Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 4701–4707
Marseille, 20-25 June 2022

Building a Dataset for Automatically Learning to Detect Questions Requiring Clarification

Ivano Lauriola, Kevin Small, Alessandro Moschitti
Amazon Alexa Web Information
Manhattan beach, CA, United States
{lauivano, smakevin, amosch}@amazon.com

Abstract

Question Answering (QA) systems aim to return correct and concise answers in response to user questions. QA research generally assumes all questions are intelligible and unambiguous, which is unrealistic in practice as questions frequently encountered by virtual assistants are ambiguous or noisy. In this work, we propose to make QA systems more robust via the following two-step process: (1) classify if the input question is intelligible and (2) for such questions with contextual ambiguity, return a clarification question. We describe a new open-domain clarification corpus containing user questions sampled from Quora, which is useful for building machine learning approaches to solving these tasks.

Keywords: Question Answering, Conversational, Clarification

1. Introduction

QA is a widely-studied research area aimed at automatically generating a correct and concise answer to an input question. Over the past three decades, several QA corpora and corresponding machine learning methods have been developed. However, available datasets generally focus on well-formed and understandable questions such that an answer can be manually produced without additional context (Voorhees and Tice, 2000; Yang et al., 2015; Yao et al., 2013; Rajpurkar et al., 2016; Kwiatkowski et al., 2019). However, these datasets do not meet the requirements for building commercial virtual assistants, where the input traffic often includes unintelligible or ambiguous questions due to reasons including automatic speech recognition errors, questions requiring commonsense reasoning, and peculiarities of speech in a live conversation.

After preliminary analyses of real user-generated traffic from a popular virtual assistant, we estimate that about 8.2% of the asked questions are unintelligible and more than 5% require clarification. Also, In our manual analysis we found that only a small percentage of unintelligible or ambigiuous questions received an acceptable answer, and mainly this was obtained by chance. Given the importance of these questions in practice, existing works have proposed corpora to train models that can spot questions that are unintelligible or questions requiring clarification (QRC) and produce a crowd-sourced follow-up question for further information when appropriate. However, these works have limitations for commercial requirements including a single domain focus (Li et al., 2016; Rao and Daumé, 2018), a generation process that does not reflect realistic use cases (Guo et al., 2017; Xu et al., 2019), or they are of limited size (De Boni and Manandhar, 2003; Stoyanchev et al., 2014; Aliannejadi et al., 2019). A relevant resource is MIMICS (Zamani et al., 2020): a collection of search clarification datasets sampled from the Bing logs. However, our task significantly differs from those related MIMICS as (i) we consider questions reflecting the interactions with a virtual assistant, which differ from queries used in search engines, and (ii) MIMICS only focuses on selection/generation of the clarification questions, which is only a fraction of the system we target.

To address these challenges, we designed and collected a new question classification corpus containing 29,869 annotated questions. Each question has multiple labels, including:

- Intelligible or unintelligible - is the input question understandable, clear, and comprehensible? Does it contain critical grammar error compromising the meaning or its intent?
- Require clarification (QRC) - is the input question ambiguous? Do we need a clarification to correctly provide an answer?
- Follow-up - for QRC, up to three follow-up questions are available.

The dataset is designed to help the train of models and the development of conversational QA systems for virtual assistants. To this end, we consider a system that, given an input question, (i) analyzes and understands the question (is intelligible?), (ii) evaluates possible ambiguities (does it require a clarification?), and (iii) provides an output. The latter can be either the answer or a follow-up question if the input is identified as QRC. Some examples of questions and their annotations are shown in Table 1.

As QRC are rare, we use a selective-sampling based approach to find these examples to mitigate annotation effort. We stress the fact that producing a dataset both sharable with the research community and representative of real-world questions is particularly challenging (e.g., we cannot use Alexa traffic for data collection due

Question	Is IN.	QRC	Follow-up
Who is the prime minister?	Yes	Yes	The prime minister of which country?
Who is the Italian prime minister?	Yes	No	-
Who won the Nobel Prize?	Yes	Yes	The Nobel in which field?
Who won the last Nobel in physics?	Yes	No	-
What does	No	-	-

Table 1: Examples of questions and annotations for: Is Intelligible (IN), requires clarification (QRC), and the follow-up question.

to privacy concerns). One important contribution of our work regards careful sampling of web questions that are unintelligible and/or requires clarification, where we use a selective sampling approach to obtain a sufficiently sized sample. For this purpose, we focus on data sourced from Quora, as this provides a very large database of available candidate questions, although our approach can be applied to any question collection. We propose to build classifiers using Transformer models fine-tuned on an initial, small question set. Then, we exploit such classifiers to rerank the Quora questions in terms of a score that estimates the need for questions to require clarification. This method allows us to sample a question set where the probability to find the target type of questions is much higher, thus reducing the effort of the annotators to find true positives. Given this sample, we also annotate possible clarification questions (if needed) and if the question is intelligible, for which, the clarification question should be different.

To validate our study, we train transformer-based classifiers for identifying unintelligible questions and QRC using this dataset. It should be noted that our experiments aim at providing (i) more insights on our corpus, e.g., showing the complexity of the tasks; and (ii) robust and reliable baselines for future research. Our results show that: (i) it is possible to detect a significant portion of QRC by just looking at the question (without using additional information); and (ii) our selective sampling approach can reduce the amount of effort required to build the desired corpus without introducing significant bias. These results suggest that our corpus, which we will release to the research community, is pontentialy very useful to advance the conversational QA research.

2. Related work

Recently, the detection of questions that require clarification and the generation of a clarification question has become increasingly popular tasks in the literature, and several corpora have been proposed.

One of the first corpora constructed for this purpose was proposed by De Boni and Manandhar (2003), consisting of 253 open-domain annotated questions from the *context* task in TREC-10 QA workshop. To the best of our knowledge, the aforementioned corpus was the first open-domain resource for clarification questions. More recently, Stoyanchev et al. (2014) collected questions using American English utterances from an open-domain speech-to-speech translation system (Akbacak et al., 2009), the IraqComm corpus. Although larger than the previous candidate, the corpus had only 794 questions.

Partially motivated by the limitations of these corpora, larger-scale ones have also been developed. Rao and Daumé (2018) introduced a corpus containing questions from StackExchange, an online forum where people post questions and knowledge about Operative Systems, and users intensively ask for clarifications (Q: *I have a problem x with my laptop.* A: *Which O.S. are you using?*). While this corpus consists of 77K questions, they are restricted to a single, narrow domain which limits wider applicability.

Similar corpora have been proposed by Guo et al. (2017) and Li et al. (2016), consisting of 100K and 180K questions respectively. Moreover, those corpora allow for training a classifier not only to discover incomplete questions that require clarification but also to formulate and to ask a clarification question, enabling a conversational approach to solving this problem. Aliannejadi et al. (2019) tackle the problem from an IR perspective, and they showed that asking a single clarification question leads to over 170% retrieval performance improvement in terms of P@1. The resulting corpus, Qulac, is publicly available and it consists of 2,639 questions from TREC Web Track 2009-2012. Finally, a Knowledge Base (KB) has been used by Xu et al. (2019) to solve ambiguities. In that case, a corpus consisting of 40K questions was released with the limitation that ambiguous questions were artificially generated from the KB.

As previously mentioned, a relevant resource in this field is is MIMICS (Zamani et al., 2020): a collection of search clarification datasets sampled from the Bing logs. However, our task significantly differs from those related MIMICS as (i) we consider questions reflecting the interactions with a virtual assistant, which differ from queries used in search engines, and (ii) MIMICS mainly focuses on selection/generation of the clarification questions.

To summarize, existing approaches for this task suffer from (i) limited amount of examples, (ii) limited domain (Movie, OS...), or (iii) the gap between the resource and real-world industrial applications (artificial or curated questions).

The definition of a conversational pipeline for QA based on clarification questions has been widely studied in the literature. Lautraite et al. (2021), for instance, described a system that, given an input questions, produces an answer. If the confidence score of the system is below a certain threshold, then the system asks the user if the answer is correct or it provides some suggestions or FAQ. Our system is significantly different as we firstly evaluate if a question can be answered. If so, the system retrieves and provides an answer. This strategy prevents the retrieval and the generation of the answer (the core of a QA system) if the input question cannot be easily answered, improving the efficiency.

3. Data collection

Based on observed limitations of existing resources, we require that: (i) the corpus must reflect the characteristics of human-virtual assistant interactions, and (ii) the annotation procedure should be economical and scalable. Specifically, identifying QRC and the generation of a follow-up question may be highly complex tasks requiring a relatively large amount of training data.

The first key ingredient for developing a clarification corpus is a source of open-domain raw questions. We discarded questions from existing QA or reading comprehension corpora for two reasons: (i) those corpora contain curated questions that do not meet our first requirement; and (ii) QA datasets have a limited quantity of available questions, which would limit the capability of our corpus. A suitable source for open-domain candidate questions is Quora, where questions are written by human users in a realistic setting and there is a virtually infinite supply.[1] To simplify our study, we considered the 4.7M of unlabelled questions from the Quora Kaggle challenge (Chen et al., 2018). It should be noted that, since we used Quora, which is a web forum, we do not consider unintelligibility caused by speech-transcription problems. We observed that, through a manual evaluation, the prevalence of QRC from Quora is notably low at 5-8%, thus a random sample will contain few QRC. To obtain a sufficient quantity of clarification questions economically, we designed an iterative annotation procedure based on crowd-sourcing and selective sampling[2] that prefers annotation of QRC. Specifically, each round of annotation requires that we: (i) train a classifier with available annotated questions; (ii) select a subset of diverse unlabeled questions with a high probability of being QRC; and (iii) annotate these questions via Amazon Mechanical Turk (AMT) as described in detail below.

Step 1 (Training) We begin by training a classifier to select a pool of candidate questions for annotation by attempting to identify intelligible QRC. Using a BERT (Devlin et al., 2019) pre-trained transformer fine-tuned on 90% of the available data using binary cross-entropy loss and the remaining 10% to terminate the learning procedure when the loss reaches a plateau. We used an *Internal QRC dataset* (IQD) to train the first classifier. The corpus consists of 3,022 annotated questions, out of which 2,774 were intelligible. Only 147 intelligible questions require clarification (i.e., 5% of the initial pool). The details of the corpus are introduced in the next paragraphs. After a preliminary experimentation phase we observed that several unintelligible questions are predicted as QRC. In order to mitigate this issue, we decided to consider unintelligible questions as negative examples.

Step 2 (Ranking) We classify all the available Quora questions with the model produced in Step 1, and use its scores to rank the questions. A higher rank indicates an increased likelihood that the question is intelligible and may require clarification. However, noting that, as the initial set was very small, the top-ranked questions were frequently similar to each other and thus capturing very few topics. To improve the selected questions, we designed a simple word-matching filter: we discard a question if it shares two or more non-function word tokens with a question preceding it in the ranking, limiting the comparison with the 100 preceding questions for computational reasons. We use this ranking to select the top N candidates, according to the budget set for this annotation task.

Step 3 (Annotation) Once the examples are selected for annotation, we designed a Human Intelligence Task (HIT) on AMT. This is a partially subjective task where different annotators may provide different answers. To this end, we collected three different annotations for each HIT. Given an input question, each HIT consists of three key steps:

1. the annotator is asked if the question is intelligible or not ("*Is the question clear and comprehensible?*");
2. In the affirmative case, the annotator is asked if the input question requires clarification or not ("*Does the question require clarification?*");
3. if a clarification is required the annotator writes a clarification question ("*If the question requires a clarification, please write a follow-up question useful to solve the ambiguity*").

We required annotators with an acceptance rate of 90% on other AMT tasks and "good English skills" to be eligible, and we discarded annotators (and associated HITs) if they do not follow the exact procedure (e.g., providing a clarification question for an unintelligible

[1] About 67 million as of March 2020

[2] Note that we do not use the term *active learning*, as the goal here is to sample the data to produce a reusable annotation dataset and not necessarily improve the classifier, which is more closely related to active search (Ma et al., 2015).

	IQD	Quora 1	Quora 2	Quora 3
Questions	3,022	4,896	19,992	4,980
HITs	3,022	14,688	59,914	14,940
Intelligible	2,774 91.8%	7,598 51.72%	49,591 82.77%	8,333 55.77 %
Requires clarification	147 5.10%	1,407 9.58%	8,103 **13.53%**	1,186 7.94%
Data used to train ranker	-	IQD	Quora 1	random

Table 2: Statistics regarding collected dataset batches.

question). We trained the turkers with multiple examples of HITs. Some of them are shown in Table 1. The compensation was set to 0,10$ per annotation.

We repeated the annotation procedure two times, collecting two different batches with 4,896 and 19,992 questions, consisting of 14,688 and 59,914 HIT annotations, respectively. Finally, we collect a third random batch of 4,980 questions to contrast the properties of the data collected with selective vs. random sampling. Our results show that selective sampling can improve the selection of QRC by +70% (from 7.94% random sample to 13.53%) when using a ranker trained on the first batch (4,896 questions). Statistics on the three batches and the initial seed are presented in Table 2. Specifically, the table shows the number of total questions and the number of annotations per batch, how many HITs marked questions as intelligible or require clarification, and the dataset used for selecting the sample.

IQD is a dataset developed internally with dedicated annotators for QA applications. Amongst other annotation instructions, we followed similar annotation guidelines as provided to the AMT workers for annotating Quora. However, this dataset is clearly more accurate as these are professional annotators trained for this task. After the IQD column, we see three different batches from Quora annotated by AMT workers. Their main difference is the classifier (and the data) used to select the pool of questions for the annotation. Questions from Quora 1 have been selected by a ranker trained on IQD, questions from Quora 2 have been selected by a ranker trained on Quora 1, and questions from Quora 3 have been randomly sampled.

We note that the percentage of clarification questions is notably small. Interestingly, if we use a classifier trained on a batch from Quora, we can almost double the percentage of QRC in the selected sample, from 7.94% (random) to 13.53% (classifier-based). Note that IQD and Quora show different distributions in terms of classes: although the percentage of intelligible QRC is similar, Quora contains a smaller percentage of intelligible questions, about 55.77% (see Quora 3). However, this difference depends on the peculiar sampling with which IQD was produced, thus it does not necessarily mean that Quora questions do not reflect the characteristics of human-virtual assistant interactions.

3.1. Error analysis

A further consideration concerns the quality of the produced data. We observed that the highest agreement between annotators occurs both with simple questions, such as “*What does a kitten eat?*” (Intelligible) or “*How should I prepare for tests?*” (QRC: which tests are you talking about?), or question containing evident semantic issues, e.g., “*Whose team and not c+?*” (Unintelligible). However, we also spot various annotation errors. For instance, the annotators agreed that the question “*How big is 25 feet?*” is intelligible, but they did not ask for a clarification question such as “*Compared to what?*”.

Table 3 shows some examples of questions from our collected corpus and the associated annotations for the Is Intelligible and QRC tasks.

Specifically, the table emphasizes three groups of questions. The first group (A1/2/3) describes questions with optimal alignment between 3 different annotators. The first two questions, i.e.: A1 - “*How do I make it happen?*” and A2 - “*Does bones decompose?*”, are clearly intelligible and well-formed. However, only the first one (A1) requires a clarification question because the topic/context of the discussion is not clear. Differently, A2 is extremely specific and does not require clarification.

The second group (U1/2/3) contains questions without agreement between annotators. Let us consider the question U1 - “*How do I remove subtitles from a episode?*”. Even if the question is, apparently, easily understandable, it is unintelligible for one annotator.

Moreover, notwithstanding the other two annotators did not ask for a clarification, we believe that the question contains a relevant ambiguity depending on the service (and its GUI) that is being used to watch the episode. Is the user watching a movie on Amazon

ID	Question	Is IN.	QRC
A1	How do I make it happen?	3	3
A2	Does bones decompose?	3	0
A3	Why do across have human rights?	0	0
U1	How do I remove subtitles from a episode?	2	0
U2	Did a therapist change your?	1	0
U3	How could I present a new product to the customer?	2	2
E1	How does monkey die?	0	0
E2	Is Coke Zero better than Coke?	3	1
E3	Why is empty set a subset of every set?	1	0

Table 3: Examples of questions and annotations. Each question receives 3 different annotations. The values exposed indicate how many annotators marked the question as Is Intelligible and QRC.

Prime Video, Netflix, or something else? Similarly, U3 - "*How could I present a new product to the customer?*" is not intelligible for one annotator. However, differently from the previous case, the two remaining annotators asked for clarification. As you can see, most of these annotation errors may be easily mitigated by the adoption of a majority agreement rule.

Differently from these cases, the last group of questions (E1/2/3) shows examples of annotations whose errors may significantly affect the overall quality of the corpus. In these cases, annotators have a good agreement in favor of a wrong class. For instance, the question E1 - "*How does monkey die?*" is unreasonably annotated as unintelligible even if its intent is clear. Focusing on question E2 - "*Is Coke Zero better than Coke?*", we agree with the annotators concerning the intelligibility, but we believe that the same is a QRC. Does *better* refer to the taste, the amount of sugar, or something else? Question E3 - "*Why is empty set a subset of every set?*" has a further problem given by a lack of (mathematical) knowledge.

E3 is a classical question that requires mathematical competence. The question is clearly intelligible and it does not require a clarification, but two annotators marked the question as unintelligible.

After a manual evaluation on the collected dataset, we believe that most of annotation errors are mitigated by the majority class.

3.2. Data split

Finally, We aggregated the three batches that we annotated and we randomly split them into training, development, and test sets. We use two different annotator agreement rules to determine labels associated with each classifier:

Majority rule Each label is assigned according to the majority vote of the three annotators. A given question is Intelligible if and only if at least 2 out of 3 annotators marked the question as intelligible. The same holds for QRC. For instance, Question U3 from Table 3 is considered Intelligible and QRC.

Total agreement Questions for which there is no complete agreement between annotators are discarded. Questions U1, U2, U3, E2, and E3 are, for instance, discarded.

Details regarding the dataset splits and agreement results are shown in Table 4 for each classification task.

4. Experiments

Our work mainly regards describing our proposed resource. For this purpose, we carried out experiments to study the effectiveness of our corpus when used for training state-of-the-art neural models (i.e., Transformer-based models) and to provide a reliable baseline for future research. Specifically, we consider developing classifiers that could be useful for implementing the following workflow of a virtual assistant. First, the customer asks the QA system for a general question. In response, the system analyzes the question. If it is not intelligible, then it asks the customer for rephrasing the question. Otherwise, it evaluates the question and decides if it can be answered. In the affirmative case, the standard QA system is run to provide an answer. Otherwise, the system produces a follow-up question asking for clarification, collects the new information, and answers the question.

In this paper, we focused on the classifiers required to recognize QRC and Intelligible questions. This allows us to provide a reliable baseline for these two tasks for future development. At the moment of writing, we do not provide results concerning the generation of the follow-up questions as the data collected for that specific task may not be sufficient without other annotation iterations.

We built two different binary classifiers based on pre-trained Transformer models (Vaswani et al., 2017). The first task consists of a binary classification problem, where the input questions are classified into two classes, *intelligible* and *unintelligible*. We started from a pre-trained RoBERTa (Liu et al., 2019) model and fine-tuned it on the training set using: a 10^{-6} learning rate, binary cross-entropy loss, the linear warm-up scheduler, and early stopping applied to our validation set. Secondly, we built the *requires clarification* classifier with the same approach as before using RoBERTa pre-trained model. In this case, we only use intelligible questions. Although the task of identifying QRC is part of the pipeline, here we consider the two tasks individually to better emphasize their complexity and peculiarities. The development of a complete QA pipeline is out of the scope of this paper.

The results are reported in Table 5. We note that the *intelligible* classifier can achieve a good F1 (i.e., 69.74 and 77.63, according to majority or total agreement labeling, respectively) on the Quora dataset. This can be a very useful classifier as it can avoid unnecessary computation within QA systems when there is lim-

Intelligible

Rule	Training	Development	Test
MR	25,000 (18,402)	2,270 (1,704)	2,500 (1,878)
TA	16,628 (13,811)	1,486 (1,262)	1,692 (1,435)

Requires clarification

Rule	Training	Development	Test
MR	18,402 (1,488)	1,704 (125)	1,878 (164)
TA	12,214 (207)	1,075 (18)	1,226 (23)

Table 4: Number of instances in the training, development and test sets for the *intelligible* and *questions requiring clarification (QRC)* tasks. Positive examples are shown in parentheses. MR=Majority Rule, TA=Total Agreement.

ited opportunity to get an answer right. However, the same classifier does not generalize well when applied to IQD data as the F1 drastically drops from 69.74 to 40.22 and from 77.63 to 38.39, depending on the labeling rule. Not surprisingly, the test on Quora + IQD reveals that fine-tuning on the target domain data is always recommended: we can improve the F1 by 17% (absolute) by simply using a small amount of data from IQD. Additionally, we evaluated the QRC classifier with AUC score since it can provide a more reliable performance indicator when the label distribution is relatively skewed as in the case of QRC classification (AUC is independent of threshold values). The results show that it is possible to achieve promising performance, e.g., 69.07 and 73.32, on the Quora dataset. Interestingly, we observe a small drop in performance (6-7% AUC) when testing the same classifier on IQD data. This result suggest that the classifier trained on Quora can be effectively applied to IQD, and the poor F1 computed on IQD is attributable to a bad classification threshold.

Similar to the previous experiment, fine-tuning on the target data highly improves the model, achieving AUC of 70.87 and 68.92.

Next, we conducted an experiment with the QRC classifier, fine-tuning and testing it exclusively on IQD (i.e., IQD → IQD), achieving an AUC of 58.05. This result clearly shows that, differently from the *intelligible* task, the adoption of the larger Quora dataset is valuable in practice, and it can be effectively used to improve (+11/12% AUC) the performance of our QA system with a limited annotation cost. Our corpus can represent a realistic human-machine interaction that may happen with virtual assistants. This is important as the IQD annotation is more costly to produce, while our cheaper annotation can enable the design of systems using automatic detection of QRC.

A final consideration regarding the labeling rules. The total agreement produces consistent annotations able to capture clear cases of unintelligible or QRC. Thus, the models tested on domain data, i.e., Quora test set, benefit from such consistent rule. However, when we use an out domain test set, i.e., IQD, the models trained with total agreement do not show a clear superiority with respect to those trained with the majority rule. This is expected as the data and annotation differ in out domain testing, thus less rigid rules could more probably match the annotation between IQD and Quora.

Is Intelligible (F1)

Rule	Q → Q	Q → IQD	Q + IQD → IQD
majority	69.74	40.22	57.13
agreement	77.63	38.39	55.21

Requires Clarification (AUC)

Rule	Q → Q	Q → IQD	Q + IQD → IQD
majority	69.07	63.45	70.87
agreement	73.32	66.25	68.92

Table 5: F1 and AUC of the intelligible and QRC classifiers derived on Quora (Q) or IQD test sets. The notation $X \rightarrow Y$ refers to models trained on X and tested on Y.

5. Conclusion

In this paper, we have presented a novel resource for question classification tasks that can enable the design of promising dialog scenarios. The corpus contains annotations concerning the comprehensibility of 29,869 questions, their need for clarification, and up to three follow-up questions, which can be used to solve ambiguities of human-machine conversations. Additionally, we designed a cheap and scalable annotation procedure to find questions that require a clarification (QRC), which are typically rare, about 5-8% of a random sample. Then, we trained state-of-the-art neural architectures on our proposed dataset, showing interesting insights and providing a reliable baseline for future development. Our experiments show (i) the complexity of classifying unintelligible questions and QRC, and (ii) the ability of our corpus to represent real traffic typical of virtual assistants. Finally, our study opens interesting research directions, ranging from using intelligible classifiers for reducing the QA service cost, to improving the interaction with customers by also enhancing the accuracy of the information the system provides them. The provided clarification questions can be used, jointly with other resources, to evaluate automatic clarification question generation, enabling simple human-machine conversations.

In the future, we plan to (i) scale up (10x) the dataset with additional annotation rounds on Amazon Mechanical Turk, (ii) develop a complete QA system based on the conversational strategy described in this paper, and (iii) delve into the generation of follow-up questions. We release our corpus to the research community[3].

6. Bibliographical References

Akbacak, M., Franco, H., Frandsen, M., Hasan, S., Jameel, H., Kathol, A., Khadivi, S., Lei, X., Mandal, A., Mansour, S., et al. (2009). Recent advances in sri's iraqcomm™ iraqi arabic-english speech-to-speech translation system. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4809–4812. IEEE.

Aliannejadi, M., Zamani, H., Crestani, F., and Croft, W. B. (2019). Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 475–484.

Chen, Z., Zhang, H., Zhang, X., and Zhao, L. (2018). Quora question pairs.

De Boni, M. and Manandhar, S. (2003). An analysis of clarification dialogue for question answering.

[3] https://github.com/alexa/wqa-clarification-corpus/

In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–55. 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 *NAACL-HLT*.

Guo, X., Klinger, T., Rosenbaum, C. M., Bigus, J. P., Campbell, M., Kawas, B., Talamadupula, K., Tesauro, G., and Singh, S. (2017). Learning to query, reason, and answer questions on ambiguous texts. In *ICLR*.

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Kelcey, M., Devlin, J., Lee, K., Toutanova, K. N., Jones, L., Chang, M.-W., Dai, A., Uszkoreit, J., Le, Q., and Petrov, S. (2019). Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*.

Lautraite, H., Naji, N., Marceau, L., Queudot, M., and Charton, E. (2021). Multi-stage clarification in conversational ai: The case of question-answering dialogue systems. *arXiv preprint arXiv:2110.15235*.

Li, J., Miller, A. H., Chopra, S., Ranzato, M., and Weston, J. (2016). Dialogue learning with human-in-the-loop. *ArXiv*, abs/1611.09823.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ma, Y., Huang, T.-K., and Schneider, J. G. (2015). Active search and bandits on graphs using sigma-optimality. In *Proc. of UAI*, pages 542–551.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the conference on empirical methods in natural language processing*.

Rao, S. and Daumé, H. (2018). Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information. In *ACL*.

Stoyanchev, S., Liu, A., and Hirschberg, J. (2014). Towards natural clarification questions in dialogue systems. In *AISB symposium on questions, discourse and dialogue*, volume 20.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Voorhees, E. M. and Tice, D. M. (2000). Building a question answering test collection.

Xu, J., Wang, Y., Tang, D., Duan, N., Yang, P., Zeng, Q., Zhou, M., and Sun, X. (2019). Asking clarification questions in knowledge-based question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1618–1629, Hong Kong, China, November. Association for Computational Linguistics.

Yang, Y., Yih, W.-t., and Meek, C. (2015). WikiQA: A challenge dataset for open-domain question answering. In *Proceedings of the conference on empirical methods in natural language processing*, pages 2013–2018.

Yao, X., Van Durme, B., Callison-Burch, C., and Clark, P. (2013). Answer extraction as sequence tagging with tree edit distance. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 858–867.

Zamani, H., Lueck, G., Chen, E., Quispe, R., Luu, F., and Craswell, N. (2020). Mimics: A large-scale data collection for search clarification. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 3189–3196.