Automatic Follow-up Question Generation for Asynchronous Interviews

Pooja Rao S B and **Manish Agnihotri** and **Dinesh Babu Jayagopi**
International Institute of Information Technology Bangalore
Karnataka, India

Abstract

The user experience of an asynchronous video interview system is often deemed non-interactive and one-sided. Interview candidates anticipate them to be natural and coherent like a traditional face-to-face interview. One aspect of improving the interaction is by asking relevant follow-up questions based on the previously asked questions, and its answers. We propose a follow-up question generation model capable of generating relevant and diverse follow-up questions. We develop a 3D virtual interviewing system, *Maya*, equipped with follow-up question generator. Many existing asynchronous interviewing systems pose questions that are fixed and scripted. *Maya*, on the contrary, reacts with relevant follow-up questions, a relatively unexplored dimension in virtual interviewing systems. We leverage the implicit knowledge from deep pre-trained language models along with a small corpus of interview questions to generate rich and diverse follow-up questions in natural language. The generated questions achieve 77% relevance with human evaluation. We compare our follow-up question generation model with strong baselines of neural network and rule-based systems and show that it produces better quality questions.

1 Introduction

The conventional hiring process is laden with challenges like prolonged hiring, lack of interviewers, expensive labour, scheduling conflicts etc. Traditional face-to-face interviews lack the ability to scale. Recent advances in machine learning has enabled automation in the field of recruitment. Recruiters are heeding to innovative choices like Asynchronous Video Interviews (AVI). Asynchronous interviews have a time-lapse between the communicating parties. These are usually conducted via online video interviews using internet-enabled digital devices. The feasibility and ease of automatic assessment of the AVIs when compared to in-person interviews (Rasipuram et al., 2016) is persuading the wide spread use of the system.

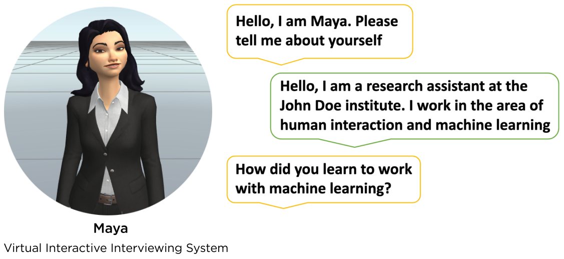

Hello, I am Maya. Please tell me about yourself
Hello, I am a research assistant at the John Doe institute. I work in the area of human interaction and machine learning
How did you learn to work with machine learning?
Maya
Virtual Interactive Interviewing System

Figure 1: *Maya* - Interactive Interviewing System

Limited prompting and follow-up, and no elaboration on questions is one of the components of structured interviews (Levashina et al., 2014). The current generation of asynchronous interview systems adopt structure and pose predefined questions selected from a relatively large set. However, with large scale adoption of these systems, it may eventually become repetitive and uninteresting for recruiters and candidates alike. The highly structured attribute of AVIs increases predictability, reduces variability, and makes them monotonous (Schmidt et al., 2016). Hence, it might be crucial to find the right balance between structure and probing. The adoption of planned or limited probing might help interviewers collect additional information related to the job, which may lead to increased interview validity (Levashina et al., 2014).

Levashina et al. (Levashina et al., 2014) define follow-up question as the one that is intended to augment an inadequate or incomplete response provided by the applicant, or to seek additional or clarifying information. Asynchronous communication does not enable coordinated turn-taking by interactants (Potosky, 2008). Integrating limited number of follow-up questions during the asynchronous interviews promises to solve the problem. A relevant follow-up question not only improves the interac-

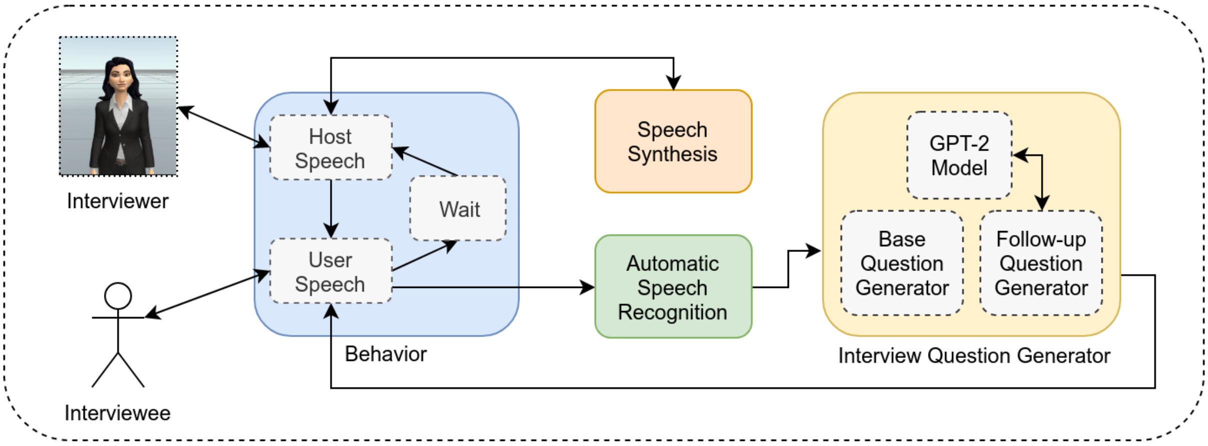


Figure 2: Framework of Interviewing System

tion between the interviewer and the interviewee but also makes it less predictable as the follow-up question is dynamic based on the interviewee's answer.

Based on these factors, we propose *Maya*, a 3D virtual interviewing system for behavioural domain. Specifically, the main contributions of this work are as follows. First, we present *Maya*, an interactive interviewing system equipped with a Follow-up Question Generation. We develop a framework for using large-scale transformer language model to generate relevant and diverse follow-up questions. Second, we perform experiments comparing Follow-up Question Generation (FQG) model with other strong question generation/selection models and show that the proposed model outperforms them by large margins with human evaluation. Finally, we perform experiments to study the robustness of the proposed model to errors in automatic speech recognition (ASR). The results indicate that *Maya* is able to produce high quality follow-up questions and hold an interactive interview with the candidate. We deploy a web-based minimalist virtual interview interface.[1]

2 Related Work

2.1 Natural Language Question Generation

Question Generation (QG), defined as the task to automatically generate questions from some form of text input (Rus and Graesser, 2009), has attracted attention since the First Question Generation Shared Task Evaluation Challenge (Rus et al., 2009). Recently, neural networks have enabled end-to-end training of question generation models influenced by the sequence-to-sequence (Seq2Seq) data-driven learning methods (Sutskever et al., 2014). Serban et al., (Serban et al., 2016) train a neural system to generate simple natural questions from structured triples - subject, relation, object. Du et al., (Du et al., 2017) use encoder-decoder model with attention to generate questions on the machine comprehension dataset SQuAD (Rajpurkar et al., 2016). QG-net (Wang et al., 2018b) is an RNN-based encoder-decoder model, trained on SQuAD, designed to generate questions from educational content.

Follow-up question generation in interviews is a new task and one study explores this (Su et al., 2018). Su et al., adopt a pattern-based Seq2Seq model on a small interview corpus in Chinese. They use a word clustering based method to build a word class table and transform all sentences in the corpus to patterns. Convolutional neural tensor network based (Qiu and Huang, 2015) sentence selection model is used on the answers to select a sentence to generate follow-up question patterns. These patterns are filled with words from the word class table to obtain potential follow-up questions. A statistical language model is used to choose a question by ranking. In contrast, we develop a follow-up question generation model utilizing knowledge from large-scale language model and a small corpus which does not involve pattern matching and template filling.

2.2 Language Model Pretraining

Pre-training on massive amounts of text in an unsupervised form has led to state-of-the-art advancements on diverse natural language processing tasks

[1] The demo of the system can be found at – https://www.youtube.com/watch?v=gdPxdi82nV0

System	Agent	Nonverbal Interaction	Verbal Interaction	Follow-up Q
Rao S B et al., 2017	Text Medium	No interaction	Fixed Script of Questions	No
SPECIES (Nunamaker et al., 2011)	Embodied Agent	Head Movement and Facial Expressions	Template based	Yes
MACH (Hoque et al., 2013)	Embodied Agent	Head Nodding and Smile Sharing	Fixed Script of Questions	No
TARDIS (Anderson et al., 2013)	Embodied Agent	Body Motions, Gestures and Facial Expressions	Fixed Script of Questions	No
ERICA (Kawahara, 2018)	Robotic Agent	Head Movement, Gestures and Eye Gaze	Template based	Yes
Maya (Ours)	Embodied Agent	Gestures, Facial Expressions and Follow-up Question	Dynamic Question Generation	Yes

Table 1: A comparison of asynchronous interview systems. The verbal interaction in *Maya* differs from other works with a follow-up question mechanism as it uses a question generation model rather than using template-based question selection method

(Devlin et al., 2018) (Radford et al., 2018). Currently, these pre-training steps are all variants of language modelling objectives. Howard and Ruder (Howard and Ruder, 2018) train a language model on huge amounts of Wikipedia data and fine-tune this on a target task with a smaller amount of labelled in-domain data. Several works follow this approach of fine-tuning and achieve impressive results. ELMo (Peters et al., 2018) is a bidirectional language model predicting the next and the previous tokens using bi-LSTM networks (Huang et al., 2015). OpenAI's GPT (Radford et al., 2018) train a unidirectional language model on massive text data. BERT (Devlin et al., 2018) is a masked language model trained with an additional objective of next sentence prediction. These models have attained state-of-the-art results on many downstream NLP tasks including the GLUE benchmark (Wang et al., 2018a). Pre-training with GPT model has also been used in generative tasks such as end-to-end dialog systems (Wolf et al., 2019) and automatic knowledge base completion (Bosselut et al., 2019) obtaining remarkable improvements over the models trained only with the in-domain data. Both the works use the transformer language model GPT for initialization. Our work builds on this to develop a Follow-up Question Generation model.

2.3 Agent-based Interviewing Systems

The use of intelligent virtual agents in dialogue systems has notably increased (Swartout et al., 2013) as it allows for a more interactive and immersive experience than traditional voice and text-based systems (López-Cózar et al., 2014). One primary application of virtual agents are in the Asynchronous Video Interviews (AVIs). A job interview is aimed to analyze the hiring feasibility of an interviewee, while a training interview gives accurate feedback about their performance.

While the initial works in AVIs were restricted to the skill assessment (Nguyen et al.), (Rao S B et al., 2017), improving the interview experience has gained momentum. One standard approach is the usage of virtual agents as interviewers instead of textual prompts to conduct interviews (Nunamaker et al., 2011). This approach makes the interview experience more interactive.

SPECIES (Nunamaker et al., 2011) introduced the usage of Embodied Conversational Agents in automated interviews. One of the goals was to study the difference in perceptions with varying attributes of agent. MACH (Hoque et al., 2013) and TARDIS (Anderson et al., 2013) are coaching-based conversational agents. Both of them focus on skill assessment and non-verbal behavior analysis to improve the feedback to interviewees significantly, but the questions are taken from a fixed pool of questions and do not take into account the interviewee's response. ERICA (Kawahara, 2018), consists of a robotic agent who has the capabilities of human-like eye gaze, head movement and gestures, and a statement-response system which is response retrieval method based on pattern and focus token matching. Although the behavior synthesis is a notable improvement, it still lacks robustness in dialogue generation.

While a lot has been done in automatic analysis of interviewee's response (Hemamou et al., 2019) to improve the quality of the interview, not much has been done to make the interview more verbally interactive. All the previous works have either used a fixed script of questions or used a pattern matching based question selection. We aim to improve the question generation system to make it more personal and response-based by generating relevant and grammatically correct follow-up questions.

3 Follow-up Question Generation - FQG

Follow-up Question Generation model is an adaptation framework for generating follow-up questions using language models by training it on an in-domain corpus of question, response and follow-up triplets. These data samples help FQG to learn the question structure and the relation between the triplets, and the knowledge from the language model pre-training produces novel questions.

3.1 Task

The training samples of $\{q, r, f\}$ in natural language, where q is the interviewer question, r is the candidate response and f is the follow-up question, are assumed to be given to the model. The task is to generate f given q and r as inputs.

3.2 Transformer Language Model

In this work, we use the transformer language model architecture, Generative Pre-trained Transformer (GPT-2) introduced in Radford et al. (Radford et al., 2019). This is very similar to the decoder part of the original transformer encoder-decoder model of Vaswani et al. (Vaswani et al., 2017). It uses multiple transformer layers each containing two sub-layers. First is the multi-headed self-attention mechanism over the input context tokens followed by position-wise feed-forward layers to produce an output distribution over target tokens. Our model is based on the recently published PyTorch adaptation of GPT-2.[2]

We initialize the Follow-up Question Generation model with 12-layer decoder-only transformer with 12 self-attention heads containing 768 dimensional states. The parameters are initialized to the smallest version of the GPT-2 model weights open-sourced by Radford et al. 2019 (Radford et al., 2019). The GPT-2 model is pre-trained on the WebText dataset which contains the text of 45 million links from internet (Radford et al., 2019).

3.3 Dataset

In order to train the FQG model, we need the training samples – $\{q, r, f\}$ triplets. We utilize the asynchronous interview dataset from Rao S. B et al. (Rao S B et al., 2017). This dataset consists of behavioural interviews of university students through asynchronous medium of video and written, referred to as the Asynchronous Video Interview dataset - AVI dataset and Asynchronous Written Interview dataset - AWI dataset respectively. We conduct a restricted crowd-sourcing to obtain follow-up questions using interview snippets from AWI dataset. We instruct the volunteers to write a follow-up question based on the presented snippet of interviewer question and the candidate response. Thus, we obtain a follow-up question dataset with more than 1000 samples, each sample containing the triplet of a question, response and a follow-up. The dataset can be found at https://ms-by-research-thesis.s3.amazonaws.com/followMLdata.xlsx

3.4 Fine-tuning

We fine-tune the GPT-2 language model using the dataset described above. 80% of the data is used for training and the rest is used for validation. The input to the model constitutes of tokens from each of the $\{q, r, f\}$ concatenated in a sequence. A set of input embeddings is constructed for this sequence. The word and position embeddings are learnt in the pre-training phase. We use an additional set of embeddings, speaker embeddings to indicate whether the token belongs to question, response or the follow-up. These embeddings are learnt during the fine-tuning phase. The input to the model is the sum of all three types – word, position and speaker embeddings for each token. Figure 3 illustrates how the tokens in $\{q, r, f\}$ are organised to form the speaker embeddings.

Following (Wolf et al., 2019), (Devlin et al., 2018), the fine-tuning is done by optimizing two loss functions – a language modelling loss, and a next-question classification loss. The language modelling loss is the commonly used cross-entropy loss. The last hidden state of the self-attention model is fed into a softmax layer over all the tokens in the vocabulary to obtain next token probabilities. These probabilities are then scored using the cross-entropy loss where the human written follow-up question tokens are used as labels.

A next-question classifier is trained to recognize the correct next question among the distractors of random questions. We append the dataset consisting of correct follow-up questions with randomly sampled questions from a pool of 200 (same as the ones used in Section 5), acting as distractors. This trains the model to learn a sense of sentence ordering. The classifier is a linear layer apply-

[2] https://github.com/huggingface/transformers

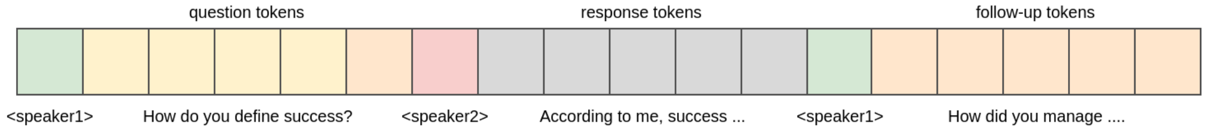


Figure 3: Input representation for training Follow-up Question Generation model

ing a linear transformation to the last hidden state of self-attention model to compute a value. Using the computed values, a softmax layer obtains the classification probabilities. Then we apply a cross-entropy loss to correctly classify the correct follow-up question. We use $n = 2$ as the number of choices for classification making it a binary classification task. The parameters of the transformer language model and the next-question classifier layer are fine-tuned jointly to maximize the log-probability of the correct label.

3.4.1 Decoding details

We use the top-k random sampling strategy for decoding (Fan et al., 2018). At each timestep, the probability of each word in the vocabulary being the next likely word is given. The decoder randomly samples a word from the *k* most likely candidates. Here *k* is a hyperparameter determined to be *k*=10 experimentally.

3.5 Results

We report the results of the follow-up question generation model in terms of perplexity (Bengio et al., 2003). We also report the classification accuracy of next-question classification task. Perplexity is usually used to measure the quality of language models. It indicates how well the model predicts the next word correctly. Our model obtains an average validation perplexity of 20.6 and average validation accuracy of 63.1%. These values can be deemed reasonable considering the small size of the in-domain dataset used for fine-tuning. It may also be due to the fact that the questions generated are novel and relevant leveraging the knowledge from the pre-training step which may not be present in the human written follow-up questions.

4 Experiments

In this section we showcase the efficiency of the FQG model through quantitative and qualitative analysis. First, we compare FQG with strong baselines. Second, we quantitatively confirm the relevance of the follow-up questions through human evaluation. Next, we investigate the robustness of the FQG model to errors in speech. Finally, we qualitatively examine the results of the FQG model.

4.1 Experimental Setup

We compare the FQG with two strong baselines. One is a rule-based system based on similarity measure and other is the reader-generator based QG-Net model (Wang et al., 2018b).

4.1.1 Similarity-based Question Selector

This model is a rule-based pre-defined question selector which selects questions from a pool of 200 behavioural questions (same as the ones used in Section 5) based on cosine similarity measure. We calculate the cosine similarity metric between the original interview question and each of the questions from the pool. We consider the top-10 most similar questions and randomly select one to be the follow-up question. This question selector loosely mimics the different rule-based question selectors in the existing systems.

4.1.2 QG-Net

QG-net is a Seq2Seq model with a context reader and question generator. The context reader is a bi-LSTM network which processes each word in the input context and turns it into a fix-sized representation. The question generator is a uni-directional LSTM which generates the question word-by-word incorporating pointer network (See et al., 2017) on the generator vocabulary. This model design enables the generator to output questions that focus on specific parts of input text. The *focus tokens* are encoded with each input word as an additional feature using one-hot encoding indicating whether the word is a focus token. QG-Net is trained on SQuAD dataset consisting of context, question and span of answer tokens within the context. QG-Net uses these answer tokens as focus tokens. Linguistic features like the POS tags, named entity and word case are also used as additional features. We refer the readers to the original paper for a detailed overview (Wang et al., 2018b). QG-Net effectively adapts a general purpose question generation model trained on SQuAD to generate questions from educational content, addressing the problem of insufficient training data. Hence we choose this as our neural network baseline model. In our case the candidate response is the context and the follow-up question is the question to be generated.

Since QG-Net model expects a sentence with its focus tokens as input, the interview question-answer pairs have to undergo preparatory techniques like finding focus of the answer and extractive summarization before feeding into the QG-Net model. We use the QG-Net model trained on SQuAD dataset released by (Wang et al., 2018b).

Finding Focus of the Answer QG-net uses a binary valued indicator as an added feature to indicate whether a word in context is important to generate a question, regarded as *focus tokens*. We employ a simple technique similar to Hu et al., (Hu et al., 2018) to automatically find these tokens. There exist overlapping tokens in the question (Q) and answer (A) pairs, seen as topics shared between them, that can be considered as focus tokens.

After removal of the stop words, A and Q are represented as a sequence of tokens *[*$a_1, .., a_n$*]* and *[*$q_1, .., q_m$*]* respectively. We consider all the tokens in A as candidates for focus tokens and all the tokens in Q as voters polling for the candidates. GloVe (Pennington et al., 2014) vectors are used to represent tokens from Q and A. The i^{th} answer token a_i gets a cumulative score S_i from all the tokens in the question calculated as

$$S_i = \sum_{j=1}^{m} p_{ij}.sim(a_i, q_j)$$

$$p_{ij} = \begin{cases} 1, & sim(a_i, q_j) > \lambda \\ 0, & \text{otherwise} \end{cases}$$

where $sim(a_i, q_j)$ is the cosine similarity between a_i and q_i. If the averaged S_i is above a certain threshold, a_i is included in the *focus*. This process is repeated for every answer token.

Extractive Summarisation The input to the QG model should be a representative of the response and give information for a potential follow-up. We employ a simple extractive summarization technique on the sentences of the answer. We use the method described above to find the focus of each sentence. We then compare the focus of each sentence with the focus of other sentences using the cosine similarity measure. R and S are two sentences from the candidate response with their focus tokens represented as *[*$fr_1, ..., fr_p$*]* and *[*$fs_1, ..., fs_q$*]* respectively. The cumulative score for each focus token of R is calculated as

$$W_i = \sum_{j=1}^{q} p_{ij}.sim(fr_i, fs_j) \quad N = \sum_{i=1}^{p} W_i$$

where p_{ij} is the indicative variable same as described above. If N crosses a certain percentage of the mean length of two sentences R and S, they are considered to be similar.

Once we have the pair(s) of similar sentences, we choose the one with more information content (more number of focus tokens) as the summary sentence. If more than one pair of sentences are similar to each other, *S* (pre-determined) number of sentences with the highest frequency of similar sentences is considered. The summary sentence along with the focus words is fed to the trained QG-Net model to generate questions.

4.2 Human Evaluation

To evaluate the quality of the generated follow-up questions and compare it against the baselines, we get human annotations. Human annotators involved in this study are non-native English speakers and graduate students with a background in Computer Science and Digital Society. We randomly sample 100 unseen question-answer pairs from the AWI dataset and generate one follow-up question (FQ) per QA pair from all three models– Similarity-based Question Selector, QG-Net question generation and GPT-2 based Follow-up Question Generation. We present the QA pair along with the follow-up questions generated by each model to the human annotators. They are asked to rank the questions based on their preference in the order of two metrics– relevance of FQ to the given interview QA pair and their grammar.

We consider the statistical mode of the ranking from three annotators for each follow-up question. When the mode is not unique i.e, when all three annotators choose a different rank, we resolve the disagreement by getting an extra set of rankings from an experienced interviewer. This is the case for about 10% of the annotations.

The results are shown in Figure 4. The bar plot indicates the count of mode of the ranks from evaluators for each of the model. FQG model significantly outperforms (well beyond p=0.01 level) the other two models with 54% of questions securing Rank 1, followed by 34% from QG-Net. 50% of the questions from SQS secure Rank 2. It can be observed that grammatically correct selected questions from SQS are preferred second after FQG

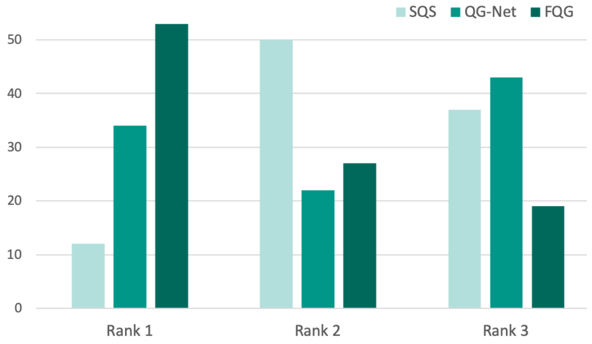


Figure 4: Human ranking of preferred follow-up questions from FQG comparing with two other baseline models based on relevance and grammar. The bar indicates the frequency of rankings, indicating that the FQG model is the most preferred for highest ranking.

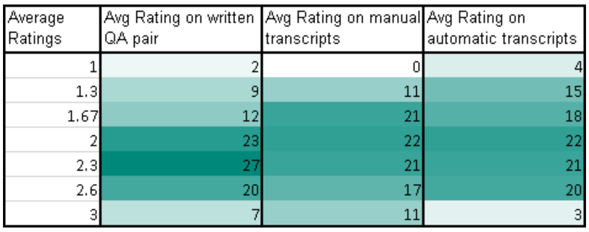
model than the gramatically incorrect and somewhat relevant questions from QG-Net model. We conclude that FQG model generates relevant and grammatically correct follow-up questions more often than the existing baselines.

We further strengthen the evaluation of FQG model by obtaining individual human ratings for the follow-up questions. Three human annotators evaluate the quality of the questions on a scale of 1-3, 1 being the lowest. The annotators are instructed to rate the questions based on grammar and relevance of the question to the original interview question and answer. We consider the average ratings from three annotators for evaluation. Figure 5 gives the statistics of the average ratings for the follow-up questions generated. 77% of the questions are scored ≥ 2. And 27% are rated ≥ 2.5. This shows that the FQG model generates superior quality follow-up questions and are scored well by humans.

4.3 Robustness to Errors in Speech

Investigating the robustness of Follow-up Question Generator has an important motivation. The model is trained on human-written triplets of $\{q, r, f\}$ whereas it will be inferred on the candidates's response obtained from ASR transcript in the virtual interviewing system. Hence, analyzing how follow-up question generation varies for ASR transcripts when compared with human transcripts helps to investigate the robustness of FQG model.

We use the asynchronous interface-based video interview dataset from Rasipuram et al. (Rasipuram et al., 2016) for this purpose as they have manual transcriptions of the interviews. We randomly select 103 question answer pairs. We also obtain automatic transcriptions for the same pairs of 103 question answers using the Web Speech API (Shires, 2019). We generate a follow-up question for each of this pair. This gives us 206 triplets of question, response, and follow-up questions, 103 each for manual and automatic transcripts.



Average Ratings	Avg Rating on written QA pair	Avg Rating on manual transcripts	Avg Rating on automatic transcripts
1	2	0	4
1.3	9	11	15
1.67	12	21	18
2	23	22	22
2.3	27	21	21
2.6	20	17	20
3	7	11	3

Figure 5: Frequency distribution of average human ratings on the quality of generated follow-up questions from the FQG on a scale of 1-3 on the different types of question-answer pair inputs (hand-typed text, manually and automatically transcribed spoken text).

Three human annotators evaluate the quality of the question on a scale of 1-3, 1 being the lowest. The annotators are displayed with the questions and answers from the manual transcripts and the follow-up questions generated on both manual and automatic transcripts to rate. We consider the average rating of the three annotators for evaluation.

Figure 5 shows count of the average ratings for follow-up questions on manual and automatic transcripts. 69% of the questions generated on manual transcripts and 64% of the questions generated on ASR transcripts get a score of ≥ 2. This implies that the FQG model generates relatively good quality follow-up questions on both manual and automatic transcripts.

4.4 Qualitative Analysis

Table 2 compares the follow-up questions generated by the follow-up question generation model with the two strong baselines, similarity-based question selector and QG-Net. It is seen that question generated by FQG is a grammatically correct and relevant follow-up in all cases. In regard with grammar, we can see that the QG-Net model struggles the most. FQG model, despite the grammatical errors in the answer, always adheres to grammar. Grammatical adherence is irrelevant with SQS model as it selects from a pool of grammatically correct questions.

In terms of relevance, we find that the FQG model always generates a follow-up relevant to the question or answer or both in all cases. SQS often produces follow-ups remotely related or unre-

Question: What is the biggest challenge you have faced so far? **Answer**: The biggest challenge that i have faced so far is IIT- JEE examination after completing my class 12. I prepared for it for two years. Unfortunately I didn't cracked it. **SQS**: Where do you see yourself 3 years from now? **QG-Net**: did you think was cracked ? **FQG**: What are the best lessons you have learned from this experience?
Question: Short self-presentation: Introduce yourself **Answer**: Hello, I am X, a passionate software engineer. Till my 12th standard I studied at my native place itself. My strongest plus point according to me is that I dont trust anyone easily and negative point is that I cant resist anything wrong happening to anyone around me.... **SQS**: What has been the greatest disappointment in your life? **QG-Net**: what is you that you can't resist happening? **FQG**: How do you manage your negative point?
Question: What has been the greatest disappointment in your life? **Answer**: I didn't get any high disappoint from my life. I was disappoint from myself when I didn't performed that well in GATE exam and not able to get admission in X, but I got admission in Y and now I feel that it was okay and not feel disappoint. **SQS**: What changes would you make if you came on board? **QG-Net**: what did you do in gate exam ? **FQG**: How would you explain your failure in GATE exam?
Question: How would you be an asset to this organisation? **Answer**: I think I have a hunger to learn and solve a problem. For me, each problem is a puzzle which I need to solve and I will try till I finish it. If I don't know something, I will learn it and get the job done. My hardworking and passionate qualities can be asset to your company. **SQS**: How quickly do you adapt to new technology? **QG-Net**: what do you have a hunger to do ? **FQG**: How do you think you can improve your skills?

Table 2: Comparison of examples generated from three different models for various question-answer pairs. SQS: Similarity-based Question Selector, QG-Net: QG-Net, FQG: Follow-up Question Generation

lated to the answers. Follow-ups by QG-Net model does not always comply to relevance. It generates unrelated or incomplete or grammatically incorrect relevant questions. We can also note that QG-Net model often generates questions which are already answered (eg. Questions 2, 4). We also present some examples generated only by the FQG model on randomly sampled QA pairs in Appendix A.

5 Maya - Interactive Interviewing System

Our interactive interviewing system, *Maya*, consists of two main components – 3D Virtual Interviewer and Interview Question Generator. The first is an Amazon Sumerian (Walker, 2017) based 3D virtual interviewing agent which asks questions and collects the interviewee's responses. We use ASR (Web Speech API (Shires, 2019)) to transcribe the user speech and this text data is fed to the second component, question generator, hosted on a server. Using Amazon Polly text-to-speech toolkit, the virtual agent communicates the generated question to the interviewee. The Interview Question Generator component contains two modules which communicates with the 3D virtual interviewer. *Base question selector* selects a question randomly from 200 questions commonly asked in an HR interview. Next question is a follow-up question generated by the *follow-up question generator*. In our experiments, we limit the number of follow-up question to one. The follow-up question is based on single previous response from the candidate and not the history. We consider one follow-up question as a proxy to planned or controlled probing.

6 Conclusion

We introduce *Maya*, a virtual agent-based interviewing system equipped with verbal interactivity from follow-up question generation. We leverage the implicit knowledge of a large scale transformer language model fine-tuned on follow-up questions dataset to generate relevant, novel and diverse questions based on the candidates' response in an interview. With availability of limited data, this approach scales as it uses external knowledge from a language model trained on a huge corpus. With human evaluation, we show that the questions generated are of good quality. We can also see that the FQG model is often robust to the errors of speech recognition. We restrict the generation of follow-up questions to one as existing research suggests the advantage of limited probing and follow-up. But the model is capable of generating multiple follow-up questions based on the previous response. The FQG model is not limited to behavioural domain but can also be trained on any other domain descriptive questions to generate follow-up questions.

One important future direction of this work can be modelling the problem as generation of follow-up question considering the complete history of the conversation and not just the previous question and response. A user study could be organised to validate the advantages of including the follow-up questions to boost the interaction.

Acknowledgments

This work was partially funded by SERB Young Scientist grant (Grant no: YSS2015001074) of Dr. Jayagopi, Karnataka government's MINRO grant and a grant from Accenture Technology Labs.

We thank all the participants who contributed for data collection. We would also like to thank all the reviewers for their insightful comments and suggestions.

References

Keith Anderson, Elisabeth André, Tobias Baur, Sara Bernardini, Mathieu Chollet, Evi Chryssafidou, Ionut Damian, Cathy Ennis, Arjan Egges, Patrick Gebhard, et al. 2013. The tardis framework: intelligent virtual agents for social coaching in job interviews. In *International Conference on Advances in Computer Entertainment Technology*, pages 476–491. Springer.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Çelikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. *ArXiv*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR*.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *ACL*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *ACL*.

Léo Hemamou, Ghazi Felhi, Vincent Vandenbussche, Jean-Claude Martin, and Chloé Clavel. 2019. Hirenet: a hierarchical attention model for the automatic analysis of asynchronous video job interviews.

Mohammed Ehsan Hoque, Matthieu Courgeon, Jean-Claude Martin, Bilge Mutlu, and Rosalind W Picard. 2013. Mach: My automated conversation coach. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 697–706. ACM.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *ACL*.

Wenpeng Hu, Bing Liu, Jinwen Ma, Dongyan Zhao, and Rui Yan. 2018. Aspect-based question generation. In *ICLR Workshop*.

Zhiheng Huang, Wei Liang Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *ArXiv*.

Tatsuya Kawahara. 2018. Spoken dialogue system for a human-like conversational robot erica. In *International Workshop Spoken Dialogue Systems*.

Julia Levashina, Christopher J Hartwell, Frederick P Morgeson, and Michael A Campion. 2014. The structured employment interview: Narrative and quantitative review of the research literature. *Personnel Psychology*, 67(1):241–293.

Ramón López-Cózar, Zoraida Callejas, David Griol, and José F Quesada. 2014. Review of spoken dialogue systems. *Loquens*, 1(2):012.

Laurent Son Nguyen, Denise Frauendorfer, Marianne Schmid Mast, and Daniel Gatica-Perez. Hire me: Computational inference of hirability in employment interviews based on nonverbal behavior. *IEEE transactions on multimedia*.

Jay F Nunamaker, Douglas C Derrick, Aaron C Elkins, Judee K Burgoon, and Mark W Patton. 2011. Embodied conversational agent-based kiosk for automated interviewing. *Journal of Management Information Systems*, 28(1):17–48.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke S. Zettlemoyer. 2018. Deep contextualized word representations. *ArXiv*.

Denise Potosky. 2008. A conceptual framework for the role of the administration medium in the personnel assessment process. *Academy of Management Review*, 33(3):629–648.

Xipeng Qiu and Xuanjing Huang. 2015. Convolutional neural tensor network architecture for community-based question answering. In *IJCAI*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy S. Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*.

Pooja Rao S B, Sowmya Rasipuram, Rahul Das, and Dinesh Babu Jayagopi. 2017. Automatic assessment of communication skill in non-conventional interview settings: a comparative study. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 221–229. ACM.

Sowmya Rasipuram, Pooja Rao S. B., and Dinesh Babu Jayagopi. 2016. Asynchronous video interviews vs. face-to-face interviews for communication skill measurement: A systematic study. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI '16, pages 370–377, New York, NY, USA. ACM.

Vasile Rus, Wyse Brendan, Paul Piwek, Mihai Lintean, Svetlana Stoyanchev, and Cristian Moldovan. 2009. The question generation shared task and evaluation challenge. In *The University of Memphis. National Science Foundation*.

Vasile Rus and Arthur C. Graesser. 2009. The question generation shared task and evaluation challenge. In *The University of Memphis. National Science Foundation*.

Frank L Schmidt, IS Oh, and JA Shaffer. 2016. The validity and utility of selection methods in personnel psychology: Practical and theoretical implications of 100 years... *Fox School of Business Research Paper*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *ACL*.

Iulian Serban, Alberto García-Durán, Çaglar Gülçehre, Sungjin Ahn, A. P. Sarath Chandar, Aaron C. Courville, and Yoshua Bengio. 2016. Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus. *CoRR*.

Glen Shires. 2019. Web speech api: Draft community group report. [Online; posted 17-July-2019].

Ming-Hsiang Su, Chung-Hsien Wu, Kun-Yi Huang, Qian-Bei Hong, and Huai-Hung Huang. 2018. Follow-up question generation using pattern-based seq2seq with a small corpus for interview coaching. In *Proc. Interspeech 2018*, pages 1006–1010.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

William Swartout, Ron Artstein, Eric Forbell, Susan Foutz, H Chad Lane, Belinda Lange, Jacquelyn Morie, Dan Noren, Skip Rizzo, and David Traum. 2013. Virtual humans for learning. *AI Magazine*, 34(4):13–30.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Tara Walker. 2017. Presenting amazon sumerian: An easy way to create vr, ar, and 3d experiences. [Online; posted 27-November-2017].

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018a. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*.

Zichao Wang, Andrew S. Lan, Weili Nie, Andrew E. Waters, Phillip J. Grimaldi, and Richard G. Baraniuk. 2018b. Qg-net: a data-driven question generation model for educational content. In *L@S*.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *ArXiv*.

A Qualitative Examples

We also present some examples generated only by the FQG model on randomly sampled QA pairs in Table 3. As seen in the first part of the table, the questions generated on the validation samples are relevant and grammatically correct but differ from the human written follow-up questions. This shows the richness, diversity and variety in the questions apart from the training data. The second part of the table shows the follow-up questions generated on completely unseen data. The model generalizes well on the unseen data and generates grammatically correct and suitable questions. The third part of the table shows the questions generated on the text from the ASR. As it can be seen, the questions are relevant and appropriate, inspite of the errors in the transcriptions. It is to be noted that the original base question is not exposed to the errors of the ASR and can be a reliable input to the FQG model.

<table>
<tr><th colspan="4">Validation Examples</th></tr>
<tr><th>Question</th><th>Candidate Response</th><th>Generated Follow-up</th><th>Human Generated Follow-up</th></tr>
<tr><td>How would you be an asset to this organisation?</td><td>I can be an asset as I am patient while handling people and situations. I can participate and coordinate well with teams I get to work with. I can take in criticisms and work on to improve myself. I like to learn new things from anybody and will take necessary help whenever needed. I quite friendly with everyone.</td><td>How would you handle criticism?</td><td>Can criticism always be seen as a means for self-improvement?</td></tr>
<tr><td>Have you ever had to sell an idea to your co-workers? How did you do it?</td><td>I had to, when we were brainstorming ideas for a hacking session. I first tried to understand their concerns and requirements. Later I pitched my idea in accordance to that. I ensured that I understood and answered all their queries.</td><td>How did you handle the stress?</td><td>What were their first reactions?</td></tr>
<tr><td>What obstacles or difficulties have you ever faced in communicating your ideas?</td><td>I am not very expressive and can’t put thoughts into words, so i have failed in communicating my ideas and explaining it or even getting work done.</td><td>How do you cope with a difficult situation?</td><td>And what steps have you taken to overcome this hardship?</td></tr>
<tr><td>What do people most often criticize about you?</td><td>I am not very social, an introvert and can’t get along with people easily. I hvae been criticized about this.</td><td>What do you think about people who criticize you?</td><td>Have you ever tried to work on these drawbacks?</td></tr>
<tr><th colspan="4">Unseen Examples</th></tr>
<tr><th>Question</th><th>Candidate Response</th><th colspan="2">Generated Follow-up</th></tr>
<tr><td>What are your salary expectations?</td><td>As of now, I haven’t thought much about it. I’m more focused on learning the requirements for this position that I am applying for.</td><td colspan="2">How much do you think you will be able to meet these requirements?</td></tr>
<tr><td>Are you a team player?</td><td>Definitely. While I will deny the fact that I can work independently with minimal supervision, I’m also one companion every leader would ever want to be in his team. Whatever the task is, I make sure it meets and exceeds what is expected of me. I also make it a point to reach out to teammates whenever needed.</td><td colspan="2">Do you have any tips for team members?</td></tr>
<tr><td>Why do you want to work for this company?</td><td>I would be proud to work for a company like yours with such a long history of leadership in the industry. I have carried out web research and believe that the company’s products and its future projections are very impressive.</td><td colspan="2">Why do you think this company is successful?</td></tr>
<tr><th colspan="4">Unseen Examples with ASR Transcriptions</th></tr>
<tr><th>Question</th><th>Candidate Response from ASR</th><th colspan="2">Generated Follow-up</th></tr>
<tr><td>Do you consider yourself successful?</td><td>I do consider myself successful because I have been able to achieve the things that I wanted for myself so ja in the respect yes I too feel them successful</td><td colspan="2">Do you think you can achieve the things you wanted for yourself?</td></tr>
<tr><td>Tell me something about yourself.</td><td>I am a research assistant at the <name> institute I work in the area of human interaction and machine learning</td><td colspan="2">How did you learn to work with machine learning?</td></tr>
<tr><td>How do you think this job will help you fulfill your career goals?</td><td>when I apply for this job applied for this job and applied for this job many things which I like to do and I will do in the stop my career goals are just to have some work satisfaction and study whatever I want to so I think this job will help me do that and</td><td colspan="2">How would you handle a situation where you have no work satisfaction?</td></tr>
</table>

Table 3: Examples of the follow-up questions generated on the interview snippets from the validation data, unseen data and unseen data from ASR