Is that really a question?
Going beyond factoid questions in NLP

Aikaterini-Lida Kalouli and **Rebecca Kehlbeck** and **Rita Sevastjanova** and
Oliver Deussen and **Daniel Keim** and **Miriam Butt**
University of Konstanz
firstname.lastname@uni-konstanz.de

Abstract

Research in NLP has mainly focused on factoid questions, with the goal of finding quick and reliable ways of matching a query to an answer. However, human discourse involves more than that: it contains non-canonical questions deployed to achieve specific communicative goals. In this paper, we investigate this under-studied aspect of NLP by introducing a targeted task, creating an appropriate corpus for the task and providing baseline models of diverse nature. With this, we are also able to generate useful insights on the task and open the way for future research in this direction.

1 Introduction

Recently, the field of human-machine interaction has seen ground-breaking progress, with the tasks of Question-Answering (QA) and Dialog achieving even human-like performance. The probably most popular example is *Watson* (Ferrucci et al., 2013), IBM's QA system which was able to compete on the US TV program *Jeopardy!* and beat the best players of the show. Since then and particularly with the rise of Neural Networks (NN), various high-performance QA and Dialog systems have emerged. For example, on the QQP task of the GLUE benchmark (Wang et al., 2018), the currently best performing system achieves an accuracy of 90.8%. Despite this success, current QA and Dialog systems cannot be claimed to be on a par with human communication. In this paper we address one core aspect of human discourse that is under-researched within NLP: non-canonical questions.

Research in NLP has mainly focused on factoid questions, e.g., *When was Mozart born?*, with the goal of finding quick and reliable ways of matching a query to terms found in a given text collection. There has been less focus on understanding the structure of questions per se and the communicative goal they aim to achieve. State-of-the-art parsers are mainly trained on Wikipedia entries or newspaper texts, e.g., the Wall Street Journal, genres which do not contain many questions. Thus, the tools trained on them are not effective in dealing with questions, let alone distinguishing between different types. Even within more computational settings that include deep linguistic knowledge, e.g., PARC's Bridge QA system (Bobrow et al., 2007) which uses a sophisticated LFG parser and semantic analysis, the actual nature and structure of different types of questions is not studied in detail.

However, if we are aiming at human-like NLP systems, it is essential to be able to efficiently deal with the fine nuances of non-factoid questions (Dayal, 2016). Questions might be posed

- as a (sarcastic, playful) comment, e.g., *Have you ever cooked an egg?* (rhetorical)
- to repeat what was said or to express incredulity/surprise, e.g., *He went where?* (echo)
- to make a decision, e.g., *What shall we have for dinner?* (deliberative)
- to deliberate rather than ask or to rather ask oneself than others, e.g., *Do I even want to go out?* (self-addressed)
- to request or order something, e.g., *Can you pass me the salt?* (ability/inclination)
- to suggest that a certain answer should be given in reply, e.g., *Don't you think that calling names is wrong?* (suggestive)
- to assert something, e.g., *You are coming, aren't you?* (tag)
- to quote the words of somebody else, e.g., *And he said, "Why do you bother?"* (quoted)
- to structure the discourse, e.g., *What has this taught us? It ...* (discourse-structuring)
- etc.

Proceedings of the 14th International Conference on Computational Semantics, pages 132–143
June 17–18, 2021. ©2021 Association for Computational Linguistics

The importance of these communicative goals in everyday discourse can be seen in systems like personal assistants, chatbots and social media. For example, personal assistants like Siri, Alexa and Google should be able to distinguish an ability question of the kind *Can you play XYZ?* from a rhetorical question such as *Can you be even more stupid?* Similarly, chatbots offering psychotherapeutic help (Ly et al., 2017; Håvik et al., 2019) should be able to differentiate between a factoid question such as *Is this a symptom for my condition?* and a self-addressed question, e.g., *Why can't I do anything right?* In social media platforms like Twitter, apart from the canonical questions of the type *Do you know how to tell if a brachiopod is alive?*, we also find non-canonical ones like *why am I lucky?* Paul et al. (2011) show that 42% of all questions on English Twitter are rhetorical.

To enable NLP systems to capture non-factoid uses of questions, we propose the task of *Question-Type Identification* (QTI). The task can be defined as follows: given a question, determine whether it is an information-seeking question (ISQ) or a non information-seeking question (NISQ). The former type of question, also known as a canonical or factoid question, is posed to elicit information, e.g., *What will the weather be like tomorrow?* In contrast, questions that achieve other communicative goals are considered non-canonical, non-information-seeking. NISQs do not constitute a homogeneous class, but are heterogeneous, comprising sub-types that are sometimes difficult to keep apart (Dayal, 2016). But even at the coarse-grained level of distinguishing ISQs from NISQs, the task is difficult: surface forms and structural cues are not particularly helpful; instead, Bartels (1999) and Dayal (2016) find that prosody and context are key factors in question classification.

Our ultimate objective in this paper is to provide an empirical evaluation of learning-centered approaches to QTI, setting baselines for the task and proposing it as a tool for the evaluation of QA and Dialog systems. However, to the best of our knowledge, there are currently no openly available QTI corpora that can permit such an assessment. The little previous research on the task has not contributed suitable corpora, leading to comparability issues. To address this, this paper introduces RQueT (*rocket*), the Resource of Question Types, a collection of questions in-the-wild labeled for their ISQ-NISQ type. As the first of its kind, the resource of 2000 annotated questions allows for initial machine-/deep-learning experimentation and opens the way for more research in this direction.

In this paper, we use this corpus to evaluate a variety of models in a wide range of settings, including simple linear classifiers, language models and other neural network architectures. We find that simple linear classifiers can compete with state-of-the-art transformer models like BERT (Devlin et al., 2019), while a neural network model, combining features from BERT and the simple classifiers, can outperform the rest of the settings.

Our contributions in this paper are three-fold. First, we provide the first openly-available QTI corpus, aiming at introducing the task and comprising an initial benchmark. Second, we establish suitable baselines for QTI, comparing systems of very different nature. Finally, we generate linguistic insights on the task and set the scene for future research in this area.

2 Relevant Work

Within modern theoretical linguistics, a large body of research exists on questions. Some first analyses focused on the most well-known types, i.e., deliberative, rhetorical and tag questions (Wheatley, 1955; Sadock, 1971; Cattell, 1973; Bolinger, 1978, to name only a few). Recently, researchers have studied the effect of prosody on the type of question as well as the interaction of prosody and semantics on the different types (Bartels, 1999; Dayal, 2016; Biezma and Rawlins, 2017; Beltrama et al., 2019; Eckardt, 2020, to name a few). It should also be noted that research in developing detailed pragmatic annotation schemes for human dialogs, thus also addressing questions, has a long tradition, e.g., Jurafsky et al. (1997); Novielli and Strapparava (2009); Bunt et al. (2016); Asher et al. (2016). However, most of this work is too broad and at the same time too fine-grained for our purposes: on the one hand, it does not focus on questions and thus these are not studied in the desired depth and on the other, the annotation performed is sometimes too fine-grained for computational approaches. Thus, we do not report further on this literature.

In computational linguistics, questions have mainly been studied within QA/Dialog systems, (e.g., Alloatti et al. (2019); Su et al. (2019)), and within Question Generation, (e.g., Sasazawa et al. (2019); Chan and Fan (2019)). Only a limited amount of research has focused on (versions of) the QTI task. One strand of research has used social media data – mostly Twitter – training simple classifier models (Harper et al., 2009; Li et al., 2011; Zhao and Mei, 2013; Ranganath et al., 2016). Although this body of work reports on interesting methods and findings, the research does not follow a consistent task definition, analysing slightly different things that range from "distinguishing informational and conversational questions", "analysis of information needs on Twitter" to the identification of rhetorical questions. Additionally, they do not evaluate on a common dataset, making comparisons difficult. Furthermore, they all deal with social media data, which, despite its own challenges (e.g., shortness, ungrammaticality, typos), is enriched with further markers like usernames, hashtags and urls, which can be successfully used for the classification. A different approach to the task is pursued by Paul et al. (2011), who crowdsources human annotations for a large amount of Twitter questions, without applying any automatic recognition. More recently, the efforts by Zymla (2014), Bhattasali et al. (2015) and Kalouli et al. (2018) are more reproducible. The former develops a rule-based approach to identify rhetorical questions in German Twitter data, while Bhattasali et al. (2015) implements a machine-learning system to identify rhetorical questions in the Switchboard Dialogue Act Corpus. In Kalouli et al. (2018) a rule-based multilingual approach is applied on a parallel corpus based on the Bible.

3 RQueT: a New Corpus for QTI

The above overview of relevant work indicates that creating suitable training datasets is challenging, mainly due to the sparsity of available data. Social media data can be found in large numbers and contains questions of both types (Wang and Chua, 2010), but often the context in which the questions are found is missing or very limited, making their classification difficult even for humans. On the other hand, corpora with well-edited text such as newspapers, books and speeches are generally less suitable, as questions, in particular NISQs, tend to appear more often in spontaneous, unedited communication. Thus, to create a suitable benchmark, we need to devise a corpus fulfilling three desiderata: a) containing naturally-occurring data, b) featuring enough questions of both types, and c) providing enough context for disambiguation.

3.1 Data Collection

To this end, we find that the CNN transcripts[1] fulfill all three desiderata. We randomly sampled 2000 questions of the years 2006–2015, from settings featuring a live discussion/interview between the host of a show and guests. Questions are detected based on the presence of a question mark; this method misses the so-called "declarative" questions (Beun, 1989), which neither end with a question mark nor have the syntactic structure of a question, but this compromise is necessary for this first attempt on a larger-scale corpus. Given the importance of the context for the distinction of the question types (Dayal, 2016), along with the question, we also extracted two sentences before and two sentences after the question as context. For each of these sentences as well as for the question itself, we additionally collected speaker information. Table 1 shows an excerpt of our corpus. Unfortunately, due to copyright reasons, we can only provide a shortened version of this corpus containing only 1768 questions; this can be gained via the CNN transcripts corpus made available by Sood (2017).[2] The results reported here concern this subcorpus, but we also provide the results of the entire corpus of 2000 questions in Appendix A. Our corpus is split in a 80/20 fashion, with a training set of 1588 and a test set of 180 questions (or 1800/200 for the entire corpus, respectively).

3.2 Data Annotation

The RQueT corpus is annotated with a binary scheme of ISQ/NISQ and does not contain a finer-grained annotation of the specific sub-type of NISQ. We find it necessary to first establish the task in its binary formulation. Each question of our corpus was annotated by three graduate students of computational linguistics. The annotators were only given the definition of each type of question and an example, as presented in Section 1, and no further instructions. The lack of more detailed instructions was deliberate: for one, we wanted to see how easy and intuitive the task is for humans given that they perform it in daily communication. For another, to the best of our knowledge, there are no previous annotation guidelines or best-practices available.

The final label of each question was determined by majority vote, with an inter-annotator agreement of 89.3% and Fleiss Kappa at 0.58. This moderate

[1] http://transcripts.cnn.com/TRANSCRIPTS/

[2] See https://github.com/kkalouli/RQueT

Sentence	Text	Speaker	QT
Ctx 2 Before	*This is humor.*	S. BAXTER	
Ctx 1 Before	*I think women, female candidates, have to be able to take those shots.*	S. BAXTER	
Question	*John Edwards got joked at for his $400 hair cut, was it?*	S. BAXTER	NISQ
Ctx 1 After	*And you know, he was called a Brett Girl.*	S. BAXTER	
Ctx 2 After	This, is you know, the cut and thrust of politics.	S. BAXTER	

Table 1: Sample of the corpus format. Each row contains a sentence and its context before and after. The question and its context also hold the speaker information. Each question is separately annotated for its type.

agreement reflects the difficulty of the task even for humans and hints at the improvement potential of the corpus through further context, e.g., in the form of intonation and prosody (see e.g., Bartels 1999). The resulting corpus is an (almost) balanced set of 944 (1076 for the entire corpus) ISQ and 824 (924 for the entire corpus) NISQ. The same balance is also preserved in the training and test splits. Table 2 gives an overview of RQueT.

4 RQueT as a Benchmarking Platform

We used the RQueT corpus to evaluate a variety of models,[3] establishing appropriate baselines and generating insights about the nature and peculiarities of the task.

4.1 Lexicalized and Unlexicalized Features

Following previous literature (Harper et al., 2009; Li et al., 2011; Zymla, 2014; Bhattasali et al., 2015; Ranganath et al., 2016) and our own intuitions, we extracted 6 kinds of features, 2 lexicalized and 4 unlexicalized, a total of 16 distinct features:

1. lexicalized: bigrams and trigrams of the surface forms of the question itself (*Q*), of the context-before (*ctxB1* and *ctxB2*, for the first and second sentence before the question, respectively) and of the context-after (*ctxA1* and *ctxA2*, for the first and second sentence after the question, respectively)
2. lexicalized: bigrams and trigrams of the POS tags of the surface forms of the question itself (*Q*), of the context-before (*ctxB1*, *ctxB2*) and of the context-after (*ctxA1* and *ctxA2*)
3. unlexicalized: the length difference between the question and its first context-before (*lenDiffQB*) and the question and its first context-after (*lenDiffQA*), as real-valued features
4. unlexicalized: the overlap between the words in the question and its first context-before/after, both as an absolute count (*wOverBAbs* and *wOverAAbs* for context before/after, respectively) and as a percentage (*wOverBPerc* and *wOverAPerc* for context before/after, respectively)
5. unlexicalized: a binary feature capturing whether the speaker of the question is the same as the speaker of the context-before/after (*speakerB* and *speakerA*, respectively)
6. unlexicalized: the cosine similarity of the InferSent (Conneau et al., 2017) embedding of the question to the embedding of the first context-before/after[4] (*similQB* and *similQA*, respectively).

	ISQ	**NISQ**	**All**
Train	847 (969)	741 (831)	1588 (1800)
Test	97 (107)	83 (93)	180 (200)
Total	944 (1076)	824 (924)	1768 (2000)

Table 2: Distribution of question type in the shortened and the entire RQueT corpus, respectively.

We used these feature combinations to train three linear classifiers for each setting: a Naive Bayes classifier (NB), a Support Vector Machine (SVM) and a Decision Tree (DT). These traditional classifiers were trained with the *LightSide* workbecnh.[5] The Stanford CoreNLP toolkit (Toutanova et al., 2003) was used for POS tagging.

4.2 Fine-tuning Pretrained BERT

Given the success of contextualized language models and their efficient modeling of semantic information, e.g., Jawahar et al. (2019); Lin et al. (2019), we experiment with BERT (Devlin et al., 2019) for this task. Since the semantic relations between the question and its context are considered the most significant predictors of QT, contextualized models

[3] https://github.com/kkalouli/RQueT

[4] Here we opt for the non-contextualized InferSent embeddings because contextualized embeddings like BERT inherently exhibit high similarities (Devlin et al., 2019).

[5] http://ankara.lti.cs.cmu.edu/side/

should be able to establish a clear baseline. The QTI task can be largely seen as a sequence classification task, much as Natural Language Inference and QA. Thus, we format the corpus into appropriate BERT sequences, i.e., question-only sequence or question – context-before or question – context-after sequence, and fine-tune the pretrained BERT (base) model on that input. We explicitly fine-tune the parameters recommended by the authors. The best models train for 2 epochs, have a batch size of 32 and a learning rate of 2e-5. By fine-tuning the embeddings, we simultaneously solve the QTI task, which is the performance we report on in this setting. The fine-tuning is conducted through *HuggingFace*.[6]

4.3 BERT Embeddings as Fixed Features

The fine-tuned BERT embeddings of Section 4.2 can be extracted as fixed features to initialize further classifier models (cf. Devlin et al. 2019). We input them to the same linear classifiers used in section 4.1, i.e., NB, SVM and DT, but also use them for neural net (NN) classifiers because such architectures are particularly efficient in capturing the high-dimensionality of these inputs. To utilize the most representative fine-tuned BERT embeddings, we experiment with the average token embeddings of layer 11 and the *[CLS]* embedding of layer 11. We chose layer 11 as the higher layers of BERT have been shown to mostly capture semantic aspects, while the last layer has been found to be very close to the actual classification task and thus less suitable (Jawahar et al., 2019; Lin et al., 2019). We found that the *[CLS]* embedding performs better and thus, we only report on this setting.

Moreover, as shown in Section 5, some of the unlexicalized features of Section 4.1 lead to competitive performance with the pretrained BERT models. Thus, we decided to investigate whether the most predictive unlexicalized feature can be efficiently combined with the BERT fine-tuned embeddings and lead to an even higher performance. To this end, each linear classifier and NN model was also trained on an *extended* vector, comprising the CLS-layer11 fine-tuned BERT embedding of the respective model, i.e., only of the question (*Q-Embedding*), of the question and its (first) context-before (*Q-ctxB-Embedding*) and of the question and its (first) context-after (*Q-ctxA-Embedding*) as a fixed vector, and an additional dimension for the binary encoded unlexicalized feature.

We experimented with three NN architectures and NN-specific parameters were determined via a grid search separately for each model. Each NN was optimized through a held-out validation set (20% of the training set). First, we trained a Multi-Layer Perceptron (MLP) with a ReLU activation and the Adam optimizer. Second, we trained a feed-forward (FF) NN with 5 dense hidden layers and the RMSprop optimizer. Last, we trained an LSTM with 2 hidden layers and the RMSprop optimizer. Both the FF and the LSTM use a sigmoid activation for the output layer, suitable for the binary classification. All NNs were trained with *sklearn*.

5 Results and Analysis

5.1 Quantitative Observations

The results of the training settings are presented in Table 3. Recall that these results concern the corpus of 1768 questions. The results on the entire corpus can be found in Appendix A. For space reasons, we only present the most significant settings and results. For the lexicalized features, all models use both the surface and the POS n-grams as their combination proved best — the separate settings are omitted for brevity, so e.g., *Q tokens/POS* stands for a) the question's bigrams and trigrams and b) the question's POS bigrams and trigrams. All performance reported in Table 3 represents the accuracy of the models.

The careful benchmarking presented in Table 3 allows for various observations. We start off with the diverse combinations of lexicalized and unlexicalized features. First, we see that training only on the question, i.e., on its n-grams and POS tags, can serve as a suitable baseline with an accuracy of 62.7% for NB. Adding the first context-before improves performance and further adding the second context-before improves it even further at 72.7% for NB. A similar performance leap is observed when the first context-after is added to the question (73.3% for NB), while further adding the second context-after does not change the picture. Since adding the first context-before and -after to the question increases accuracy, we also report on the setting where both first context-before and -after are added to the question. This does indeed boost the performance even more, reaching an accuracy of 75% for NB. Given that the second context-before is beneficial for the *Q+ctxB1+ctxB2* setting, we add it to the previously best model of 75%

[6] https://huggingface.co/

Lexicalized					Unlexicalized				BERT Embeds			Classifiers						
Q tokens/POS	**ctxB1 tokens/POS**	**ctxB2 tokens/POS**	**ctxA1 tokens/POS**	**ctxA2 tokens/POS**	**speakerB**	**speakerA**	**similQA**	**lenDiffA**	**Q-Embed**	**Q-ctxB-Embed**	**Q-ctxA-Embed**	**NB**	**SVM**	**DT**	**MLP**	**FF**	**LSTM**	**BERT Fine-Tuning**
✓												62.7	61.1	63.3	-	-	-	-
✓	✓											68.8	69.4	58.5	-	-	-	-
✓	✓	✓										72.7	70	61.1	-	-	-	-
✓			✓									73.3	65	66.1	-	-	-	-
✓			✓	✓								68.8	68.8	63.3	-	-	-	-
✓	✓		✓									**75**	62.7	62.7	-	-	-	-
✓	✓	✓	✓									66.1	66.6	58.5	-	-	-	-
✓	✓	✓	✓	✓								65	67.2	58.8	-	-	-	-
					✓							57.2	57.2	57.2	57.2	57.2	56.9	-
						✓						**77.7**	**77.7**	**77.7**	**77.7**	**77.7**	**77.7**	-
					✓	✓						77.7	77.7	77.7	77.7	77.7	77.7	-
						✓	✓	✓				77.7	77.7	77.7	77.7	77.7	77.7	-
✓	✓	✓			✓							73.3	69.4	61.1	-	-	-	-
✓			✓			✓						75	73.3	**76.1**	-	-	-	-
✓			✓		✓	✓						75.5	72.7	**76.1**	-	-	-	-
✓	✓		✓			✓						74.4	71.6	62.7	-	-	-	-
✓	✓	✓	✓			✓						67.2	76.1	75.5	-	-	-	-
✓	✓		✓			✓	✓					74.4	71.6	75.5	-	-	-	-
✓	✓		✓			✓		✓				74.4	71.6	75.5	-	-	-	-
									PT			-	-	-	-	-	-	76.1
										PT		-	-	-	-	-	-	78.3
											PT	-	-	-	-	-	-	80.1
									FN			**77.7**	72.7	72.7	71.1	75.5	75.5	-
						✓			FN			77.7	80	**83.8**	80	78.8	80	-
										FN		76.6	**82.2**	72.2	77.2	80	81.1	-
					✓					FN		76.6	**81.6**	72.2	81.1	80	78.3	-
											FN	83.3	**83.3**	77.7	81.1	82.7	76.6	-
						✓					FN	83.3	83.3	81.1	82.2	**84.4**	80	-
✓	✓		✓			✓			✓	✓	✓	**Ensemble: *88.3***						

Table 3: Accuracy of the various classifiers and feature combinations (settings). A checkmark means that this feature was present in this setting. *PT* stands for the pretrained BERT embeddings and *FN* for the fine-tuned ones. Bolded figures are the best performances across types of classifiers. The stared figure is the best performing ensemble model across settings. *wOverAbs* and *wOverPerc* are omitted for brevity.

and find out that their combination rather harms the accuracy. Experimenting with both contexts-before and -after and the question does not lead to any improvements either. The combinations of the lexicalized features show that the best setting is the one where the question is enriched by its first context-before and -after (75%).

We make a striking observation with respect to the unlexicalized features. Training only on the speaker-after, i.e., on whether the speaker of the question is the same as the speaker of the first context-after, and ignoring entirely the question and context representation is able to correctly predict the QT in 77.7% of the cases. This even outperforms the best setting of the lexicalized features. The speaker-before does not seem to have the same expressive power and training on both speaker features does not benefit performance either. We also find that the rest of the unlexicalized features do not have any impact on performance because training on each of them alone hardly outperforms the simple *Q tokens/POS* baseline, while by training on all unlexicalized features together we do not achieve better results than simply training on speaker-after.[7]

Based on the finding that the speaker-after is so powerful, we trained hybrid combinations of lexicalized features and the speaker information. First, the speaker-before is added to the *Q+ctxB1+ctxB2*, which is the best setting of contexts-before, but we do not observe any significant performance change. This is expected given that speaker-before alone does not have a strong performance. Then, the speaker-after is added to the setting *Q+ctxA1* and

[7] These settings are omitted from the table for brevity.

the performance reaches 76.1% (for DT), approaching the best score of speaker-after. The addition of speaker-before to this last setting does not improve performance. On the other hand, adding the speaker-after information to the best lexicalized setting (*Q+ctxB1+ctxA1*) does not have an effect, probably due to a complex interaction between the context-before and the speaker. This performance does not benefit either from adding the second context-before (which proved beneficial before) or adding the other unlexicalized features.[8]

Moving on, we employ the pretrained BERT embeddings to solve the QTI task. Here, we can see that the model containing the question and the context-after (*Q-ctxA-Embedding*) is the best one with 80.1%, followed by the model containing the question and the context-before (*Q-ctxB-Embedding*, 78.3). Worst-performing is the model based only on the question (*Q-Embedding*). This simple fine-tuning task shows that contextualized embeddings like BERT are able to capture the QT more efficiently than lexicalized and unlexicalized features – they even slightly outperform the powerful speaker feature. This means that utilizing these fine-tuned embeddings as fixed input vectors for further classifiers can lead to even better results, and especially, their combination with the predictive speaker information can prove beneficial.

In this last classification setting, we observe that the classifiers trained only on the fine-tuned BERT embeddings deliver similar performance to the fine-tuning task itself. This finding reproduces what is reported by Devlin et al. (2019). However, the real value of using this feature-based approach is highlighted through the addition of the speaker information to the contextualized vectors. The speaker information boosts performance both in the setting of *fine-tuned Q-Embedding* and in the setting *fine-tuned Q-ctxA-Embedding*. In fact, the latter is the best performing model of all with an accuracy of 84.4%. Adding the speaker-before information to the *fine-tuned Q-ctxB-Embedding* does not have an impact on performance due to the low impact of the speaker-before feature itself.

5.2 Qualitative Interpretation

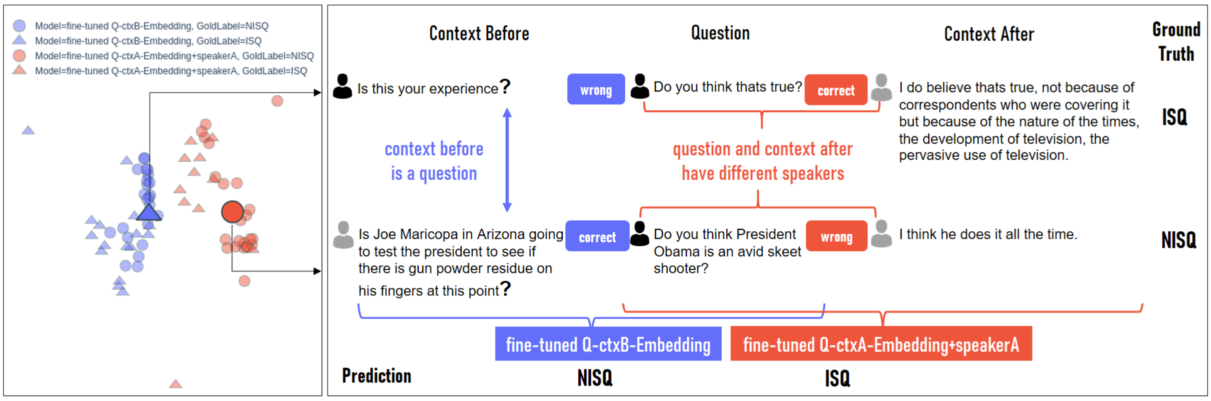
The results presented offer us interesting insights for this novel task. First, they confirm the previous finding of the theoretical and computational literature that context is essential in determining the question type. Both the lexicalized and the embeddings settings improve when context is added. Concerning the lexicalized settings, we conclude that the surface and syntactic cues present within the question and its first context-after are more powerful than the cues present within the question and the first context-before. This is consistent with the intuition that whatever follows a question tends to have a more similar structure to the question itself than whatever precedes it: no matter if the utterer of the question continues talking or if another person addresses the question, the attempt is to stay as close to the question as possible, to either achieve a specific communication goal or to actually answer the question, respectively. However, our experiments also show that combining the first context-before and -after with the question does indeed capture the most structural cues, generating the insight that one sentence before and after the question is sufficient context for the task at hand. Interestingly, we can confirm that the second context-after is not useful to the classification of the QT, probably being too dissimilar to the question itself. Table 4 shows examples of the most predictive structural cues for the best setting of the lexicalized classifiers (*Q+ctxB1+ctxA1*).

ISQ	*you_feel, what_do_you, do_you_agree, make_of_that, you_expect, me_ask_you, why_did_you, how_did_you*
NISQ	*why_arent't, and_should_we, COMMA_how_about, how_could, do_we_want, can_we*

Table 4: Structural features with the most influence in the model *Q+ctxB1+ctxA1*.

Training on non-linguistic unlexicalized features does not boost performance. However, our work provides strong evidence that the speaker meta-information is of significant importance for the classification. This does not seem to be a peculiarity of this dataset as later experimentation with a further English dataset and with a German corpus shows that the speaker information is consistently a powerful predictor. Additionally, we can confirm from Appendix A that the speaker feature has the same behavior, when trained and tested on the entire corpus. To the best of our knowledge, previous literature has not detected the strength of this feature. From the prediction power of this feature, it

[8] Although the unlexicalized features had shown no significant performance, they were added here to check for interaction effects between them and the lexicalized features.


Model=fine-tuned Q-ctxB-Embedding, GoldLabel=NISQ
Model=fine-tuned Q-ctxB-Embedding, GoldLabel=ISQ
Model=fine-tuned Q-ctxA-Embedding+speakerA, GoldLabel=NISQ
Model=fine-tuned Q-ctxA-Embedding+speakerA, GoldLabel=ISQ
Context Before
Question
Context After
Ground Truth
wrong
correct
ISQ
context before is a question
question and context after have different speakers
correct
wrong
NISQ
fine-tuned Q-ctxB-Embedding
fine-tuned Q-ctxA-Embedding+speakerA
Prediction
NISQ
ISQ

Is this your experience?

Do you think thats true?

I do believe thats true, not because of correspondents who were covering it but because of the nature of the times, the development of television, the pervasive use of television.

Is Joe Maricopa in Arizona going to test the president to see if there is gun powder residue on his fingers at this point?

Do you think President Obama is an avid skeet shooter?

I think he does it all the time.

Figure 1: Interactive visualization of the wrongly predicted instances of the models *fine-tuned Q-ctxB-Embedding* and *fine-tuned Q-ctxA-Embedding+speakerA*. Based on this visualization, we can observe sentences with similar patterns and how these are learned from the models. Some sentences are ambiguous having both patterns; thus, we need a third model for our ensemble.

might seem that information on the question and its context is not necessary at all. However, we show that the addition of the linguistic information of the question and its context through the fine-tuned embeddings provides a clear boost for the performance. The importance of similar linguistic unlexicalized features has to be investigated in future work. In fact, for the current work, we also experimented with the topic information, i.e., based on topic modeling, we extracted a binary feature capturing whether the topic of the question and the context-after is the same or not. However, this feature did not prove useful in any of the settings and was thus omitted from the analysis. Future work will have to investigate whether a better topic model leads to a more expressive binary feature and whether other such features, such as sentiment extracted from a sentiment classification model, can prove powerful predictors.

Concerning the distributional and NN methods, this is the first work employing such techniques for the task and confirming the findings of the more traditional machine learning settings. Fine-tuning the pretrained BERT embeddings reproduces what we showed for the standard classifiers: the context and especially the context-after boosts the performance. This finding is also confirmed when treating the fine-tuned BERT embeddings as standard feature vectors and further training on them. Most importantly, this setting allows for the expansion of the feature vector with the speaker information: this then leads to the best performance. Unsurprisingly, the speaker-before is not beneficial for the classification, as it was not itself a strong predictor. Finally, we also observe that the results reported for this smaller corpus are parallel to the results reported for the entire corpus (see Appendix A).

5.3 Further Extension & Optimization

By studying Table 3 the question arises whether our best-performing model of *fine-tuned Q-ctxA-Embedding+speakerA* can be further improved and crucially, whether the context-before can be of value. With our lexicalized models, we show that the best models are those exploiting the information of the context-before, in addition to the question and the context-after. However, all of our BERT-based models have been trained either on the combination of question and context-before or on the combination of question and context-after, but never the combination of all three. The inherent nature of the BERT model, which requires the input sequence to consist of a pair, i.e., at most two distinct sentences separated by the special token *[SEP]*, is not optimized for a triple input. On the other hand, "tricking" BERT into considering the context-before and the question as one sentence delivers poor results. Thus, we decided to exploit the power of visualization to see whether an ensemble model combining our so far best performing model of *fine-tuned Q-ctxA-Embedding+speakerA* with our context-before BERT-based model *fine-tuned Q-ctxB-Embedding* would be beneficial.

To this end, we created a small interactive Python visualization to compare the two models, using UMAP (McInnes et al., 2018) as a dimensionality reduction technique and visualizing the datapoints in a 2D scatter plot. We computed positions jointly for both models and projected them into the same 2D space using cosine similariy as the distance measure. As we are interested in potential common wrong predictions between the models, we only visualize wrongly classified samples, and group them by two criteria: the model used (color-encoded) and the gold label (symbol-encoded).

Examining the visualization of Figure 1 (left) we observe that there is no overlap between the wrongly predicted labels of the two models. This means that training an ensemble model is a promising way forward. Additionally, through the interactive visualization, we are guided to the most suitable ensemble model. Particularly, we see some common patterns for the wrongly predicted labels for each of the models. The *fine-tuned Q-ctxA-Embedding+speakerA* has a better performance in predicting ISQ, whereby the decision seems to be influenced by the speaker feature (i.e., if the question and context-after have different speakers, the model predicts ISQ). However, the *fine-tuned Q-ctxB-Embedding* model seems to learn a pattern of a context-before being a question; in such cases, the target question is predicted as NISQ. In the ground truth we have ambiguous cases though, where questions have both patterns. Thus, although it seems that the two models fail on different instances and that they could thus be combined in an ensemble, they would alone likely fail in predicting the ambiguous/controversial question instances. Instead, surface and POS features of the questions and their contexts should be able to differentiate between some of the controversial cases. To test this, we created an ensemble model consisting of the two models and the best lexicalized model holding such features (*Q+ctxB1+ctxA1*). First, this ensemble model checks whether *fine-tuned Q-ctxA-Embedding+speakerA* and *fine-tuned Q-ctxB-Embedding* predict the same label. If so, it adopts this label too. Otherwise, it picks up the prediction of *Q+ctxB1+ctxA1*. With this ensemble approach, we are indeed able to improve our so-far best model by 4%, reaching an accuracy of 88.3%, as shown in the last entry of Table 3.

At this point, two questions arise. First, the reader might wonder whether this result means that the task is virtually “solved”. Recall that the inter-annotator agreement was measured at 89.3% and thus, it might seem that our ensemble model is able to be competitive with that. However, this is not the case: if we observe the Fleiss Kappa, we see that it only demonstrates moderate agreement. This could be due to the difficulty of the task, as mentioned before, but it also shows that the task formulation has room for improvement. In a post-annotation session, our annotators reported that some of the uncertainty and disagreement could be tackled with multi-modal data, where also audio or video data of the corresponding questions is provided. Additionally, higher agreement could have been achieved with more annotators. Thus, our current work offers room for improvement, while providing strong baselines. Second, the question is raised whether this feature combination is indeed the best setting for all purposes of this task; the answer to this depends on what the ultimate goal of this task is. If the ultimate goal is application-based, where a model needs to determine whether a question requires a factoid answer (or not) in a real-life conversation, the trained model should not include the context-after as a feature as this would exactly be what we want to determine based on the model’s decision. However, if the goal is to automatically classify questions of a given corpus to generate linguistic insights, then the trained model can include all features. The evaluation undertaken here serves both these purposes by detailing all settings. On the one hand, we show that the models achieve high performance even when removing the context-after and that therefore an application-based setting is possible. On the other hand, we also discover which feature combination will lead to the best predictions, generating theoretical insights and enabling more research in this direction.

6 Conclusion

In this paper, we argued for the need of the Question-Type Identification task, in which questions are distinguished based on the communicative goals they are set to achieve. We also provided the first corpus to be used as a benchmark. Additionally, we studied the impact of different features and established diverse baselines, highlighting the peculiarities of the task. Finally, we were able to generate new insights, which we aim to take up on in our future work.

Acknowledgements

We thank the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for funding within project BU 1806/10-2 “Questions Visualized” of the FOR2111 “Questions at the Interfaces”. We also thank our annotators, as well as the anonymous reviewers for their helpful comments.

References

Francesca Alloatti, Luigi Di Caro, and Gianpiero Sportelli. 2019. Real Life Application of a Question Answering System Using BERT Language Model. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 250–253, Stockholm, Sweden. Association for Computational Linguistics.

Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. Discourse structure and dialogue acts in multiparty dialogue: the STAC corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2721–2727, Portorož, Slovenia. European Language Resources Association (ELRA).

Christine Bartels. 1999. *The intonation of English statements and questions*. New York: Garland Publishing.

Andrea Beltrama, Erlinde Meertens, and Maribel Romero. 2019. Decomposing cornering effects: an experimental study. *Proceedings of Sinn und Bedeutung*, 22(1):175–190.

Robbert-Jan Beun. 1989. *The recognition of declarative questions in information dialogues*. Ph.D. thesis, Tilburg University. Pagination: 139.

Shohini Bhattasali, Jeremy Cytryn, Elana Feldman, and Joonsuk Park. 2015. Automatic Identification of Rhetorical Questions. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 743–749, Beijing, China. Association for Computational Linguistics.

María Biezma and Kyle Rawlins. 2017. Rhetorical questions: Severing asking from questioning. *Semantics and Linguistic Theory*, 27:302.

Daniel G. Bobrow, Bob Cheslow, Cleo Condoravdi, Lauri Karttunen, Tracy Holloway King, Rowan Nairn, Valeria de Paiva, Charlotte Price, and Annie Zaenen. 2007. PARC's Bridge and Question Answering System. In *Proceedings of the Grammar Engineering Across Frameworks Workshop (GEAF 2007)*, pages 46–66, Stanford, California, USA. CSLI Publications.

Dwight Bolinger. 1978. Yes—no questions are not alternative questions. In *Questions*, pages 87–105. Springer.

Harry Bunt, Volha Petukhova, Andrei Malchanau, Kars Wijnhoven, and Alex Fang. 2016. The DialogBank. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3151–3158, Portorož, Slovenia. European Language Resources Association (ELRA).

Ray Cattell. 1973. Negative Transportation and Tag Questions. *Language*, 49(3):612–639.

Ying-Hong Chan and Yao-Chung Fan. 2019. BERT for Question Generation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 173–177, Tokyo, Japan. Association for Computational Linguistics.

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 the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Veneeta Dayal. 2016. *Questions*. Oxford University Press, Oxford.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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. Association for Computational Linguistics.

Regine Eckardt. 2020. Conjectural questions: The case of German verb-final wohl questions. *Semantics and Pragmatics*, 13:1–17.

David Ferrucci, Anthony Levas, Sugato Bagchi, David Gondek, and Erik T. Mueller. 2013. Watson: Beyond Jeopardy! *Artificial Intelligence*, 199–200(1):93–105.

F.M. Harper, D. Moy, and J. A. Konstan. 2009. Facts or Friends? Distinguishing Informational and Conversational Questions in Social Q&A Sites. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI 2009)*, pages 759–768.

Robin Håvik, Jo Dugstad Wake, Eivind Flobak, Astri Lundervold, and Frode Guribye. 2019. A conversational interface for self-screening for adhd in adults. In *Internet Science*, pages 133–144, Cham. Springer International Publishing.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

D. Jurafsky, E. Shriberg, and D. Biasca. 1997. Switchboard SWBD-DAMSL shallow-discourse-function annotation coders manual. Technical Report Draft 13, University of Colorado, Institute of Cognitive Science.

Aikaterini-Lida Kalouli, Katharina Kaiser, Annette Hautli-Janisz, Georg A. Kaiser, and Miriam Butt. 2018. A Multilingual Approach to Question Classification. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Baichuan Li, Xiance Si, Michael R. Lyu, Irwin King, and Edward Y. Chang. 2011. Question Identification on Twitter. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, page 2477–2480, New York, NY, USA. Association for Computing Machinery.

Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open Sesame: Getting inside BERT's Linguistic Knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253, Florence, Italy. Association for Computational Linguistics.

Kien Hoa Ly, Ann-Marie Ly, and Gerhard Andersson. 2017. A fully automated conversational agent for promoting mental well-being: A pilot RCT using mixed methods. *Internet Interventions*, 10:39–46.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. 2018. UMAP: Uniform Manifold Approximation and Projection. *The Journal of Open Source Software*, 3(29):861.

Nicole Novielli and Carlo Strapparava. 2009. Towards unsupervised recognition of dialogue acts. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Student Research Workshop and Doctoral Consortium*, pages 84–89, Boulder, Colorado. Association for Computational Linguistics.

Sharoda A. Paul, Lichan Hong, and Ed H. Chi. 2011. What is a question? Crowdsourcing tweet categorization. In *CHI 2011, Workshop on Crowdsourcing and Human Computation*.

Suhas Ranganath, Xia Hu, Jiliang Tang, SuhangWang, and Huan Liu. 2016. Identifying Rhetorical Questions in Social Media. In *Proceedings of the 10th International AAAI Conference on Web and Social Media (ICWSM 2016)*.

Jerrold Sadock. 1971. Queclaratives. In *Papers from the 7th Regional Meeting of the Chicago Linguistic Society*, pages 223–232.

Yuichi Sasazawa, Sho Takase, and Naoaki Okazaki. 2019. Neural Question Generation using Interrogative Phrases. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 106–111, Tokyo, Japan. Association for Computational Linguistics.

Gaurav Sood. 2017. CNN Transcripts 2000–2014. Published by Harvard Dataverse, retrieved from `https://doi.org/10.7910/DVN/ISDPJU`.

Dan Su, Yan Xu, Genta Indra Winata, Peng Xu, Hyeondey Kim, Zihan Liu, and Pascale Fung. 2019. Generalizing Question Answering System with Pretrained Language Model Fine-tuning. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 203–211, Hong Kong, China. Association for Computational Linguistics.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Kai Wang and Tat-Seng Chua. 2010. Exploiting salient patterns for question detection and question retrieval in community based question answering. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING10)*, page 1155–1163.

J. M. O. Wheatley. 1955. Deliberative questions. *Analysis*, 15(3):49–60.

Zhe Zhao and Qiaozhu Mei. 2013. Questions about Questions: An Empirical Analysis of Information Needs on Twitter. In *Proceedings of the International World Wide Web Conference Committee (IW3C2)*, pages 1545–1555.

Mark-Matthias Zymla. 2014. Extraction and Analysis of non-canonical Questions from a Twitter-Corpus. Master's thesis, University of Konstanz.

Appendix A: Performance Results on the entire RQueT

The following table collects all performance results when training on the entire RQueT corpus of 2000 questions. Although we cannot make this whole corpus available, we would like to report on the performance to show how our findings are parallel in both variants of the corpus and that the smaller size of the corpus we make available does not obscure the overall picture.

Lexicalized					Unlexicalized				BERT Embeds			Classifiers						
Q tokens/POS	**ctxB1 tokens/POS**	**ctxB2 tokens/POS**	**ctxA1 tokens/POS**	**ctxA2 tokens/POS**	**speakerB**	**speakerA**	**similQA**	**lenDiffA**	**Q-Embed**	**Q-ctxB-Embed**	**Q-ctxA-Embed**	**NB**	**SVM**	**DT**	**MLP**	**FF**	**LSTM**	**BERT Fine-Tuning**
✓												64.5	61.5	59	-	-	-	-
✓	✓											67.5	67	62	-	-	-	-
✓	✓	✓										72.5	62.5	56.5	-	-	-	-
✓			✓									73	62	63	-	-	-	-
✓			✓	✓								71	65.5	62.5	-	-	-	-
✓	✓		✓									**75.5**	65.5	61	-	-	-	-
✓	✓	✓	✓									67.5	62	60	-	-	-	-
✓	✓	✓	✓	✓								66.5	61	58	-	-	-	-
					✓							57	57	57	57	56.9	56.9	-
						✓						**78.5**	**78.5**	**78.5**	**78.5**	**78.5**	**78.5**	-
					✓	✓						78.5	78.5	78.5	78.5	78.5	78.5	-
						✓	✓	✓				78.5	77.5	78.5	-	-	-	-
✓	✓	✓			✓							72	65	56.5	-	-	-	-
✓			✓			✓						77.5	69	75	-	-	-	-
✓			✓		✓	✓						76	70	75	-	-	-	-
✓	✓		✓			✓						**78**	73	77	-	-	-	-
✓	✓	✓	✓			✓						69	71	75.5	-	-	-	-
✓	✓		✓			✓	✓					78	74.5	76.5	-	-	-	-
✓	✓		✓			✓		✓				78	72	77	-	-	-	-
									PT			-	-	-	-	-	-	76.4
										PT		-	-	-	-	-	-	77.4
											PT	-	-	-	-	-	-	79.4
									FN			76.5	76	72.5	**77**	76.4	72.5	-
						✓			FN			77	78.5	73	**80**	79.5	76.4	-
										FN		76	78.5	78.5	**80**	79.5	79.5	-
					✓					FN		76	79	78.5	78.5	**80**	79	-
											FN	78.5	78	**79.5**	78.5	**79.5**	76.4	-
						✓					FN	78.5	80	80	81.5	**82.4**	78.5	-
✓	✓		✓			✓			✓	✓	✓	**Ensemble: *85***						

Table 5: Accuracy of the various classifiers and feature combinations (settings) on the entire RQueT corpus of 2000 questions. A checkmark means that this feature was present in this setting. *PT* stands for the pretrained BERT embeddings and *FN* for the fine-tuned ones. Bolded figures are the best performances across types of classifiers. The stared figure is the best performing ensemble model across settings. *wOverAbs* and *wOverPerc* are omitted for brevity.