Context Aware Automatic Subjective and Objective Question Generation using Fast Text to Text Transfer Learning

Arpit Agrawal[1], Pragya Shukla[2]
Institute of Engineering and Technology, DAVV Indore, India[1, 2]

***Abstract*—Online learning has gained a tremendous popularity in the last decade due to the facility to learn anytime, anything, anywhere from the ocean of web resources available. Especially the lockdown all over the world due to the Covid-19 pandemic has brought an enormous attention towards the online learning for value addition and skills development not only for the school/college students, but also to the working professionals. This massive growth in online learning has made the task of assessment very tedious and demands training, experience and resources. Automatic Question generation (AQG) techniques have been introduced to resolve this problem by deriving a question bank from the text documents. However, the performance of conventional AQG techniques is subject to the availability of large labelled training dataset. The requirement of deep linguistic knowledge for the generation of heuristic and hand-crafted rules to transform declarative sentence into interrogative sentence makes the problem further complicated. This paper presents a transfer learning-based text to text transformation model to generate the subjective and objective questions automatically from the text document. The proposed AQG model utilizes the Text-to-Text-Transfer-Transformer (T5) which reframes natural language processing tasks into a unified text-to-text-format and augments it with word sense disambiguation (WSD), ConceptNet and domain adaptation framework to improve the meaningfulness of the questions. Fast T5 library with beam-search decoding algorithm has been used here to reduce the model size and increase the speed of the model through quantization of the whole model by Open Neural Network Exchange (ONNX) framework. The keywords extraction in the proposed framework is performed using the Multipartite graphs to enhance the context awareness. The qualitative and quantitative performance of the proposed AQG model is evaluated through a comprehensive experimental analysis over the publicly available Squad dataset.**

Keywords—Automatic question generation; Text-to-Text-transfer-transformer (T5); natural language processing; word sense disambiguation (WSD); domain adaptation; multipartite graphs; beam-search decoding

I. INTRODUCTION

Assessment has always been a very crucial tool in the educational ecosystem to identify the attainment of the learning outcomes and the curriculum gaps. Various evaluation techniques and assessment methods have been proposed by the researchers and academicians to exploit various aspects of students' learning. Question-answer technique has found to be the most effective way of evaluating the students' knowledge as it serves various purposes simultaneously like offering the opportunity to practice the retrieval of information from memory, identifying the misconceptions, reinforced learning through iterative core concepts, engagement in learning activities, etc. However, deriving these questions manually is a huge task as it requires the thorough knowledge, training and depth of understanding. The challenge increases many folds when the questions need to be replaced periodically to ensure the validity and value which reduces after few rounds of usage. The emergence of e-learning has added a new dimension over the last decade through the available massive open online courses (MOOCs) and adaptive learning. These tools require a large pool of questions to ensure their effectiveness [1-3].

Considering the amount of labor, time and cost associated with the process of generating questions manually, automatic question generation (AQG) has been introduced recently to automate the process. It produces the questions automatically through the structured or unstructured knowledge source and hence enables the educators to invest the time in some other instructional activities. A well-structured and customized question bank can also be created by controlling the difficulty and cognitive level to make the testing adaptive as per the students' requirements. The type of the questions may be subjective or objective. Subjective questions can be used to assess the depth and diversity of understanding of the students, while the binary mode of assessment in objective questions may be used to evaluate the logical understanding in a faster and reliable manner. The most popular varieties of objective questions include the multiple-choice questions (MCQs), fill in the blank questions, true or false questions or matching questions. Although the subjective question-based assessment has remained the most trusted tool in the traditional education systems so as to assess the writing skills and critical reasoning of the students, the recent growth in the e-learning paradigm has attracted many universities towards the objective questions. This is due to the reason that in computer-assisted examination, accurate assessment of numerous students through subjective questions will be a very tedious task. Automatic generation of semantically meaningful and well-formed questions (subjective and objective) has the potential to greatly enhance the learning and assessment experience [4-6].

Traditionally, the process of AQG is classified in two categories, rule-based approach and deep learning approach. Rule-based approach derives the questions on the basis of the hand-crafted rules which are derived through a high level of manual interference. However, the limitation of this approach is that the rules derived for one domain may not be suitable for other domains. On the other hand, deep leaning approach utilizes the natural language processing (NLP) techniques like text abstraction, text summarization, machine translation, etc to generate the questions automatically. Many question answer scenarios and various datasets have been presented over the last decade with different size of context, different formats of answers and sizes of training datasets [7,8]. However, the performance of the AQG model is greatly subject to the understanding and reasoning about the relationship between the sentences in the contextual data. The most important requirement from a machine learning model applied for NLP task is to process the text in a way which is acquiescent to downstream the learning through a general-purpose knowledge framework which understands the text.

Recently, transfer learning has emerged as a very powerful technique in NLP which utilizes the knowledge retrieved from one task to the other related task and therefore reduces the necessity of fine-tuning dataset and improves the performance. This feature of transfer learning technique is also known as domain adaptation which is achieved through the word vector mapping between the similar words and similar vectors. The general-purpose knowledge and abilities in transfer learning is achieved through the pre-training of the entire model over a data rich task. This knowledge is further been transferred to the downstream tasks afterwards. Generally, supervised learning on a large labeled dataset is used to pre-train the transfer learning model for computer vision applications and unsupervised learning on unlabeled data is utilized for the same in NLP applications. The enormous amount of text data available on internet may be used in transfer learning to train the network. Various transfer learning models have been proposed for the NLP applications like GPT, ELMo, BERT, XLNET, ALBERT, RoBERTa, etc. [9-12]. All of these models have shown promise for doing particular tasks, but when asked to fulfill a more comprehensive set of requirements, they fell short. The performance of these models differs from one task to the next due to the fact that even the procedures, practices, and processes that they use are not consistent. Therefore, in order to comprehend transfer learning more thoroughly, a methodology that is both consistent and methodical is essential.

Various unifying approaches like language modeling, span extraction, casting of all text as question answering, etc have been proposed by the researchers in last few years for NLP tasks. But, Text-to-Text-Transfer-Transformer (T5) has evolved as the most powerful unified framework which treats every text processing problem as a “text-to-text” problem [13]. It takes text as input, process or transforms it and generates text as output. The unification has made it possible to apply the same model, objectives, training and decoding procedure to any NLP task at hand like text abstraction, document summarization, question-answer generation, text classification, sentiment analysis from text, etc. The encoder-decoder model in T5 is pretrained on a multi-purpose blend of supervised and unsupervised tasks. The encoded input is fed through a cross-attention layer to generate the autoregressive decoded output using the scaler embedding. The training algorithm in T5 is teacher forcing and hence requires an input sequence and target sequence compulsorily.

However, due to the sequential nature of T5 model, the text-to-text transformation is naturally slow. The model speed even reduces for the larger T5 models and makes the implementation really difficult with limited resources. Augmenting the transformer model with WSD to generate the objective questions in terms of multiple choice, fill-in-the-blank, true/false or pair matching questions enhances the complexity further. This paper presents a fast automatic question generation model for objective and subjective questions by using the fastT5 model which is capable of inferencing faster than the conventional transformer for a reduced model size. The proposed model is run on the onnx runtime which quantizes the whole process and gives the model as output in a single line of code. The flexibility to customize the whole model as per the application has been achieved in this work through the PyTorch Lightning library. It is a lightweight, scalable and high-performance deep learning framework which provides a flexible interface for PyTorch which can easily work on distributed hardware while keeping the models hardware skeptical. The keywords extraction in the proposed framework is performed using the Multipartite graphs to enhance the context awareness. The speed of finding and replacing the keywords in objective questions generation is further improved by using the FlashText library.

The major contribution of the proposed work lies within the application of FastT5 transfer learning model for the subjective and objective question generation automatically. The augmentation of word sense disambiguation (WSD) and domain adaptation framework with the proposed model has improved the meaningfulness of the questions. As the model is required to understand and reason about the relationship between the sentences in the story, the context awareness in the proposed work is enhanced using the multipartite graph-based keyword extraction and FlashText library and the flexibility is achieved through the PyTorch Lightning library. The performance of the proposed technique is evaluated thoroughly through an experimental analysis over SQuAD dataset. Specifically, “teacher forcing” methodology has been used here to train the model with a maximum likelihood objective regardless of the task. Due to the quantized model over onnxruntime, the proposed system is lightweight and faster with good BLEU score.

Rest of the paper is organized as follows: Section II deals with the literature survey through the analysis of related work. The mathematical framework of the text-to-text transformer is given in Section III. The proposed automatic subjective and objective question generation using the proposed fastT5 model is discussed in Section IV. Effectiveness of the proposed strategy is illustrated through the experimental analysis in Section V while Section VI concludes the paper.

II. RELATED WORK

The potential of AQG to change the complete paradigm of online education system, information retrieval systems and interactive support systems has attracted a lot of researchers towards it. Rus et al. [14] has defined the problem of AQG as “the task of automatically generating questions from various inputs such as raw text, database, or semantic representation”. The increasing availability of the digital information on internet and the amazing advancement in the field of NLP has driven the pace of research in the area of AQG. Conventionally, the problem of AQG is classified in two categories rule-based approaches and neural network approaches (also called neural approaches). In rule-based approaches, human designed syntactic rules are used to transform the declarative sentences in text to the interrogative sentences. It typically applies some lexical transformation around the main verb in the sentence. Heilman and Smith [15] presented a trained logistic model to generate the questions from a paragraph by deriving the transformation rules to produce multiple declarative sentences. These sentences are further converted into questions by syntactic and lexical transformations. Dhole and Manning [16] proposed a collection of semantic and syntactic rules-based heuristics model and named it Syn-QG. They addressed the problem of rule-based approach of generating the questions on Blooms Taxonomy level 1 like “what” and “yes or no”. The proposed Syn-QG model utilizes the VerbNet to generate a set of semantically richer questions. However, this approach requires extensive knowledge in linguistics and well-designed transformation rules to convert the declarative sentences into questions.

Wang et al. [17] proposed a parser based AQG model for medical knowledge evaluation. The medical terms in the articles are extracted and the unstructured entries are classified into different fields using the parsers. They formed more than 100 templates to match the parsed data entries to the question template. However, this method required an extensive labor and the respective templates are domain specific and cannot be applied to other applications. Fabbri et al. [18] developed a context-answer pair based syntactic dataset and used it for the training of deep learning model. The accuracy and effectiveness of this model was dependent on the quality of the dataset and was not generalized for any application. The generated questions were also monotonous.

RNN Encoder-decoder based framework was proposed by Zhou et al. [19] for AQG using attention mechanism for generating natural language questions. The network in the encoder is bidirectional Gated Recurrent Unit (GRU) and left-to-right GRU in the decoder. A multi-perspective context matching algorithm has been proposed by Song et.al. [20] in the sequence-to-sequence LSTM encoder-decoder framework with the copy mechanism for the AQG. Two sets of hidden vectors were generated by encoding the context passage and answer pair through two bi-directional LSTMs. Yuan et al. [21] proposed a combination of supervised learning and reinforcement learning to train a model for AQG. The proposed framework has trained the model first with an objective function of minimizing the cross-entropy loss and then maximized the reward function using the policy gradient method.

Kim et al. [22] derived an answer masking based AQG framework to encode the answers and the sentences separately where the answer text is replaced special tokens. The stack of attention layers with a dot-product based alignment score is used as attention module which cooperates with the answer encoder. The output gate of the LSTM network utilized retrieval style word generator to predict the token. Recently the potential of transfer learning has been exploited by Mitkov et. al. [23] to generate the multi-choice questions answering. They have showed that the unsupervised transfer learning can be helpful in NLP based applications through iterative self-labeling technique. The transferability of knowledge in encoder and decoder has been explored by See et.al. [24] through a thorough experimental analysis with source question dataset and target dataset. They have used a sequence-to-sequence pointer-generator network over a smaller sized dataset and evaluated the performance over semi-supervised learning.

Most of the prior works in the field of AQG has assumed the access to the sufficiently large dataset for the training and validation. The time taken in the training of the existing models and transformers is also very large. The techniques proposed in the traditional techniques were mostly domain specific and were either suitable for subjective question generation (long answers and short answers) or for objective question generation. Motivated by these facts, our work aims to provide a general purpose, fast and more meaningful AQG model for subjective and objective questions generation using transfer learning.

III. T5 Transformer Model Description

Transfer learning has gained a lot of attention over the last years due to its potential to utilize the knowledge leaned from one task for the inferential study of another task and thereby reducing the necessity of fine tuning the dataset. It typically reuses the learned weights of a base network which is trained with a large dataset to the target network by changing the training objectives over a smaller dataset. Typically, the objective of transfer learning is to enhance the learning characteristics through the target conditional probability distribution $P(Y_T|X_T)$ in a target domain D_T utilizing the knowledge attained from the source domain $D_S$and Source task T_S. **T**ext-**t**o-**T**ext **T**ransfer **T**ransformer (T5) is an advance model based on transfer learning converts every problem of NLP like question answering, text classification, translation, question generation, etc. to a text-to-text problem [25]. The model takes text as input and gets trained with the input so as to generate some target text. It enables the model to use the same model, hyperparameters, loss function, activation function, etc. for any diverse set of NLP application. The typical structure of the transformer model is shown in Fig. 1 which is based on an encoder-decoder model. The input sequence of symbols represented by $X=(x_1,x_2,\dots, x_n)$ is mapped to a sequence of continuous representation $Z=(z_1,z_2,\dots, z_n)$ at the encoder. Taking the sequence z, decoder generates an output sequence of symbols $Y=(y_1,y_2,\dots,y_n)$ with one element at a time. The model works in autoregressive style to generate the next symbol by utilizing the previously generated symbols.

The encoder comprises of a sequentially connected stack of N identical encoder layers where the output of one layer is the input to the next layer. The input token sequences fed to the encoder first concerted into vectors of size d_{model} by the token embedding layer and the position embedding layer. Each block of encoder layer is further comprising of two components; a multi-head self-attention layer and a fully connected feed-forward layer. A simplified layer normalization is applied to the input of each sublayer where the activation function is rescaled without adding bias. Residual skip connection method is applied afterwards to map each sub-component's input to the output. Dropout is applied within the feed-forward network, on the skip connection, on the attention weights, and at the input and output of the entire stack. Self-attention mechanism plays a vital role in transformer model which relates different positions of a single sequence to compute the representation of the sequence. It typically utilizes the weighted average of the rest of the sequences to replace each element of a sequence. The process of multi-head self-attention mechanism can be represented as

$$MultiHeadAttn(\tilde{Q},\tilde{K},\tilde{V}) = [\phi_1;\phi_2;\phi_3;\ldots;\phi_h]W^O \qquad (1)$$

and

$$\phi_i = Attention(\tilde{Q}W_i^Q,\tilde{K}W_i^K,\tilde{V}W_i^V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \qquad (2)$$

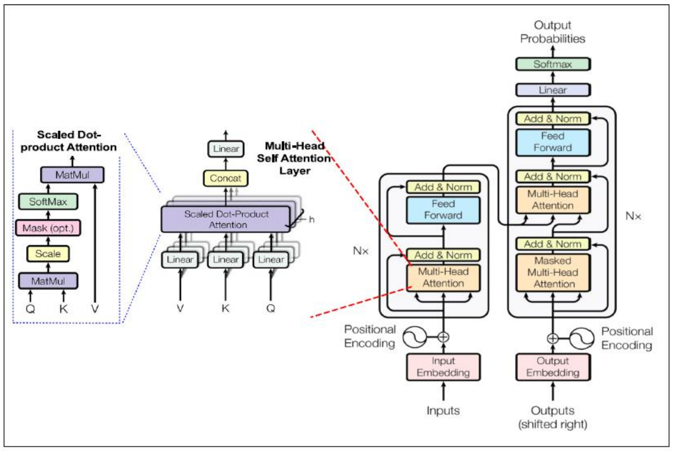


Fig. 1. Text-to-text-transformer model [25].

Similar to the encoder, the decoder structure also comprises of a stack of identical layers. However, each layer in decoder comprises of three sublayers. The third extra sub-layer in decoder is of the form of autoregressive or casual self-attention which performs the multi-head attention over the output of encoder stack and allows the model to attend to past outputs. The residual connections are employed around each sub-layer followed by layer normalization. The self-attention sub-layer is modified here to prevent the positions from attending to subsequent positions. The output of the final decoder block is given to the dense layer with 'softmax' output. The weights of this dense layer are further shared with the input embedding matrix. All attention mechanisms here are fragmented into independent "heads" whose outputs are concatenated before being further processed.

The computation complexity in case of convolutional neural network based like Extended Neural GPU, ByteNet and ConvS2S increases with the increase in the distance between the two arbitrary inputs or output positions [26-28]. It varies linearly for ConvS2S and logarithmically for ByteNet which makes it extremely difficult to establish the learning among the dependencies between distant positions. This issue has been resolved in T5 transformer architecture by limiting the computation to a constant number by establishing a tradeoff with reduced resolution. Considering the fact that self-attention is an order independent operation, an explicit position is provided in the transformer architecture. Relative position embedding is used here instead of fixed embedding to provide the learned embedding with respect to the offset between the "key" and "query". A simplified form of position embedding is used in T5 model where the attention weights is computed by adding a scaler "embedding" to the corresponding logit. The model efficiency is further enhanced by sharing the position embedding parameters across all the layers. The major difference in the structure of T5 model with the conventional transformer model is that the layer normalization in T5 is placed outside the residual path. It also utilizes different position embedding scheme and removes the Layer norm bias as compared to the traditional model without affecting the performance considerably due to the orthogonal changes in the structure.

IV. Proposed Methodology

The proposed framework is designed for the automatic subjective and objective questions using the fast T5 Model as shown in Fig. 2.

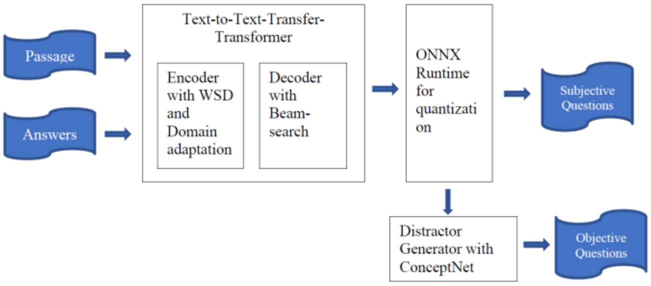

Passage
Answers
Text-to-Text-Transfer-Transformer
Encoder with WSD and Domain adaptation
Decoder with Beam-search
ONNX Runtime for quantization
Subjective Questions
Distractor Generator with ConceptNet
Objective Questions

Fig. 2. Proposed model for subjective and objective question generation.

The methodology used for both the cases is discussed here in two parts. Part 1 will discuss the AQG for Subjective question generation and part 2 will discuss the objective questions generation.

Part 1. AQG for Subjective questions: It comprises of four phases of processing namely problem formulation, data preprocessing, baseline model training and domain adaptation

A. Problem Formulation

Considering w_k as the word in the input sentence $S = \{w_k\}_{k=1}^{L^S}$ with length L^S, the problem is formulated as the generation of natural question Q with an objective of maximization of the conditional probability of the predicted question sequence Q given the input sentence S as

$$P(Q|S,\theta) = \prod_{t=1}^{L^Q} P\left(q_t|S,\left\{\{q_\tau\}_{\tau=1}^{t-1}\right\},\theta\right), \qquad (3)$$

where L^Q and q_t denote the length of the output question Q and words within Q respectively, θ represents the set of parameters of the prediction model. It is assumed here that the answer of the generated question is a consecutive segment in S so as to generate factual questions.

B. Data Preprocessing

The dataset comprises of three fields (C, Q, A) where C is the prefix which is a string indicating the task to perform, Q and A are the question (target text: The target sequence) and answer (input text: The input text sequence) from the context respectively. As mentioned earlier, our source data set, Squad [29], is a general-purpose QA dataset containing questions generated from Wikipedia pages that cover various topics. Preprocessing is performed to derive the pair of input sequence S and the corresponding question from the context. The larger paragraph is first of all transformed to the abstractive form through text abstraction process. The sentence which contains the answer A is extracted from the short paragraph C. Sentence-question (S,Q) pair is generated through this process which can be utilized for the training of the transformer model.

C. Baseline Model

The baseline model used in the proposed work is T5 model which converts every NLP problem into a text-to-text problem. The architecture of the model is discussed in earlier section. However, the sequential manner of text-to-text transformation is naturally slow and makes the implementation in real time challenging. The performance deteriorates further for larger models as the decoder in T5 model is utilized repeatedly for inference. To overcome this challenge, the conventional T5 Transformer model in this paper is augmented with fast T5 library which makes the inference of the conventional transformer faster. The high inference speed is achieved by running it on onnx runtime which quantize the whole model to decrease its size. FastT5 library converts the pretrained model to Open Neural Network Exchange (onnx) for quantization and generates a model as output which can run in a single line of code through a cross-platform inference framework, onnx runtime. It is a training machine-learning accelerator which offers compatibility among different hardware, drivers and operating systems to optimize the overall performance. The resultant quantized model is lightweight models which offer almost the same accuracy with low latency (due to graph optimization) as compared to the conventional transformer models.

The encoder and decoder are exported to the onnx model separately to reduce the computation complexity and the memory requirement considerably. It is also observed in the conventional transformer model that a constant number of inputs are given to the models, but first level of decoder does not take the previous key values. In contrast, the past key values are provided to the other layers of decoder. This uneven computation issue is addressed in fastT5 model by creating two different decoders; one for the first step without past key values and other for the other steps with past key values. The augmentation of word sense disambiguation (WSD) [30] and domain adaptation framework with the proposed model has improved the meaningfulness of the questions. As the model is required to understand and reason about the relationship between the sentences in the story, the context awareness in the proposed work is enhanced using the multipartite graph-based keyword extraction and FlashText library and the flexibility is achieved through the PyTorch Lightning library. Beam search algorithm has been used in this model in the decoder which tracks the *n* (number of beams) most likely hypotheses (based on word probability) at each timestep and finally chooses the hypothesis with the highest overall probability. Teacher forcing has been used in the model as the error propagation-based learning strategy during the training phase of the T5 model where the inference of a new token is depending upon the previous predicted tokens and current hidden state. This strategy modifies the training process by using the true tokens partially instead of always using the generated tokens.

D. Domain Adaptation

Domain adaptation is the process to learn a predictive function F_t which can be used to map the knowledge attained from the source domain D_S for task T_S to the target domain D_T for task T_T. The transfer of knowledge is done in such a way that domain distribution discrepancy between D_S and D_T is reduced to a minimum possible value. Supervised domain adaptation has been used in this work where the best model M_b is selected on the basis of best model evaluation parameters over the validation dataset in target domain for a given number of epochs. The best model is further fine-tuned for the target domain.

Part 2. AQG for Objective questions: Automatic generation of objective questions comprises of three phases of processing namely problem formulation, data preprocessing, and baseline model and distractor generation

E. Problem Formulation

Considering $C = \{w_t^c\}_{t=1}^{t=L_c}$ as the contextual passage, $Q = \{w_t^q\}_{t=1}^{t=L_q}$ as the question and $A = \{w_t^a\}_{t=1}^{t=L_a}$ as the correct answer with lengths $L_c, L_q \; and \; L_a$ respectively. The problem here is to design a transfer learning model M to generate a distractor $D = \{w_t^d\}_{t=1}^{t=L_d}$ about the question. The determination of best possible distractors $\bar{D}$ is done in such a way that the conditional likelihood is maximized given by $\bar{D} = \arg\max_{D} \log \Pr(\mathrm{D} | \mathrm{C}, Q, A)$

F. Data Preprocessing

The preprocessing stage for objective questions generation is similar to that discussed earlier for subjective questions. The dataset comprising of three fields (C, Q, A) but the difference is that the field A will be a word or a phrase. The length of A as compared to subjective question generation scenario is very small instead. The dataset will be transformed to Sentence-question (S,Q) pair which can be utilized for the training of the transformer model.

G. Baseline Model

The baseline model for objective question generation is FastT5 model where the encoder and decoder are exported to the onnx model separately to achieve the faster speed and low complexity. However, the model is augmented with distractor generating algorithm to generate other options which are similar to the correct answer but are wrong answers and are used to befuddle the examinee. The keywords extraction in the proposed framework is performed using the multipartite graphs to enhance the context awareness. An objective question, especially a multiple-choice question or fill in the blank question, comprises of three important parts; a question stem, a correct answer and distractors. Due to the limited scope of varying stem or answer, a right set of distractors can greatly control the level and relevance of the automatically generated questions. For MCQs, this task has been performed in this paper through the ConceptNet which is a freely available semantic framework designed to enable the computer understand the meaning of the commonly used words. Word embeddings has been created by ConceptNet [30] by representing the word meanings as vectors. These word embeddings are free, multilingual, aligned across languages, and designed to avoid representing harmful stereotypes. It is a graph knowledge base $G \subseteq C \times R \times C$ where C and R represent the natural language concepts and commonsense relations respectively. ConceptNet contains 32 million triplets where each triplet instance in the respective graph (c_1, r, c_2) represents a commonsense knowledge such as '(Commitment, Leadsto, Success)'. The words and relations for which the created MCQs are referred as seed words and seed relations, respectively. In the generated graph, the directed edge from node *c* to node *p* with relation *r*, *p* is referred as a parent of node *c* and *c* as a child of node *p* with relation *r*. The siblings of a node, with respect to a specific relation, are defined as all the children of its parent node, except for the node itself. The selection of distractors for objective questions is done on the basis of hypothesis that the distractors should not share any common property with correct answer. It has been ensured by finding non-overlapping graph communities within words. To attain these leading nodes of non-overlapping community, a one hop expansion is performed in that community and repeated words are removed. The community for each seed word along with its leading nodes is identified to derive the objective questions. Depending on the existence of path between the respective leading node and seed word, distractors from the same community can be chosen using the same seed relation. These generated distractors can be used to derive multiple choice questions, fill-in-the-blanks or True-false questions.

V. Experiment Results and Discussion

An experimental analysis has been performed to evaluate the performance of the proposed AQG model. Stanford Question Answering Dataset (SQUAD) has been used in this work which comprises of a rich set of 107785 questions collected from the crowd workers on 536 Wikipedia articles. It is generated through the question-answer pairs from the paragraphs of Wikipedia articles. The selection of Squad in this paper is to derive the questions from a larger spectrum of fields. The available SQUAD dataset consists of two sets: a training set and development set which has been divided further for training, testing and validation. The dataset is having three fields namely context, question and answer. The part of the dataset which is accessible has 490 articles which is divided randomly as given in Table I.

TABLE I. Statistics of Data Split

Data Split	Articles	Passage-answer pairs	Avg. passage tokens	Avg. question tokens
Train	442	75668	154.32	12.26
Validation	24	10566	159.61	12.58
Test	24	11877	133.64	12.55

The dataset has been used here for the subjective and objective questions generation. The data is first processed using the Wordpiece tokenizer which is a data-driven sub-word level method and consists of 30522 tokens trained from Google. Context selection has been implemented in this work by truncating the input tokens as per the need. The maximum limit of number of tokens in the tokenizer has been chosen 512 tokens as maximum limit so as to attain the optimal training speed and number of truncated data. The limit for number of question tokens has been set to 96 here. The context and answer are given as input to the proposed fast T5 model and question is the output of the transformer.

The proposed T5 model and its training procedures have been implemented using the PyTorch v1.5.0, Pytorch-Lightning frameworks along with the Transformersv3.0.2 from Hugging Face which is a python library providing various transformer architectures. Cross entropy loss and teacher training has been used here for the efficient training performance whereas GPT2 has been used in the decoder. The encoder and decoder of the transformer are exported to the onnx model separately to reduce the computation complexity and the memory requirement considerably. The reasoning and understanding between the sentences in the story to attain the context awareness is enhanced using the multipartite graph-based keyword extraction and FlashText library and the flexibility is achieved through the PyTorch Lightning library. The decoder has utilized the Beam search algorithm in this model so as to reduce the risk of hidden high probability word sequences by keeping the most likely number beams of hypotheses at each time step and help in choosing the hypothesis that has the overall highest probability.

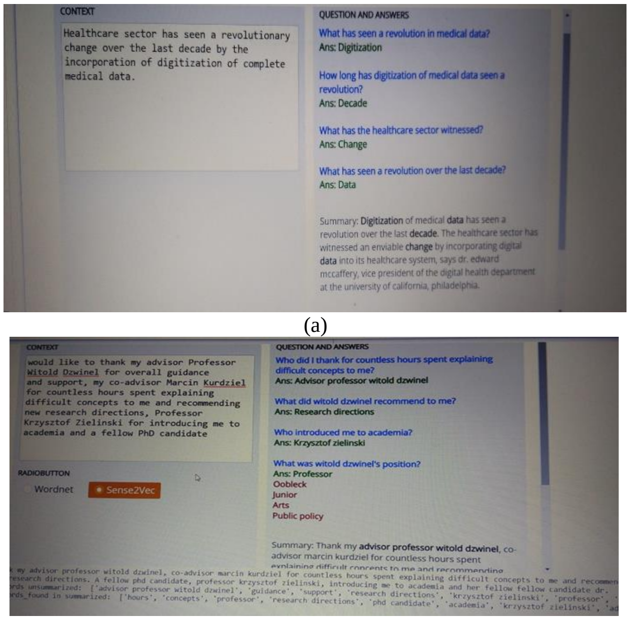
The model training and the evaluation of the proposed framework is performed with Google Cloud Compute Engine. One virtual machine with one 16 GB NVIDIA Tesla T4 GPU has also been used. The sum of token embeddings, segment embeddings and position embedding are set to 0.1. The dropout probability of the self-attention layer and all fully connected layers is also kept 0.1 similar to the original transformer.

The initial learning rate is set to $5 \times 10^{-5}$for the training of the FastT5 model with Adam optimizer. The batch size for the model is kept 64 and is trained only with 2 epochs with equal number of iterations due to the GPU memory constraints. The quality of question generation has been evaluated by BLEU-4 score during the training process which is an automatic performance measuring parameter. It resembles to the 4-gram precision of the hypothesis against the corresponding reference. The performance of the proposed fastT5 model based AQG has also been compared with some conventional models over BLEU-4 score with 2000 samples. The respective comparison has been shown in Table II.

TABLE II. COMPARATIVE ANALYSIS OF THE PROPOSED WORK

Model	BLEU-4
NQG[31]	12.28
M2S+cp [32]	13.98
Ass2s [33]	16.20
S2S-a-at-mp-gsa [34]	16.38
Proposed model	20.28

A graphical user interface (GUI) has been designed in this work using Gradio to present a user-friendly interface to utilize the model as shown in Fig. 3(a) and 3(b). It creates customizable UI components quickly around your TensorFlow or PyTorch models. The GUI shows four fields, namely, context, RadioButton(corpus type), question and answer, and summary field. The user needs to enter the context in the context field and select the suitable corpus; the generated questions and the summary will be automatically displayed in the respective fields. The sense2vec is a powerful variation of word2vec which improves the performance of algorithms like syntactic dependency parsing while significantly reducing computational overhead for calculating the representations of word senses. It enables the model to be implemented for the specific and typical context as well.



CONTEXT

Healthcare sector has seen a revolutionary change over the last decade by the incorporation of digitization of complete medical data.

QUESTION AND ANSWERS

What has seen a revolution in medical data?
Ans: Digitization

How long has digitization of medical data seen a revolution?
Ans: Decade

What has the healthcare sector witnessed?
Ans: Change

What has seen a revolution over the last decade?
Ans: Data

Summary: Digitization of medical data has seen a revolution over the last decade. The healthcare sector has witnessed an enviable change by incorporating digital data into its healthcare system, says dr. edward mccaffery, vice president of the digital health department at the university of california, philadelphia.

(a)

CONTEXT

would like to thank my advisor Professor Witold Dzwinel for overall guidance and support, my co-advisor Marcin Kurdziel for countless hours spent explaining difficult concepts to me and recommending new research directions, Professor Krzysztof Zielinski for introducing me to academia and a fellow PhD candidate

RADIOBUTTON

○ Wordnet ● Sense2Vec

QUESTION AND ANSWERS

Who did I thank for countless hours spent explaining difficult concepts to me?
Ans: Advisor professor witold dzwinel

What did witold dzwinel recommend to me?
Ans: Research directions

Who introduced me to academia?
Ans: Krzysztof zielinski

What was witold dzwinel's position?
Ans: Professor
Oobleck
Junior
Arts
Public policy

Summary: Thank my advisor professor witold dzwinel, co-advisor marcin kurdziel for countless hours spent explaining difficult concepts to me and recommending

```
k my advisor professor witold dzwinel, co-advisor marcin kurdziel for countless hours spent explaining difficult concepts to me and recommen
research directions. A fellow phd candidate, professor krzysztof zielinski, introducing me to academia and her fellow fellow candidate dr.
rds unsummarized:  ['advisor professor witold dzwinel', 'guidance', 'support', 'research directions', 'krzysztof zielinski', 'professor', '
rds_found in summarized:  ['hours', 'concepts', 'professor', 'research directions', 'phd candidate', 'academia', 'krzysztof zielinski', 'ad
```

(b)

Fig. 3. (a) Graphical user interface designed in gradio for AQG
(b) Graphical user interface designed in gradio for AQG.

VI. CONCLUSION

A challenging problem of automatic subjective and objective question generation with context awareness has been addressed in this work using unified text to text transfer learning. It presents a fastT5 based transformer model which reframes natural language processing tasks into a unified text-to-text-format and augments it with word sense disambiguation (WSD), concept net and domain adaptation framework to improve the meaningfulness of the questions. The proposed T5 model and its training procedures have been implemented using the PyTorch v1.5.0, Pytorch-Lightning frameworks along with the Transformersv3.0.2 from Hugging Face which is a python library providing various transformer architectures. Beam-search decoding algorithm has been used here to reduce the model size and increase the speed of the model through quantization of the whole model by Open Neural Network Exchange (onnx) framework. The keywords extraction in the proposed framework is performed using the Multipartite graphs to enhance the context awareness. It can be used to derive multiple choice questions, fill-in-the-blanks or True-false questions. The qualitative and quantitative performance of the proposed AQG model is evaluated through a comprehensive experimental analysis over the publicly available SQuAD dataset. The research can further be extended in future with the more advance text transformers. The computation complexity can also be addressed in the future research work.

REFERENCES

[1] Lane, H. C. and Vanlehn, K. "Teaching the tacit knowledge of programming to novices with natural language tutoring", Journal Computer Science Education, 15, pp. 183–201, 2005.

[2] Graesser, A. C., Rus, V., D'Mello, S. K., and Jackson, G. T., "AutoTutor: Learning through natural language dialogue that adapts to the cognitive and affective states of the learner", In D. H. Robinson & G. Schraw (Eds.), Recent innovations in educational technology that facilitate student learning, Information Age Publishing, pp. 95-125, 2008.

[3] Heath, T. and Bizer, C., "Linked Data: Evolving the Web into a Global Data Space", Morgan & Claypool Publishers, 2011

[4] Ali, H., Chali, Y. and Hasan, S. A., "Automation of question generation from sentences. In Boyer, K. E. and Piwek, P. (eds.), Proceedings of the 3rd Workshop on Question Generation, held at ITS 2010, pp.58-67, 2010.

[5] Chen, W., Aist, G., and Mostow, J., "Generating questions automatically from Informational text", In: Craig, S. D. & Dicheva, D. (eds.), Proceedings of the 2nd Workshop on Question Generation, held at AIED 2009, pp.17-24. 2009

[6] Jouault, C., and Seta, K., "Building a Semantic Open Learning Space with Adaptive Question Generation Support", In Proceedings of the 21st International Conference on Computers in Education, 2013.

[7] Amidei, J., Piwek, P., Willis, A., "Evaluation methodologies in automatic question generation", , In Proceedings of The 11th International Natural Language Generation Conference (pp. 307–317). Tilburg University: Association for Computational Linguistics, pp. 2013-2018, 2018.

[8] Chen, G., Yang, J., Hauff, C., Houben, G.-J. , "Learningq: A large-scale dataset for educational question generation", In Twelfth International AAAI Conference on Web and SocialMedia, pp. 481–490, 2018.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[10] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), Advances in Neural Information Processing Systems 32, pp. 13063–13075. 2019.

[11] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692, 2019.

[12] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite BERT for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942, 2019.

[13] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683, 2019.

[14] Rus, V., Cai, Z. & Graesser, A., "Question Generation: Example of A Multi-year Evaluation Campaign", In: Rus, V. and A. Graesser (eds.), Online Proceedings of 1st Question Generation Workshop, NSF, Arlington, VA. 2008.

[15] Michael Heilman and Noah A. Smith. "Good Question! Statistical Ranking for Question Generation". In: HLT-NAACL. 2010.

[16] Kaustubh D. Dhole and Christopher D. Manning. "Syn-QG: Syntactic and Shallow Semantic Rules for Question Generation". In: ArXiv abs/2004.08694 (2020).

[17] Weiming Wang, T. Hao, and W. Liu. "Automatic Question Generation for Learning Evaluation in Medicine". In: Advances in Web Based Learning – ICWL 2007 4823 (2007), pp. 242 –251.

[18] A. R. Fabbri et al. "Template-Based Question Generation from Retrieved Sentences for Improved Unsupervised Question Answering". In: ACL. 2020.

[19] Qingyu Zhou et al. "Neural Question Generation from Text: A Preliminary Study". In: NLPCC. 2017.

[20] Linfeng Song et al. "Leveraging Context Information for Natural Question Generation". In: NAACL-HLT. 2018.

[21] Xingdi Yuan et al. "Machine Comprehension by Text-to-Text Neural Question Generation". In: Rep4NLP@ACL. 2017.

[22] Yanghoon Kim et al. "Improving neural question generation using answer separation". In: Proceedings of the AAAI Conference on Artificial Intelligence 33 (2019), pp. 6602–6609.

[23] Mitkov, R., Ha, L.A., Varga, A., Rello, L.: Semantic similarity of distractors in multiplechoice tests: extrinsic evaluation. In: Proceedings of the EACL 2009 Workshop on GEometical Models of Natural Language Semantics, pp. 49–56 (2009).

[24] Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1073– 1083.

[25] Ashish Vaswani et al. "Attention is All you Need". In: Advances in Neural Information Processing Systems 30. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 5998–6008.

[26] Łukasz Kaiser and Samy Bengio. Can active memory replace attention? In Advances in Neural Information Processing Systems, (NIPS), 2016.

[27] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. Neural machine translation in linear time. arXiv preprint arXiv:1610.10099v2, 2017.

[28] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning. arXiv preprint arXiv:1705.03122v2, 2017.

[29] Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text, In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (pp. 2383–2392). Austin: Association for Computational Linguistics.

[30] Robyn Speer and Catherine Havasi. Representing general relational knowledge in ConceptNet 5. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), pp. 3679–3686, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA).

[31] Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. In NLPCC

[32] Linfeng Song, ZhiguoWang, Wael Hamza, Yue Zhang, and Daniel Gildea. 2018. Leveraging context information for natural question generation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 569–574.

[33] Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. 2018. Improving neural question generation using answer separation. In AAAI.

[34] Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018b. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3901–3910.