“I’ll be there for you”: The One with Understanding Indirect Answers

Cathrine Damgaard,[♡] **Paulina Toborek,**[♡] **Trine Eriksen**[♡] and **Barbara Plank**
Department of Computer Science
IT University of Copenhagen
{catd, pato, trer, bapl}@itu.dk

Abstract

Indirect answers are replies to polar questions without the direct use of word cues such as ‘yes’ and ‘no’. Humans are very good at understanding indirect answers, such as ‘I gotta go home sometime’, when asked ‘You wanna crash on the couch?’. Understanding indirect answers is a challenging problem for dialogue systems. In this paper, we introduce a new English corpus to study the problem of understanding indirect answers. Instead of crowdsourcing both polar questions and answers, we collect questions and indirect answers from transcripts of a prominent TV series and manually annotate them for answer type. The resulting dataset contains 5,930 question-answer pairs. We release both aggregated and raw human annotations. We present a set of experiments in which we evaluate Convolutional Neural Networks (CNNs) for this task, including a cross-dataset evaluation and experiments with learning from disagreements in annotation. Our results show that the task of interpreting indirect answers remains challenging, yet we obtain encouraging improvements when explicitly modeling human disagreement.

1 Introduction

Humans are very good at interpreting indirect answers to polar questions. In conversations, even if direct answers are possible, humans often prefer indirect answers due to cooperativeness and to advance the dialogue (Stenström, 1984). For dialogue systems and Natural Language Processing (NLP) more generally, however, interpreting indirect answers remains a challenge (Clark et al., 2019). Recent seminal work introduced Circa, a new large-scale dataset containing pairs of polar questions and indirect answers in English (Louis et al., 2020). This allows for data-driven experiments in this question-answering domain.

Q: Hey. Everything ok?
A: I’m just mad at my agent.
L: No, No, Yes

Q: Are you back from Minsk?
A: Well, just for a couple of days.
L: Yes, No, Yes, subject to some conditions

Table 1: Examples from the dataset with polar question (Q), indirect answer (A) and annotator labels (L).

Understanding indirect answers is a pragmatic problem, and even though humans typically have little difficulty in interpreting indirect answers, they may not all agree on a possible interpretation. Work on learning from human disagreement has shown that incorporating disagreement from human annotation is not only noise, but can provide valuable information (Plank et al., 2014; Aroyo and Welty, 2015; Rodrigues and Pereira, 2018).

Motivated by these two lines of research, we propose a new dataset for studying indirect answers. We provide both aggregated (ground truth/gold) annotations and the raw annotations. This allows us to study the effect of learning to integrate human disagreement into understanding indirect answers. Our dataset, called Friends-QIA, was created by collecting question and answer pairs from transcripts of a popular TV series. The data collection differs in comparison to the recently introduced Circa corpus. In their study, a set of 10 dialogue prompts were defined, and both questions and answers were collected by crowdsourcing (Louis et al., 2020). Following this step, the annotation was crowdsourced again, resulting in a set of labels which was later conflated into a relaxed set of six classes. Instead, we opted to collect the data from transcripts and manually annotate the question-answer pairs in house (by three of the authors of this paper, all highly proficient in English). Examples from our dataset are provided in Table 1. A data statement is provided in Appendix A.

[♡] The authors contributed equally to this work.

Proceedings of the 2nd Workshop on Computational Approaches to Discourse, pages 1–11
November 10–11, 2021. ©2021 Association for Computational Linguistics

Contributions In this paper, we a) introduce a new dataset, FRIENDS-QIA, with 5,930 polar question-indirect answer pairs in English;[1] b) study the effectiveness of neural classifiers based on Convolutional Neural Networks, both with traditional pre-trained word embeddings and contextualized BERT embeddings; c) provide results on cross-dataset evaluation, for which we train a model on both CIRCA and FRIENDS-QIA; and d) show that modeling human disagreement via deep learning from crowds is beneficial for this task.

2 Related Work

The task of understanding indirect answers in Natural Language Processing is relatively new and has not been attempted by many yet. To enable progress on the task, Louis et al. (2020) created and released the first large-scale English language corpus, CIRCA, consisting of 34,268 polar questions and their corresponding indirect answers. The difficulty of interpreting indirect answers is, however, widely studied in some early papers, (e.g. Green and Carberry, 1992, 1999). Following this work, de Marneffe et al. (2009) propose a logical inference model with probabilistic methods. They realize the influence of discourse conditions as well as the difference between the literal meaning of the answer and the interpretation by the two speakers. Hockey et al. (1997) further underline the complexity of interpreting indirect answers to polar questions. They explore the existing Edinburgh map task corpus (Thompson et al., 1993) which consists of two-person dialogues already coded for dialogue structure. Clark et al. (2019) recently also explore the understanding of indirect answers and report on the difficulty of the task. They create a new dataset, BOOLQ, by combining search queries from the Google search engine as questions and passages on Wikipedia pages as answers. They attempt to classify such indirect answers by training BERT-based neural models. Louis et al. (2020) further experiment with training BERT models from scratch and by transfer learning from BOOLQ, building on top of Clark et al. (2019). Their newly created dataset, CIRCA, includes 10 question prompt types, spanning a wide variety of communication situations. Their study shows promising results, and inspired our work.

Preparing a dataset for classification tasks often requires collecting labels from multiple annotators. A unanimous gold standard label cannot always be clearly achieved (e.g. Aroyo and Welty, 2015; Rodrigues et al., 2013; Palomaki et al., 2018; Pavlick and Kwiatkowski, 2019). The problem of learning from multiple annotators has become more important and several attempts have been made to deal with biases present in such data. Yan et al. (2014) look into different levels of expertise among annotators and how a model can learn from this, taking into consideration the biases present while labeling the dataset. By measuring the inter-annotator agreement and incorporating the annotator uncertainty in model training, Plank et al. (2014) show that modelling annotator disagreement can be useful even in cases of seemingly more objective annotation tasks, like part-of-speech tagging. An interesting and similar approach was suggested by Rodrigues and Pereira (2018). Their method, deep learning from crowds (DLFC), adds a crowd layer on top of a neural network which takes advantage of the reliability and bias from different annotators. It assumes access to the raw annotations from the training data. This is in contrast to the weighting approach proposed by Plank et al. (2014), who integrate aggregated disagreement from a sample. As we have the full data available with multiple annotations in FRIENDS-QIA, we here experiment with the deep learning from crowds (DLFC) approach proposed by Rodrigues and Pereira (2018). The key idea is to train a model directly from the noisy annotator-specific annotations modeled as additional per-annotator auxiliary tasks on top of the gold distribution. By using backpropagation and creating annotator-specific weights in the crowd layer, it allows us to include the label bias in the model's learning. There is emerging interest in learning from disagreement, and we refer the reader to a recent survey on this topic (Uma et al., 2021).

3 FRIENDS-QIA Corpus

We introduce a new corpus called FRIENDS-QIA for studying indirect answers. The three steps involved in corpus construction are: data collection, data preprocessing and data annotation.

Instead of crowdsourcing the data collection as done by Louis et al. (2020), we attempt to exploit already existing data by simply looking for useful question-answer (QA) pairs in dialogues. We use the existing transcriptions of the American televi-

[1] FRIENDS-QIA is available at: https://github.com/friendsQIA/Friends_QIA.

1: Yes **Monica:** You still work at the multiplex? **Chip:** Oh, like I'd give up that job! Free popcorn and candy, anytime I want. I can get you free posters for your room.	2: No **Monica:** He is right, isn't he? **Chandler:** Y'know what, I think this might be one of the times he's wrong.	3: Yes, subject to some conditions **Joey:** Now if a cow should die of natural causes, I can have one of those right? **Phoebe:** Not if I get there first.
4: Neither yes nor no **Phoebe:** Okay—ooh, but are you going to have time to read it? **Rachel:** Oh, I read that in high school.	5: Other **Monica:** I need more swordfish. Can you get me some more swordfish? **Kitchen Worker:** I don't speak English.	6: N/A **Joey:** You know more than one Fun Bobby? **Chandler:** I happen to know a Fun Bob.

Table 2: Examples of QA pairs from FRIENDS-QIA for each of the 6 labels.

sion sitcom "Friends".[2] This data provides rich dialogues from conversations covering regular, everyday topics. FRIENDS-QIA includes 10 seasons consisting of 17–25 episodes each. This sums to a total of 228 episodes of the TV series. The full dataset, including metadata, is described in Appendix C.

3.1 Data Collection

To collect the QA pairs we manually scan all episodes of "Friends" for polar questions and retrieve both the question and the answer as well as which character said each of the two. A few examples are listed below.

Example 3.1
Joey: *Hey Pheebs, you wanna help?*
Phoebe: *Oh, I wish I could, but I don't want to.*

Example 3.2
Rachel: *Hey! So, did you quit?*
Chandler: *No, I almost did, couldn't leave Ross there without a spotter!*

Example 3.3
Joey: *(intrigued) Really?*
Mr. Treeger: *Yeah, you could dance real good with her, she's the same size as me.*

As the examples show, some answers directly give away cues (e.g., Example 3.2 starts with a "No,"). We preprocess the data manually, to obfuscate direct cues and enrich short questions with information from the context, to keep the setup similar to prior work and model single question-answer pairs. The data preprocessing motivation and details are outlined next.

3.2 Data Preprocessing

To prepare the QA pairs in a useful format for the task at hand, we assign each collected pair to one of four categories. The categories are included in the final FRIENDS-QIA dataset and describe if, and in what way, the question or answer had to be modified. Modifications include removing direct cues to the answers 'yes' or 'no', such as 'yeah' or 'nope' (a full list is provided in Appendix B), adding context to the question in case we deem it insufficient, as well as removing metadata and irrelevant parts of the question or answer. The four categories are listed below along with their respective description.

1. Able to use question and answer, exactly as they are
2. Have to only remove yes/no/nope/yeah (etc.) in the answer
3. Clarifying/adding context to the question (or rarely the answer)
4. Questions with an answer only containing a yes/no/nope/yeah (etc.)

Example 3.1 is a QA pair which is useful for our task exactly as it is. It has a polar question and a perfectly indirect answer with no direct mappers or cues, so we assign this QA pair to category 1.

However, in Example 3.2 we remove a bit of irrelevant context, namely the "Hey!" in the question. Furthermore, the answer contains an actual "No", which should not be present in an indirect answer, so this is also removed. These modifications cause the QA pair to be assigned to category 2. The modified example is shown below.

Example 3.4
Rachel: *So, did you quit?*

[2] The transcriptions are available at https://fangj.github.io/friends/

Chandler: *I almost did, couldn't leave Ross there without a spotter!*

Example 3.3 is from category 3, where we need to do a larger modification to make the QA pair useful. This modification could for example be to add context, which is usually found in the transcript lines immediately above the question. The modified example is shown below.

Example 3.5
Joey: *Really? Marge has a girlfriend?*
Mr. Treeger: *You could dance real good with her, she's the same size as me.*

Category 4 contains QA pairs where the answer consists of only a direct mapper or cue alone, so it cannot easily be modified into an indirect answer. Therefore, this category is ultimately excluded from the final FRIENDS-QIA dataset.

3.3 Data Annotation

We adopt the RELAXED label set as introduced by Louis et al. (2020). It consists of six classes: YES, NO, YES SUBJECT TO SOME CONDITIONS, NEITHER YES NOR NO, OTHER and N/A. A description of each class as used in FRIENDS-QIA is as follows:

1. YES: The answer is either a definite yes, a probably yes or a sometimes yes
2. NO: The answer is either a definite no or a probably no
3. YES, SUBJECT TO SOME CONDITIONS: The answer is only yes/no, if a certain condition is satisfied
4. NEITHER YES NOR NO: The answer does not imply yes nor no
5. OTHER: The answer is not related to the question
6. N/A: Lack of majority agreement

The label distribution of FRIENDS-QIA is shown in Figure 1.

3.3.1 Annotator Agreement and Aggregation

We measure the agreement and thereby the reliability of the annotations in FRIENDS-QIA. The three annotators independently labeled all QA pairs in the dataset. The raw agreement distribution of the annotations is listed in Table 3, which includes the agreement for each of the final three categories described in Section 3.2 as well as for the entire FRIENDS-QIA (all three categories). The resulting Fleiss Kappa score of the full dataset is 0.8833.

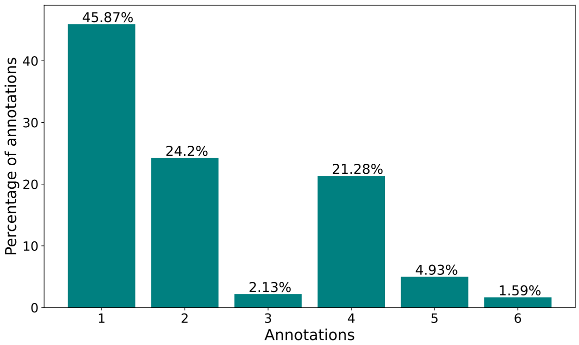


Figure 1: Gold label distribution.
1: YES, 2: NO, 3: YES, SUBJECT TO SOME CONDITIONS, 4: NEITHER YES NOR NO, 5: OTHER, 6: N/A.

	All agree	Two agree	All disagree
Full dataset	75.02%	23.39%	1.59%
Category 1	72.79%	25.48%	1.73%
Category 2	77.80%	20.70%	1.50%
Category 3	77.17%	21.54%	1.29%

Table 3: Annotator agreement.

We use majority voting to aggregate the gold standard from the three annotations. In few cases (1.59%), all three annotators disagreed on the label, and we map these to N/A. Examples from FRIENDS-QIA are provided in Table 2, where the QA pairs with label 1–5 were given the same label by at least two annotators, whereas the example with label 6 is a case of full annotator disagreement.

For compatibility with CIRCA, we exclude OTHER (as we found it annotated when the question was not polar in CIRCA) and N/A, which is very infrequent in FRIENDS-QIA (only 1.59% of the labels). We note here that OTHER is used in FRIENDS-QIA in cases where the answer is not related to the question —which is possible due to the dialogue context on which FRIENDS-QIA is based.

Dataset	FRIENDS-QIA	CIRCA
All	5,930	32,993
Train	4,744	26,394
Dev	593	3,300
Test	593	3,299

Table 4: Data split and dataset sizes using labels 1–4.

The resulting dataset used in the experiments contains 5,930 question-answer pairs over four labels. The dataset sizes after splitting it are listed in Table 4, including a comparison to the CIRCA dataset (Louis et al., 2020). The splits for both datasets are stratified to ensure similar label distribution across all three sets.

Table 5 provides additional insights into the two datasets, by providing measures on vocabulary size, length of the instance as well as type-token ratio (TTR). We observe that FRIENDS-QIA is richer in terms of vocabulary, has higher lexical variety and contains on average longer utterances, despite the smaller overall size compared to CIRCA.

Statistic	Q	A	Q + A
FRIENDS-QIA			
Vocabulary size	4,207	4,843	6, 373
Maximum length	80	188	195
Average length	11	13	24
Type-token ratio	0.06	0.06	0.04
CIRCA			
Vocabulary size	2,003	7,317	7,499
Maximum length	25	27	38
Average length	7	6	14
Type-token ratio	0.01	0.03	0.02

Table 5: Vocabulary statistics (in number of tokens).

4 Experiments

We experiment with different variants of a Convolutional Neural Network (CNN). In particular, we explore the use of BERT embeddings, a crowd layer and combining datasets to obtain higher performance on FRIENDS-QIA.

Base CNN Our CNNs are implemented with inspiration from Kim (2014), who uses 1-dimensional convolutions in parallel. As shown in Figure 2, our implementation of the parallel convolutions consists of a convolutional 1d, a max pooling, a flatten and a dropout layer, after which they are concatenated and fed to the final output layer which uses softmax as activation function. The convolutional layer applies the ReLU activation function after which the maximum of each filter is selected. The dropout layer applies a dropout rate of 0.5 which is the same rate we use on the final layer when we apply regularization with a constraint of the L2-norm on the weights. The models either take English BERT embeddings (Devlin et al., 2019) as input or use GloVe embeddings (Pennington et al., 2014). The models are optimized using the Adam optimizer (Kingma and Ba, 2015). This general architecture is what all of our CNN variants are built upon. In preliminary experiments we took a portion of the training data as hyperparameter tuning set. We performed a grid search on the base CNN to find the optimal hyperparameters. For the CNN with BERT no further hyperparameter tuning was done.

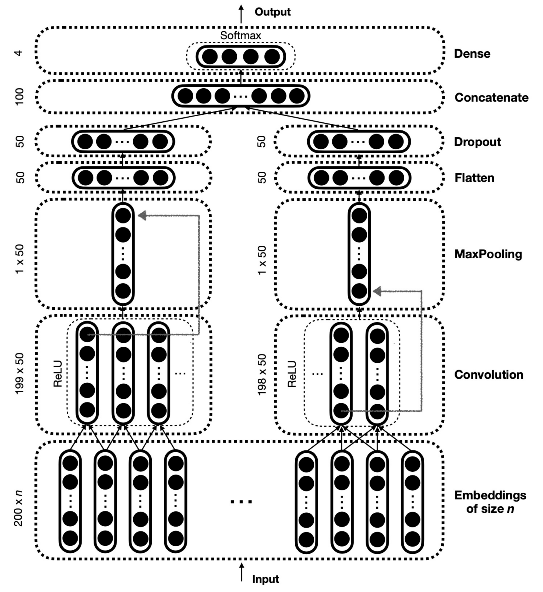


Figure 2: CNN architecture.

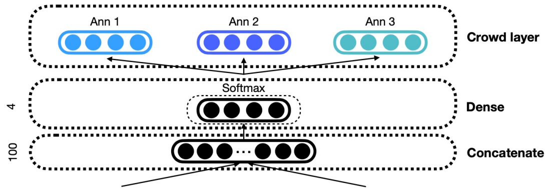


Figure 3: Illustration of deep learning from crowds proposed by Rodrigues and Pereira (2018).

Crowd Layer Figure 3 illustrates the key idea of the DLFC approach we adopt on top of our CNN. Training the models with a crowd layer is closely based on the paper and code by Rodrigues and Pereira (2018). Following their implementation, our crowd layer is applied on top of the existing network. It takes as input the output of the dense layer and uses the annotator-specific labels, each modeled as a separate task, to propagate errors through the network and adjust the gradients. The layer is applied on the already trained and saved base model with an annotator-specific weight matrix ($\mathbf{W}^r$), and further training is performed with the crowd layer. We found this setup to perform best. For the CNN+crowd layer model we use the same parameters as when training the base model. After training, the crowd layer is removed to allow for the final classification using the original dense layer with a softmax activation function.

Evaluation Accuracy and macro-average F1-score are the evaluation metrics. We first report results on the development set, and report the performance of the best models on the test portion. As baselines, we provide results of a majority baseline and a Naive Bayes model with word trigrams.

5 Results

The main results on the development set are given in Table 6. It provides results of a majority baseline, the CNN with GloVe and BERT embeddings, and the CNN trained with the crowd layer. We first discuss results for models trained on FRIENDS-QIA, followed by results on training with FRIENDS-QIA *and* CIRCA.

	Accuracy	F1-score
Majority baseline	49.07	16.46
Train on FRIENDS-QIA:		
Naive Bayes	52.45	37.08
CNN	54.86	33.02
CNN (Q only)	49.92	23.46
CNN (A only)	55.09	35.32
CNN, multi-input	53.91	35.61
CNN + crowd layer	**55.71**	**39.38**
CNN with BERT	**64.08**	49.16
CNN with BERT (Q only)	43.79	26.96
CNN with BERT (A only)	52.22	29.96
CNN with BERT, multi-input	61.27	50.31
CNN with BERT + crowd layer	63.46	**55.00**
Train on FRIENDS-QIA + CIRCA:		
Naive Bayes	52.11	45.67
CNN	53.51	43.16
CNN with BERT	61.27	48.65

Table 6: Results on the FRIENDS-QIA development data.

Take-aways There are four take-aways. First, we observe the difficulty of the task. This can be seen from the low baseline results: The majority baseline reaches an accuracy of 49.07 and F1-score of 16.46. The Naive Bayes model reaches an accuracy of 52.45 and F1-score of 37.08.

Second, we test several variants of the base CNN with GloVe embeddings: one which takes the concatenation of question and answer as input (CNN), one that models question and answer separately (CNN, multi-input), and a CNN using only the answer or the question. The latter provides information on how much signal is represented in the answer (or question) alone. As the results in Table 6 show, multi-input modeling is not consistently the best model (on both metrics). Training the CNN on answers alone is highly predictive and performs substantially better than the question alone. This corroborates findings by Louis et al. (2020): the answer alone is highly predictive for the task (yet not sufficient). Overall, the best models are obtained when considering both question and answer, reaching an accuracy of 54.86 and F1-score of 33.02 on the development set with base CNN.

Third, we observe that modeling the annotator uncertainty is beneficial. When we add the crowd layer to the base CNN (CNN + crowd layer), we observe both improved accuracy and F1-score, reaching 55.71 and 39.38, respectively. This is encouraging, as it shows that the disagreement in human annotations is informative for this task.

Finally, we observe that using BERT embeddings as input representations consistently and remarkably improves the performance of the model. The CNN model with BERT reaches an accuracy of 64.08 and F1-score of 49.16, which is an absolute improvement of over 9% in accuracy and over 16% in F1-score. What is, however, striking, is that this trend does not hold for the BERT-based model trained on answer or question alone; we attribute this to the limited context, but this result warrants further investigation. Especially the F1-score is not even better than the base CNN model with GloVe embeddings on answers alone. Nevertheless, BERT representations overall improve the best models that take question-answer pairs as input, which are necessary for the overall best performance on the task.

We observe that the crowd layer for the CNN taking the concatenation of question and answer as input (CNN) further improves upon BERT representations, for which we obtain an F1-score of 55 and the overall best performance with the CNN with BERT *and* crowd layer. The accuracy with the crowd layer does not improve considerably, which shows that the crowd layer is not equally helpful for all classes. We hypothesize that the crowd layer is particularly helpful for infrequent classes, which are the most difficult for this task and which we observed the highest annotator disagreement on. This is in fact the case, as we will discuss in Section 6.

Training on FRIENDS-QIA and CIRCA We investigated whether we could improve the model further by using also the CIRCA data as additional training data. The bottom rows of Table 6 show that this is not the case; taking the union of the two datasets is generally not beneficial. The limited added vocabulary of CIRCA does not yield better generalization within FRIENDS-QIA.

	Accuracy	F1-score
Majority baseline	49.07	16.46
Train on FRIENDS-QIA:		
CNN with BERT	**61.33**	45.65
CNN with BERT, multi-input	61.10	45.53
CNN with BERT + crowd layer	60.32	**47.89**
Train on FRIENDS-QIA + CIRCA:		
CNN with BERT	58.52	41.82

Table 7: Results on the FRIENDS-QIA test data.

Test Set Results Table 7 shows the results of evaluating the best models and baselines on the test set. The results corroborate our findings from the development set. Training with the crowd layer is beneficial, and improves F1-score from 45.65 to 47.89. Similarly as on the development set, overall accuracy slightly drops with the crowd layer (it is within a 1% range). Training on the union of the two datasets does not outperform a model trained on FRIENDS-QIA alone.

6 Discussion

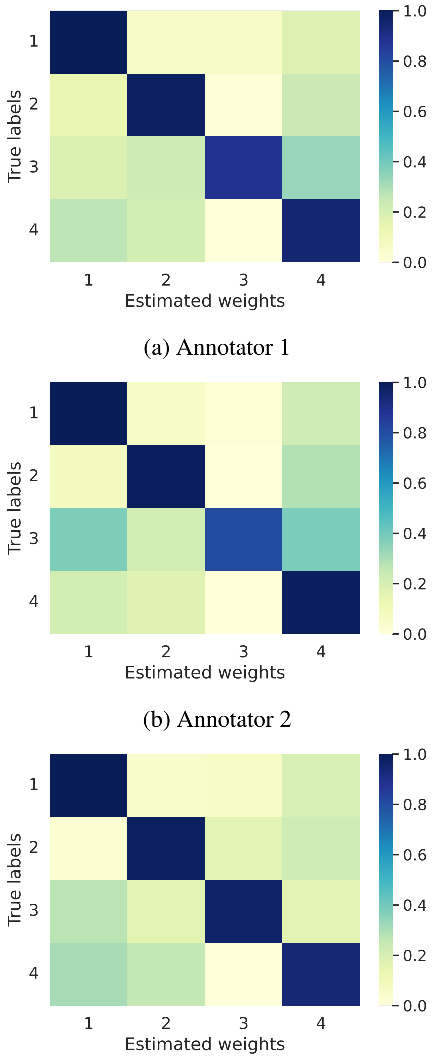
In the following section, we provide additional insights on the task.

Crowd Layer Analysis We observe a consistent improvement of overall macro F1-score, at a cost of a slight drop in accuracy. Therefore we analyze the per-class F1-score for the BERT-based model trained with and without the crowd layer. Moreover, we analyze the resulting estimated annotator bias matrices obtained from the crowd layer.

	1	2	3	4
CNN with BERT	73.42	51.93	14.23	57.06
CNN with BERT + crowd layer	72.28	53.98	38.72	55.05

Table 8: Per-class results on the development set.
1: YES, 2: NO, 3: YES, SUBJECT TO SOME CONDITIONS, 4: NEITHER YES NOR NO.

The results are presented in Table 8. The results confirm that the crowd layer is particularly useful for more easily confused (in this task low frequency) classes, namely 3 (YES, SUBJECT TO SOME CONDITIONS). It helps on 2 (NO) as well. The crowd layer hurts the most frequent class, which aligns well with the overall gain in macro F1-score at a slight cost in accuracy.



(a) Annotator 1

(b) Annotator 2

(c) Annotator 3

Figure 4: Normalized weight matrices from the crowd layer for each annotator.
1: YES, 2: NO, 3: YES, SUBJECT TO SOME CONDITIONS, 4: NEITHER YES NOR NO.

Figure 4 shows the three annotator-specific weight matrices extracted from the crowd layer. Overall, the weight matrices show several patterns. The dark diagonals mean that the estimated weights are high and the annotators agree with the gold standard. This is also reflected in the raw agreement as shown in Table 3. The crowd layer generally models each annotator similarly with a few exceptions. One of them is a clear uncertainty of label 3 (YES, SUBJECT TO SOME CONDITIONS). This fits exactly with the label distribution in the FRIENDS-QIA, where class 3 is heavily underrepresented. For annotator 2, the estimated weights are visibly lower for the gold standard 3 (YES, SUBJECT TO SOME CONDITIONS). That means that this annotator tends to give other labels, mostly 1 (YES) and 4 (NEITHER YES NOR NO), when the actual gold standard is 3 (YES, SUBJECT TO SOME CONDITIONS). The annotators 1 and 2 assign label 4 (NEITHER YES NOR NO) more often, in cases where the gold standard was actually one of the other labels. This pattern is not as apparent for annotator 3, which means that this annotator agreed slightly more with the gold labels than the other two annotators.

Incorrect Predictions Figure 5 shows the annotator agreement for correct and incorrect predictions of CNN with BERT (using FRIENDS-QIA on the test set). It is clear that the model is more likely to predict the wrong label, when the annotators are also in disagreement with each other.

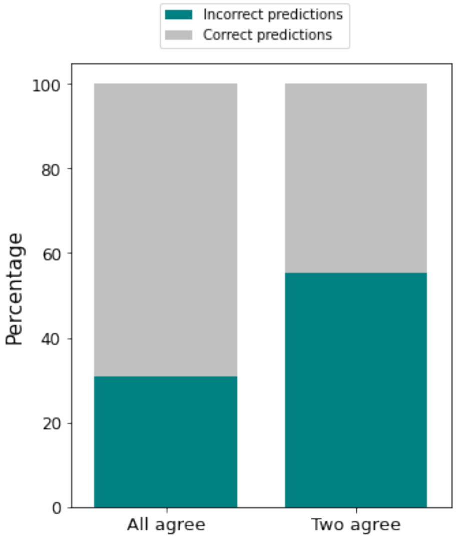


Figure 5: Correct and incorrect predictions of CNN with BERT vs. annotator agreement.

Furthermore, we show the amount of correct and incorrect predictions of CNN with BERT for each category (which were described in Section 3.2) in Figure 6. The amounts are generally similar, but category 3, which required the largest modifications such as adding context to the question, is slightly harder to classify than category 1 and 2, which required either no modification or removing direct cues in the answer, respectively.

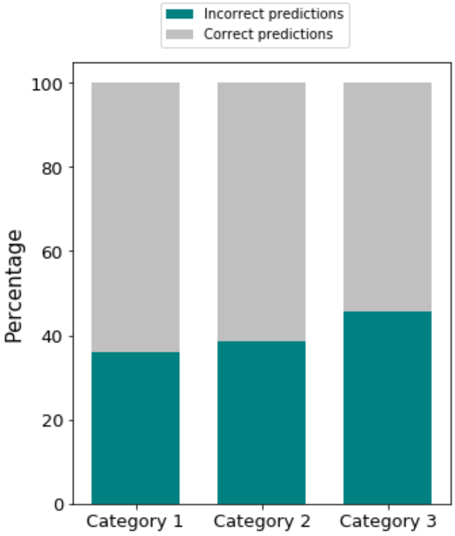


Figure 6: Correct and incorrect predictions of CNN with BERT vs. category.

FRIENDS-QIA versus CIRCA In general, we see a lower performance when we train and evaluate on FRIENDS-QIA, compared to using the CIRCA data for training and evaluation (we also trained models on CIRCA alone and evaluate on FRIENDS-QIA but leave them out for space reasons). The difference in performance is largely due to the general differences between the two datasets, and we describe a few of those differences as well as the reasons for them in the following paragraphs.

First of all, the data is collected in a different way, which greatly affects the content of the questions and answers. CIRCA is created specifically for the task at hand by Louis et al. (2020), namely understanding indirect answers, and was created under specific, topic-restricted settings, consisting of 10 different dialogue prompts. FRIENDS-QIA is obtained from a more open domain (yet, confined to typical TV series dialogues) and includes a broader context (also reflected in the vocabulary size differences). For example, it includes cases of both sarcasm and irony as well as additional information in the utterances, which might not be related to the actual question or answer.

Secondly, FRIENDS-QIA includes multi-sentence responses. Given the dialogue context, we observe cases where the speaker might change their mind in-between the sentences, further complicating the task of interpreting the questions and answers. This is very different from CIRCA due to the restricted and written setup, resulting in the CIRCA data being much more concise in meaning and structure than FRIENDS-QIA.

Third of all, a major difference between CIRCA and FRIENDS-QIA is the data size. FRIENDS-QIA contains much fewer QA pairs than CIRCA, which makes it considerably more difficult for the CNNs to learn well from. Additionally, FRIENDS-QIA has fewer annotations contributing to the gold standard (three versus five annotations in CIRCA), yet each of the three corresponds to a single annotator for FRIENDS-QIA, which is not the case in CIRCA.

7 Conclusions and Future Work

In this paper, we present FRIENDS-QIA, a corpus for studying indirect answers in English dialogues. We propose to mine TV series transcripts of a well-known TV series. Recent work proposed this challenging answer-understanding task, and collected CIRCA, a dataset on question-answer pairs from 10 dialogue prompts constructed instead using crowdsourcing (Louis et al., 2020). Motivated by their work, we propose FRIENDS-QIA. It contains a total of 5,930 question-answer pairs, and is released both with a majority label and the raw annotations.

Our results with CNNs show that a model trained with BERT embeddings outperforms a CNN trained with GloVe word representations. Most interestingly, this is, however, only the case for a model that considers both question and answer. Training on the answers alone provides reasonable signal for the task, but is not sufficient to resolve the indirect answer.

Understanding indirect answers is a challenging pragmatic task, and even human annotators might not agree on a single gold label. We experiment with a way to leverage disagreement in labeling, which proves encouraging: the macro F1-score of the two best CNN models further improves, when fine-tuning with a crowd layer that encodes individual annotator preferences. This is encouraging, as — to the best of our knowledge — human disagreement has not yet been leveraged in modeling for understanding indirect answers; our dataset can be considered a starting point for further research in learning from disagreement.

Overall, the data in FRIENDS-QIA was originally written in a manuscript for the TV series, then uttered in a spoken dialogue context on the show and ultimately transcribed again. Our results show that understanding indirect answers remains a challenging task. We are missing out on influential factors such as intonation, body language and discourse relations from this dialogue context, when we only process the QA pairs in written form. Modeling such factors is interesting and challenging future work.

Acknowledgements

We would like to thank the anonymous reviewers for their thorough feedback, as well as ITU's High-performance Computing Cluster team for providing computational resources used in this work. This research is in part supported by the Independent Research Fund Denmark (DFF) grant 9063-00077B.

References

Lora Aroyo and Chris Welty. 2015. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Magazine*, 36(1):15–24.

Emily M. Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. 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 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Marie-Catherine de Marneffe, Scott Grimm, and Christopher Potts. 2009. Not a simple yes or no: Uncertainty in indirect answers. In *Proceedings of the SIGDIAL 2009 Conference*, pages 136–143, London, UK. Association for Computational Linguistics.

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.

Nancy Green and Sandra Carberry. 1992. Conversational implicatures in indirect replies. In *30th Annual Meeting of the Association for Computational Linguistics*, pages 64–71, Newark, Delaware, USA. Association for Computational Linguistics.

Nancy Green and Sandra Carberry. 1999. Interpreting and generating indirect answers. *Computational Linguistics*, 25(3):389–435.

Beth Ann Hockey, Deborah Rossen-Knill, Beverly Spejewski, Matthew Stone, and Stephen Isard. 1997. Can you predict responses to yes/no questions? yes, no, and stuff. In *Fifth European Conference on Speech Communication and Technology*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. volume abs/1412.6980.

Annie Louis, Dan Roth, and Filip Radlinski. 2020. "I'd rather just go to bed": Understanding indirect answers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7411–7425, Online. Association for Computational Linguistics.

Jennimaria Palomaki, Olivia Rhinehart, and Michael Tseng. 2018. A case for a range of acceptable annotations. In *SAD/CrowdBias@ HCOMP*, pages 19–31.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, page 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 742–751, Gothenburg, Sweden. Association for Computational Linguistics.

Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. 2013. Learning from multiple annotators: Distinguishing good from random labelers. *Pattern Recognition Letters*, 34(12):1428–1436.

Filipe Rodrigues and Francisco C. Pereira. 2018. Deep learning from crowds. In *Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*. Association for the Advancement of Artificial Intelligence.

Anna-Brita Stenström. 1984. *Questions and responses in English conversation*. Krieger Pub Co.

Henry S. Thompson, Anne H. Anderson, Ellen Gurman Bard, Gwyneth Doherty-sneddon, Alison Newlands, and Cathy Sotillo. 1993. The HCRC map task corpus: natural dialogue for speech recognition. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Alexandra N. Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2021. Learning from disagreement: A survey. *The Journal of Artificial Intelligence Research*, Forthcoming.

Yan Yan, Rómer Rosales, Glenn Fung, Ramanathan Subramanian, and Jennifer Dy. 2014. Learning from multiple annotators with varying expertise. *Machine Learning*, page 291–327.

A FRIENDS-QIA Data Statement

Following (Bender and Friedman, 2018), the following outlines the data statement for FRIENDS-QIA:

A. CURATION RATIONALE: Collection of examples of polar questions and curated answers for identification of indirect answer types.

B. LANGUAGE VARIETY: US (en-US) mainstream English.

C. SPEAKER DEMOGRAPHIC: Characters in a TV series.

D. ANNOTATOR DEMOGRAPHIC: All annotators are female, highly proficient in English. Native languages include Danish and Polish. Socioeconomic status: higher-education student.

D. SPEECH SITUATION: Both standard and colloquial US English, i.e., spontaneous speech.

D. TEXT CHARACTERISTICS: Sentences from transcripts of a TV series.

PROVENANCE APPENDIX: The data originates from `https://fangj.github.io/friends/`.

B FRIENDS-QIA Direct Mappers

B.1 Direct mappers which are removed

To avoid having direct mappers or cues in the answers, we remove all occurrences of the words listed below, if they "stand alone".

- Yes / yeah / yup / yep / yeah-eah / yah
- No / nope / nah / na / noo(...) / na-ah

B.2 Direct mappers which are *not* removed

We had to draw a line somewhere, between what to keep and what to remove. We ultimately decided to only remove variants of "yes" and "no" listed in the previous section. This results in keeping terms such as the ones listed below, which otherwise in certain cases might also act as a direct mapper or cue to the answer meaning "yes" or "no".

- Of course (not) / absolutely (not) / totally (not) / definitely (not) / certainly (not) / obviously (not) / apparently (not) / I guess (not)
- Alright / all right / affirmative / that's right / right / that's correct / sure / exactly / fine / please / okay / OK / 'kay / exclusively
- No way / no can do
- Thank you / thanks / no thanks / I don't know / maybe / kind of / (not) really (not)
- Uhuh / a-huh / mm-hm / mm-mh

C FRIENDS-QIA Variables

Here we list the variables which FRIENDS-QIA contains as well as a description of them.

- SEASON is the season of the TV series from which the QA pair was extracted.
- EPISODE is the episode (of the season) of the TV series from which the QA pair was extracted.
- CATEGORY tells the category which the QA pair was assigned to during data collection in order to know which preprocessing had to be performed.
- Q_PERSON is the name of the character of "Friends" who asked the question.
- A_PERSON is the name of the character of "Friends" who said the answer.
- Q_ORIGINAL is the original, non-modified question taken directly from the transcription.
- A_ORIGINAL is the original, non-modified answer taken directly from the transcription.
- Q_MODIFIED is the modified question (might be exactly the same as Q_ORIGINAL, if no modification was needed).
- A_MODIFIED is the modified answer (might be exactly the same as A_ORIGINAL, if no modification was needed).
- ANNOTATION_1 is the annotation given by annotator 1.
- ANNOTATION_2 is the annotation given by annotator 2.
- ANNOTATION_3 is the annotation given by annotator 3.
- GOLDSTANDARD is the aggregated gold label from the three annotators.