Learning to Identify Follow-Up Questions in Conversational Question Answering

Souvik Kundu, Qian Lin, and **Hwee Tou Ng**
Department of Computer Science, National University of Singapore
souvik@u.nus.edu, qlin@u.nus.edu, nght@comp.nus.edu.sg

Abstract

Despite recent progress in conversational question answering, most prior work does not focus on follow-up questions. Practical conversational question answering systems often receive follow-up questions in an ongoing conversation, and it is crucial for a system to be able to determine whether a question is a follow-up question of the current conversation, for more effective answer finding subsequently. In this paper, we introduce a new follow-up question identification task. We propose a three-way attentive pooling network that determines the suitability of a follow-up question by capturing pair-wise interactions between the associated passage, the conversation history, and a candidate follow-up question. It enables the model to capture topic continuity and topic shift while scoring a particular candidate follow-up question. Experiments show that our proposed three-way attentive pooling network outperforms all baseline systems by significant margins.

1 Introduction

Conversational question answering (QA) mimics the process of natural human-to-human conversation. Recently, conversational QA has gained much attention, where a system needs to answer a series of interrelated questions from an associated text passage or a structured knowledge graph (Choi et al., 2018; Reddy et al., 2019; Saha et al., 2018). However, most conversational QA tasks do not explicitly focus on requiring a model to identify the follow-up questions. A practical conversational QA system must possess the ability to understand the conversation history well, and to identify whether the current question is a follow-up of that particular conversation. Consider a user who is trying to have a conversation with a machine (e.g., Siri, Google Home, Alexa, Cortana, etc). First, the user asks a question and the machine answers it. When the user asks the second question, it is very important for the machine to understand whether it is a follow-up of the first question and its answer. Further, this needs to be determined for every question posed by the user in that ongoing conversation. By identifying whether the question is a follow-up question, a machine determines whether the conversation history is relevant to the question. Based on this decision, it is expected to use a suitable answer finding strategy for answering the question. Additionally, a QA system first retrieves some relevant documents using an information retrieval (IR) engine to answer a question. If a follow-up question identifier predicts the question as an invalid follow-up question given the retrieved documents, it can communicate to the IR engine to retrieve additional supporting documents.

Passage: ... script for Verhoeven's first American film, Flesh and Blood (1985), which starred Rutger Hauer and Jennifer Jason Leigh. Verhoeven moved to Hollywood for a wider range of opportunities in filmmaking. Working in the U.S. he made a serious change in style, directing big-budget, very violent, special-effects-heavy smashes RoboCop and Total Recall. RoboCop, for ... Verhoeven followed those successes with the equally intense and provocative Basic Instinct (1992) ... received two Academy Awards nominations, for Film Editing and for Original Music ...
Conversation history:
Q: What was the first film Verhoeven did in the US?
A: Flesh and Blood
Q: What genre of films did he make?
A: big-budget, very violent, special-effects-heavy smashes
Candidate follow-up question examples:
What year did his first film debut? – **Valid**
Did he make any films during his final years? – **Invalid**
What did she do after her debut film? – **Invalid**

Figure 1: Examples illustrating the follow-up question identification task.

A few example instances are given in Figure 1 to illustrate the follow-up question identification task in a conversational reading comprehension setting.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 959–968

We present a new dataset for learning to identify follow-up questions, namely **LIF**. Given a text passage as knowledge and a series of question-answer pairs as conversation history, it requires a model to identify whether a candidate follow-up question is valid or invalid. The proposed dataset requires a model to understand both topic continuity and topic shift to correctly identify a follow-up question. For instance, in the first example given in Figure 1, a model needs to capture the topic continuity from the first question-answer pair (i.e., *first film* is *Flesh and Blood*) and the topic shift from the second question-answer pair (i.e., *genre of films*) of the conversation history. The candidate follow-up question in the second example is invalid since the associated passage does not provide any information about *his final years*. The last follow-up question example is invalid since *Verhoeven* is a *he*, not *she*.

There has been some research in the past which focuses on identifying what part of the conversation history is important for processing follow-up questions (Bertomeu et al., 2006; Kirschner and Bernardi, 2007). However, the recently proposed neural network-based models for conversational QA have not explicitly focused on follow-up questions. In this paper, we propose a three-way attentive pooling network for follow-up question identification in a conversational reading comprehension setting. It evaluates each candidate follow-up question based on two perspectives – topic shift and topic continuity. The proposed model makes use of two attention matrices, which are conditioned over the associated passage, to capture topic shift in a follow-up question. It also relies on another attention matrix to capture topic continuity, directly from the previous question-answer pairs in the conversation history. For comparison, we have developed several strong baseline systems for follow-up question identification.

The contributions of this paper are as follows:

1. We propose a new task for follow-up question identification in a conversational reading comprehension setting which supports automatic evaluation.

2. We present a new dataset, namely LIF, which is derived from the recently released conversational QA dataset QuAC (Choi et al., 2018).

3. We propose a three-way attentive pooling network which aims to capture topic shift and topic continuity for follow-up question identification. The proposed model significantly outperforms all the baseline systems.

2 Task Overview

Given a passage, a sequence of question-answer pairs in a conversation history, and a candidate follow-up question, the task is to identify whether or not the candidate follow-up question is a valid follow-up question. We denote the passage as $\mathcal{P}$ which consists of T tokens. Let the sequence of previous questions and their corresponding answers be denoted as $\{\mathcal{Q}_1, \mathcal{Q}_2, \ \ldots, \ \mathcal{Q}_M\}$ and $\{\mathcal{A}_1, \mathcal{A}_2, \ \ldots, \ \mathcal{A}_M\}$, where M is the number of previous question-answer pairs in the conversation history. The candidate follow-up question is denoted as $\mathcal{C}$. We formulate this task as a binary classification task, which is to classify $\mathcal{C}$ as *valid* or *invalid*. In the remainder of this paper, we denote the length of the candidate follow-up question as V. In our model, we concatenate all previous questions and their answers with special separator tokens as follows: $\mathcal{Q}_1 \mid \mathcal{A}_1 \mid\mid \mathcal{Q}_2 \mid \mathcal{A}_2 \mid\mid \ \ldots \ \mid\mid \mathcal{Q}_M \mid \mathcal{A}_M$. The combined length of the previous question-answer pairs in the conversation history is denoted as U.

3 LIF Dataset

In this section, we describe how we prepared the LIF dataset, followed by an analysis of the dataset.

3.1 Data Preparation

We rely on the QuAC dataset (Choi et al., 2018) to prepare the LIF dataset. Each question in the QuAC dataset is assigned one of three categories: *should ask*, *could ask*, or *should not ask* a follow-up question. We construct the *valid* instances of the dataset using the *should ask* follow-up question instances. Since the test set of QuAC is hidden, we split the QuAC development set into two halves to generate the development set and the test set of LIF. The split is done at the passage level to ensure that there is no overlap in the passages used in the development and test set.

To create each instance in LIF from QuAC, we take the associated passage, the previous question-answer pairs till it says *should ask* a follow-up question, and the next question as the gold *valid* candidate follow-up question. For each instance, we sample *invalid* follow-up questions from two sources:

1. Questions from other conversations in QuAC which can serve as potential distractors, and

2. Non-follow-up questions from the same conversation in QuAC which occurs after the gold *valid* follow-up question.

The sampling from the first source involves a two-step filtering process. We first compare the cosine similarity between the associated passage and all the questions from the other conversations by using embeddings generated by *InferSent* (Conneau et al., 2017). We take the top 200 questions based on higher similarity scores. In the second step, we concatenate the gold *valid* candidate follow-up question with the question-answer pairs in the conversation history to form an *augmented* follow-up question. Then, we calculate the token overlap count between each ranked question obtained in the first step and the *augmented* follow-up question. We normalize the token overlap count by dividing it by the length of the ranked question (after removing stop words). For each *valid* instance, we fix a threshold and take at least one but up to two questions with the highest normalized token overlap count as *invalid* candidate follow-up questions.

We also introduce potential distractors from the same conversation in QuAC. We check through the remaining question-answer pairs which occur after the *valid* follow-up question. We tag a question as an *invalid* candidate if the question appears just before it is labeled with *should not ask* a follow-up question. Throughout the *invalid* question sampling process, we exclude generic follow-up questions containing keywords such as *what else*, *any other*, *interesting aspects* and so on, to avoid selecting follow-up questions which can be potentially *valid* (e.g., *Any other insteresting aspects about this article?*).

For the training and the development sets, we combine all candidate follow-up questions from both other conversations and the same conversation. We keep three test sets with candidates from different sources: from both other conversations and the same conversation (**Test-I**), from other conversations only (**Test-II**), and from the same conversation only (**Test-III**). The overall dataset statistics are given in Table 1. We randomly sampled 100 *invalid* follow-up questions from Test-I set, and manually checked them. We verified that 97% of them are truly *invalid*.

LIF	Train/Dev/Test-I/Test-II/Test-III
#Instances	126,632/5,861/5,992/5,247/2,685
Avg #prev QA	3.6/3.7/3.7/3.9/3.5
Avg passage len	447.4/521.9/533.2/533.7/532.0
Avg question len	7.2/7.3/7.3/7.3/7.3
Avg answer len	16.3/15.8/15.6/15.7/15.6
Avg FUQ† len	8.8/8.4/8.4/8.6/7.4

Table 1: LIF dataset statistics. †follow-up question

3.2 Challenges of the Dataset

To identify whether a question is a valid follow-up question, a model needs the ability to capture its relevance to the associated passage and the conversation history. The model is required to identify whether the subject of the question is the same as in the associated passage or in the conversation history, which is often distracted by the introduction of pronouns (e.g., *I*, *he*, *she*) and possessive pronouns (e.g., *my*, *his*, *her*). Such resolution of pronouns is a critical aspect while determining the validity of a follow-up question. It also needs to examine whether the actions and the characteristics of the subject described in the candidate follow-up question can be logically inferred from the associated passage or the conversation history. Moreover, capturing topic continuity and topic shift is necessary to determine the validity of a follow-up question. The subjects and their actions or characteristics in the *invalid* follow-up questions are often mentioned in the passages, but associated with different topics.

3.3 Data Analysis

We randomly sampled 100 *invalid* instances from the Test-I set, and manually analyzed them based on different properties as given in Table 2. We found that 35% of the *invalid* questions have identical topics as the associated passages, 42% of the questions require pronoun resolution, 11% of the questions have the same subject entity as the gold follow-up question, and 5% of the questions have the same subject entity as the last question in the conversation history. Pronouns in 8% of the *invalid* questions match the pronouns in the corresponding *valid* follow-up questions, and match the last question in the conversation history for another 8% of the cases. For 7% of the cases, the question types are the same as the *valid* questions, and for 6% of the cases they are the same as the last question in the conversation history. We also observed that 4% of the *invalid* questions mention the same actions as in the corresponding *valid* ones, and they are the same as the last question in the conversation

Properties	%	Example
Identical topic	35	$\mathcal{P}$: ... the **band** released their second **album** ... $\tilde{\mathcal{Q}}$: Is "A Rush of Blood to the Head" their **album** name?
Pronoun resolution	42	$\mathcal{Q}$: Was **he** the wealthiest person? $\tilde{\mathcal{Q}}$: Did **she** go to college?
Entity match (gold)	11	$\mathcal{G}$: What was in the **song** that caused a feud? $\tilde{\mathcal{Q}}$: What was some of the **songs** on this album?
Entity match (last)	5	$\mathcal{L}$: Did her writing win any **awards**? $\tilde{\mathcal{Q}}$: Did he win any **awards**?
Pronoun match (gold)	8	$\mathcal{G}$: How many goals did **he** make? $\tilde{\mathcal{Q}}$: Was **he** married for many years?
Pronoun match (last)	8	$\mathcal{L}$: Where was **he** born? $\tilde{\mathcal{Q}}$: Why did **he** live in Exile?
Q-type match (gold)	7	$\mathcal{G}$: In **what year** did this happen? $\tilde{\mathcal{Q}}$: **What year** did he enact the reproductive health act?
Q-type match (last)	6	$\mathcal{L}$: **What happened** after that fight? $\tilde{\mathcal{Q}}$: **What happened** in this episode?
Action match (gold)	4	$\mathcal{G}$: Did he **go** on any tours? $\tilde{\mathcal{Q}}$: When did Mr Brando **go** to New York?
Action match (last)	3	$\mathcal{L}$: When did he **release**? $\tilde{\mathcal{Q}}$: When was their next album **released**?

Table 2: An analysis of the LIF dataset. The percentages do not add up to 100% since many examples consist of multiple properties. ($\tilde{\mathcal{Q}}$ – *invalid* follow-up question; $\mathcal{P}$ – associated passage; $\mathcal{G}$ – gold *valid* follow-up question; $\mathcal{L}$ – last question in the conversation history.)

history for 3% of the cases. The distribution of these properties shows the challenges in tackling this task.

4 Three-Way Attentive Pooling Network

In this section, we describe our proposed three-way attentive pooling network[1]. First, we apply an embedding layer to the associated passage, the conversation history, and the candidate follow-up question. Further, they are encoded to derive sequence-level encoding vectors. Then the proposed three-way attentive pooling network is applied to score each candidate follow-up question.

4.1 Embedding and Encoding

We use both character and word embeddings[2]. Similar to Kim (2014), we obtain the character-level embedding using convolutional neural networks (CNN). First, characters are embedded as vectors using a character-based lookup table, which are fed to a CNN, and whose size is the input channel size of the CNN. Then the CNN outputs are max-pooled over the entire width to obtain a fixed-size vector for each token. We use pre-trained vectors from GloVe (Pennington et al., 2014) to obtain a fixed-length word embedding vector for each token. Finally, both word and character embeddings are concatenated to obtain the final embeddings.

For encoding the conversation history and the candidate follow-up question, we use bidirectional LSTMs (Hochreiter and Schmidhuber, 1997). We represent the sequence-level encoding of the conversation history and the candidate follow-up question as $\mathbf{Q} \in \mathbb{R}^{U \times H}$ and $\mathbf{C} \in \mathbb{R}^{V \times H}$, respectively, where H is the number of hidden units. Similarly, we compute the sequence-level passage encoding, resulting in $\mathbf{D} \in \mathbb{R}^{T \times H}$. Then a similarity matrix $\mathbf{A} \in \mathbb{R}^{T \times U}$ is derived, where $\mathbf{A} = \mathbf{D}\,\mathbf{Q}^{\top}$.

4.1.1 Joint Encoding

We then jointly encode the passage and the conversation history. We apply a row-wise softmax function on $\mathbf{A}$ to obtain $\mathbf{R} \in \mathbb{R}^{T \times U}$. Now, for all the passage words, the aggregated representation of the conversation history is given as $\mathbf{G} = \mathbf{R}\mathbf{Q} \in \mathbb{R}^{T \times H}$. The aggregated vectors corresponding to the passage words in $\mathbf{G}$ are then concatenated with the passage vectors in $\mathbf{D}$, followed by another BiLSTM to obtain a joint representation $\mathbf{V} \in \mathbb{R}^{T \times H}$.

4.1.2 Multi-Factor Attention

In addition, multi-factor self-attentive encoding (Kundu and Ng, 2018) is applied on the joint representation. If m represents the number of factors, multi-factor attention $\mathbf{F}^{[1:m]} \in \mathbb{R}^{T \times m \times T}$ is formulated as:

$$\mathbf{F}^{[1:m]} = \mathbf{V}\mathbf{W}_f^{[1:m]}\mathbf{V}^{\top} \quad (1)$$

where $\mathbf{W}_f^{[1:m]} \in \mathbb{R}^{H \times m \times H}$ is a 3-way tensor. A max-pooling operation is performed on $\mathbf{F}^{[1:m]}$, over the number of factors, resulting in the self-attention matrix $\mathbf{F} \in \mathbb{R}^{T \times T}$. We normalize $\mathbf{F}$ by applying a row-wise softmax function, resulting in $\tilde{\mathbf{F}} \in \mathbb{R}^{T \times T}$. Now the self-attentive encoding can be given as $\mathbf{M} = \tilde{\mathbf{F}}\mathbf{V} \in \mathbb{R}^{T \times H}$. The self-attentive encoding vectors are then concatenated with the joint encoding vectors, and a feed-forward

[1]The source code and data are released at https://github.com/nusnlp/LIF

[2]We also experimented with ELMO and BERT but did not observe any consistent improvement.

neural network-based gating is applied to control the overall impact, resulting in $\mathbf{Y} \in \mathbb{R}^{T \times 2H}$. The final passage encoding $\mathbf{P} \in \mathbb{R}^{T \times H}$ is obtained by applying another BiLSTM layer on $\mathbf{Y}$.

4.2 Three-Way Attentive Pooling

Now, we use our proposed three-way attentive pooling network to score every candidate follow-up question. The architecture of the network is depicted in Figure 2.

Attentive pooling (AP) was first proposed by dos Santos et al. (2016) and successfully used for the answer sentence selection task. AP is essentially an attention mechanism that enables joint learning of the representations of a pair of inputs as well as their similarity measurement. The primary idea is to project the paired inputs into a common representation space to compare them more plausibly even if both inputs are not semantically comparable, such as a question-answer pair. In this paper, we extend the idea of attentive pooling network to the proposed three-way attentive pooling network for the follow-up question identification task, where the model needs to capture the suitability of a candidate follow-up question by comparing with the conversation history and the associated passage. In particular, the proposed model aims to capture topic shift and topic continuation in the follow-up question. dos Santos et al. (2016) used a single attention matrix to compare a pair of inputs. In contrast, our proposed model relies on three attention matrices, where the two additional attention matrices make use of the associated passage. Moreover, our proposed model is developed to deal with a more complex follow-up question identification task, in contrast to the proposed model in dos Santos et al. (2016). We score each candidate follow-up question based on its relevance to the conversation history in two different perspectives: (1) considering the associated passage (i.e., knowledge) and (2) without considering the passage.

Attention Matrix Computation

In this step, we compute three different attention matrices for capturing the similarity between the conversation history and the candidate follow-up question – two matrices when the associated passage is taken into consideration, and another one when the passage is not considered. The attention matrix $\mathbf{A}_{q,p} \in \mathbb{R}^{T \times U}$, which captures the token-wise contextual similarity between the conversation history and the passage, is given as:

$$\mathbf{A}_{q,p} = \text{f}_{\text{attn}}(\mathbf{Q}, \mathbf{P}) \ , \tag{2}$$

where the $\text{f}_{\text{attn}}(.)$ function can be written as $\text{f}_{\text{attn}}(\mathbf{Q}, \mathbf{P}) = \mathbf{P}\ \mathbf{Q}^{\top}$. Intuitively, $A_{q,p}(i, j)$ captures the contextual similarity score between the i-th token in the passage (i.e., i-th row of $\mathbf{P}$) and the j-th token in the conversation history (i.e., j-th row of $\mathbf{Q}$). Similarly, the attention matrix $\mathbf{A}_{c,p} \in \mathbb{R}^{T \times V}$, which captures the contextual similarity of a candidate follow-up question and the associated passage, is given as:

$$\mathbf{A}_{c,p} = \text{f}_{\text{attn}}(\mathbf{C}, \mathbf{P}) \tag{3}$$

Note that, $\mathbf{A}_{q,p}$ and $\mathbf{A}_{c,p}$ will be used jointly to capture the similarity between $\mathbf{Q}$ and $\mathbf{C}$, given $\mathbf{P}$.

The attention matrix $\mathbf{A}_{c,q} \in \mathbb{R}^{U \times V}$, which captures the similarity between a candidate follow-up question and the conversation history without considering the associated passage, is given as:

$$\mathbf{A}_{c,q} = \text{f}_{\text{attn}}(\mathbf{C}, \mathbf{Q}) \tag{4}$$

Attention Pooling

After obtaining the attention matrices, we apply column-wise or row-wise max-pooling. When the associated passage is considered to capture the similarity between the conversation history and the candidate follow-up question, we perform column-wise max-pooling over $\mathbf{A}_{q,p}$ and $\mathbf{A}_{c,p}$, followed by normalization with softmax, resulting in $\mathbf{r}_{qp} \in \mathbb{R}^{U}$ and $\mathbf{r}_{cp} \in \mathbb{R}^{V}$, respectively. For instance, $\mathbf{r}_{qp}$ is given as ($1 \leq i \leq U$):

$$\mathbf{r}_{qp} = \text{softmax}\ (\ldots,\ \max_{1\ \leq j\ \leq T}\ [A_{q,p}(j, i)],\ \ldots) \tag{5}$$

Intuitively, the i-th element in $\mathbf{r}_{qp}$ represents the relative importance score of the contextual encoding of the i-th token in the conversation history with respect to the passage encoding vectors. Every element of $\mathbf{r}_{cp}$ can be interpreted in the same fashion. When the associated passage encoding is not considered, we perform both row-wise and column-wise max-pooling over $\mathbf{A}_{c,q}$ to generate $\mathbf{r}_{qc} \in \mathbb{R}^{U}$ and $\mathbf{r}_{cq} \in \mathbb{R}^{V}$, respectively.

Candidate Scoring

In this step, we score each candidate follow-up question. Each candidate $\mathcal{C}$ is scored based on two perspectives – with and without consideration of

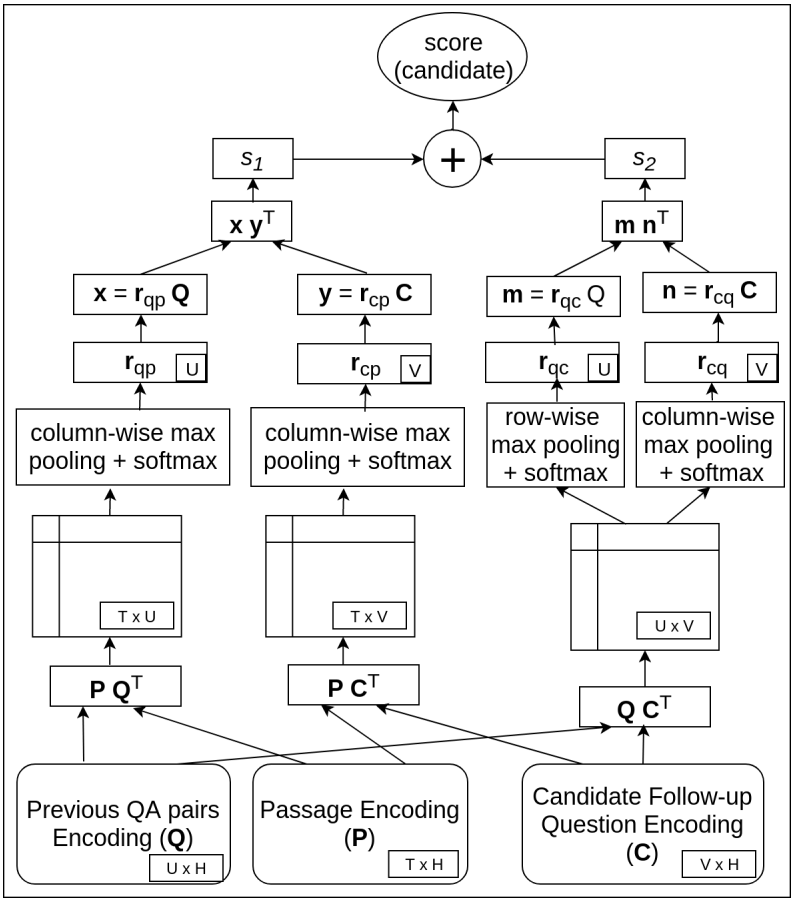

Figure 2: Architecture of the three-way attentive pooling network.

the associated passage encoding $\mathbf{P}$:

$$\begin{aligned}\text{score}(\mathcal{C}) &= s_1 \ + \ s_2 \\ &= \text{f}_{\text{sim}}(\mathbf{C}, \mathbf{Q} \mid \mathbf{P}) \ + \ \text{f}_{\text{sim}}(\mathbf{C}, \mathbf{Q}) \ , \quad (6)\end{aligned}$$

where $\mathbf{C}$ is the encoding of $\mathcal{C}$. The similarity function $\text{f}_{\text{sim}}(\mathbf{C}, \mathbf{Q} \mid \mathbf{P}) = \mathbf{x}\mathbf{y}^\top$, where $\mathbf{x} = \mathbf{r}_{qp}\ \mathbf{Q} \in \mathbb{R}^H$ and $\mathbf{y} = \mathbf{r}_{cp}\ \mathbf{C} \in \mathbb{R}^H$. The other similarity function $\text{f}_{\text{sim}}(\mathbf{C}, \mathbf{Q}) = \mathbf{m}\ \mathbf{n}^\top$, where $\mathbf{m} = \mathbf{r}_{qc}\ \mathbf{Q} \in \mathbb{R}^H$ and $\mathbf{n} = \mathbf{r}_{cq}\ \mathbf{C} \in \mathbb{R}^H$.

We use binary cross entropy loss for training the model. For prediction, we find a threshold to maximize the scores on the development set. For the test instances, we use the threshold to predict whether a follow-up question is valid or invalid.

5 Baseline Models

We develop several rule-based, statistical machine learning, and neural baseline models. For all the models, a threshold is determined based on the best performance on the development set.

5.1 Rule-Based Models

We develop two models based on word overlap counts – between the candidate follow-up question and the passage, and between the candidate follow-up question and the conversation history. We normalize the count values based on the length of the candidate follow-up question.

Next, we develop two models based on the contextual similarity scores using *InferSent* sentence embeddings (Conneau et al., 2017). The two models compare the candidate follow-up question with the associated passage and the conversation history, respectively. The similarity scores are computed based on vector cosine similarity.

We also develop another rule-based model using tf-idf weighted token overlap scores. We prepend the last question from the conversation history to the candidate follow-up question and add the tf-idf of overlapping words between the concatenated context and the passage.

5.2 Statistical Machine Learning Models

We handcraft two sets of features for the statistical machine learning models. One set of features consists of tf-idf weighted GloVe vectors. Since we adopt 300 dimensional GloVe vectors in our experiments, these features are of dimension 300.

Models	Dev V-P/-R/-F1/Macro F1	Test-I V-P/-R/-F1/Macro F1	Test-II V-P/-R/-F1/Macro F1	Test-III V-P/-R/-F1/Macro F1
Norm. overlap (Psg)	34.4/46.4/39.5/50.9	36.0/52.1/42.6/52.5	40.5/60.3/48.5/52.3	71.5/65.4/68.3/48.6
Norm. overlap (Hist)	34.1/40.7/37.1/51.0	33.8/43.1/37.9/50.8	40.6/33.2/36.6/52.2	78.6/67.4/72.6/58.3
InferSent (Psg)	28.4/42.7/34.1/44.2	28.5/47.0/35.5/43.6	30.0/40.6/34.5/42.0	72.3/71.4/71.9/50.1
InferSent (Hist)	22.0/11.9/15.5/43.5	25.2/12.7/16.9/45.1	26.6/10.5/15.1/42.8	72.1/69.6/70.8/49.7
Tf-idf + Overlap	32.6/61.7/42.7/45.5	32.5/66.3/43.7/44.6	37.5/66.3/47.9/46.7	72.1/85.6/78.2/48.2
Logistic Regression				
Tf-idf + GloVe	58.4/67.5/62.6/71.1	53.5/61.1/57.0/67.1	63.7/66.0/64.8/71.8	73.5/95.1/82.9/50.1
Overlap count	41.0/61.9/49.4/57.1	39.7/58.4/47.3/56.1	49.1/57.6/53.0/60.7	73.1/98.5/83.9/47.0
CNN-Maxpool	61.6/67.3/64.3/72.9	58.0/62.3/60.1/69.9	69.0/62.4/65.5/73.4	78.4/62.2/69.4/56.5
CNN-Attnpool	52.8/56.9/54.8/65.7	48.3/54.0/51.0/62.7	56.2/54.3/55.2/64.9	77.6/53.7/63.5/53.0
LSTM-MaxPool	75.1/70.0/72.4/79.9	72.9/66.1/69.3/77.8	89.9/66.1/76.2/82.5	79.3/66.1/72.1/58.7
LSTM-AttnPool	72.6/65.7/69.0/77.5	72.1/66.2/66.8/76.2	89.2/62.2/73.3/80.6	79.0/62.2/69.6/57.1
BERT	74.2/76.4/75.3/81.5	72.4/76.1/74.2/80.7	88.5/76.1/81.8/86.2	79.9/76.1/78.0/62.6
Three-way AP	76.2/77.3/76.8/82.7	74.4/75.7/75.0/81.4	89.0/75.7/81.8/86.2	81.9/75.7/78.7/65.0

Table 3: Comparison results for the follow-up question identification task. We compare the performance of three-way attentive pooling network with several rule-based, statistical machine learning, and neural models (V – Valid, P – Precision, R – Recall, Psg – Passage, Hist – Conversation history).

Another set of features consists of word overlap counts. We compute the pairwise word overlap counts among the candidate follow-up question, the associated passage, and the conversation history. The overlap count-based features are of dimension 3. We experiment with logistic regression using the derived features.

5.3 Neural Models

We also develop several neural baseline models. We first concatenate the associated passage, the conversation history, and the candidate follow-up question, followed by embedding (the same as described earlier). Then, we apply sequence-level encoding with either BiLSTM or CNN. For CNN, we use equal numbers of unigram, bigram, and trigram filters, and the outputs are concatenated to obtain the final encoding. Next, we apply either global max-pooling or attentive pooling to obtain an aggregated vector representation, followed by a feed-forward layer to score the candidate follow-up question. Let the sequence encoding of the concatenated text be $\mathbf{E} \in \mathbb{R}^{L \times H}$, and $\mathbf{e}_t$ be the tth row of $\mathbf{E}$. The aggregated vector $\tilde{\mathbf{e}} \in \mathbb{R}^H$ for attentive-pooling can be obtained as:

$$a_t \propto \exp(\mathbf{e}_t \mathbf{w}^\top); \quad \tilde{\mathbf{e}} = \mathbf{a}\,\mathbf{E}, \qquad (7)$$

where $\mathbf{w} \in \mathbb{R}^H$ is a learnable vector. We also develop a baseline model using BERT (Devlin et al., 2019). We first concatenate all the inputs and then apply BERT to derive the contextual vectors. Next, we aggregate them into a single vector using attention. Then a feed-forward layer is used to score each candidate follow-up question.

6 Experiments

In this section, we present the experimental settings, results, and performance analysis.

6.1 Experimental Settings

We do not update the GloVe vectors during training. We use 100-dimension character-level embedding vectors. The number of hidden units in all the LSTMs is 150 ($H = 300$). We use dropout (Srivastava et al., 2014) with probability 0.3. Following Kundu and Ng (2018), we set the number of factors as 4 in multi-factor attentive encoding. We use the Adam optimizer (Kingma and Ba, 2015) with learning rate 0.001 and clipnorm 5. Following Choi et al. (2018), we consider at most 3 previous question-answer pairs in the conversation history. This being a binary classification task, we use precision, recall, F1, and macro F1 as evaluation metrics. All scores reported in this paper are in %.

6.2 Results

Table 3 shows that our proposed model outperforms the competing baseline models by significant margins across all test sets. We perform statistical significance tests using paired t-test and bootstrap resampling. Performance of our proposed model is significantly better ($p < 0.01$) than the best baseline system which provides the highest Macro-F1 score on Test-I. The LSTM-based neural baselines perform better than the rule-based and statistical machine learning models in most cases. On Test-III, the statistical models tend to predict *valid*, and the number of valid instances is much higher than the invalid instances (about 75%:25%), resulting

Model	V-P	V-R	V-F1	Macro F1
– History	72.7	67.0	69.7	77.9
– Knowledge	75.8	73.8	74.8	81.4
– $\mathbf{A}_{c,q}$	71.8	75.8	73.7	80.2
– Multi-factor Attn	75.6	76.4	76.0	82.1
– Joint encoding	75.3	76.6	76.0	82.1
– Char embedding	74.2	72.3	73.2	80.2
Three-way AP	76.2	77.3	76.8	82.7

Table 4: An ablation study on the development set.

in high Valid F1 scores. These baseline systems (while performing well on valid questions) perform poorly when evaluated using Macro F1 which measures performance on both valid and invalid follow up questions. Macro F1 is the overall evaluation metric used to compare all systems. Overall, identifying follow-up questions from the same conversation (Test-III) is harder compared to other conversations (Test-II).

We perform an ablation study as shown in Table 4. The proposed model performs worst when we do not consider the conversation history. This is because the question-answer pairs in the conversation history help to determine topic continuity while identifying a valid follow-up question. The performance also drops when we do not consider the associated passage (i.e., knowledge) because it helps to capture topic shift. The performance also degrades when we remove $\mathbf{A}_{c,q}$. It performs better than the model where we do not consider the conversation history at all, as the conversation history is taken into consideration in passage encoding. The performance also degrades when we remove other components such as multi-factor attentive encoding, joint encoding, and character embedding.

6.3 Qualitative Analysis

The proposed model aims to capture topic continuity and topic shift by using a three-way attentive pooling network. Attention pooling on $\mathbf{A}_{q,p}$ and $\mathbf{A}_{c,p}$ aims to capture topic shift in the follow-up question for a given conversation history. Consider the first example in Table 5. When we do not consider the passage, it could not identify the follow-up question correctly while our proposed model correctly identifies the topic shift to the duration of the riot by validating with the passage words *after four days* and *restore order and take back the prison on September 13*. In the second example, while our model could correctly identify topic continuity through *Schuur*, the model without history fails to identify the follow-up question.

We performed an error analysis where our proposed model failed to identify the follow-up questions. We randomly sampled 50 such instances (25 *valid* and 25 *invalid*) from the development set. We found that 32% of them require pronoun resolution for the subject in the follow-up questions. 38% of the instances require validation of the actions/characteristics of the subjects (e.g., *did they have any children?* vs. *gave birth to her daughter*). 14% of the errors occur when it requires matching objects or predicates which occur in different forms (e.g., *hatred* vs *hate*, *television* vs *TV*). For the remaining 16% of the cases, it could not correctly capture the topic shift.

7 Related Work

Many data-driven machine learning methods have been shown to be effective for tasks relevant for dialog such as dialog policy learning (Young et al., 2013), dialog state tracking (Henderson et al., 2013; Williams et al., 2013; Kim et al., 2016), and natural language generation (Sordoni et al., 2015; Li et al., 2016; Bordes et al., 2017). Most of the recent dialog systems are either not goal oriented (e.g., simple chit-chat bots), or domain-specific if they are goal oriented (e.g., IT help desk). In the last few years, there has been a surge of interest in conversational question answering. Saha et al. (2018) released a Complex Sequential Question Answering (CSQA) dataset for learning conversations through a series of interrelated QA pairs by inferencing over a knowledge graph. Choi et al. (2018) released a large-scale conversational QA dataset, namely question answering in context (QuAC), which mimics a student-teacher interactive scenario. Reddy et al. (2019) released the CoQA dataset and many systems were evaluated on it. Zhu et al. (2018) proposed SDNet to fuse context into traditional reading comprehension models. Huang et al. (2019) proposed a "Flow" mechanism that can incorporate intermediate representations generated during the process of answering previous questions, through an alternating parallel processing structure. In a conversation setting, given the previous QA pairs as conversation history, while these models focus on answering the next question, our work is focused on identifying follow-up questions. Recently, Saeidi et al. (2018) proposed a dataset for regulatory texts that requires a model to ask follow-up clarification questions. However, the answers are limited to *yes* or *no*, which makes the task rather

Passage: On September 9, 1971, prisoners at the state penitentiary at Attica, NY, took control of a cell block and seized thirty-nine correctional officers as hostages. After four days of negotiations, Department of Correctional Services Commissioner Russell Oswald agreed to most of the inmates' demands for various reforms but refused to grant complete amnesty to the rioters, with passage out of the country and removal of the prison's superintendent. When negotiations stalled and the hostages appeared to be in imminent danger, Rockefeller ordered New York State Police and national guard troops to restore order and take back the prison on September 13. . . . **History**: Q: Where was the Attica Prison? A: On September 9, 1971, prisoners at the state penitentiary at Attica, NY, took . . . Q: Why did they riot? A: the inmates' demands for various reforms but refused to grant complete amnesty to the rioters, with passage out of the country and removal of the prison's superintendent. **Candidate follow-up question**: How long did the riot last? – **Valid**
Passage: In 1975, at age 22, Schuur auditioned for drummer/bandleader Ed Shaughnessy. Escorted by her twin brother, she went backstage to . . . singing the blues. . . . He hired her to be the vocalist in his orchestra, "Energy Force". Jazz trumpeter Dizzy . . . **History**: Q: When was Schuur discovered? A: In 1975, at age 22, Schuur auditioned for drummer/bandleader Ed Shaughnessy. . . . **Candidate follow-up question**: Was she hired by Ed Shaughnessy? – **Valid**

Table 5: Examples taken from the LIF development set where our model correctly identified a valid follow-up question.

restrictive. Moreover, while Saeidi et al. (2018) focuses on generating a clarification question in response to a question of a conversation, we focus on identifying whether a question is a follow-up question of a conversation.

8 Conclusion

In this paper, we present a new follow-up question identification task in a conversational setting. We developed a dataset, namely LIF, which is derived from the previously released QuAC dataset. Notably, the proposed dataset supports automatic evaluation. We proposed a novel three-way attentive pooling network which identifies whether a follow-up question is valid or invalid by considering the associated knowledge in a passage and the conversation history. Additionally, we developed several strong baseline systems, and showed that our proposed three-way attentive pooling network outperforms all the baseline systems. Incorporating our three-way attentive pooling network into open domain conversational QA systems will be interesting future work.

Acknowledgments

This research is supported by the National Research Foundation Singapore under its AI Singapore Programme (Award Number: AISG-RP-2018-007).

References

Núria Bertomeu, Hans Uszkoreit, Anette Frank, Hans-Ulrich Krieger, and Brigitte Jörg. 2006. Contextual phenomena and thematic relations in database QA dialogues: Results from a Wizard-of-Oz experiment. In *Proceedings of the Interactive Question Answering Workshop at HLT-NAACL*.

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *Proceedings of ICLR*.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of EMNLP*.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of EMNLP*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*.

Matthew Henderson, Blaise Thomson, and Steve Young. 2013. Deep neural network approach for the dialog state tracking challenge. In *Proceedings of SIGDIAL*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hsin-Yuan Huang, Eunsol Choi, and Wen tau Yih. 2019. FlowQA: Grasping flow in history for conversational machine comprehension. In *Proceedings of ICLR*.

Seokhwan Kim, Luis Fernando D'Haro, Rafael E. Banchs, Jason Williams, and Matthew Henderson. 2016. The fourth dialog state tracking challenge. In *Proceedings of IWSDS*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of EMNLP*.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Manuel Kirschner and Raffaella Bernardi. 2007. An empirical view on IQA follow-up questions. In *Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue*.

Souvik Kundu and Hwee Tou Ng. 2018. A question-focused multi-factor attention network for question answering. In *Proceedings of AAAI*.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016. Deep reinforcement learning for dialogue generation. In *Proceedings of EMNLP*.

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

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the ACL*.

Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of EMNLP*.

Amrita Saha, Vardaan Pahuja, Mitesh M. Khapra, Karthik Sankaranarayanan, and Sarath Chandar. 2018. Complex sequential question answering: Towards learning to converse over linked question answer pairs with a knowledge graph. In *Proceedings of AAAI*.

Cicero dos Santos, Ming Tan, Bing Xiang, and Bowen Zhou. 2016. Attentive pooling networks. *arXiv preprint arXiv:1602.03609*.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of NAACL*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958.

Jason Williams, Antoine Raux, Deepak Ramachadran, and Alan Black. 2013. The dialog state tracking challenge. In *Proceedings of SIGDIAL*.

Steve Young, Milica Gašić, Blaise Thomson, and Jason D. Williams. 2013. POMDP-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5).

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2018. SDNet: Contextualized attention-based deep network for conversational question answering. *arXiv preprint arXiv:1812.03593*.