*Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022)*, pages 7321–7327
Marseille, 20-25 June 2022

# BehanceQA: A New Dataset for Identifying Question-Answer Pairs in Video Transcripts

**Amir Pouran Ben Veyseh[1], Viet Dac Lai[1], Franck Dernoncourt[2], Thien Huu Nguyen[1]**
[1]Dept. of Computer and Information Science, University of Oregon, Eugene, OR, USA
[2]Adobe Research, Seattle, WA, USA
{vietl,apouranb,thien}@cs.uoregon.edu, franck.dernoncourt@adobe.com

## Abstract

Question-Answer (QA) is one of the effective methods for storing knowledge which can be used for future retrieval. As such, identifying mentions of questions and their answers in text is necessary for a knowledge construction and retrieval systems. In the literature, QA identification has been well studied in the NLP community. However, most of the prior works are restricted to formal written documents such as papers or websites. As such, Questions and Answers that are presented in informal/noisy documents have not been adequately studied. One of the domains that can significantly benefit from QA identification is the domain of livestreaming video transcripts that involve abundant QA pairs to provide valuable knowledge for future users and services. Since video transcripts are often transcribed automatically for scale, they are prone to errors. Combined with the informal nature of discussion in a video, prior QA identification systems might not be able to perform well in this domain. To enable comprehensive research in this domain, we present a large-scale QA identification dataset annotated by human over transcripts of 500 hours of streamed videos. We employ Behance.net to collect the videos and their automatically obtained transcripts. Furthermore, we conduct extensive analysis on the annotated dataset to understand the complexity of QA identification for livestreaming video transcripts. Our experiments show that the annotated dataset presents unique challenges for existing methods and more research is necessary to explore more effective methods. The dataset and the models developed in this work will be publicly released for future research. Dataset is available from https://github.com/amirveyseh/BehanceQA/tree/main.

**Keywords:** Question Answering, Video Transcripts

## 1. Introduction

One of the efficient methods to acquire knowledge about a new topic is via Questions. Due to their applicability, preparing high quality Question-Answer (QA) pairs has been used as an effective method for storing the knowledge to be used in future retrieval. To this end, identifying QA pairs from raw text is a necessary pre-processing step. This task is called Question-Answer Identification (QAI) that aims to find all important questions along with their answers provided in text. For instance, in the text excerpt "*Now, we will explore the rotation techniques. How can you rotate the image by certain degrees? For that you can use CTR+R shortcut key.*", a QAI system needs to recognize the question "*How can you rotate the image by certain degrees?*" along with its answer "*For that you can use CTR+R shortcut key.*". Note that questions and answers (if available) should be matched with each other in QAI. A QAI system can be used for constructing a knowledge base of QA pairs.

Due to the importance of QAI, this task has been studied well in the NLP community (Richardson et al., 2013; Hermann et al., 2015; Rajpurkar et al., 2016; Joshi et al., 2017; He et al., 2018; Rogers et al., 2021). However, most of the existing work are restricted to the formally and well written documents such books or websites. As such, the challenges for QAI in informal and noisy settings are less explored. One of the domains with informal text that can benefit from QAI is the domain of livestreaming video transcripts. Tutorial videos or streamed videos in which the streamers discuss specific technical topics are replete with QA pairs that can provide valuable knowledge for future users. However, in order to identify QA pairs, a system will need to first automatically transcribe the videos to obtain transcript texts in scale to enable text processing models for QAI. Unfortunately, automatically generated transcripts are prone to errors and do not provide correct punctuation which will pose significant challenges to existing QAI systems. Moreover, in a streamed video, the streamer might diverge from the main topic, e.g., chitchats or off-topic discussions, making it harder for a QAI model to recognize the relevant questions or answers. Due to such challenges and lack of resource to study them, in this work we present a large-scale manually-annotated QAI dataset for the domain of video transcripts.

We collect 298 videos, covering 500 hours of streamed videos, from the Behance platform. Behance is a website for artists to share their creative projects using Adobe Creative Cloud products, e.g., Photoshop. Most of the content in these websites is transmitted verbally in English, thus processing them requires effective text processing methods. In order to obtain video transcripts, we employ the Microsoft Automatic Speech Recognition (ASR) tool. Transcript of each video is manually annotated with expert annotators with background knowledge in creative projects and data annotation. For each transcript, the questions mentioned by the streamer are annotated. Annotators also identify an-

| Statistics | Train | Test | Dev | Total |
|---|---|---|---|---|
| # Sentences | 220,520 | 23,273 | 14,104 | 257,897 |
| # Questions | 13,903 | 1,470 | 993 | 16,366 |
| # Answers | 4,371 | 527 | 303 | 5,272 |
| Avg Question Length | 7.78 | 6.13 | 8.88 | 7.79 |
| Avg Answer Length | 18.35 | 17.22 | 15.48 | 18.05 |

Table 1: Statistics of BehanceQA. Numbers of questions and answers represent the numbers of word spans annotated as questions or answers. Average length of questions and answers are presented by numbers of words.

swer spans for questions (in case they exist). In total, 257,897 sentences are annotated from which 13,903 questions and 5.272 answers are identified. Our dataset is called BehanceQA.

We conduct extensive analysis on BehanceQA to shed more light on the challenging nature of QAI in the domain of video transcripts. Our analysis show that while current methods can effectively find questions, they struggle with identifying answers in video transcripts. Moreover, we observe that joint models for QAI outperforms pipeline solutions for answer detection. As such, more research is needed to study the relation between answers and questions for QAI in video transcript domain. To promote future research in this direction, we will release the prepared dataset along with the code for our experiments.

## 2. Dataset

### 2.1. Data Collection

We propose to use the videos streamed on the prominent video-hosting platform Behance[1] to obtain transcripts that will be annotated with questions and answers for BehanceQA in this research. Behance is a website where artists may showcase their work created with Adobe Creative Cloud products (e.g., Photoshop, Illustrator, etc.). The majority of the content is conveyed verbally in English. Each video lasts anything between a few minutes and several hours. We begin by gathering 298 videos with a total duration of 500 hours in the first step. A video lasts about 48 minutes on average. The transcript documents of the videos are then obtained using the Microsoft Automatic Speech Recognition program for each video. On average, a video transcript has 7,219 words. The transcripts are split into sentences to make the annotation process easier (i.e., using the utterances generated by the ASR tool). A transcript, on average, has 621 sentences. To improve this sentence spitting step, we can consider better punctuation restoration systems for video transcripts in future work (Lai et al., 2022).

### 2.2. Annotation

We recruited 10 annotators from the Upwork crowdsourcing platform. As Upwork allows the freelancers to submit their resumes to bid for the job, we choose the most experienced writers and proof readers with prior experience on data annotation and graphic creativity tools such as Adobe Photoshop and Adobe Illustrator. The annotators are provided with a detail guideline with examples of questions and answers. The BRAT annotation tool (Stenetorp et al., 2012) is used to assist the annotation. The BRAT tool allows us to pre-annotate the input texts with some prior knowledge. Hence, relying on the question marks from ASR, we initialize the texts presented to the BRAT tool with the annotation where all sentences ending with question marks are marked as questions. The annotators are allowed to delete/modify the initialized marking if it is incorrect. Based on our observation, this significantly reduces the annotation efforts. To clarify, each sentence in an input document is shown in its own line in our annotation framework. To perform the annotation, the annotators will select sequences of consecutive words as spans for questions or answers. We only annotate a span as an answer if it answers some annotated question. We do not ask annotators to link an answer span to its corresponding question in our data as we find that an annotated answer can be perfectly linked with its preceding annotated question. As such, the QAI problem in our dataset can be transformed into the problem of detecting question and answer spans in video transcript documents.

We randomly select a subset of documents for QAI annotation in BehanceQA. In the first phase of the annotation, the annotators independently co-annotate 20% of the selected documents, achieving the Cohen's Kappa scores of 0.72 that indicates a substantial agreement among annotators. The annotators then discuss to resolve the conflicts over the annotated data. Next, the remaining 80% of data is distributed to the annotators to perform separate annotation and generate the final version of our dataset BehanceQA. To facilitate model development and evaluation, we split the dataset into 3 portions for training/development/test data. Table 1 shows detailed statistics of BehanceQA.

### 2.3. Challenges

Annotating questions and answers in video transcripts is a challenging task. Specifically, during the annotation process, the following sources of confusion has been detected:

- **Answer Span Disagreement**: In video transcripts, the answers to questions are expressed in the context of other topics that might provide background information but are not necessarily related to the question. Moreover, the steamers might provide multiple answers to a question. As such, identifying correct spans of answers is challenging. For instance, in the text excerpt "*How would I change his eyes? Oh, they are very dark. Since it is in a different layer we can directly change its color. But let me use my new brush to*

[1] www.behance.net

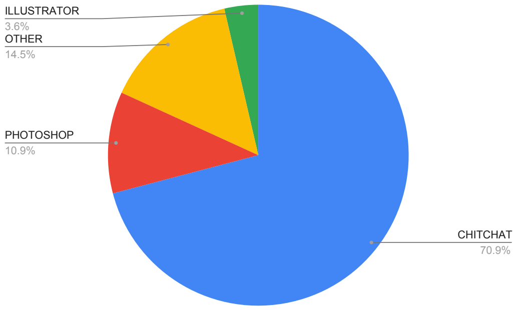

Figure 1: Distribution of question types in a sample of 150 questions selected from the training set.

*edit it.*" with one question "*How would I change his eyes?*". The text span "*they are very dark*", provides background information but is not an essential part of the answer for the question. Also, in this sample, the streamer provides two possible answers to the question, i.e., "*Since it is in a different layer we can directly change its color.*" and "*But let me use my new brush to edit it.*". As such, annotators might disagree on which words should be selected as the answer span. To resolve this conflict, we require the answer spans to directly reply the questions to avoid texts with background or indirect information. To this end, in our running example, "*Since it is in a different layer we can directly change its color.*" should be annotated as the answer span.

- **Rhetorical Questions**: In a video transcript, the streamer might ask questions that are not intended to be answered later. In fact, these questions are expected to attract audience attention and persuade them to agree with the streamer. For instance, in the text "*They look much better, don't you think so?*", the presented question is a rhetorical question whose goal is to convince the audience. Some annotators might select the rhetorical questions while others ignore them. In order to resolve this conflict, in our discussion, we require the annotators to avoid rhetorical questions.

### 2.4. Analysis

To provide more insights into the dataset, we manually study the types of annotated questions for a randomly selected set of 150 questions. In our analysis, we are able to identify four types of questions in the annotated data: (1) Photoshop, (2) Illustrator, (3) Other Products, and (4) Chitchat. In particular, the first three question types involve an explicit information request about a feature of the products of interest, focusing on Photoshop, Illustrator, or Other Products in our dataset. The fourth question type, on the other hand, refers to general questions that are related to a specific product, including chitchat or causal questions. The distribution of the four question types is shown in Figure 1. It shows that the majority of questions are of Chitchat type in livestreaming video transcripts. Note that identifying chitchat questions in text might be a more challenging task as they can be expressed in more diverse formats and topics, thus presenting a unique challenge in our dataset.

## 3. Experiments

To study the challenges of the proposed dataset, we evaluate the performance of typical models for QAI on BenhanceQA. Specifically, we study the performance for question detection, answer detection and the overall task. To this end, we examine two groups of models in this section:

**Pipeline**: In the pipeline approach, a model first extracts the questions in the transcript. Then, given the extracted questions, an separate answer detection model is employed to recognize the answers to each question. Concretely, for question detection, we study two baseline models: (a) **RULE**$^{Q}$: In this model, we employ the punctuation available in the transcripts to detect the questions. Specifically, sentences ending in a question mark are selected as question; and (b) **BERT**$^{Q}$: This model leverage BERT (Devlin et al., 2019), a recent state-of-the-art text encoder to perform question detection. In particular, the model first split input texts in chunks that can fit into the length limit of BERT (i.e., 512 sub-tokens) and each chunk will be analyzed to recognized questions separately using the sequence labeling framework and the BIO tagging schema. As such, each word in the chunks will be assigned a label to indicate whether it is at the beginning, inside or outside a question span. In this model, each chunk $W = w_1, w_2, \ldots, w_n$ ($w_i$ is the $i$-th word in the chunk) is sent into the BERT$_{base}$ model using the format of $[[CLS], w_1, w_2, \ldots, w_n]$. The BERT model will return a representation vector for each word $w_i inW$ using the average of the vectors for the word-pieces of $w_i$ in the last transformer layer of BERT. Finally, the word representations will be consumed by a feed-forward network to predict a BIO label for question detection for each word in the chunk.

For the answer detection models, we also explore two models: (a) **BERT**$^{A}$: A separate BERT$_{base}$ model is employed for answer detection. For each detected question $Q = q_1, q_2, \ldots, q_m$ (of $m$ words) from the question detection model, we obtain a sequence of words $C = c_1, c_2, \ldots, c_k$ (of $k$ words) in the document that directly follow $Q$ so that the total length of $Q$ and $C$ can maximally fill in the length limit of BERT. Afterward, the concatenation of the question $Q$ and context $C$, i.e., in the form of $[[CLS], q_1, \ldots, q_m, [SEP], c_1, \ldots, c_k]$, will be sent into BERT to obtain representation vectors for the words in $C$. Finally, the representations for $c_i \in C$ will also be consumed by a feed-forward network to predict a BIO tag for each word $c_i$ to capture answer spans in $C$. Note that the inclusion of $Q$ in the input for

| Model | Question | | | Answer | | | Overall | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 |
| RULE$^Q$-BERT$^A$ (Pipeline) | 43.83 | 51.72 | 47.45 | 36.72 | 2.80 | 5.22 | 41.47 | 37.99 | 39.65 |
| RULE$^Q$-BERT+CRF$^A$ (Pipeline) | 43.83 | 51.72 | 47.45 | 25.98 | 2.38 | 4.37 | 38.52 | 37.88 | 38.19 |
| BERT$^Q$-BERT$^A$ (Pipeline) | 81.31 | 93.37 | 86.92 | 42.25 | 4.70 | 8.47 | 69.95 | 68.49 | 69.21 |
| BERT$^Q$-BERT+CRF$^A$ (Pipeline) | 81.31 | 93.37 | 86.92 | 41.13 | 4.16 | 7.56 | 69.64 | 68.34 | 68.98 |
| Joint_BERT (Joint) | 85.22 | 78.12 | 81.52 | 41.19 | 6.53 | 11.28 | 72.47 | 58.00 | 64.43 |
| Joint_BERT+CRF (Joint) | 84.16 | 76.88 | 80.36 | 40.39 | 5.79 | 10.13 | 71.50 | 56.91 | 63.37 |

Table 2: Performance of the models on the test set of the proposed dataset in terms of Precision, Recall and F1 score for correctly predicting boundaries of questions and answers. The first group are Pipeline models. The names of pipeline models follow the format of "Question Model"-"Answer Model". The second group are Joint models.

BERT allows the representation vectors for the words in $C$ to condition on the detected question $Q$ for more question-customized information; (b) **BERT+CRF$^A$**: This model is similar to the previous BERT$^A$ model for answer detection. However, a Conditional Random Field layer (CRF) is stacked on top of the BERT model to capture the label dependency for the words in $C$ for sequence labeling for answer detection. In the pipeline approach, by combining two options for question detection and two options for anser detection, we obtain four different complete models for QAI in our dataset. To denote the full pipeline model, the question and answer model names are concatenated using hyphen for separation, e.g., RULE$^Q$-BERT$^A$ for a pipeline model with RULE$^Q$ for question model and BERT$^A$ for answer model.

**Joint**: The second group of models involve joint models in which the boundaries of questions and answers are simultaneously predicted in a single model. We also use the sequence labeling framework to solve this joint prediction problem where the BIO tagging is extended to include labels for both questions and answers, i.e., $\{B\_Question, I\_Question, B\_Answer, I\_Answer, O\}$. First, we also split the input texts into chunks to fit the length limit of BERT as done in the BERT model for question detection in the pipeline approach. Next, the input text for each chunk $W$, i.e., $[[CLS], w_1, w_2, \ldots, w_n]$, is encoded using the BERT$_{base}$ model and a feed-forward network is applied on the top to perform word classification. This model is called **Joint_BERT** for the joint modeling approach. In addition, we also explore the inclusion of the CRF layer on top of BERT, leading to the **Joint_BERT+CRF** model to solve joint prediction of question and answer spans.

To evaluate the performance of the models, we employ Precision, Recall and F1 scores for identifying correct boundaries of questions, answers, and their aggregation as the performance measures. We employ the development set of BehanceQA to tune the hyper-parameters for the models. The test set is utilized for model comparison. In particular, we select the learning rate of $2e$-5 for the Adam optimizer and the mini-batch size of 128 for training. For the feed-forward networks in the models, we employ two layers (selected in the set $[1, 2, 3, 4]$) with 256 hidden units in each layer (selected in the set $[64, 128, 256, 512]$).

### 3.1. Results

The results of the experiments are presented in Table 2. The first observation from the table is that the RULE$^Q$ model for question detection performs very poorly, thus leading to significantly worse performance of answer and overall scores for the corresponding Pipeline models. In addition, between question and answer detection tasks, the latter is more challenging, as both the Pipeline and Joint models have significantly lower performance for answer detection compared to question detection. This is expected since in addition to contextual information, the answer model will need to appropriately capture the questions to determine correct answer spans. Also, the answer spans are more ambiguous (as discussed in Section 2.3) and involves longer sequences than the question spans (i.e., see Table 1), thus contributing to the more complication of answer boundaries for QAI.

Comparing the Joint and Pipeline models, we find that the former performs better for answer detection. In particular, for answer detection, the best Joint model, i.e., Joint_BERT, improves the answer F1 score of the best Pipeline model, i.e., BERT$^Q$-BERT$^A$ by $2.81\%$. It thus suggests that there are inter-dependencies between representations for answers and questions, and jointly modeling answers and questions will improve the representations for answers to deliver better performance. In contrast, for question detection, it turns out that the Pipeline models can achieve better performance than the Joint models. Specifically, the best Pipeline model BERT$^Q$-BERT+CRF$^A$ for question detection is 5.4% better than the best question F1 score of the Joint models (i.e., Joint_BERT). We hypothesize that due to the difficulty of answer detection, the question detection in the Joint models could be distracted by the influence of answer detection to affect the performance. In all, it implies that Joint and Pipeline models have their own advantages and disadvantages and more research effort is necessary to design appropriate models to balance the two tasks to optimize the performance. Also, from the table, it is clear that the CRF layer is not helpful for both the answer detection model in the Pipeline ap-

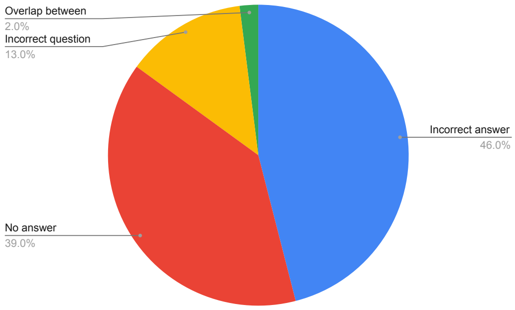

Figure 2: Error analysis for answer detection (analysis is conducted on 5% of the development set).

proaches and the Joint models as including CRF would decrease the performance. We hypothesize that it is due to the ability of the BERT model to encode sufficient information in the word representation, making the CRF layer redundant for QAI in our dataset. Overall, the BERT$^Q$-BERT$^A$ model in the Pipeline architecture achieves the best trade-off performance between question and answer detection (i.e., 69.21% for the F1 score). However, as this best performance is still far from being perfect, our dataset BehanceQA offer ample opportunities for future research to improve its performance and understand texts from livestreaming video transcripts.

### 3.2. Analysis

As discussed in Section 3.1, the performance of the models for answer detection is significantly lower than those the question detection. To study what contributes the most to the errors for answer detection, we conduct an error analysis on 5% of the development set. For this analysis, we analyze the outputs of the best performing model for answer detection, i.e., Joint_BERT. Based on our analysis, we are able to categorize the errors into 4 major groups whose distribution is presented in Figure 2. Specifically, we find the following errors for answer detection (numbers in parentheses represent the proportions of the errors): (a) **Incorrect Answer**(46%): In this category, the questions are correctly recognized by the joint model; however, the predicted boundaries of the answers are not accurate; (b) **No Answer**(39%): For this case, the model incorrectly predict no answer in the input texts. We find that in this type of error, the answers tend to be implicitly provided and requires deeper understanding of the texts; (c) **Incorrect Question**(13%): We observe that for these cases when the model predicts the questions incorrectly, it also cannot identify the answers. Specifically, for 13% of the errors in the development set, the incorrect predictions of the question result in incorrect answer detection too; and (d) **Overlap between Q&A**(2%): For some rare scenarios, the model is confused between questions and answers, i.e., it incorrectly predicts part of an question as belonging to the answer. Using this analysis, future research can devise better architecture to capture transcript contexts to improve the performance on BehanceQA.

## 4. Related Work

Question and answer identification (QAI) is an important tasks in NLP (Shrestha and McKeown, 2004; Li et al., 2011; Richardson et al., 2013; He et al., 2018; Rachha and Vanmane, 2020; Khan et al., 2020; Rogers et al., 2021). This task can be helpful to extract important information from texts to populate knowledge bases, or directly create FQA repositories from input texts (West et al., 2014; Sakata et al., 2019). Due to its importance, there have been different works dedicated to QAI (Wang et al., 2010; Khan et al., 2020; Du and Cardie, 2018). However, one limitation of the existing work is that they are evaluated only on formal text such as news, web-blogs or books (Richardson et al., 2013; Hermann et al., 2015; Rajpurkar et al., 2016; Joshi et al., 2017; He et al., 2018; Rogers et al., 2021). As such, the challenges of QAI in informal and noisy text such as video transcripts are under explored. To fill this gap, we present the first large-scale human-annotated QAI dataset for the domain of video transcripts.

## 5. Conclusion

In this work, we present a new dataset, i.e., BehanceQA, for identifying questions and answers in video transcripts. This is a new domain that has been neglected in prior QAI research. To this end, we collect 298 videos from the Behance platform which contains streamed videos for creative image editing. The collected data, consisting of more than 220,000 transcribed sentences, is manually labeled with questions and answers. The dataset provides more than 16,000 questions and 5,000 answers. We also conduct thorough experiments to study the challenges of QAI in the proposed dataset. Our analysis shows that while joint modeling has better performance for answer detection, a pipeline model performs better for identifying both questions and answers. Finally, we conduct an analysis to study the poor performance for answer detection in BehanceQA which can provide suggestions for future research. In the future, we plan to extend our dataset to include other NLP tasks for video transcripts.

## Ethical Consideration

In this work we present a dataset on the transcripts of a publicly accessible video-streaming platform, i.e., "*Behance*"[2]. Complying with the discussion presented by (Benton et al., 2017), research with human subjects information is exempted from the required full Institutional Review Board (IRB) review if the data is already available from public sources or if the identity of the subjects cannot be recovered. However, to protect the identity of the streamers and any other people whose

[2] www.behance.net

information are shared in the video transcripts, we impose extra processing on the transcribed documents before presenting them to annotators and publicly releasing it later. First, in this dataset, we remove username or any other identity-related information of the streamers in the transcripts to prevent disclosing their identity. In addition, to reduce the risk of disclosing the information of other people in the transcripts, in the final version of the dataset, we exclude the transcripts that explicitly or implicitly refer to the identify of the target people. Finally, we will only provide textual data (i.e., transcript documents and annotation information) to the annotators and later users, hence the other content of the videos (e.g., images, audios) are not revealed to protect human identity.

## Acknowledgement

This research has been supported by the Army Research Office (ARO) grant W911NF-21-1-0112 and the NSF grant CNS-1747798 to the IUCRC Center for Big Learning. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ARO, the Department of Defense, or the U.S. Government. This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

## Bibliographical References

Benton, A., Coppersmith, G., and Dredze, M. (2017). Ethical research protocols for social media health research. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 94–102, Valencia, Spain, April. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Du, X. and Cardie, C. (2018). Harvesting paragraph-level question-answer pairs from wikipedia. In *arXiv preprint arXiv:1805.05942*.

He, W., Liu, K., Liu, J., Lyu, Y., Zhao, S., Xiao, X., Liu, Y., Wang, Y., Wu, H., She, Q., Liu, X., Wu, T., and Wang, H. (2018). DuReader: a Chinese machine reading comprehension dataset from real-world applications. In *Proceedings of the Workshop on Machine Reading for Question Answering*. Association for Computational Linguistics.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching machines to read and comprehend. volume 28, pages 1693–1701.

Joshi, M., Choi, E., Weld, D., and Zettlemoyer, L. (2017). TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Khan, A., Ibrahim, I., Uddin, M. I., Zubair, M., Ahmad, S., Firdausi, A., Dzulqarnain, M., and Zaindin, M. (2020). Machine learning approach for answer detection in discussion forums: an application of big data analytics. In *Scientific Programming*, volume 2020. Hindawi.

Lai, V. D., Veyseh, A. P. B., Dernoncourt, F., and Nguyen, T. H. (2022). Behancepr: A punctuation restoration dataset for livestreaming video transcript. In *Findings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Li, B., Si, X., Lyu, M. R., King, I., and Chang, E. Y. (2011). Question identification on twitter. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2477–2480.

Rachha, A. and Vanmane, G. (2020). Detecting insincere questions from text: A transfer learning approach. In *arXiv preprint arXiv:2012.07587*.

Rajpurkar, P., Zhang, J., Lopyrev, K., and 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*. Association for Computational Linguistics.

Richardson, M., Burges, C. J., and Renshaw, E. (2013). MCTest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, Seattle, Washington, USA, October. Association for Computational Linguistics.

Rogers, A., Gardner, M., and Augenstein, I. (2021). Qa dataset explosion: A taxonomy of nlp resources for question answering and reading comprehension. In *arXiv preprint arXiv:2107.12708*.

Sakata, W., Shibata, T., Tanaka, R., and Kurohashi, S. (2019). Faq retrieval using query-question similarity and bert-based query-answer relevance. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1113–1116.

Shrestha, L. and McKeown, K. (2004). Detection of question-answer pairs in email conversations. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 889–895.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France, April. Association for Computational Linguistics.

Wang, B., Wang, X., Sun, C.-J., Liu, B., and Sun, L. (2010). Modeling semantic relevance for question-answer pairs in web social communities. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1230–1238.

West, R., Gabrilovich, E., Murphy, K., Sun, S., Gupta, R., and Lin, D. (2014). Knowledge base completion via search-based question answering. In *Proceedings of the 23rd international conference on World wide web*, pages 515–526.