MEETINGQA: Extractive Question-Answering on Meeting Transcripts

Archiki Prasad[1] **Trung Bui**[2] **Seunghyun Yoon**[2] **Hanieh Deilamsalehy**[2]
Franck Dernoncourt[2] **Mohit Bansal**[1]
[1]UNC Chapel Hill [2]Adobe Research
{archiki, mbansal}@cs.unc.edu {bui, syoon, deilamsa, dernonco}@adobe.com

Abstract

With the ubiquitous use of online meeting platforms and robust automatic speech recognition systems, meeting transcripts have emerged as a promising domain for natural language tasks. Most recent works on meeting transcripts primarily focus on summarization and extraction of action items. However, meeting discussions also have a useful question-answering (QA) component, crucial to understanding the discourse or meeting content, and can be used to build interactive interfaces on top of long transcripts. Hence, in this work, we leverage this inherent QA component of meeting discussions and introduce MEETINGQA, an extractive QA dataset comprising of questions asked by meeting participants and corresponding responses. As a result, questions can be open-ended and actively seek discussions, while the answers can be multi-span and distributed across multiple speakers. Our comprehensive empirical study of several robust baselines including long-context language models and recent instruction-tuned models reveals that models perform poorly on this task (F1 = 57.3) and severely lag behind human performance (F1 = 84.6), thus presenting a challenging new task for the community to improve upon.[1]

1 Introduction

Millions of meetings occur every day worldwide, which results in vast amounts of meeting transcripts. Meeting transcripts are typically long documents, often domain-specific depending on the subject matter, and contain a lot of information. Basic tasks such as catching up with a missed meeting, looking up a specific discussion or response to a query can be time-consuming. These tasks can be facilitated by NLP systems, including summarization and question-answering. To this end, several publicly available small-scale corpora of meeting transcripts have been released (Carletta et al., 2005; Janin et al., 2003; Garofolo et al., 2004, *inter alia*).

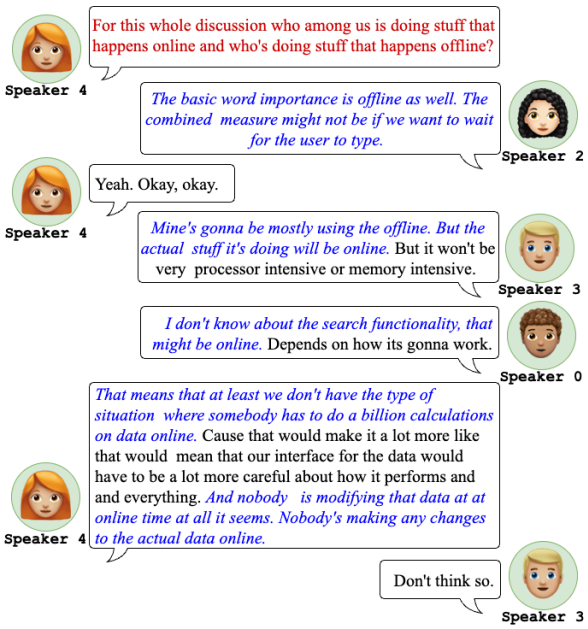

Figure 1: Representative example from meeting transcript segment in MEETINGQA. The question and annotated answer are highlighted in red and *blue* respectively.

Prior NLP work on meeting transcripts mainly focuses on summarization (Oya et al., 2014; Li et al., 2019; Zhu et al., 2020, *inter alia*). However, lack of annotated data impedes research on other important NLP tasks in this domain. To address this gap, we introduce a question-answering (QA) task based on conversations in meeting transcripts. Specifically, we consider *questions asked by participants during the meeting* and aim to extract corresponding answer spans from relevant discussions among meeting participants (refer to Figure 1). This task has several practical applications such as building an interactive meeting browser/interface for navigating through transcripts and informing tasks such as meeting summarization and handling action items involving QA pairs (Kathol and Tur, 2008; August et al., 2022).

[1]MEETINGQA data and code is available at https://archiki.github.io/meetingqa.html

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 15000–15025
July 9-14, 2023

While standard QA datasets consist of human generated questions either based on short supplied contexts (Rajpurkar et al., 2016, 2018; Rogers et al., 2021) or are answered using a large collection of documents (Joshi et al., 2017; Kwiatkowski et al., 2019; Zhu et al., 2021b), our task setting is challenging yet interesting in several ways. First, meeting transcripts are long documents and QA systems still struggle to understand long contexts (Pang et al., 2022; Soleimani et al., 2021). Second, successfully answering questions asked within meetings requires robust understanding of the conversation and discourse that takes place both before and after a question. Third, the multi-party spoken text falls under a different domain when compared to typical text documents. While standard long documents rarely include any meaningful (non-rhetorical) questions, multi-party meetings often involve discussions asked by one participant and answered by the rest, allowing us to use these questions to create a QA dataset. Furthermore, the conversational nature of transcribed text differs from written documents and may contain disfluencies and other artifacts. Finally, instead of using annotator-generated questions (like in Wu et al. (2022)), questions asked by participants are more open-ended and discussion-seeking, with interesting answer types that can be multi-span and/or contributed by multiple speakers (e.g., Figure 1).

To this end, we first introduce our dataset MEETINGQA, created by annotating meetings transcripts from the popular AMI (Augmented Multi-party Interaction) corpus, containing over 100 hours of meetings (Carletta et al., 2005), via a robust annotation pipeline . MEETINGQA comprises of 7,735 questions asked by participants across 166 meetings. Unlike other datasets, questions in MEETINGQA are less concise (12 words on average) and reflect queries asked in a conversational setting. The answers include realistic situations such as rhetorical questions, multiple discontiguous spans and/or contributions from multiple speakers.

Next, on MEETINGQA dataset, we test diverse models designed for long input contexts such as Longformer (Beltagy et al., 2020), and BigBird (Zaheer et al., 2020) as well as RoBERTa (Liu et al., 2019), and DeBERTa-v3 (He et al., 2020) with as much meeting context surrounding the question as possible. To incorporate the multi-span nature of answers in our dataset, we design and experiment with multi-span variants of the aforementioned models. Furthermore, we also investigate how well recent instruction-tuned large language models fare at answering questions from MEETINGQA. Lastly, we create a silver-annotation pipeline using MEDIASUM (Zhu et al., 2021a), a corpus containing 463.6K short interview transcripts, to provide additional training data. We find that the best performance is achieved by finetuned short-context models (F1 = 57.3). Overall, we show that models struggle to identify rhetorical questions and selecting which utterances constitute the answer. Thus, model performance significantly trails behind human performance on MEETINGQA (F1 = 84.6), leaving a large potential for future improvements on this challenging task.

2 Our Dataset: MEETINGQA

We first describe our data collection process in Section 2.1 and then provide an extensive analysis of MEETINGQA in Section 2.2.

2.1 Data Collection

Question Selection. We leverage the punctuated text to identify possible questions (ending with '?'). We also filter out questions containing ≤ 2 words as we manually find them to be either meaningless or rhetorical. While questions are marked to facilitate annotators, we encourage them to find missed potential questions due to incorrect punctuation.

Answer Annotation. For each possible question, we ask annotators to label the set of sentences (each identified by a unique number) from the meeting transcript that form the answer. Additionally, we also collect meta-data about the question. First, we ask the annotators to label if the question was *meaningful*, used to filter out rhetorical, unanswered or logistical questions and incorrect punctuations. Some speakers can ask consecutive or multiple questions in the same turn that are often related and answered together. In such scenarios, we allow annotators to combine questions and provide a common answer from the meeting transcript. The annotators mark these questions using the *combined question* attribute. Finally, since our questions are conversation segments, they may not be self-contained. Hence, we ask annotators to mention the *question context* sentences (if any) separately. We refer readers to Appendix A for more details and examples from MEETINGQA.

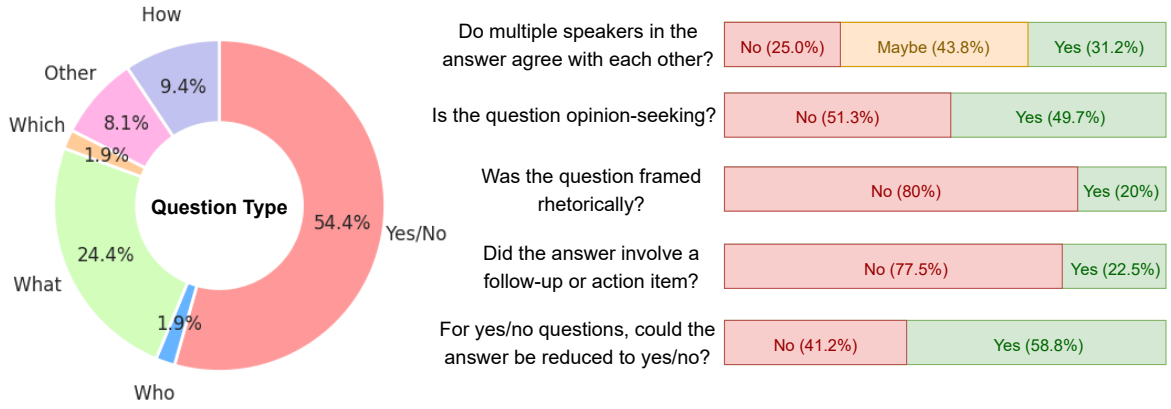

Figure 2: Analysis of 200 randomly selected questions from MEETINGQA. On left, we show the distribution of question types. On right, we show additional properties of *answerable* questions: level of agreement in multi-speaker answers, subjectivity of questions, question-framing, follow-ups and answer ambiguity in yes/no questions.

Annotation Process. All annotators were hired by a professional crowdsourcing company TELUS.[2] The company obtained consent from the crowd workers and conducted ethical reviews. To train annotators, we provide comprehensive instructions for each type of annotation with several manually annotated examples from a small subset of transcripts and different possible scenarios curated by the first author. The annotations were collected in multiple batches, starting with the first batch containing a small subset of 250 questions. We iteratively provided extensive feedback to the crowdworkers on their annotations and resolved existing issues till the annotations were satisfactory. Next, we assigned three independent annotators to each question, and calculated Krippendorff's $\alpha = 0.73$ (Krippendorff, 1980) using MASI-distance (Passonneau, 2006), indicating substantial agreement. We then collected annotations for the remaining questions in two additional batches using one annotator per question followed by a quality assurance stage to validate the outcome of the annotations. Overall, we spent $10,427 in the annotation process, amounting to $61 per meeting. For additional details refer to Appendix A.

2.2 Dataset Information and Analysis

After filtering and quality control, we were left with a total of 7,735 questions from 166 meetings ($\approx$ 100 hours of meeting recordings).

Size and Splits. We split our dataset into train, dev, and test sets such that questions in each split come from distinct meetings. Table 1 shows dataset statistics across different answer types, namely *unanswerable*, *multi-span*, and *multi-speaker* (described below). Due to relatively small number of meetings in the AMI corpus and diversity in meeting content, our test set contains a larger fraction of questions from the dataset as opposed to the conventional 80:10:10 split across train/dev/test sets.

	Train	Dev	Test
Number of Meetings	64	48	54
Number of Questions	3007	2252	2476
w/ No Answer	956	621	764
w/ Multi-Span Answers	787	548	663
w/ Multi-Speaker Answers	1016	737	840
Avg. Questions per Meeting	46.98	46.92	45.85

Table 1: Dataset statistics of MEETINGQA.

Question Types. Unlike most QA datasets, questions in MEETINGQA are extracted directly from the meeting transcripts. Consequently, we find that questions may not be concise, and may not begin with 'wh' prefixes, making our dataset challenging yet interesting for the community. We perform a manual analysis of question types based on 200 randomly selected questions from the test set in Figure 2 (left). First, we observe that a majority of questions in MEETINGQA are framed in a 'yes/no' manner, followed by 'what' and 'how' questions that are typically information-seeking. We find that in a discussion-heavy setting such as ours, yes/no questions elicit a detailed response that cannot be reduced to a direct 'yes/no' response in over 40% of the cases (see Figure 2 (right)). Further, manual analysis shows that nearly half the questions are subjective, i.e., seeking opinions of meeting participants, and as high as 20% of answerable questions

[2] https://www.telusinternational.com/

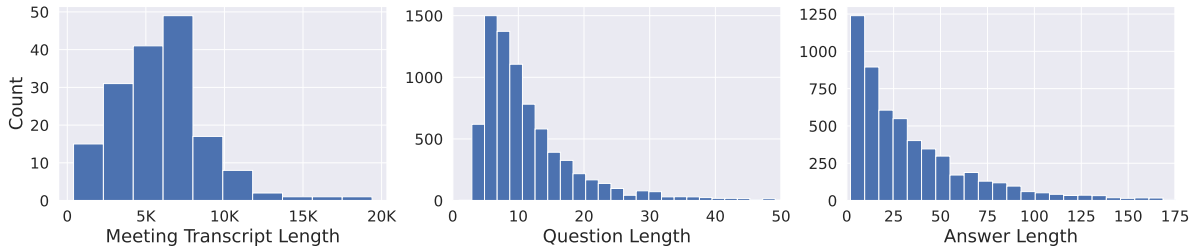

Figure 3: Number of words (length) in a meeting transcript, question and answer (for answerable questions) in MEETINGQA. The average length of a transcript, question, and answer is 5.9K, 12, and 35 words respectively, while the maximum length of a transcript, question, and answer is 19.4K, 155, and 305 words respectively.

are framed rhetorically. Appendix A contains additional tri-gram-based analysis of questions.

Length. Figure 3 shows the distribution of the length of meeting transcripts, questions, and answers in MEETINGQA. On average, each meeting transcript comprises of 5.8K words which constitute as long documents unlikely to fit entirely in the input context of typical pretrained language models (Devlin et al., 2019; Liu et al., 2019; He et al., 2020). Further, questions and their answers contain an average of 12, and 35 words respectively.

Answer Types. Due to the nature of meeting conversations and questions asked by participants, most answers are direct responses or follow-up discussions. However, some questions are rhetorical or do not elicit any discussion. These questions are *unanswerable* (30% of MEETINGQA). Among answerable questions, we note two scenarios of interest: *multi-span* and *multi-speaker* answers. Multi-span answers contain non-consecutive and discontinuous utterances or sentences, typically in the form of relevant discussion interleaved with irrelevant chit-chat (see examples in Appendix A). Additionally, multi-speaker answers occur when multiple participants contribute to answering a question which is typical in a discussion. Note that multi-speaker and multi-span answer cases are not mutually exclusive (refer to Figure 1 for an example). We find that 40% of all answers (excluding unanswerable questions) in our dataset are multi-span and 48% of answers are multi-speaker in nature. Moreover, Figure 2 (right) shows from our manual analysis that a considerable amount of disagreement exists among speakers in multi-speaker answers, with approximately 70% of cases displaying some form of disagreement. Notably, 22% of answers involve additional follow-up or action items, which are specific to the context of meetings.

Human Performance. We estimate human performance on MEETINGQA using a random subsample of 250 questions from the test split. Each question is assigned a different annotator who had not previously annotated the meeting containing that question. Scoring the provided answers relative to the reference answers in our dataset, yields an F1 of 84.6. This breaks down to F1 of 80.7 and 86.3 for unanswerable and answerable questions respectively. The F1 score for multi-span and multi-speaker answers is 88.1 and 87.7 respectively.

3 Methods

In this section, we investigate the difficulty level of our new MEETINGQA for state-of-the-art QA systems and establish strong baseline results. We describe strategies for retrieving contexts from transcripts in Section 3.1, followed by different QA models in Section 3.2, and silver data annotation for data augmentation methods in Section 3.3.

3.1 Retrieving Contexts from Transcripts

Given that meeting transcripts are very long documents, it is infeasible to input the entire transcript as context to typical QA models. Thus, we first select a smaller transcript segment that fits the model's input length limitations. We explore two strategies to retrieve contexts as described below.

Location-based Context Retrieval. We use the relative location of the question in the meeting transcript to retrieve a context by fitting as many (complete) sentences as possible under a fixed length budget (measured in words). Further, we split this budget into two components: *prefix* and *suffix* referring to the sentences that precede and succeed the question respectively. We set the prefix budget to 50 words and the suffix budget to 250 words respectively, resulting in a total budget of 300 words.[3]

[3]Ensures context fits into QA models limited 512 tokens.

Retrieval Method	Answer-Span Overlap	Upper Bound	
		F1	IoU
Location	99.20	99.99	99.98
ROUGE-1	14.81	23.21	19.95
Embedding Cos. Sim.	32.45	38.24	34.17

Table 2: Upper-bound performance of different retrieval methods for answerable questions split. Answer-span overlap measures the relative number of sentences in the annotated answer span present in the context (%).

Note that the suffix budget is significantly larger than the prefix budget since we expect to find answers in sentences following the question. The sentences before the question only provide additional context to the ongoing discussion.

Score-based Context Retrieval. Alternatively, we use the question as a query and compare it to other sentences from the entire transcript via two scoring methods consistent with Pang et al. (2021). First, we retrieve sentences using ROUGE-1 score relative to the question. Second, we use cosine similarity based on sentence embeddings (Reimers and Gurevych, 2019). We concatenate sentences in the order they appear in the transcript until reaching the total length budget. Similar to location-based retrieval, we set the total budget to 300 words.

Results of Context Retrieval. Table 2 compares both retrieval methods using the same total length budget on the answerable questions split. We observe that the sentence-level overlap between extracted contexts and annotated answers for score-based retrieval is significantly lower than for location-based retrieval. We use this overlap to compute the maximum achievable performance of QA systems for each type of retrieval. Correspondingly, we find similar trends in upper-bound performance metrics (discussed in Section 4) with location-based contexts (near-perfect max F1) considerably outperforming score-based contexts (max F1 < 40). Therefore, for short-context models, we henceforth use location-based contexts.

3.2 Models for Meeting-based QA

We primarily focus on extractive models including both short and long-context models. Given the transcript or a segment from it (context) and the question, models are tasked with extracting answer-span(s) from the context. We use two high-performing short-context models RoBERTa and DeBERTaV3, each supporting up to 512 tokens, with extracted context from Section 3.1. Additionally, we explore Longformer and BigBird which support longer sequences of up to 4096 tokens by utilizing a combination of sliding window and global attention mechanisms. Further, the Longformer Encoder-Decoder (LED) model supports up to 16,384 input tokens. These models allow us to use most or all portions of the transcript needed for answering the questions as the context. In case of an overflow, we use as many utterances from the transcript around the question as possible and truncate the rest. Note that these models output a single answer-span by default. Therefore, for multi-span answers, we train models to predict a span starting with first utterance and ending with the last utterance of the gold answer.

Multi-Span Models. In order to better model multi-span answers, we follow Segal et al. (2020) and pose multi-span QA as a sequence tagging task, predicting if each token in the context is part of the answer. For simplicity, we restrict ourselves to their proposed `IO` tagging. Thus, the answer prediction is a concatenation of all token-spans contiguously tagged with `I`. Similar to single-span models, we train multi-span variants of RoBERTa, DeBERTa, Longformer, and BigBird models.

Instruction-Tuned Models. Furthermore, we use FLAN-T5 (Chung et al., 2022), a publicly-available instruction-tuned model, to study zero-shot performance on our MEETINGQA. Given the relatively large size of contexts and distinct nature of our task, we rely on succinct instructions instead of few-shot demonstrations. Furthermore, due to the model's generative nature, we cannot directly use the predictions for our extractive QA task. Therefore, we adapt instruction-tuned models for our setting by employing instructions that ask models to *list sentences* instead of directly generating answers that may be less faithful to the context. Next, we filter out predicted sentences not present in the context. While this is a strict selection criterion, it removes any possible hallucinations.[4]

3.3 Silver Data Augmentation

Due to high annotation costs of gold labels, and unavailability of similar QA datasets, we investigate automatic methods to annotate answers. We match the salient features of MEETINGQA, such as meaningful questions within the transcript and

[4]Appendix E shows these choices improve overall scores.

Model	Intermediate Train Data	Overall	No Answer	Answerable		
				All	Multi-Span	Multi-Speaker
		F1 / IoU	F1†	F1 / IoU	F1 / IoU	F1 / IoU
RoBERTa-base	–	56.5 / 51.1	41.0	63.1 / 55.6	**60.8 / 50.1**	64.1 / 54.7
	SQuADv2	54.1 / 49.4	37.4	61.5 / 54.7	50.8 / 40.2	56.2 / 46.9
	+ silver	55.4 / 50.7	47.4	58.9 / 52.2	57.2 / 46.9	60.2 / 51.4
DeBERTa-base	–	**57.3 / 52.9**	55.8	58.0 / 51.6	49.6 / 39.3	55.3 / 46.7
	SQuADv2	56.5 / 52.1	51.0	58.9 / 52.6	49.6 / 39.1	55.7 / 47.2
	+ silver	55.2 / 50.4	46.7	59.0 / 52.0	51.4 / 40.5	57.7 / 48.6
Longformer-base	–	55.6 / 50.9	46.1	59.9 / 53.0	55.3 / 44.9	59.4 / 50.4
	SQuADv2	54.2 / 49.1	31.4	**64.4 / 56.9**	58.0 / 47.2	**62.6** / 53.0
	+ silver	54.9 / 50.2	51.2	56.6 / 49.8	54.5 / 44.0	58.6 / 49.9
LED-base	–	27.8 / 25.0	**59.0**	13.9 / 9.7	12.1 / 7.0	12.4 / 7.4
BigBird-base	–	53.7 / 48.6	44.4	57.8 / 50.4	58.1 / 47.5	**62.6 / 53.4**
	TriviaQA	54.5 / 49.5	35.2	63.2 / 55.9	56.3 / 45.5	60.6 / 51.1
	+ silver	54.7 / 49.8	43.7	59.6 / 52.4	57.6 / 46.9	60.5 / 51.2
Turn-based Baseline	–	35.9 / 30.4	0.5	51.8 / 43.8	42.0 / 31.5	47.3 / 40.0
Human Performance	–	**84.6 / 83.5**	**80.7**	**86.3 / 84.6**	**88.1 / 86.2**	**87.7 / 85.3**

Table 3: Comparing performance of finetuned single-span models and human performance on across answer types (best numbers in bold). Intermediate Train Data denotes the intermediate training data used, lack of which indicates direct finetuning. †All scores for unanswerable questions are equal as the reference string is empty.

multi-speaker discussions using the MEDIASUM dataset (Zhu et al., 2021a). This dataset contains 463.6K short multi-party interview transcripts, detailed speaker information, and identifies a host or interviewer who steers the discussion via questions.

We begin by identifying the host speaker and focusing on their questions. Next, we predict which speaker(s) would answer the question by identifying speaker entities mentioned in utterances or from previous dialogue turns. Finally, we search utterances from the identified speakers until a stopping criterion is met and label it as the answer. Due to the assumptions made in the above process, models trained directly on this data could overfit on spurious correlations (Jia and Liang, 2017; Wang and Bansal, 2018). Thus, we apply various perturbations to the context such as separating the question and answer utterances, converting to unanswerable questions by removing relevant sentences, creating more speaker transitions, and masking speaker names. Refer to Appendix F for additional details.

4 Experiments and Results

Evaluation Metrics. Following Rajpurkar et al. (2016) we report macro-averaged F1 on the entire test set as well as on specific answer types (Section 2.2).[5] However, F1 treats sequences as bag-of-words, and thus, there can be a non-significant overlap between a random span and the target span for large span lengths. To address this, Soleimani et al. (2021) propose reporting Intersection-over-Union (IoU) defined as:

$$\text{IoU} = |p \cap t| \Big/ |p \cup t|,$$

where p and t are the predicted and target spans, respectively. Since our answer spans are much longer than those in SQuAD (refer to Figure 3), we also report macro-averaged IoU to measure performance.

Training Settings. We measure performance of various models in both finetuned and zero-shot settings. First, we directly finetune the base pretrained model on the model on MEETINGQA. Next, to supplement training data we explore intermediate-training (Phang et al., 2018; Pruksachatkun et al., 2020) with SQuAD v2.0 (Rajpurkar et al., 2018)[6] or a combination including silver data from Section 3.3 prior to finetuning on MEETINGQA, increasing the training data by 5x and 10x respectively. Additional details on checkpoints, hyperparameters, and training are present in Appendix B.

Turn-based Baseline. We devise a straightforward algorithm called turn-based baseline that is inspired by the automatic silver data annotation

[5]We also report exact match (EM) scores in Appendix C.

[6]SQuADv2.0 is used for all models except BigBird, for which we use TriviaQA due to lack of reliable existing model checkpoint on HuggingFace (Wolf et al., 2019).

Model	Int. Train Data	Overall	No Answer	Answerable		
				All	Multi-Span	Multi-Speaker
		F1 / IoU	F1	F1 / IoU	F1 / IoU	F1 / IoU
RoBERTa-base	–	54.0 / 48.1	41.1	59.8 / 51.4	58.2 / 47.2	60.9 / 50.9
	silver	**55.1 / 50.0**	40.1	61.9 / 54.5	56.4 / 45.8	60.0 / 50.2
DeBERTa-base	–	54.5 / 47.9	35.3	63.0 / 53.8	62.9 / 51.1	64.9 / 53.6
	silver	55.1 / 49.8	36.1	**63.6 / 56.1**	**63.0 / 52.7**	**66.1 / 56.5**
Longformer-base	–	53.8 / 48.2	39.4	60.3 / 52.3	58.8 / 48.3	62.0 / 52.0
	silver	52.3 / 48.0	**57.2**	50.2 / 44.0	47.2 / 38.0	49.0 / 40.8
BigBird-base	–	49.6 / 43.4	28.3	59.2 / 50.2	57.3 / 45.5	60.9 / 50.2
	silver	53.5 / 48.0	36.4	61.2 / 53.2	61.3 / 50.9	63.9 / 54.1

Table 4: Comparing performance of finetuned multi-span models across evaluation metrics and answer types.

algorithm explained in Section 3.3. In the turn-based baseline, when a speaker asks a question, the predicted answer includes all the subsequent utterances of other speakers until the same speaker gets another turn (stopping criterion). Note that, turn-based baseline assumes all questions can be answered and always provides single-span answers, although the predictions may be multi-speaker.

4.1 Results and Discussion

We report performance of various fine-tuned single-span, multi-span models in Tables 3, and 4 respectively on the test split of MEETINGQA. Further, we evaluate zero-shot performance in Table 5. We summarize our findings below and refer readers to Appendix C for additional results.

Main Baselines and Comparison with Human Performance. Results from Tables 3 and 4 show that single-span models (narrowly) outperform the multi-span models, with the best overall performance achieved by single-span variant of DeBERTa-base (overall F1 $= 57.3$). Other single-span variants of Longformer and BigBird achieve higher performance on answerable questions (up to F1 $= 64.4$) but have lesser overall performance due to lower F1 scores on unanswerable questions.[7] Comparing to the human performance (overall F1 $= 84.6$), we find at least a 25 point difference in overall F1 of all finetuned models. Across various answer types, the difference in F1 scores is still at least 20 points. Similar trends holds for EM and IoU metrics too.[8] In the zero-shot setting (refer to Table 5), the difference in overall scores with respect to human performance is even greater (≥ 44 points across all metrics). Furthermore, all finetuned models outperform the turn-based baseline (with the exception of LED-base), whereas the corresponding zero-shot variants fail to outperform the turn-based baseline on overall metrics. This suggests that our dataset is challenging for current QA systems, leaving significant scope for improvement via interesting future work.

Impact of Long-Context Models. We observe that in a majority cases short-context models (especially RoBERTa) outperforms long-context models (Longformer and BigBird) by 1-2 points. Furthermore, the LED model that completely fits 90% of transcripts has significantly lower overall score ($\approx$30 F1 point difference) due to poor performance on answerable questions.[9] We believe that the ability to fit larger contexts is traded-off by well-optimized design of short-context models. This is consistent with the findings of Pang et al. (2022) and suggests better long-context models may be needed to outperform shorter extractive models.

Impact of Multi-Span Models. Table 5 shows that in the zero-shot setting, multi-span variants slightly outperform their single-span counterparts for long-context models and slightly underperform for DeBERTa. In Appendix C, we find that within answer types zero-shot performance drops for unanswerable questions while improving for multi-span and multi-speaker answers. For finetuned models (Tables 3 and 4), the overall performance of multi-span models is comparable if not slightly less than single-span variants.[10] Notably, for short-context

[7] Model predictions may be biased against (or towards) empty spans impacting score of unanswerable questions.

[8] Following the order EM $\leq$ IoU $\leq$ F1 for all models.

[9] Due to this, we do not experiment further with LED models in multi-span, intermediate-training, and zero-shot settings.

[10] Note that we do not intermediate train multi-span models on standard extractive QA tasks such SQuAD v2.0. Therefore, gold training data for multi-span models is always scarce.

Model	Int. Train Data	F1	IoU
RoBERTa-base (SS)	SQuADv2	27.9	26.0
	+ silver	34.6	31.1
DeBERTa-base (SS)	SQuADv2	19.8	17.5
	+ silver	34.2	32.1
Longformer-base (SS)	SQuADv2	15.1	9.4
	+ silver	32.5	29.6
BigBird-base (SS)	TriviaQA	7.6	3.5
	+ silver	33.7	31.2
RoBERTa-base (MS)	silver	34.9	30.9
DeBERTa-base (MS)	silver	31.6	27.5
Longformer-base (MS)	silver	35.1	31.3
BigBird-base (MS)	silver	**35.3**	**31.7**
FLAN-T5 XL	–	33.8	26.1
FLAN-T5 XL (self ans)	–	34.0	28.6
FLAN-T5 XL (ext ans)	–	25.6	23.8

Table 5: Comparing performance of zero-shot models on all questions. Single-span and multi-span models are denoted by SS and MS respectively. Identifying answerable questions using FLAN-T5 is denoted by 'self ans' and 'ext ans' denotes use of external supervised model.

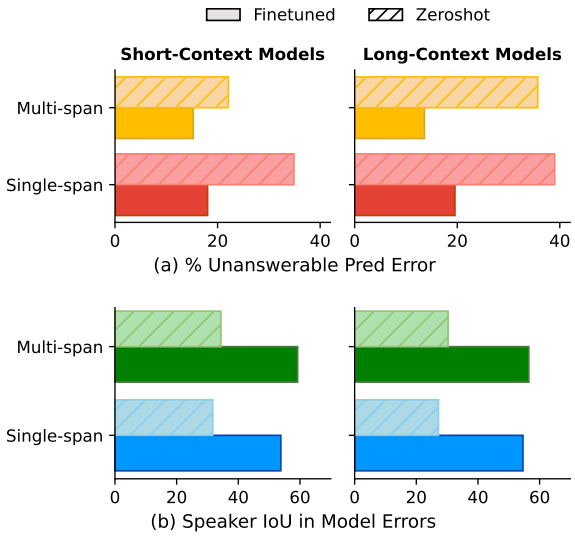

Figure 4: Error analysis of different model configurations on answerable questions. Top: percentage of errors where the model incorrectly predicts a question is unanswerable. Bottom: overlap with gold speakers in incorrect predictions on multi-speaker questions.

models, there is significant gain in performance for all answerable questions. Further, we observe that multi-span models consistently underperform on unanswerable questions (as high as 15 F1 points). Performance of multi-span model on unanswerable questions can be negatively impacted by even one false positive `I` tag, changing the prediction from unanswerable to answerable. While prior work on multi-span QA (Segal et al., 2020; Li et al., 2022) have found tagging-based approaches to outperform single-span variants, they only explore factoid questions on relatively shorter contexts. Future work can focus on improving multi-span QA for more open-ended questions like in MEETINGQA.

Impact of Intermediate Training. Silver data augmentation is effective in zero-shot settings with $\geq$15 point improvement for single-span long-context models (Table 5). For finetuned models, however, we do not observe significant improvements in overall scores from intermediate-training compared to directly finetuning. Interestingly, silver data augmentation improves performance on unanswerable questions for single-span models (except DeBERTA) and multi-span models.

Instruction-Tuned Models. Lastly, Table 5 shows zero-shot performance of instruction-tuned FLAN-T5 model. We find the FLAN-T5 XL model (3B parameters) outperforms most zero-shot single-span models and narrowly underperforms zero-shot multi-span models. Despite the design of instructions and filtering (Section 3.2), the model underperforms on unanswerable questions. Thus, we add an additional step to identify answerable questions and use model responses only for predicted answerable questions. The question classification can be done zero-shot using the same FLAN-T5 model[11] or by training an external supervised model.[12] We observe that using the FLAN-T5 model is more effective (yields best performance) than using a supervised model (6 F1 point drop) as the predictions of the latter are biased towards the question being unanswerable. Future work can further focus on accurately identifying answerable questions to improve overall performance.

Error Analysis. Next, we analyze some intriguing patterns in the errors within model predictions. Firstly, we observe that identifying rhetorical or unanswerable questions asked in a meeting is a challenging sub-task. Training a separate binary classification model that classifies whether a question is answerable based on the context from MEETINGQA yields only an F1$=$ 49.2 (see Appendix B). In Figure 4a, it becomes apparent that a significant portion of errors in predictions for answerable questions stem from the model incorrectly predicting that the question is rhetorical, particularly in the

[11] Different instruction like *based on the context, did anyone answer the given question?* This elicits a 'yes'/'no' response.

[12] Using the sequence classification head of RoBERTa-base model trained on questions from MEETINGQA (Appendix B).

zero-shot setting. Additionally, in case of multi-span answers, single-span models exhibit higher fraction of errors where predictions include sentences not present in the gold answer, in contrast to their multi-span counterparts (for details refer to Appendix D). This follows from the construction of single-span models, as described in Section 3.2. Lastly, for multi-speaker answers, we analyze the overlap in speakers (measured via IoU) of predicted and gold answers in Figure 4b. We find that even incorrect predictions of finetuned models contain roughly 55% speaker overlap with the gold answer, i.e., models can effectively predict which speakers answer the question. However, incorrect predictions in the zero-shot setting contain only 30% speaker overlap indicating that zero-shot models may struggle to predict which speakers answer the question. Future works can explore methods to effectively identify rhetorical questions and predict which speakers answer the question to improve overall performance. A more detailed analysis of errors can be found in Appendix D.

5 Related Work

Our work builds upon prior work on meeting transcripts and question answering. Rogers et al. (2021) provide a comprehensive survey of several QA datasets and formats.

Meeting Transcripts. Several other small-scale corpora of meeting recordings or transcripts are publicly available (Janin et al., 2003; Garofolo et al., 2004; Chen et al., 2005; Mostefa et al., 2007). We restrict ourselves the most popular and frequently used AMI corpus. Other works study various aspects of summarizing meeting transcripts (Mehdad et al., 2013; Wang and Cardie, 2013; Shang et al., 2018; Li et al., 2019; Zhu et al., 2020, *inter alia*) or extracting action-items (Morgan et al., 2006; Purver et al., 2007; Cohen et al., 2021). The work closest to ours uses Markov models to classify dialogue-acts as questions, answers or others (Kathol and Tur, 2008).

QA on Conversational Text. Prior work comprises of QA datasets based on small chit-chat from TV shows (Sun et al., 2019; Yang and Choi, 2019) or domain-specific chat-rooms (Li et al., 2020). The QAConv (Wu et al., 2022) dataset builds on these works with conversations from multiple domains (including MediaSum). However, these works employ human annotators for generating questions based on their understanding of the conversation resulting in straight-forward questions testing local information. Consequently, the answer spans of these datasets are significantly shorter, single-span, restricted to one speaker and often correspond to simple noun phrases (as high as 80% for QAConv). In contrast, questions asked by meeting participants are more open-ended, discussion-seeking, and correspond to longer answers ($\approx$ 7x) with complex multi-span and multi-speaker scenarios. Note that our work is different from conversational QA datasets that consist of a sequence of questions and answers simulating a conversation grounded in a short paragraph (Choi et al., 2018; Reddy et al., 2019; Campos et al., 2020).

Long-Context QA. Recent works show that QA models struggle to understand answer questions correctly using long contexts (Pang et al., 2022; Mou et al., 2021; Soleimani et al., 2021; Dasigi et al., 2021). However, unlike our work, the source (long) documents for building these datasets are taken from written-text domains such as books, film-scripts, research papers, or news articles.

6 Conclusion

In this work, we present MeetingQA, an extractive QA dataset based on meeting transcripts to identify answers to questions asked during discussion among meeting participants. Detailed analysis of the data reveals it is a challenging real-world task. Baseline experiments with a wide variety of models show the current performance lags behind human performance by at least 25 and 44 overall F1 points for finetuned and zeroshot models respectively. This demonstrates that current QA systems find our task challenging, leaving tremendous scope for improvement. We hope that future works will aim to bridge this gap and our work fosters research in NLP tasks (especially QA) on other text domains such as meeting transcripts.

Acknowledgements

We thank the reviewers and the area chairs for their helpful comments and feedback. We thank TELUS International for their help with data collection. We also thank Shiyue Zhang and Swarnadeep Saha for their helpful comments. This work was partially supported by NSF-CAREER Award 1846185, and NSF-AI Engage Institute DRL-2112635. The views contained in this article are those of the authors and not of the funding agency.

Limitations

Due to the structure of MEETINGQA, the answers to questions asked by participants (if any) are present in the transcript itself, making it an extractive task. Therefore, we do not extensively explore the use of generative models since the predictions do not stick to the sentences in the transcript and could possibly include hallucinations. However, we aim to mitigate hallucinations by using instruction-tuned generative models with suitably designed instructions and enforce a strict exact match criteria for filtering any possible hallucinations. Future work can explore how to adapt or evaluate non-instruction-tuned generative models on this task and better identify hallucinations with a more relaxed filtering to improve performance. We also do not report zero-shot performance of InstructGPT (Ouyang et al., 2022) as these models are not freely accessible. Additionally, we use a simple multi-span QA adaptation technique from Segal et al. (2020), but predicting answer spans by classifying each token can be difficult to train leading to slightly lower performance (discussed in Section 4.1). We hope our dataset provides additional motivation for future work on multi-span QA. Finally, MEETINGQA only comprises of publicly available meeting transcripts in English, but our methodology of data collection and model training (using multilingual variants) should still be applicable for other languages in future work.

Ethical Considerations

The human participants in our work were recruited by an external crowd-sourcing company that ensured annotators provided informed consent, were given fair compensation, and no personally identifiable information (PII) was collected or released. We use existing publicly available meeting transcripts collected by the AMI project (Carletta et al., 2005) in controlled scenarios and filtered for offensive/toxic content. We also conducted manual inspection of a random sample from annotated transcripts and did not find any toxic content or PII. Furthermore, the collected data and experiments are conducted in English and we do not claim generalization of our findings across languages. Given the broad nature of meetings, the content can fall into a number of domains, of which only a few are represented in the AMI corpus. Therefore, we do not expect models trained on MEETINGQA to generalize to certain domains such as judicial, ethical review, congressional proceedings, etc. which involve specific jargon and rules of engagement.

References

Tal August, Lucy Lu Wang, Jonathan Bragg, Marti A Hearst, Andrew Head, and Kyle Lo. 2022. Paper plain: Making medical research papers approachable to healthcare consumers with natural language processing. *arXiv preprint arXiv:2203.00130*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jon Ander Campos, Arantxa Otegi, Aitor Soroa, Jan Deriu, Mark Cieliebak, and Eneko Agirre. 2020. DoQA - accessing domain-specific FAQs via conversational QA. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7302–7314, Online. Association for Computational Linguistics.

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. 2005. The ami meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39. Springer.

Lei Chen, R Travis Rose, Ying Qiao, Irene Kimbara, Fey Parrill, Haleema Welji, Tony Xu Han, Jilin Tu, Zhongqiang Huang, Mary Harper, et al. 2005. Vace multimodal meeting corpus. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 40–51. Springer.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Amir Cohen, Amir Kantor, Sagi Hilleli, and Eyal Kolman. 2021. Automatic rephrasing of transcripts-based action items. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2862–2873, Online. Association for Computational Linguistics.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers. In *Proceedings of the 2021 Conference of the North American Chapter of*

the Association for Computational Linguistics: Human Language Technologies, pages 4599–4610, Online. 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.

John S Garofolo, Christophe Laprun, Martial Michel, Vincent M Stanford, Elham Tabassi, et al. 2004. The nist meeting room pilot corpus. In *LREC*. Citeseer.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Paria Jamshid Lou and Mark Johnson. 2020. Improving disfluency detection by self-training a self-attentive model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3754–3763, Online. Association for Computational Linguistics.

Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, et al. 2003. The icsi meeting corpus. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 1, pages I–I. IEEE.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Andreas Kathol and Gokhan Tur. 2008. Extracting question/answer pairs in multi-party meetings. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5053–5056. IEEE.

Klaus Krippendorff. 1980. *Content analysis: An Introduction to Its Methodology*. Sage publications.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Haonan Li, Martin Tomko, Maria Vasardani, and Timothy Baldwin. 2022. MultiSpanQA: A dataset for multi-span question answering. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1250–1260, Seattle, United States. Association for Computational Linguistics.

Jiaqi Li, Ming Liu, Min-Yen Kan, Zihao Zheng, Zekun Wang, Wenqiang Lei, Ting Liu, and Bing Qin. 2020. Molweni: A challenge multiparty dialogues-based machine reading comprehension dataset with discourse structure. *arXiv preprint arXiv:2004.05080*.

Manling Li, Lingyu Zhang, Heng Ji, and Richard J. Radke. 2019. Keep meeting summaries on topic: Abstractive multi-modal meeting summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2190–2196, Florence, Italy. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Yashar Mehdad, Giuseppe Carenini, Frank Tompa, and Raymond T. Ng. 2013. Abstractive meeting summarization with entailment and fusion. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 136–146, Sofia, Bulgaria. Association for Computational Linguistics.

William Morgan, Pi-Chuan Chang, Surabhi Gupta, and Jason M. Brenier. 2006. Automatically detecting action items in audio meeting recordings. In *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, pages 96–103, Sydney, Australia. Association for Computational Linguistics.

Djamel Mostefa, Nicolas Moreau, Khalid Choukri, Gerasimos Potamianos, Stephen M Chu, Ambrish Tyagi, Josep R Casas, Jordi Turmo, Luca Cristoforetti, Francesco Tobia, et al. 2007. The chil audiovisual corpus for lecture and meeting analysis inside smart rooms. *Language resources and evaluation*, 41(3):389–407.

Xiangyang Mou, Chenghao Yang, Mo Yu, Bingsheng Yao, Xiaoxiao Guo, Saloni Potdar, and Hui Su. 2021. Narrative question answering with cutting-edge open-domain QA techniques: A comprehensive study. *Transactions of the Association for Computational Linguistics*, 9:1032–1046.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang,

Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Tatsuro Oya, Yashar Mehdad, Giuseppe Carenini, and Raymond Ng. 2014. A template-based abstractive meeting summarization: Leveraging summary and source text relationships. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, pages 45–53, Philadelphia, Pennsylvania, U.S.A. Association for Computational Linguistics.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, and Samuel Bowman. 2022. QuALITY: Question answering with long input texts, yes! In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5336–5358, Seattle, United States. Association for Computational Linguistics.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, et al. 2021. Quality: Question answering with long input texts, yes! *arXiv preprint arXiv:2112.08608*.

Rebecca Passonneau. 2006. Measuring agreement on set-valued items (MASI) for semantic and pragmatic annotation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

Jason Phang, Thibault Févry, and Samuel R Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *arXiv preprint arXiv:1811.01088*.

Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R. Bowman. 2020. Intermediate-task transfer learning with pretrained language models: When and why does it work? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5231–5247, Online. Association for Computational Linguistics.

Matthew Purver, John Dowding, John Niekrasz, Patrick Ehlen, Sharareh Noorbaloochi, and Stanley Peters. 2007. Detecting and summarizing action items in multi-party dialogue. In *Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue*, pages 18–25, Antwerp, Belgium. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Anna Rogers, Matt Gardner, and Isabelle Augenstein. 2021. Qa dataset explosion: A taxonomy of nlp resources for question answering and reading comprehension. *arXiv preprint arXiv:2107.12708*.

Elad Segal, Avia Efrat, Mor Shoham, Amir Globerson, and Jonathan Berant. 2020. A simple and effective model for answering multi-span questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3074–3080, Online. Association for Computational Linguistics.

Guokan Shang, Wensi Ding, Zekun Zhang, Antoine Tixier, Polykarpos Meladianos, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2018. Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 664–674, Melbourne, Australia. Association for Computational Linguistics.

Amir Soleimani, Christof Monz, and Marcel Worring. 2021. NLQuAD: A non-factoid long question answering data set. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1245–1255, Online. Association for Computational Linguistics.

Kai Sun, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. 2019. DREAM: A challenge data set and models for dialogue-based reading comprehension. *Transactions of the Association for Computational Linguistics*, 7:217–231.

Lu Wang and Claire Cardie. 2013. Domain-independent abstract generation for focused meeting summarization. In *Proceedings of the 51st Annual Meeting of*

the Association for Computational Linguistics (Volume 1: Long Papers), pages 1395–1405, Sofia, Bulgaria. Association for Computational Linguistics.

Yicheng Wang and Mohit Bansal. 2018. Robust machine comprehension models via adversarial training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 575–581, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Chien-Sheng Wu, Andrea Madotto, Wenhao Liu, Pascale Fung, and Caiming Xiong. 2022. QAConv: Question answering on informative conversations. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5389–5411, Dublin, Ireland. Association for Computational Linguistics.

Zhengzhe Yang and Jinho D. Choi. 2019. FriendsQA: Open-domain question answering on TV show transcripts. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 188–197, Stockholm, Sweden. Association for Computational Linguistics.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33:17283–17297.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021a. MediaSum: A large-scale media interview dataset for dialogue summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5927–5934, Online. Association for Computational Linguistics.

Chenguang Zhu, Ruochen Xu, Michael Zeng, and Xuedong Huang. 2020. A hierarchical network for abstractive meeting summarization with cross-domain pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 194–203, Online. Association for Computational Linguistics.

Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021b. Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*.

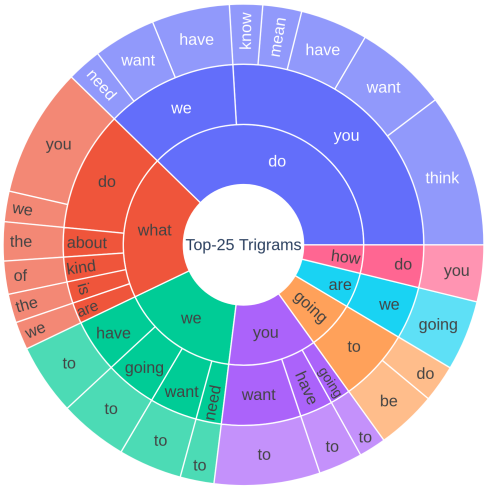

Figure 5: Top-25 most frequently occurring trigrams in questions from MEETINGQA.

A Additional Details on MEETINGQA

A.1 Tri-gram Analysis of Question Types

In contrast to most QA datasets, questions in MEETINGQA are extracted directly from the meeting transcripts and thus are conversation segments. Consequently, we find that questions may not be concise, often use auxiliary verbs, and do not typically begin with 'wh' or 'how' prefixes, making our new QA task and dataset challenging yet interesting for the community. This makes conventional analysis of question types based on prefixes less relevant here, and instead, we compute the top-25 most common trigrams from all questions, shown in Figure 5. The three most common question patterns are: '*do you/we ...*', and '*what ...*'. Additionally, the trigrams demonstrate that our questions are open-ended and seeking opinions or thoughts from other participants that tend to elicit long responses.

A.2 Dataset format and meta-information

We provide annotations for each meeting transcript at the sentence level in '.json' format, and each sentence has 4 primary attributes: `displayText`, `speakerFaceId`, `sentenceId`, and `question` which contain the sentence text, integer identifier of the speaker (unique within a meeting), integer identifier of the sentence, and information about the sentence as a question respectively. The question attribute is relevant only if the sentence is identified as a question. It contains additional attributes: `possible`, `meaningful`, `questionContext`, `combinedQuestion`, and `answerSpan`. First, we perform "ques-

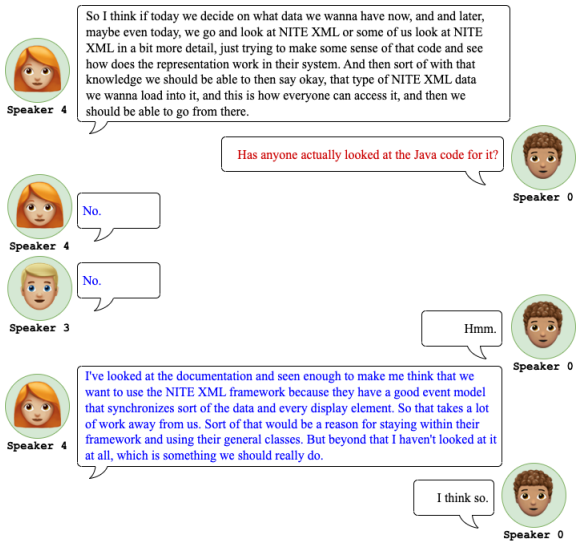

Figure 6: Illustrative QA example from a portion of meeting transcript in MEETINGQA. The question and annotated answer are highlighted in red and blue respectively. This example corresponds to a multi-speaker (`Speaker 0`, `3`, and `4`) and multi-span answer (due to chit-chat from `Speaker 0`).

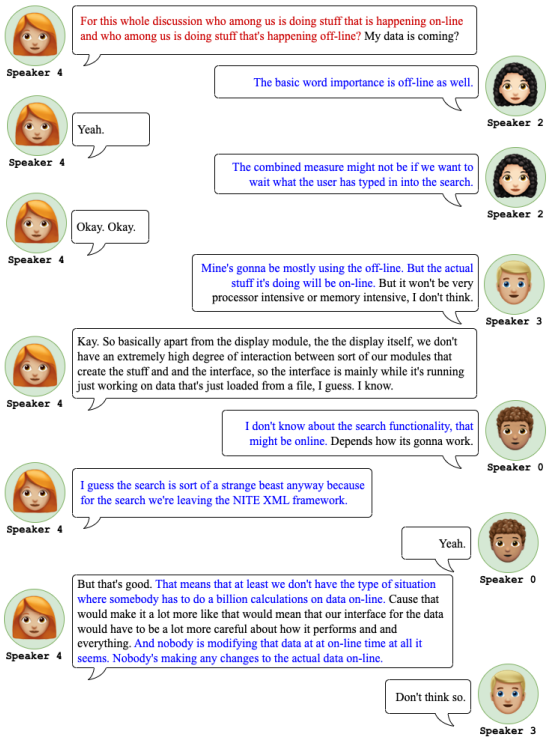

Figure 7: Illustrative QA example from a portion of meeting transcript in MEETINGQA. The question and annotated answer are highlighted in red and blue respectively. This example corresponds to a multi-speaker (`Speaker 0`, `2`, `3`, and `4`) and multi-span answer (due to chit-chat from `Speaker 0`, `3`, and `4`). The second question asked by `Speaker 4`, *"My data is coming?"*, is unanswerable/rhetorical and labeled with `meaningful = False`.

tion selection" described in Section 2.1 and set the possible tag as `True` to guide the annotators. The remaining attributes are set to default values `meaningful = False`, `answerSpan = []`, `questionContext = []`, and `combinedQuestion = []` prior to annotation. Annotators modify the `question` attribute during the course of the annotation and can even mark additional questions outsider our question selection criteria by setting `possible = True`. They lable the remaining attributes according to the "answer annotation" steps mentioned in Section 2.1. The list type attributes `questionContext`, `combinedQuestion`, and `answerSpan` contain sentences specified using the value of the corresponding `sentenceId` attributes. The domain of meeting transcripts (from AMI corpus) is a combination of elicited scenario-driven data, and natural data. We refer interested readers to the AMI project page for more information about the topic or scenario of each meeting.[13]

We find that out of 7.7K questions in MEETINGQA, only 66 ($< 1\%$) additional questions were identified by the annotators that were missed by our question selection criteria. Further, 751 questions (9.7%) were annotated with additional context sentences via `questionContext` and a total of 784 (10.1%) were combined with another question via `combinedQuestion` attribute. Among the latter, an average of 2.2 (maximum 4) questions were combined and these questions were an average of 1.5 sentences apart. The average length of `questionContext` (when annotated) was 1.7 sentences (maximum 3) which preceded the question by 1.7 sentences. Note that for the purposes of QA evaluation, we only use the `possible` and `answerSpan` attributes. The remaining attributes serve as meta-information to understand the dataset better and can facilitate error analysis and/or future work. Also to come up with overall question counts, we ignore the `combinedQuestion` attributes and count all the questions individually. Therefore, this attribute serves as an indicator of when and why different questions share the same answer. Empirically, we note that the combined

[13]Refer to documentation links: `https://groups.inf.ed.ac.uk/ami/corpus/scenariomeetings.shtml`, and `https://groups.inf.ed.ac.uk/ami/corpus/nonscenariomeetings.shtml`.

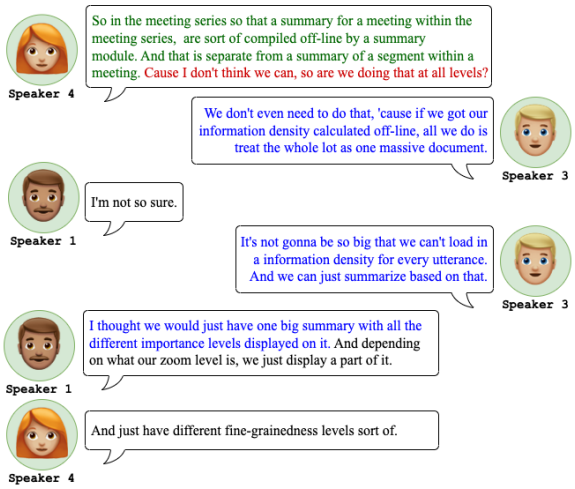

Figure 8: Illustrative QA example from a portion of meeting transcript in MEETINGQA. The question and annotated answer are highlighted in red and blue respectively. The first two sentences (in green) provide necessary context to understand the question. This example corresponds to a multi-speaker (`Speaker 1`, `3`, and `4`) and multi-span answer.

questions and question context typically fit within the contexts created using location-based retrieval (Section 3.1) and are present in the input fed into QA models in the vast majority of cases.

A.3 Additional Annotated Examples

Next, we show multiple examples of snippets from meeting transcripts with QA components present in MEETINGQA in Figures 6-11. Figure 7 also contains an example of an unanswerable question asked by `Speaker 4` (“*my data is coming?*”) which is either rhetorical or corresponds to incorrect punctuation. In such cases, annotators label `meaningful = False` and an empty/null answer annotation (`answerSpan` = []). On the other hand, Figure 11 also contains two consecutive questions asked by `Speaker 1` but the annotators mark both as `meaningful = True`, but choose to combine them via (`combinedQuestion`) and share a common answer. This because the first question is more generic, and the second question builds on top of it, by providing a specific example of what is loaded and what isn't. Further, in Figure 8 we provide an example of question which needs additional context sentences annotated via `questionContext`. Figures 6, 9, and 11 are diverse instances of multi-speaker and multi-span answers in our dataset.

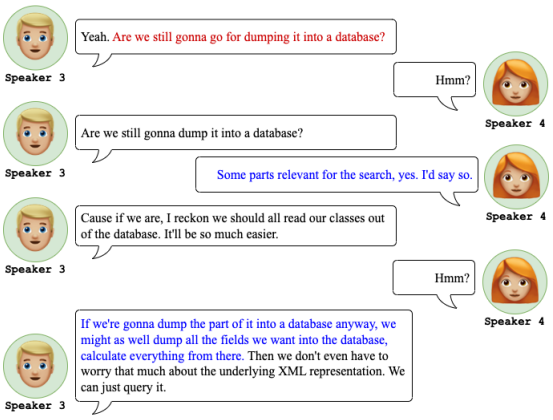

Figure 9: Illustrative QA example from a portion of meeting transcript in MEETINGQA. The question and annotated answer are highlighted in red and blue respectively. This example corresponds to a multi-speaker (`Speaker 3`, and `4`) and multi-span answer.

A.4 More on Data Collection

AMI Transcript Preprocessing. The AMI corpus is a collection of 171 meeting transcripts containing manually annotated and punctuated speaker-specific XML files for each meeting. We parse these XML files and combine utterances from multiple speakers by aligning the start times into a single transcript (with speaker information) corresponding to each meeting. We then use a disfluency detector model to identify and remove disfluencies from the utterances (Jamshid Lou and Johnson, 2020).

Annotator Recruitment and Training. All annotators are hired by a professional crowdsourcing company TELUS.[14] The company obtained consents from the crowdworkers before the annotation process and conducted ethical reviews. The company recruited 18 annotators, all based in the United States and native English speakers, who had previously successfully participated in text-based annotation projects. In addition to the instruction document (shared in the supplementary) curated by the first author, TELUS conducted a series of (virtual) meetings to deliver instructions, conduct example walk-through of the annotation and clarify doubts. At the end of initial training, a small batch of 5 meetings was provided to each of the annotators to calibrate performance. The responses were then compared to the good quality annotations performed by 2 project leads at TELUS manually in consultation with the authors. Feedback was pro-

[14] https://www.telusinternational.com/

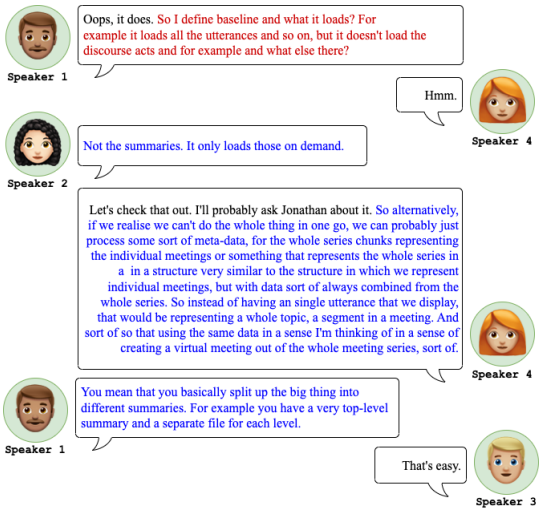

Figure 10: Illustrative QA example from a portion of meeting transcript in MEETINGQA. The question and annotated answer are highlighted in red and blue respectively. This example corresponds to a multi-speaker (`Speaker 1`, `2`, `3`, and `4`) and multi-span answer. `Speaker 1` asks two questions which are combined (via `combinedAnswer`) and share a common answer (same entry in `answerSpan`).

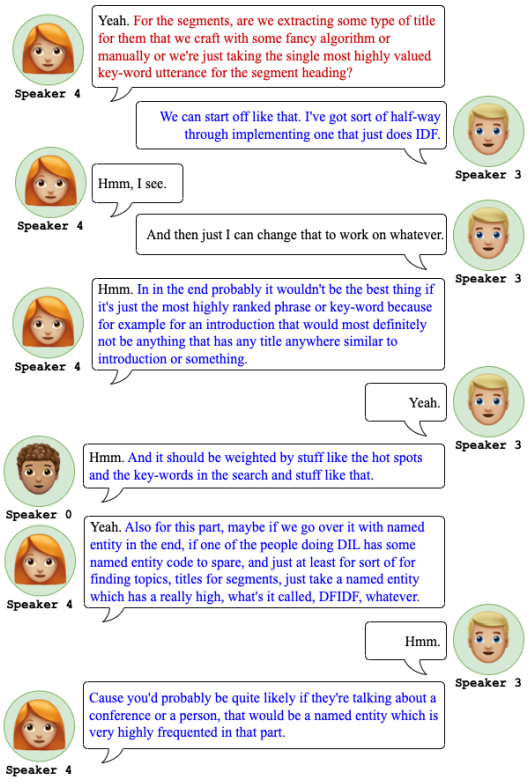

Figure 11: Illustrative QA example from a portion of meeting transcript in MEETINGQA. The question and annotated answer are highlighted in red and blue respectively. This example corresponds to a multi-speaker (`Speaker 0`, `3`, and `4`) and multi-span answer (due to chit-chat from `Speaker 0`, `3`, and `4`).

vided to the annotators to improve the quality of annotations, and based on their final responses the top-6 best performing annotators were selected to work on the project.

Quality Control. From the pool of selected annotators, project leads recruited two of the best performing and experienced annotators to help with quality control. Any annotated meeting transcript was assigned to either of these two annotators for review. Between the two, meeting annotated by one was assigned to the other for review. After the review, minor errors in annotation were fixed directly, otherwise major errors were sent back to the respective annotators for a re-annotation of the question and in some rare cases the annotation was redone by the reviewers. At the end of annotation batch (total of 4), the transcripts were sent to the authors who extensively reviewed them and provided feedback. We also looked for typos, and other issues which were fixed promptly. The TELUS project leads did not find any toxic and offensive content at their end and no such concerns were reported in the quality control stage. Further, all communication with the annotators is done by the crowdsourcing company and no personally identifiable information (PII) is released to the authors. Additionally, their execution platform contains unique identifiers for all annotators ensuring their PII is not released along with the annotated data. Finally, based on the feedback from annotators, we removed all questions corresponding to 5 meetings as the meeting content was hard to follow.

Time Taken. On an average across meeting transcript and annotators, it took ≈ 1 hour to annotate each meeting transcript which averages to ≈ 1.3 minutes per question. However, the per meeting annotation time strongly correlated with the length of the transcript (number of sentences). Among questions, annotators spent more time on answerable questions. This is because if the question is marked unanswerable (not meaningful), they did not have find exact answer span (in sentences) and annotated other meta-information. The project leads also ensured that the amount of time taken by each annotators was consistent with internal estimated by the intial batch used during annotator recruitment. The total production time for this project (for TELUS) was 186.5 hours.

Model	Int. Train Data	Exact Match (EM)			
		Overall	Answerable		
			All	M-Span	M-Speaker
RoBERTa (base)	–	28.8	23.3	0	12.1
	SQuADv2	31.0	**28.2**	0	12.0
	+ silver	30.3	22.7	0	11.6
DeBERTa (base)	–	**34.9**	25.6	0	**12.9**
	SQuADv2	34.6	27.3	0	**12.9**
	+ silver	31.4	24.5	0	12.2
Longformer (base)	–	31.0	24.3	0	11.7
	SQuADv2	27.3	25.5	0	11.8
	+ silver	30.8	21.7	0	13.0
BigBird (base)	–	31.0	24.3	0	11.7
	TriviaQA	34.6	27.3	0	**12.9**
	+ silver	31.4	24.5	0	12.2
Human	–	**75.2**	**73.0**	**72.5**	**66.4**

Table 6: Comparing exact match scores of finetuned single-span models and human performance and for different answer-types. M-Span and M-Speaker denote muli-span and multi-speaker splits respecitvely. No Answer EM (same as No Answer F1) present in Table 3.

Model	Int. Train Data	Exact Match (EM)			
		Overall	Answerable		
			All	M-Span	M-Speaker
RoBERTa (base)	–	26.0	19.3	7.0	11.3
	silver	30.3	**26.0**	8.0	12.8
DeBERTa (base)	–	25.2	20.6	7.9	11.2
	silver	29.2	26.1	**13.3**	**17.9**
Longformer (base)	–	26.4	20.6	7.6	10.1
	silver	**32.0**	20.7	5.0	12.0
BigBird (base)	–	20.0	16.3	7.7	9.1
	silver	25.8	21.0	4.8	12.9
Human	–	**75.2**	**73.0**	**72.5**	**66.4**

Table 7: Comparing exact match scores of finetuned multi-span models and human performance and for different answer-types. M-Span and M-Speaker denote muli-span and multi-speaker splits respecitvely. No Answer EM (same as No Answer F1) present in Table 4.

Compensation. The annotators were compensated through a fixed hourly rate defined for each participant. No additional bonus was provided to incentivize faster turnaround times. The average hourly wage for participants was roughly $20/hour in compliance with all the federal and local local laws to ensure fair payment.

B Experimental Details

GPU Compute. For training and/or inference we used a combination of 6 NVIDIA A10 24 GB GPUs and 2 NVIDIA RTX A6000 48GB GPUs. Directly finetuning on MEETINGQA starting from a pretrained checkpoints is quite fast and takes not more than 4 GPU hours (depending on the batch size). Intermediate training on the silver-annotated data (5 epochs) takes about 12-18 GPU hours (training time is higher for long-context models).

Hyperparameters. For the short-context models (RoBERTa and DeBERTa-v3) we used max sequence length of 512, (stride of 128 but not utilized due to location-based context retrieval), and batch size of 16. For long context models, we used max sequence length of 4096, stride of 128, and batch size varied between 8 and 16 (depending on GPU availability). Note that there is no stride for multi-span models. For model training on MEETINGQA, we use a learning rate of 3e-5, warmup ratio of 0.2, and train for 15 epochs with an early stopping criteria set to a patience of 2 epochs (F1 on dev split). For intermediate training with silver data, we use the same hyperparameters except we train for 8 epochs with the same early stop criteria. The hyperparameters values used are pretty standard and were not tuned explicitly for MEETINGQA.

Model Sizes. RoBERTa base and large models comprise of 125M and 355M parameters respectively, while DeBERTa base and large models comprise of 86M and 304M parameters. On the other hand, Longformer-base comprises of 149M parameters. Since BigBird is initialized with the RoBERTa checkpoints they share the same model size. Finally, instruction-tuned models FLAN-T5 LARGE, XL and XXL consist of 770M, 3B, and 11B model parameters respectively.

Pretrained Checkpoints. For models without intermediate-training we use the standard checkpoints for all models available on HuggingFace. For the score-based context retrieval in Section 3.1, we use HuggingFace's `evaluate` library for computing ROUGE-1 and the `multi-qa-MiniLM-L6-cos-v1` model from the `sentence-transformers` python package for embedding cosine similarity. During silver data annotation, we used the `en_core_web_sm` from `spacy` package for NER. For intermediate training (SS) we used the following pretrained checkpoints (base size):

- RoBERTa: deepset/roberta-base-squad2
- DeBERTa: deepset/deberta-v3-base-squad2

Model	Int. Train Data	Overall	No Answer	Answerable		
				All	Multi-Span	Multi-Speaker
		F1 / EM / IoU	F1†	F1 / EM / IoU	F1 / EM / IoU	F1 / EM / IoU
RoBERTa-base (SS)	SQuADv2	27.9 / **25.9** / 26.0	**80.2**	4.6 / 1.6 / 1.8	2.9 / 0 / 1.2	3.6 / 0.1 / 1.6
	+ silver	34.6 / 20.7 / 31.1	32.6	35.4 /15.4 / 30.4	26.3 / 0 / 19.2	28.1 / 1.7 / 21.0
DeBERTa-base (SS)	SQuADv2	19.8 / 16.2 / 17.5	50.3	6.2 / 1.0 / 2.9	5.6 / 0 / 2.6	6.0 / 0 / 2.9
	+ silver	34.2 / 25.4 / **32.1**	63.1	21.3 / 8.6 / 18.3	15.5 / 0 / 11.6	16.5 / 1.4 / 12.6
Longformer-base (SS)	SQuADv2	15.1 / 0 / 9.4	0.1	21.8 / 0 / 13.5	**32.8** / 0 / 21.3	28.8 / 0 / 18.3
	+ silver	32.5 / 20.5 / 29.6	39.8	29.2 / 11.9 / 25.0	23.3 / 0 /17.4	23.3 / 1.8 / 17.5
BigBird-base (SS)	SQuADv2	7.6 / 0.8 / 3.5	0.1	10.9 / 1.2/ 5.0	9.6 / 0 / 5.0	10.7 / 0 / 5.5
	+ silver	33.7 / 23.8 / 31.2	53.3	25.0 / 10.7 / 21.4	18.4 / 0 / 13.5	20.0 / 1.9 / 15.1
RoBERTa-base (MS)	silver	34.9 / 19.8 / 30.9	24.0	39.8 / **17.9** / **34.0**	29.2 / **0.3** / 20.9	29.4 / 0.5 / 21.2
DeBERTa-base (MS)	silver	31.6 / 17.0 / 27.5	15.8	38.6 / 17.5 / 32.7	27.2 / 0.2 / 19.4	27.4 / 0 / 19.2
Longformer-base (MS)	silver	35.1 / 21.3 / 31.3	32.6	36.2 / 16.3 / 30.9	25.7 / 0 / 18.5	27.3 / 0.4 / 19.9
BigBird-base (MS)	silver	**35.3** / 20.9 / 31.7	35.5	35.2 / 14.4 / 30.1	26.4 / 0.2 / 19.2	28.1 / 2.0 / 20.8
FLAN-T5 LARGE	–	26.0 / 12.4 / 20.6	17.4	29.8 / 10.2 / 22.0	23.9 / 0.2 / 15.7	25.6 / 1.6 / 17.1
FLAN-T5 LARGE (self ans)	–	26.3 / 13.0 / 21	20.0	29.1 9.9 / 21.4 /	23.5 / 0.2 / 15.4	24.9 / 1.6 / 16.6
FLAN-T5 LARGE (ext ans)	–	22.8 / 20.9 / 21.7	62.0	5.7 / 2.9 / 4.1	2.5 / 0 / 1.7	3.5 / 0.1/ 2.3
FLAN-T5 XL	–	33.8 / 17 / 26.1	15.6	**41.9** / 17.6 / 30.8	20.8 / 0.2 / 13.3	23.7 / **2.3** / 16.3
FLAN-T5 XL (self ans)	–	34.0 / 22.2 / 28.6	45.3	28.9 / 11.9 / 21.1	20.8 / 0.2 / 13.3	23.7 / **2.3** / 16.3
FLAN-T5 XL (ext ans)	–	25.6 / 22.8 / 23.8	62.0	9.4 / 5.3 / 6.8	4.4 / 0 / 2.8	5.0 / 0.6 / 3.5
FLAN-T5 XXL	–	31.0 / 15.1 / 24.2	26.2	33.1 / 10.2 / 23.3	32.7 / **0.3** / **22.6**	**31.4** / 1.0 / **21.4**
FLAN-T5 XXL (self ans)	–	31.6 / 19.3 / 26.2	44.6	25.7 / 7.9 / 18.0	25.1 / 0.3 / 17.3	24.4 / 0.7 / 16.5
FLAN-T5 XXL (ext ans)	–	24.3 / 22.2 / 22.8	62.0	6.0 / 2.9 / 3.8	3.5 / 0 / 2.3	3.7 / 0 / 2.4

Table 8: Comparing performance of zero-shot models for different answer-types. Single-span and multi-span models trained on intermediate training data are denoted by SS and MS respectively. Identifying answerable questions using FLAN-T5 is denoted by 'self ans' whereas 'ext ans' denotes use of external supervised model. †F1, EM and IoU are the same for unanswerable questions as the reference is an empty string.

- Longformer: mrm8488/longformer-base-4096-finetuned-squadv2
- BigBird: google/bigbird-base-trivia-itc

Licensing. We used the AMI dataset that has `CC-BY-4.0` license. Our released data will have the `CC-BY-NC` license. We do not violate the constraints put in the MEDIASUM dataset to use interview files for reasearch purposes only.

Instructions/Prompts. For instruction-tuned FLAN models we use the following prompt template to generate sentences from the context that answer the question.

> `[CONTEXT]`
> Based on the conversation above, which sentences from the converstation answer `[SPEAKER]`'s question: `[QUESTION]`

Here, `[.]` is a placeholder filled in separately for each instance/question. Additionally, for the 'self ask' setting, we first use a prompt (shown below) to get the model to output if the question is answerable. If the model outputs "no", filter out those questions and use an empty string as the predictions. We find answers to the remaining questions using the prompt above.

> `[CONTEXT]` Based on the conversation above, did anyone answer `[SPEAKER]`'s question: `[QUESTION]` Respond "yes" if answered, "no" otherwise.

Binary Answerable Classification Model. As mentioned in Section 4.1, we train a separate supervised RoBERTa-base model to detect if a question is answerable. This is formulated as a binary classification task, therefore we train the sequence classification head on questions from MEETINGQA. We use the same hyperparameters as for single-span RoBERTa models mentioned above. The final performance of this model, is not as strong with an overall F1 = 49.2. This indicates that even a simple binary task formulation from MEETINGQA is challenging and requires thorough understanding of meeting discussions.

C Additional Results

Building on Tables 3 and 4, which contain F1 and IoU scores, we present the exact match (EM) scores

	Model	Int. Train Data	Overall	No Answer	Answerable		
					All	Multi-Span	Multi-Speaker
			F1 / EM / IoU	F1†	F1 / EM / IoU	F1 / EM / IoU	F1 / EM / IoU
Finetuned	RoBERTa-large (SS)	–	54.7 / **34.5** / 50.7	59.8	52.4 / 23.2 / 46.6	45.6 / 0 / 36.5	50.8 / 11.1 / 43.1
		SQuADv2	55.6 / 32.3 / 51.0	59.4	53.8 / 20.2 / 47.3	52.8 / 0 / 43.3	54.9 / 10.0 / 46.2
		+ silver	55.3 / 31.7 / 50.7	**63.4**	51.7 / 17.5 / 45.1	53.4 / 0 / 43.9	54.4 / 7.7 / 45.5
	DeBERTa-large (SS)	–	56.1 / 30.9 / 51.0	43.1	**61.9 / 25.5 / 54.5**	55.2 / 0 / 44.3	59.6 / 12.4 / 50.1
		SQuADv2	57.2 / 32.0 / **52.4**	52.0	59.5 / 23.0 / 52.6	55.6 / 0 / 45.8	59.4 / 11.8 / 50.9
		+ silver	55.7 / 31.5 / 51.0	52.1	57.3 / 22.4 / 50.5	55.2 / 0 / 44.9	58.5 / 10.4 / 49.3
	RoBERTa-large (MS)	–	48.6 / 18.3 / 38.7	47.2	49.2 / 5.4 / 34.9	50.2 / 1.1 / 34.9	50.1 / 1.2 / 34.5
		silver	49.4 / 19.3 / 40.1	48.9	50.6 / 6.8 / 35.7	52.0 / 2.1 / 36.2	51.7 / 2.3 / 35.8
	DeBERTa-large (MS)	–	**56.8** / 29.4 / 50.2	47.1	61.2 / 21.5 / 53.0	58.8 / 6.0 / 47.9	62.2 / 9.9 / 52.0
		silver	57.5 / 31.2 / 51.4	48.4	62.3 / 22.9 / 54.2	**59.6 / 7.1 / 48.3**	**63.5 / 10.7 / 52.9**
Zero-shot	RoBERTa-large (SS)	SQuADv2	31.7 / 29.7 / 29.9	**92.4**	4.6 / 1.8 / 2.0	2.6 / 0 / 1.1	3.4 / 0 / 1.5
		+ silver	33.4 / 20.6 / 30.0	38.0	31.3 / 12.8 / 26.4	24.2 / 0 / 17.3	24.7 / 0.1 / 17.9
	DeBERTa-large (SS)	SQuADv2	27.3 / 23.5 / 24.8	72.4	7.1 / 1.7 / 3.5	6.0 / 0 / 3.0	6.0 / 0.1 / 3.0
		+ silver	**35.8 / 24.4 / 32.9**	52.2	28.5 / 12.0 / 24.3	21.0 / 0 / 15.2	21.7 / 1.0 / 16.3
	RoBERTa-large (MS)	silver	35.2 / 20.3 / 31.7	25.1	**39.7 / 17.2 / 33.6**	**28.5** / 0 / **20.2**	**29.2 / 0 / 20.7**
	DeBERTa-large (MS)	silver	32.1 / 17.4 / 28.2	16.6	38.3 / 16.9 / 32.2	27.1 / 0 / 19.3	27.2 / 0 / 18.7

Table 9: Comparing performance of RoBERTa and DeBERTa large models for different answer-types. Single-span and multi-span models trained on intermediate training data are denoted by SS and MS respectively. †F1, EM and IoU are the same for unanswerable questions as the reference is an empty string.

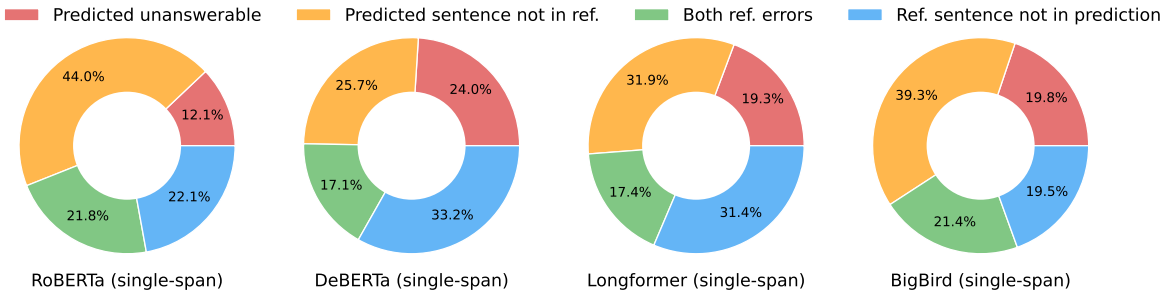

Figure 12: Error distribution (answerable Qs) for various single-span models finetuned directly on MEETINGQA.

in Tables 6 and 7 for finetuned single-span and multi-span respectively. While the relative trends across the models remains the same, we find that the EM scores are the lowest because it reflects predictions that perfectly match the reference. Another noteworthy observation is that the EM scores of all single-span models on the multi-span split is 0. This can be explained by the training procedure of single-span models (described in Section 3.2). The models are trained to predict a single "super-span" starting from the first sentence in the reference to the last sentence in the reference. Therefore, even in the theoretical best-case-scenario, the models would predict a single super-span containing all the reference sentences interleaved by irrelevant sentences for questions with multi-span answers. We analyze errors due to this in Appendix D.

In Table 5, we only present the overall scores for various models. All the scores on different splits are given in Table 8. We observe that for all single-span models (except RoBERTa on unanswerable questions) adding silver data in intermediate training helps improve performance across all splits. Furthermore, the multi-speaker and multi-span splits consistently pose a challenge for all models evaluated in a zero-shot setting. Also, within answer-types performance drops for unanswerable questions while improving for multi-span and multi-speaker answers. The challenge posed by unanswerable questions can be explained by the multi-span adaptation (Section 3.2). By posing question answering as a token-classification task, even one false positive (`I` tag instead of all `O`s) in token label, changes the answer prediction from unanswerable to answerable. Finally, we note that when using a pipeline-approach of isolation unanswerable questions separately, we find the errors in this step cascade and are reflected in the per-

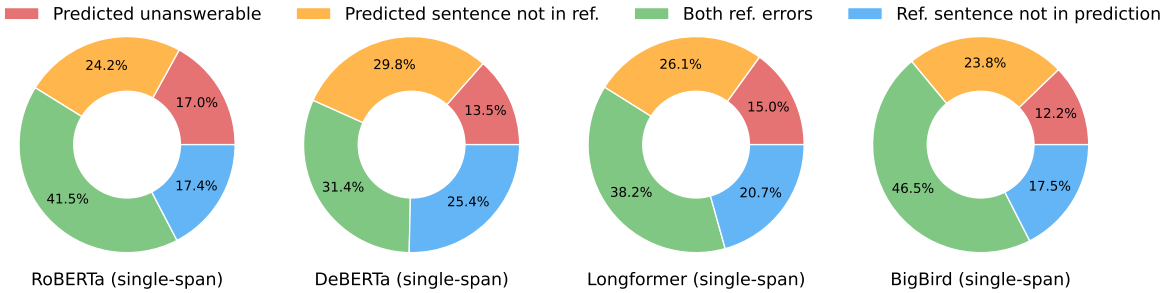

Figure 13: Error distribution (answerable Qs) for various multi-span models finetuned directly on MEETINGQA.

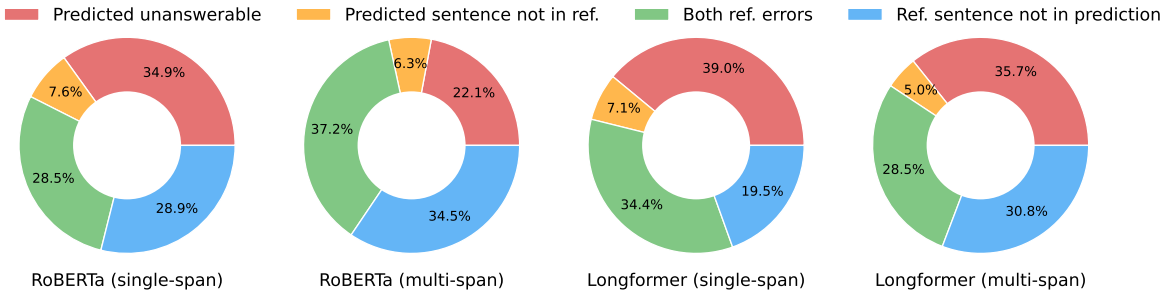

Figure 14: Error distribution (answerable Qs) for various single-span and multi-span models evaluated zero-shot on MEETINGQA. Single-span models use SQuADv2 + silver data for intermediate training whereas multi-span models use only silver data for intermediate training.

formance on answerable questions of the overall system. These systems perform better on unanswerable questions (not identified by regular instruction and filtering), however the false-positives decrease performance on answerable questions, even more so when using an external supervised model.

In Table 8, we see clear scaling of performance as we move from FLAN-T5 large (770M) to FLAN-T5 XL (3B). However, for FLAN-T5 XXL (11B) the performance of unanswerable, multi-span and multi-speaker questions increases ($\geq$ 8 F1 points) but performance on other answerable questions decreases ($\approx$ 9 F1 points) which in turn reduces the overall performance as compared to FLAN-T5 xl. Table 9 evaluates the performance of RoBERTa-large and DeBERTa-large architectures for single-span and multi-span models in both finetuned and zero-shot settings. The corresponding performance of the base models can be found in Tables 3, 4, and 5 respectively. We do not observe any significant increase in performance when using the larger checkpoints, thus leaving ample room for future work to bridge the gap between model and human performance on MEETINGQA.

D Error Analysis

In this section, we analyze error patterns across models discussed in Sections 3.2 and 4 in detail. First, we note that for the unanswerable questions split, any error corresponds to the model predicting a non-empty answer span. The frequency of this for a given model can be calculated by 100 − F1 score for this split (provided in Tables 3, 4, and 8). However, for answerable questions, errors in model predictions are diverse as categorized below.

I. Prediction is an empty-span (unanswerable)

II. Predicted span contains a sentence *not* present in the gold or annotated reference span

III. At least one of the sentences in the reference span is *not* present in the predicted span

IV. Combination of errors with respect to reference span (both II and III)

Therefore, whenever the model prediction does not exactly match the annotated reference span, we can put it in one of the above 4 categories. We perform this analysis for various finetuned single-span, finetuned multi-span models as well as zeroshot single-span, multi-span and instruction tuned models discussed in Sections 3.2 and 4. For brevity, we

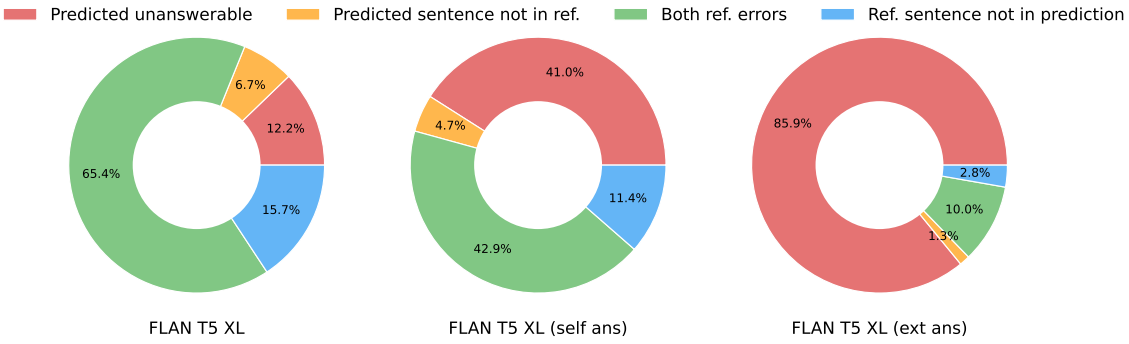

Figure 15: Error distribution (answerable Qs) for different configurations of Instruction-tuned FLAN-T5 models.

	Model	Training Data	Multi-Span Split	Multi-Speaker Split	
			% Preds.	% Preds.	Speaker IoU
Finetuned	RoBERTa-base (SS)	MEETINGQA	0.0	65.04	58.71
	DeBERTa-base (SS)	MEETINGQA	0.0	43.44	48.88
	Longformer-base (SS)	MEETINGQA	0.0	52.29	52.04
	Bigbird-base (SS)	MEETINGQA	0.0	63.86	57.15
	RoBERTa-base (MS)	MEETINGQA	52.58	48.66	54.72
	DeBERTa-base (MS)	MEETINGQA	70.86	54.48	63.76
	Longformer-base (MS)	MEETINGQA	53.98	50.54	57.27
	Bigbird-base (MS)	MEETINGQA	71.12	42.72	55.70
Zero-shot	RoBERTa-base (SS)	SQuADv2 + silver	0.0	8.72	31.71
	Longformer-base (SS)	SQuADv2 + silver	0.0	12.36	27.19
	RoBERTa-base (MS)	silver	5.30	3.47	34.35
	Longformer-base (MS)	silver	1.51	1.2	30.30
	FLAN-T5	–	11.01	10.71	31.11
	FLAN-T5 (self ans)	–	9.50	8.69	22.75
	FLAN-T5 (ext ans)	–	1.51	2.02	4.85

Table 10: Error analysis for various models and configurations on multi-span and multi-speaker splits.

pick representative models from different possible combinations of intermediate training data. This is illustrated in Figures 12-15 with error I shown in red, error II shown in yellow, error III shown in blue, and error IV shown in green.

Table 3 shows very similar performance different intermediate training data configurations for a given model architecture. Thus, we present error distribution for single-span models directly finetuned on MEETINGQA in Figure 12. We find that most of the errors belong to categories II-IV. The DeBERTa model has a relatively high unanswerable prediction error which is primarily because its predictions skew towards unanswerable as explained by the F1 score on No Answer (unanswerable) split in Table 3. Next, in Figure 13 we show the error distributions on the corresponding multi-span models finetuned directly on MEETINGQA. We observe that, for all model (except RoBERTa) the frequency of incorrectly predicting unanswerable goes down as well as the prediction span containing sentences outside the reference (error II). However, the frequency of hybrid error IV increases significantly. This can partly be explained by the design of single-span and multi-span models. As mentioned in Section 3.2, training data of the single span model involves creating a single "super-span" starting from the first sentence in the reference to the last sentence in the reference. This by construction involves error II and supervision on this data directs the model to include irrelevant sentences in the answer span if it is sandwiched between two relevant sentences. Also, for multi-span models an unanswerable prediction implies all tokens are labeled with the `O` tag, and even one false positive (`I` tag) would make the prediction answerable. Due to this, one can expect these models to mispredict empty spans less frequently.

Interestingly, when we look at zero-shot performance of single-span and multi-span models in Figure 14, we find relatively high frequency of error I and very low frequency of error II. The errors III

and hybrid IV also become more common. These models which are trained only on intermediate data, do not generalize in their ability to predict when a question is unanswerable. Further, we find that in the zero-shot evaluation setting, model predictions are shorter than the reference span by at least one sentence on average ($\approx$ 2 sentences for multi-span split). This indicates in zero-shot evaluation models are more likely to predict a part of the answer than output spans that covers all sentences in the reference, containing extra sentences that lie outside the reference.

Finally, we look at the errors of instruction-tuned FLAN-T5 models in Figure 15. When using FLAN-T5 with appropriate instruction and filtering we find that most of the errors are hybrid, i.e. predicted sentences do not cover the reference span entirely and also contains irrelevant sentences. When we add the self-ans pipeline on top of it with additional instructions to spot unanswerable questions, the predictions contain more empty spans (relatively) which is reflected in the increase in frequency of error I and the No Answer F1 (in Table 8). Surprisingly, when we use an external supervised model to predict unanswerable questions, it contributes to the vast majority of errors in the pipeline (error I). This is consistent with the fact that the test F1 score of this model on the task of classifying questions as answerable or not was only 49.2.

So far, we have analyzed errors in predictions for all the answerable questions. Next, we focus our attention on questions with multi-span and multi-speaker answers. Within the multi-span split, we calculate the fraction of incorrect predictions (as per exact match) that are multi-span, denoted by *multi-span preds (%)*. Similarly, for multi-speaker split, we calculate the fraction of incorrect predictions (as per exact match) that are multi-speaker in nature, denoted by *multi-speaker preds (%)*. Further, we compare the list of speakers in the reference and predicted spans using Jaccard similarity (IoU) denoted as *speaker IoU*. We compute and report these metrics for all the aforementioned models in Table 10.

As expected, due to the single-span training, none of the predictions of the single-span models are multi-span in nature. On the other hand, even incorrect predictions of the finetuned multi-span models are multi-span in nature at least half of the times. However, a significant fraction, between 29-46%, of the errors in this split can be attributed to single-span predictions for various models. For zero-shot models, over 90% of incorrect predictions are single span (also vast majority of all predictions are single span). On the multi-speaker split, the incorrect predictions of finetuned models are multi-speaker in nature. However, the speaker IoU (< 65) indicates that predicted spans often miss utterances from relevant speakers in the reference and also include irrelevant utterances from other speakers. Zero-shot models on the other hand, only tend to give single-speaker responses which is the primary source for errors. Note that, relatively high frequency of error I or prediction unanswerable spans also contributes in driving down the values of these metrics (empty spans have no span or speaker information).

Method	F1	EM	IoU
list instruction	33.4	13.0	25.3
+ filtering	35.1	17.6	27.6
direct instruction	14.0	5.4	7.9
+ filtering	28.0	20.6	22.4

Table 11: Overall performance comparison of both types of instructions with and without filtering.

E Instruction-tuned Model Ablations

In Section 3.2, we describe adaptation of generative FLAN-T5 model to our extractive setting by (i) designing instructions that ask models to *list which sentences from the context contain the answer* (mentioned in Section B); and (ii) filtering out all sentences from the model response that are not present in the context to remove any possible hallucinations. To show the importance of both these steps, we first compare with an instruction eliciting a direct response (answer) from the model, mentioned below.

> `[CONTEXT]`
> Based on the conversation above, state the answer(s) given to `[SPEAKER]`'s question: `[QUESTION]`

We call this *direct instruction* as opposed to the *list instruction* mentioned used in Sections 3.2 and 4.1. Further, we examine the importance of filtering by comparing raw model responses to their filtered counterparts. The comparison on a random subset of 500 question from the dev splits is shown in Table 11. We find that our chosen type of instruction (list) significantly outperforms the direct

ID	Transcript Segment
CNN-130961	... SPEAKER 3: He wants to spread it across the country. SPEAKER 2: Let me go back to the Confederate flag issue that you raised a few moments ago, Faye. I'm getting a couple of e-mails on that where "Republican policies have done little to support African-Americans. Isn't Bush the same man who was indifferent about the Confederate flag? We do not forget so soon". And also Peter in New York says: "Bush may be trying to appeal to black voters, but if he goes down in the polls, don't be surprised if he wraps himself in that Confederate flag". This is an issue that will not go away with these candidates and with these races. So I mean, how much of an influence is this going to be? SPEAKER 0: Well, it's going to be a big influence. It's an issue that ought not to go away. And actually we don't have to look to the future. We already know what happened when Bush went down in the polls, when it was following New Hampshire, when he was nervous about the South Carolina primary when he wrapped himself in the Confederate flag. So there's already a bit of a history there. And even though in the scheme of things the Confederate flag is a very symbolic issue, but symbols do matter. Try telling Jewish Americans if they should move on, that they should forget about a swastika that a candidate who would remain silent on a swastika is deserving of their support. SPEAKER 3: Well, look, Bobbie, I think what I want to establish right here is that George Bush has made it very clear that he thinks that there is only one flag that matters, that there is only one flag that represents freedom and that there's only one flag that one should die for in this country, and that is Old Glory. And I think that he has said that he has a personal point of view on the flag, and he thinks that it is a matter that should be resolved at the state level. The state has taken down the flag due to voices... SPEAKER 2: But how does that illustrate that we're one country under one flag? ...
CNN-116	... SPEAKER 7: That poll that just flashed up on the screen suggests that she did do well because that poll seems to have been taken around the time of the Letterman interview, and it shows almost a dead heat statistically, if you take the margin of error into account. So if that poll is accurate, then she's moved up from where she was according to other polls. SPEAKER 9: Yes, she has moved up. SPEAKER 1: But she is down from a year ago. So I'm curious as to why you think that she may have lost her edge a little bit from a year ago or people don't seem to be quite as enthusiastic about her? SPEAKER 7: People always like politicians better when they're not running for office. As soon as you decide that you are actually going to run, there are a certain number of people who decide they don't like you anymore. They liked you when you were a proposed candidate, and they don't like you are your ambition comes to show. SPEAKER 9: And Bobbie, she benefited I think from the entire impeachment situation. She was, you know, standing in support of Bill Clinton, and I think her poll numbers went up. But then when it looked like she was going to be just a typical politico, and people started looking at her views on the issues, and her stumbles in Israel and so forth, the numbers just started to go down. SPEAKER 1: Well she had that victim status when she first started thinking about entering this race. SPEAKER 9: Yes, I think ...

Table 12: Representative examples from the silver annotated data. The question and the automatically identified answers are highlighted in red and blue respectively. In the first example, Speaker 0's utterances are not automatically annotated since the entity in the utterance of Speaker 2 ("Faye") corresponds to Speaker 3's name. Thus, the algorithm predicts only Speaker 3 would answer.

instruction (tends to be more abstractive). Furthermore, filtering consistently improves overall performance especially when using direct instructions possibly due to higher number of hallucinations in the corresponding model answers.

F Silver Data Augmentation Details

Section 3.3 describes the silver data annotation process to annotate publicly available interview transcripts from CNN and NPR (Zhu et al., 2021a) for extractive QA task similar to MEETINGQA. We first identify the subset of speakers that act as the host or interviewer and focus on questions asked by these speakers to generate answer annotations. Based on the utterance containing the question, we first automatically identify speaker(s) answer the question using a rule-based approach. This is done by (i) finding speakers mentioned in the question sentence or utterance using an off-the-shelf NER model (mentioned in Appendix B), (ii) identifying speakers from previous speaker turns if the same speaker takes the turn after the host speaker assuming this is a case of follow-up questions, or (iii) in the absence of first two conditions, all speaker that take turns after the host. Finally, we search utterances corresponding to identified speakers in the transcript until a stopping criterion (max number of utterances, or reach the next host utterance) is

met and label it as the answer. From the 463.6K transcripts, only 15% of the files have identify host who steer the interview and have sufficiently high frequency of questions. However, each of these transcripts result in roughly 20 annotated questions on average. For intermediate training, we sample a total of 150K questions from this set and split it randomly into train, dev, test splits in $80:10:10$ ratio. Table 12 shows a few examples of silver answer annotations for questions asked in MEDIASUM interviews.

Perturbations. First, we add random sentences between the question and answer utterances to prevent a location bias in which model predicts sentences that immediately follow the question as the answer. Second, we create scenarios where the question is unanswerable by removing annotated answer spans from the context. Third, we replace speaker names in the context with a numeric identifier because information about speaker names are not always available in the transcript including AMI dataset. For multi-span models, we further insert random sentences from elsewhere in the transcript in between annotated answers to facilitate better span selection. Finally, the number of speaker turns in MEDIASUM are 10x smaller than those in AMI dataset (refer to Table 2 of Zhu et al. (2021a)). Therefore, we create more speaker transitions by splitting a long speaker utterance into shorter utterances by multiple speakers.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 7

☑ A2. Did you discuss any potential risks of your work?
Section 8

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 2, 3 and 4 verify the claims made.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 2

☐ B1. Did you cite the creators of artifacts you used?
No response.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Appendix B

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 8, Appendix B

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Section 8, Appendix A

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 2, 8, Appendix A

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 2

C ☑ Did you run computational experiments?

Section 3-4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix B, Sec 4

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix B, Sec 4

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Sec 4, Appendix B, C

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix B

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Section 2, Appendix A

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Appendix A and supplementary data folder

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Section 2, Appendix A

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Section 2, Appendix A

☒ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Ethical Review was conducted by crowd-sourcing company.

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Appendix A