the questions because the model isn’t confident enough of either of the classes.

The layers used are Convolution, Average Pooling, Flatten, and Dense. The model has been compiled using the “Adam” optimizer[25] and “Binary Crossentropy” Loss function[26] for the purposes of binary classification.

E. Image Linking to the Relevant Questions

For every meaningful question generated, we check if the question generated contains any of the class labels present in the training dataset used to build the classification model and any other labels that a lecturer deems appropriate. For instance, one of the questions generated for the second video of the dataset in section 4.1 is “What is the leftmost graph?”. We know that the question contains the class label “graph”. After this, the source sentence which generated this question

![Fig. 2: Sample image linked to the question: “What is the leftmost graph?”
Timestamp: 05:25](https://huggingface.co/valhalla/t5-base-qg-hl)

Algorithm 1: Image Linking Algorithm

**Result:** Links Questions with Relevant Images

**Input:** Questions from the QG Model, Class Labels

**Output:** Questions Linked with (if any) Relevant Images

**Begin**

for question in generated_questions
  if any_class_labels not in question
    output the question
  else
    get timestamp of question
    run Image Classification Model for video frames in timestamp range
    output ← result of prediction
    if output < 0.25 or output>0.75
      output the question with image
    else
      delete the question //ambiguous question
  endif
endif
endfor

**End**

is located. The source sentence needs to be mapped to a timestamp contained in the subtitles. Once the timestamp of the source sentence has been found, the classification model runs against all frames on the sentence’s time range to classify the object, which in this example would be the equation. Lastly, if the object is successfully found based on the above-mentioned threshold values, a screenshot of that time frame is attached to the question. If no object is found in the given time range, then the question is discarded, as such a question would be considered a vague question. The pseudocode for this is shown in Algorithm 1, and a sample question with the linked image is shown in Fig. 2.

F. Post-processing of Questions

a) Assigning Questions: A certain number of questions need to be selected to assess a student. The number of questions depends on the video’s total length. For instance, for a 40-minute-long video, it is suggested to have 20 questions. This total number can be decided by the teacher beforehand. Each subtopic shift discussed in section 3.2 would get proportionate weightage to the amount of time it was spoken about in the video (length of subtitles in total subtitles). Finally, we propose selecting random questions from the total for each student to allow unique learning.

b) Feedback Design: Since the generated questions require students to type out their answers, qualitative feedback needs to be provided to each question. The qualitative feedback is generated by computing semantic similarity between the learner’s answer and automatically extracted model answer. Correctness of answer has been implemented based on a methodology proposed by Li et al[27]. A similarity of greater than 0.66 has been assigned as high similarity, between 0.33 and 0.66 as medium similarity an answer with lesser than 0.33 similarity has been assigned low similarity. By providing real-time feedback, the student’s learning is made more comprehensive.
IV. IMPLEMENTATION

This section explains the rationale behind the choices made while implementing the system, which is as follows:

A. Domain Selection and Creation of Videos Dataset

To narrow the focus of our proposed system, we choose the domain of “Algorithms used in Machine Learning and Graph Theory”. Since there is an increased demand for these skills, the number of video lectures on them has consequently increased, giving us many videos on which our system can be applied. We have chosen four videos in our dataset to run against the system. The details of the dataset are shown in Table 1. Further, this will help us decide the type of images used in our dataset to train the classifier.

B. Preparing Image Dataset and Training for Image Recognition Model

To deploy the classification model for our purposes, we have prepared our dataset by searching the web for images of equations and graphs relevant to our selected domain. To train our modified LeNet model, we used around 200 images of each class for accurate results. For the purposes of this paper, the model has been trained to detect 2 classes of images - “Equations” and “Graphs”. Examples of relevant images included in the dataset are shown in Fig. 3, Fig. 4, and Fig. 5.

V. EVALUATION OF THE APPROACH

The evaluation of the system has been divided into the evaluation of the classification model, evaluation of the question generation model, and evaluation of images linked to questions.

\[
p(y^{(i)}|x^{(i)}; \theta) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left( -\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2} \right).
\]

Fig. 3. Example of dataset image of class "equation"

Fig. 4. Example of dataset image of class "graph"

Fig. 5. Example of dataset image of class "graph"

A. Evaluation of Classification Model

The LeNet model was trained on our custom dataset, using an 80-20% split for training and validation. The model achieved a training accuracy of 0.9734 and a validation accuracy of 0.9664. The expected results for validation accuracy were in the range of 0.95 to 0.99, and the achieved values fit in the said range. Apart from this, the model was tested on a test dataset containing 120 randomly chosen images of both classes. The accuracy achieved on this test dataset was 0.9748. Fig. 6 and Fig 7 show images from the test dataset which were collected from YouTube Videos, paired with their predictions.

B. Evaluation of Image Linked Questions Model

We ran our system against our dataset of videos. The results for the questions generated are shown in Table 2. The length of the video is denoted in Minutes: Seconds format. The number of questions generated for each video's transcripts, along with the number of questions that have images linked to them is displayed. Two metrics that have been calculated for each video are shown below:

\[
\text{Accuracy of Image Linking} = \frac{\text{number of questions that have relevant images}}{\text{number of usable questions that have images}}
\]

\[
\text{Percentage of Relevant Images Linked Questions} = \frac{\text{number of questions that have relevant images}}{\text{total number of usable questions}}
\]

TABLE 1. Dataset of Videos

| Video Number | Domain          | Topic                                | Length of the Video | Hosting Platform | Video Location                  |
|--------------|-----------------|--------------------------------------|---------------------|------------------|---------------------------------|
| 1            | Graph Theory    | Dijkstra's Shortest Path Algorithm, Graph Theory | 4:14                | YouTube          | https://www.youtube.com/watch?v=pSqmAO-m7Lk |
| 2            | Graph Theory    | Graph Theory Introduction            | 6:40                | YouTube          | https://www.youtube.com/watch?v=eQA-m22w1TQ |
| 3            | Machine Learning| Backpropagation Algorithm           | 16:14               | Coursera         | https://www.coursera.org/lecture/machine-learning/backpropagation-algorithm |
| 4            | Machine Learning| Normal Equation                     | 12:00               | Coursera         | https://www.coursera.org/lecture/machine-learning/normal-equation-2DKxQ |
C. Observations and Analysis

We observe that a significant number of questions for each video are unusable because they are semantically inaccurate or make little sense. This can be largely attributed to three factors: The first and the most important factor is due to imperfect Speech-To-Text Translations. It was observed that transcripts often recognized certain words incorrectly, and even a simple confusion of "node A" and "node a" can lead to the QG model creating highly incorrect questions. The second factor is the conversational language used by professors. It was observed that in the conversational tone, professors tend to speak rather long sentences which may not have perfect grammar. While students are able to keep up with professors, the model is unable to make sense of these tortuous sentences. These two factors are especially relevant for the two YouTube videos (Video 1, Video 2) whose transcripts were created using the Speech-To-Text API discussed in section 3.1. The third factor is related to questions generated that are semantically correct but do not test anything relevant. An example of this is shown (the example includes the sentence from the transcript from which the question was generated, the question generated, and the expected answer):

Sentence: So, how we might wish the activation of that note is slightly different.
Question: How do we wish the activation of a superscript I subscript j is?
Answer: slightly different

Sentence: It's not like, you know, we don't really want to try to change those values.
Question: What does it look like if we don't want to change values?
Answer: we don't really want to try to change those values

Of the usable questions, we observe that some videos (e.g., Video 3) do not require image-linked questions majorly (only 6.1% of usable questions needed an image), while 51.3% of the usable questions of the video 4 needed an image. On average, we observe that 27.2% of usable questions are aided by the presence of a relevant screenshot. Further, we observe that on average, 70% of the time if an image is linked to a question, it is relevant to the question in some way. In a few cases, the questions directly target something that is in the video frame. This is shown in figure 8. In other cases, the questions indirectly target a concept that is in the Video frame. This is illustrated in figure 9. A few questions served to improve the overall clarity of the question while also refreshing the student's concepts after the lecture was completed. This is illustrated in figure 10.

From the image classifier point of view, it was observed that 100% of the image-linked questions were accurate to the timestamp of the video from which the question was created. It is important to note, however, that not all videos may have frames that have equations or graphs that are similar to the ones that the model was trained on. One way to overcome this is to increase the training data, by collecting varied types of images that represent a particular class. Additionally, a video frame could include objects of both classes in the same frame that can confuse the model as well. In such cases, even though attaching the frame might be relevant, the question may get discarded due to inaccurate predictions of the classifier. An example of such a case is shown in Fig. 11, where the model’s output was 0.9834 (indicating that the image is of class “graph”), but the question contained the word “equation” because of which it was discarded.

![Fig. 8](image1)

**Fig. 8.** Question: How many layers does the neural network have on the right? Answer: four. (Video 4)

![Fig. 9](image2)

**Fig. 9.** Question: What is the next most promising node? Answer: node 4 (Video 2)

![Fig. 10](image3)

**Fig. 10.** Question: What type of graph has a unique edge between every pair of vertices? Answer: Complete Graph (Video 2)

### TABLE 2: Results for Image Linked Questions Module

| Dataset Video | Number of Questions Generated | Number of Usable Questions | Number of Image Linked Questions | Number of Relevant Image Linked Questions | Accuracy of Image Linked Questions | Percentage of Relevant Image Linked Questions |
|---------------|-------------------------------|-----------------------------|----------------------------------|------------------------------------------|-----------------------------------|-----------------------------------------------|
| 1             | 19                            | 9                           | 5                                | 3                                        | 60%                               | 33%                                           |
| 2             | 28                            | 15                          | 9                                | 5                                        | 55%                               | 33%                                           |
| 3             | 118                           | 49                          | 5                                | 3                                        | 60%                               | 6.1%                                          |
| 4             | 76                            | 37                          | 24                               | 19                                       | 79%                               | 51.3%                                         |
| Total         | 241                           | 110                         | 43                               | 30                                       | 70%                               | 27.2%                                         |

57
VI. CONCLUSION AND FUTURE WORK

The paper describes an approach to automatically generate questions from videos to evaluate the cognitive abilities of online learners. The main activities performed for this purpose include: (i) extracting subtitles from the videos, (ii) processing text to generate questions, (iii) linking questions to images, and (iv) selecting appropriate questions by ranking their relevance to the content in the videos. The approach presented in this paper has numerous advantages such as timely generation of questions, improved test-taking experience by linking questions to images, adaptability of approach to other domains or disciplines, and a comprehensive assessment of learners. A prototype of the approach has been implemented on a few video lectures from courses in the aforementioned domains.

For the future scope in terms of the classification model, the training dataset used for training the classification model needs to be enhanced in terms of inclusion of more images that represent the highly varied nature of the classes. The model presented can be trained to detect classes other than the ones mentioned in this paper. Apart from this, the entire system can be extended to other domains but will require careful curation of the training dataset and training the model in that particular domain. For domains containing many classes, an object detection model can be used to first localize the regions of interest, following which a classification model can be employed. In terms of question generation, improvements are needed largely in terms of transcript generation and transcript accuracy. This can be done by using better transcription APIs or lecturers themselves providing transcripts for any lectures that they may have created. The presence of accurate transcripts can greatly enhance the quality of questions that are generated.

REFERENCES

[1] Heilman M. Automatic factual question generation from text (Doctoral dissertation, Carnegie Mellon University).
[2] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. Attention is all you need. InAdvances in neural information processing systems 2017 (pp. 5998-6008).
[3] Dong L, Yang N, Wang W, Wei F, Liu X, Wang Y, Gao J, Zhou M, Hon HW. Unified language model pre-training for natural language understanding and generation. arXiv preprint arXiv:1905.03197. 2019 May 8.
[4] Zhou Q, Yang N, Wei F, Tan C, Bao H, Zhou M. Neural question generation from text: A preliminary study. InNational CCF Conference on Natural Language Processing and Chinese Computing 2017 Nov 8 (pp. 662-671). Springer, Cham.
[5] Zhao Y, Ni X, Ding Y, Ke Q. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. InProceedings of the 2018 Conference on Empirical Methods in Natural Language Processing 2018 (pp. 3901-3910).

[6] Zhang S, Qu L, You S, Yang Z, Zhang J. Automatic generation of grounded visual questions. arXiv preprint arXiv:1612.06530. 2016 Dec 20.
[7] Krishna A, Blouwimpick P, Ghosh K, Sahu A, Roy S. Automatic Generation and Insertion of Assessment Items in Online Video Courses. InProceedings of the 20th International Conference on Intelligent User Interfaces Companion 2015 Mar 29 (pp. 1-4).
[8] Huang YT, Tseng YM, Sun YS, Chen MC. TEDQuiz: automatic quiz generation for TED talks video clips to assess listening comprehension. In2014 IEEE 14Th international conference on advanced learning technologies 2014 Jul 7 (pp. 350-354). IEEE.
[9] Skalban Y, Specia L, Mitkov R. Automatic question generation in multimedia-based learning. InProceedings of COLING 2012: Posters 2012 Dec (pp. 1151-1160).
[10] Druzhkiv PN, Kustikova VD. A survey of deep learning methods and software tools for image classification and object detection. Pattern Recognition and Image Analysis. 2016 Jan;26(1):9-15.
[11] Albawi S, Mohammed TA, Al-Zawi S. Understanding of a convolutional neural network. In2017 International Conference on Engineering and Technology (ICET) 2017 Aug 21 (pp. 1-6). Ieee.
[12] Altenberger F, Lenz C. A non-technical survey on deep convolutional neural network architectures. arXiv preprint arXiv:1803.02129. 2018 Mar 6.
[13] LeCun Y, Bottou L, Bengio Y, Haftner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998 Nov;86(11):2278-324.
[14] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems. 2012;25:1097-105.
[15] Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. InEuropean conference on computer vision 2014 Sep 6 (pp. 818-833). Springer, Cham.
[16] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. 2014 Sep 4.
[17] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. InProceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 1-9).
[18] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 770-778).
[19] Szegedy C, Ioffe S, Vanhoucke V, Alemi AA. Inception-v4, inception-resnet and the impact of residual connections on learning. InThirty-first AAAI conference on artificial intelligence 2017 Feb 12.
[20] Zou Z, Shi Z, Guo Y, Ye J. Object detection in 20 years: A survey. arXiv preprint arXiv:1905.05055. 2019 May 13.
[21] Hearst MA. Text Tiling: Segmenting text into multi-paragraph subtopic passages. Computational linguistics. 1997;23(1):33-64.
[22] Wankhede HS, Kiwelekaw AR. Qualitative assessment of software engineering examination questions with bloom’s taxonomy. Indian Journal of Science and Technology. 2016 Feb;9(6):1-7.
[23] Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ. Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683. 2019 Oct 23.
[24] Transformer, E., 2020. Exploring Transfer Learning with T5: the Text-To-Text Transfer Transformer. Google AI Blog.: https://ai.googleblog.com/2020/02/exploring-transfer-learning-with.html (Accessed 16 August 2021).
[25] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. 2014 Dec 22.
[26] Ruby U, Yendapalli V. Binary cross entropy with deep learning technique for image classification. International Journal of Advanced Trends in Computer Science and Engineering. 2020 Jul;9(10).
[27] Li Y, McLean D, Bandar ZA, O'Shea JD, Crockett K. Sentence similarity based on semantic nets and corpus statistics. IEEE transactions on knowledge and data engineering. 2006 Jun 26;18(8):1138-50.