

# QMSum: A New Benchmark for Query-based Multi-domain Meeting Summarization

Ming Zhong<sup>\*†</sup> Da Yin<sup>\*♣</sup> Tao Yu<sup>‡</sup> Ahmad Zaidi<sup>‡</sup>

Mutethia Mutuma<sup>†</sup> Rahul Jha<sup>¶</sup> Ahmed Hassan Awadallah<sup>¶</sup>

Asli Celikyilmaz<sup>¶</sup> Yang Liu<sup>§</sup> Xipeng Qiu<sup>†</sup> Dragomir Radev<sup>‡</sup>

<sup>†</sup>Fudan University <sup>♣</sup>University of California, Los Angeles <sup>‡</sup>Yale University

<sup>¶</sup>Microsoft Research <sup>§</sup>Microsoft Cognitive Services Research

mzhong18@fudan.edu.cn da.yin@cs.ucla.edu

{tao.yu, dragomir.radev}@yale.edu

## Abstract

Meetings are a key component of human collaboration. As increasing numbers of meetings are recorded and transcribed, meeting summaries have become essential to remind those who may or may not have attended the meetings about the key decisions made and the tasks to be completed. However, it is hard to create a single short summary that covers all the content of a long meeting involving multiple people and topics. In order to satisfy the needs of different types of users, we define a new query-based multi-domain meeting summarization task, where models have to select and summarize relevant spans of meetings in response to a query, and we introduce QMSum, a new benchmark for this task. QMSum consists of 1,808 query-summary pairs over 232 meetings in multiple domains. Besides, we investigate a *locate-then-summarize* method and evaluate a set of strong summarization baselines on the task. Experimental results and manual analysis reveal that QMSum presents significant challenges in long meeting summarization for future research. Dataset is available at <https://github.com/Yale-LILY/QMSum>.

## 1 Introduction

Meetings remain the go-to tool for collaboration, with 11 million meetings taking place each day in the USA and employees spending six hours a week, on average, in meetings (Mroz et al., 2018). The emerging landscape of remote work is making meetings even more important and simultaneously taking a toll on our productivity and well-being (Spataro, 2020). The proliferation of meetings makes it hard to stay on top of this sheer volume of information and increases the need for automated methods for accessing key information exchanged during them. Meeting summarization (Wang and Cardie, 2013; Shang et al., 2018;

<sup>\*</sup> These two authors contributed equally. The order of authorship decided by the flip of a coin.

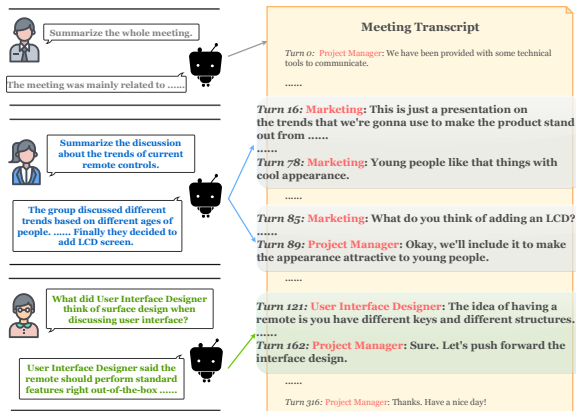


Figure 1: Examples of query-based meeting summarization task. Users are interested in different facets of the meeting. In this task, a model is required to summarize the contents that users are interested in and query.

Li et al., 2019; Zhu et al., 2020) is a task where summarization models are leveraged to generate summaries of entire meetings based on meeting transcripts. The resulting summaries distill the core contents of a meeting that helps people efficiently catch up to meetings.

Most existing work and datasets on meeting summarization (Janin et al., 2003; Carletta et al., 2005) pose the problem as a single document summarization task where a single summary is generated for the whole meeting. Unlike news articles where people may be satisfied with a high-level summary, they are more likely to seek more detailed information when it comes to meeting summaries such as topics (Li et al., 2019), opinions, actions, and decisions (Wang and Cardie, 2013). This poses the question of *whether a single paragraph is enough to summarize the content of an entire meeting?*

Figure 1 shows an example of a meeting about “remote control design”. The discussions in the meeting are multi-faceted and hence different users might be interested in different facets. For example, someone may be interested in learning about the new trends that may lead to the new product

standing out, while others may be more interested in what other attendees thought about different elements of the design. It is challenging to compress or compose a short summary that contains all the salient information. Alternatively, summarization systems should adopt a more flexible and interactive approach that allows people to express their interests and caters to their diverse intents when generating summaries (Dang, 2005, 2006; Litvak and Vanetik, 2017; Baumel et al., 2018).

With comprehensive consideration of the multi-granularity meeting contents, we propose a new task, query-based meeting summarization. To enable research in this area, we also create a high-quality multi-domain summarization dataset. In this task, as shown in Figure 1, given a query and a meeting transcript, a model is required to generate the corresponding summary. The query-based approach is a flexible setup that enables the system to satisfy different intents and different levels of granularity. Besides the annotated queries and corresponding gold summaries at different levels of granularity, our new dataset contains a rich set of annotations that include the main topics of each meeting and the ranges of relevant text spans for the annotated topics and each query. We adopt a hierarchical annotation structure that could not only assist people to find information faster, but also strengthen the models’ summarization capacity.

In this paper, we employ a two-stage meeting summarization approach: *locate-then-summarize*. Specifically, given a query, a model called *Locator* is used to locate the relevant utterances in the meeting transcripts, and then these extracted spans are used as an input to another model called *Summarizer* to generate a query-based summary. We present and evaluate several strong baselines based on state-of-the-art summarization models on QM-Sum. Our results and analysis from different perspectives reveal that the existing models struggle in solving this task, highlighting the challenges the models face when generating query-based meeting summaries. We are releasing our dataset and baselines to support additional research in query-focused meeting summarization.

Overall, our contributions are listed as follows: 1) We propose a new task, query-based multi-domain meeting summarization, and build a new benchmark QMSum with a hierarchical annotation structure. 2) We design a *locate-then-summarize* model and conduct comprehensive experiments on

its strong variants and different training settings. 3) By human evaluation, we further pose the challenges of the new task, including the impact of different query types and factuality errors.

## 2 Related Work

### 2.1 Text Summarization

Most prior work in text summarization (Rush et al., 2015; Chopra et al., 2016; Nallapati et al., 2016; See et al., 2017; Celikyilmaz et al., 2018; Chen and Bansal, 2018; Zhong et al., 2019a; Xu and Durrett, 2019; Liu and Lapata, 2019; Lebanoff et al., 2019; Cho et al., 2019; Zhong et al., 2020; Wang et al., 2020; Xu et al., 2019; Jia et al., 2020) investigate how to generate better summaries on news article data, such as CNN/DailyMail (Hermann et al., 2015), Newsroom (Grusky et al., 2018), etc. Scientific paper summarization is another important branch (Cohan et al., 2018; Yasunaga et al., 2019; An et al., 2021). Our paper mainly focuses on meeting summarization, a more challenging task compared to news summarization. With the burst of demand for meeting summarization, this task attracts more and more interests from academia (Wang and Cardie, 2013; Oya et al., 2014; Shang et al., 2018; Zhu et al., 2020) and becomes an emerging branch of text summarization area.

### 2.2 Query-based Summarization

Query-based summarization aims to generate a brief summary according to a source document and a given query. There are works studying this task (Daumé III and Marcu, 2006; Otterbacher et al., 2009; Wang et al., 2016; Litvak and Vanetik, 2017; Nema et al., 2017; Baumel et al., 2018; Ishigaki et al., 2020; Kulkarni et al., 2020; Laskar et al., 2020). However, the models focus on news (Dang, 2005, 2006), debate (Nema et al., 2017), and Wikipedia (Zhu et al., 2019). Meeting is also a genre of discourses where query-based summarization could be applied, but to our best knowledge, there are no works studying this direction.

### 2.3 Meeting Summarization

Meeting summarization has attracted a lot of interest recently (Chen and Metze, 2012; Wang and Cardie, 2013; Mehdad et al., 2013; Oya et al., 2014; Shang et al., 2018; Li et al., 2019; Zhu et al., 2020; Koay et al., 2020). Specifically, Mehdad et al. (2013) leverage entailment graphs and ranking strategy to generate meeting summaries. Wang

and Cardie (2013) attempt to make use of decisions, action items and progress to generate the whole meeting summaries. Oya et al. (2014) leverages the relationship between summaries and the meeting transcripts to extract templates and generate summaries with the guidance of the templates. Shang et al. (2018) utilize multi-sentence compression techniques to generate summaries under an unsupervised setting. Li et al. (2019) attempt to incorporate multi-modal information to facilitate the meeting summarization. Zhu et al. (2020) propose a model which builds a hierarchical structure on word-level and turn-level information and uses news summary data to alleviate the inadequacy of meeting data.

Unlike previous works, instead of merely generating summaries for the complete meeting, we propose a novel task where we focus on summarizing multi-granularity contents which cater to different people’s need for the entire meetings, and help people comprehensively understand meetings.

### 3 Data Construction

In this section, we show how we collected meeting data from three different domains: academic meetings, product meetings, and committee meetings. In addition, we show how we annotated the three types of meeting data while ensuring annotation quality for query-based meeting summarization.

#### 3.1 Data Collection

We introduce the three types of meetings that we used to annotate query-summary pairs.

**Product Meetings** AMI<sup>1</sup> (Carletta et al., 2005) is a dataset of meetings about product design in an industrial setting. It consists of 137 meetings about how to design a new remote control, from kick-off to completion over the course of a day. It contains meeting transcripts and their corresponding meeting summaries.

**Academic Meetings** ICSI<sup>2</sup> (Janin et al., 2003) dataset is an academic meeting dataset composed of 59 weekly group meetings at International Computer Science Institute (ICSI) in Berkeley, and their summaries. Different from AMI, the contents of ICSI meetings are specific to the discussions about research among students.

<sup>1</sup><http://groups.inf.ed.ac.uk/ami/download/>

<sup>2</sup><http://groups.inf.ed.ac.uk/ami/icsi/index.shtml>

**Committee Meetings** Parliamentary committee meeting is another important domain of meetings. These meetings focus on the formal discussions on a wide range of issues (e.g., the reform of the education system, public health, etc.) Also, committee meetings are publicly available, which enables us to access large quantities of meetings. We include 25 committee meetings of the Welsh Parliament<sup>3</sup> and 11 from the Parliament of Canada<sup>4</sup> in our dataset.

#### 3.2 Annotation Pipeline

After collecting meeting transcripts, we recruited annotators and required them to annotate by following annotation instruction. As illustrated in Figure 2, the annotation process is composed by three stages: topic segmentation, query generation, and query-based summarization.

**Topic Segmentation** Meeting transcripts are usually long and contain discussions about multiple topics. To assist further annotations, we asked annotators to write down the main topics discussed in the meetings, and their relevant text spans, which makes the meeting structure clear. As shown in Figure 2, “scope of the project and team building” is one of the annotated main topics, and its relevant text spans of the topic are (*Turn 25 - 50, Turn 73 - 89*). More details are listed in Appendix A.2.1.

**Query Generation** Towards the query-based task, we further asked annotators to design queries by themselves. To cater to the need for multi-granularity contents, we categorized two types of queries: queries related to general information (e.g., the contents of whole meetings, etc.) are called *general queries*; queries focusing on relatively detailed information (e.g., the discussion about certain topics, etc.) are called *specific queries*.

To alleviate the influence of extremely hard queries and focus on the evaluation of query-based summarization capacity, rather than designing queries in an unconstrained way, we asked annotators to generate queries according to the schema. Details of the query schema list are shown in Appendix A.1. The list consists of important facets people might be interested in, including overall contents of discussions, speakers’ opinions, the reasons why a speaker proposed an idea, etc., which cover the most common queries over meetings involving multiple people discussing several topics.

<sup>3</sup><https://record.assembly.wales/>

<sup>4</sup><https://www.ourcommons.ca/Committees/en/Home>

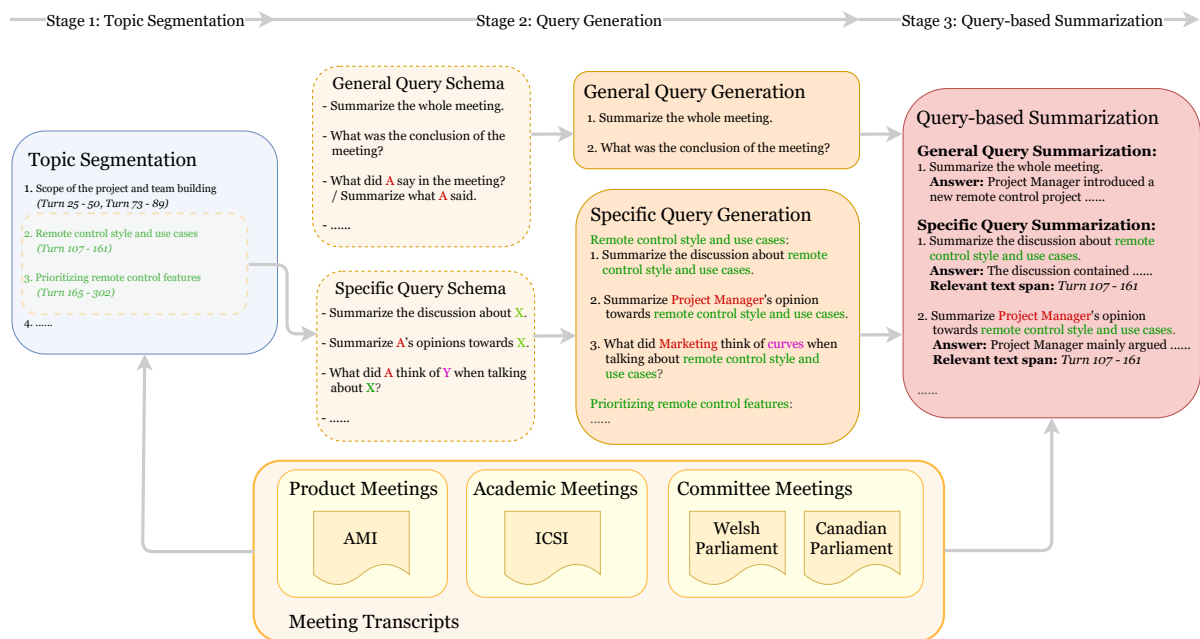


Figure 2: Overall annotation pipeline. It is divided into three stages: Stage 1 is to annotate main topics and their relevant text spans; Stage 2 is to generate queries based on query schema lists; Stage 3 is to annotate the summaries according to the queries. The pipeline was implemented upon the collected meetings of multiple domains.

To query multi-granularity meeting contents, we further divided the query schema list into general and specific ones, and asked annotators to design queries towards general and specific meeting contents, respectively. In terms of *general query generation*, the annotators were asked to design 1 - 2 general queries according to the general schema list. For *specific query generation*, annotators were asked to first select 2 - 4 main topics and their relevant text spans, and then design around 3 specific queries based on the specific schema list for each main topic. To ensure the task to be *summarization* instead of *question answering*, we asked annotators to design queries of which the relevant text spans are more than 10 turns or 200 words. Therefore, our proposed task would differ from *question answering* tasks where models merely need to extract phrases or generate answers based on short text spans, and focus on how to *summarize* based on large stretches of texts. Additional details are in Appendix A.2.2.

**Query-based Summarization** According to the designed queries and meeting transcripts, annotators were asked to do faithful summarization. Being accorded with the meeting transcripts and queries is the most important criterion. We also required annotators to write informative summarization. For example, they could add more de-

tails about the reasons why the group/committee made such decisions, and which important ideas the group/committee members proposed, etc. Besides, the annotated summaries should be abstractive, fluent and concise. We set word limits for the answers of general queries (50 - 150 words) and specific queries (20 - 100 words) to keep conciseness. More details are shown in Appendix A.2.3.

In the end, we organize all the meeting data after accomplishing the three annotation stages. Detailed annotations of one product meeting and one committee meeting are shown in Appendix A.4. Each meeting transcript is accompanied with annotated main topics, queries, their corresponding summaries, and relevant text span information.

### 3.3 Additional Details of Annotation Process

This section describes how we recruited annotators and how we review the annotations in detail.

**Annotator Recruitment** To guarantee annotation quality given the complexity of the task, instead of employing tasks on Amazon Mechanical Turk, we anonymously recruited undergraduate students who are fluent in English. The annotation team consists of 2 native speakers and 10 non-native speakers majoring in English literature.

**Annotation Review** To help the annotators fully comprehend the instruction, annotators were



Datasets	# Meetings	# Turns	# Len. of Meet.	# Len. of Sum.	# Speakers	# Queries	# Pairs
AMI	137	535.6	6007.7	296.6	4.0	-	97 / 20 / 20
ICSI	59	819.0	13317.3	488.5	6.3	-	41 / 9 / 9
Product	137	535.6	6007.7	70.5	4.0	7.2	690 / 145 / 151
Academic	59	819.0	13317.3	53.7	6.3	6.3	259 / 54 / 56
Committee	36	207.7	13761.9	80.5	34.1	12.6	308 / 73 / 72
All	232	556.8	9069.8	69.6	9.2	7.8	1,257 / 272 / 279

Table 1: Statistics of meeting summarization datasets. The top half of the table is the existing meeting datasets, and the bottom half is the statistics of QMSum. Because a meeting may have multiple queries, #Pairs here means how many query-summary pairs are contained in the train / valid / test set.

trained in a *pre-annotation* process. Annotations were reviewed across all stages in our data collection process by expert of this annotation task. More details of review standards could be found in Appendix A.3.

### 3.4 Dataset Statistics and Comparison

Statistics of the final QMSum dataset is shown in Table 1. There are several advantages of QMSum dataset, compared with the previous datasets.

**Number of Meetings and Summaries** QMSum includes 232 meetings, which is the largest meeting summarization dataset to our best knowledge. For each query, there is a manual annotation of corresponding text span in the original meeting, so there are a total of 1,808 question-summary pairs in QMSum. Following the previous work, we randomly select about 15% of the meetings as the validation set, and another 15% as the test set.

**Briefly** The average length of summaries in QMSum 69.6 is much shorter than that of previous AMI and ICSI datasets. It is because our dataset also focuses on specific contents of the meetings, and the length of their corresponding summaries would not be long. It leaves a challenge about how to precisely capture the related information and compress it into a brief summary.

**Multi-domain Setting** Previous datasets are specified to one domain. However, the model trained on the summarization data of a single domain usually has poor generalization ability (Wang et al., 2019; Zhong et al., 2019b; Chen et al., 2020). Therefore, QMSum contains meetings across multiple domains: Product, Academic and Committee meetings. We expect that our dataset could provide a venue to evaluate the model’s generalization ability on meetings of different domains and help create more robust models.

## 4 Method

In this section, we first define the task of query-based meeting summarization, then describe our two-stage *locate-then-summarize* solution in detail.

### 4.1 Problem Formulation

Existing meeting summarization methods define the task as a sequence-to-sequence problem. Specifically, each meeting transcript  $X = (x_1, x_2, \dots, x_n)$  consists of  $n$  turns, and each turn  $x_i$  represents the utterance  $u_i$  and its speaker  $s_i$ , that is,  $x_i = (u_i, s_i)$ . Additionally, each utterance contains  $l_i$  words  $u_i = (w_1, \dots, w_{l_i})$ . The object is to generate a target summary  $Y = (y_1, y_2, \dots, y_m)$  by modeling the conditional distribution  $p(y_1, y_2, \dots, y_m | (u_1, s_1), \dots, (u_n, s_n))$ .

However, meetings are usually long conversations involving multiple topics and including important decisions on many different matters, so it is necessary and practical to use queries to summarize a certain part of the meeting. Formally, we introduce a query  $Q = (w_1, \dots, w_{|Q|})$  for meeting summarization task, the objective is to generate a summary  $Y$  by modeling  $p(y_1, y_2, \dots, y_m | Q, (u_1, s_1), \dots, (u_n, s_n))$ .

### 4.2 Locator

In our two-stage pipeline, the first step requires a model to locate the relevant text spans in the meeting according to the queries, and we call this model a Locator. The reason why we need a Locator here is, most existing abstractive models cannot process long texts such as meeting transcripts. So we need to extract shorter, query-related paragraphs as input to the following Summarizer.

We mainly utilize two methods to instantiate our Locator: Pointer Network (Vinyals et al., 2015) and a hierarchical ranking-based model. Pointer

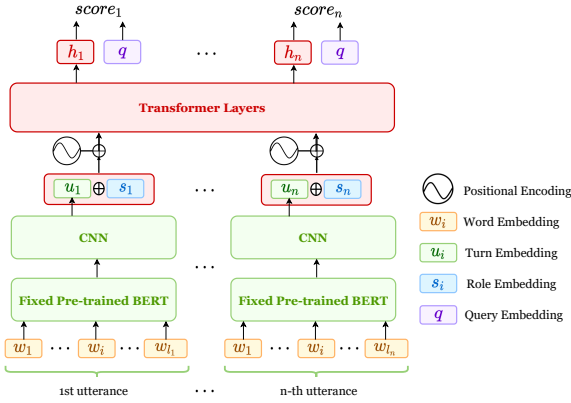


Figure 3: Hierarchical ranking-based locator structure.

Network has achieved widespread success in extractive QA tasks (Wang and Jiang, 2017). For each question, it will point to the <start, end> pair in the source document, and the span is the predicted answer. Specific to our task, Pointer Network will point to the start turn and the end turn for each query. It is worth noting that one query can correspond to multiple spans in our dataset, so we always extract three spans as the corresponding text for each query when we use Pointer Network as Locator in the experiments.

In addition, we design a hierarchical ranking-based model structure as the Locator. As shown in Figure 3, we first input the tokens in each turn to a feature-based BERT to obtain the word embedding, where feature-based means we fix the parameters of BERT, so it is actually an embedding layer. Next, CNN (Kim, 2014) is applied as a turn-level encoder to capture the local features such as bigram, trigram and so on in each turn. Here we do not use Transformer because previous work (Kedzie et al., 2018) shows that this component does not matter too much for the final performance. We combine different features to represent the utterance  $u_i$  in each turn, and concatenate the speaker embedding  $s_i$  as the turn-level representation:  $x_i = [u_i; s_i]$ , where  $[\cdot]$  denotes concatenation and  $s_i$  is a vector randomly initialized to represent the speaking style of meeting participants.

Then these turn representations will be contextualized by a document-level Transformer (Vaswani et al., 2017) encoder. Next, we introduce query embedding  $q$  which is obtained by a CNN (shared parameters with CNN in turn-level encoder) and use MLP to score each turn. We use binary cross-entropy loss to train our Locator. Finally, turns with the highest scores are selected as the relevant text spans of each query and will be inputted to the

subsequent Summarizer.

### 4.3 Summarizer

Given the relevant paragraphs, our goal in the second stage is to summarize the selected text spans based on the query. We instantiate our Summarizer with the current powerful abstractive models to explore whether the query-based meeting summarization task on our dataset is challenging. To be more specific, we choose the following three models:

**Pointer-Generator Network** (See et al., 2017) is a popular sequence-to-sequence model with copy mechanism and coverage loss, and it acts as a baseline system in many generation tasks. The input to Pointer-Generator Network (PGNet) is: “<s> Query </s> Relevant Text Spans </s>”.

**BART** (Lewis et al., 2020) is a denoising pre-trained model for language generation, translation and comprehension. It has achieved new state-of-the-art results on many generation tasks, including summarization and abstractive question answering. The input to BART is the same as PGNet.

**HMNet** (Zhu et al., 2020) is the state-of-the-art meeting summarization model. It contains a hierarchical structure to process long meeting transcripts and a role vector to depict the difference among speakers. Besides, a cross-domain pretraining process is also included in this strong model. We add a turn representing the query at the beginning of the meeting as the input of HMNet.

## 5 Experiments

In this section, we introduce the implementation details, effectiveness of Locator, experimental results and multi-domain experiments on QMSum.

### 5.1 Implementation Details

For our ranking-based Locator, the dimension of speaking embedding is 128 and the dimension of turn and query embedding is 512. Notably, we find that removing Transformers in Locator has little impact on performance, so the Locator without Transformer is used in all the experiments. To reduce the burden of the abstractive models, we utilize Locator to extract 1/6 of the original text and input them to Summarizer. The hyperparameters used by PGNet and HMNet are consistent with the original paper. Due to the limitation of computing resources, we use the base version of pre-trained models (including feature-based BERT and BART)

Models	Extracted Length			
	1/6	1/5	1/4	1/3
Random	58.86	63.20	67.56	73.81
Similarity	55.97	59.24	63.45	70.12
Pointer	61.27	65.84	70.13	75.96
Our Locator	<b>72.51</b>	<b>75.23</b>	<b>79.08</b>	<b>84.04</b>

Table 2: ROUGE-L Recall score between the predicted spans and the gold spans. 1/6 means that the turns extracted by the model account for 1/6 of the original text.

in this paper. We use fairseq library<sup>5</sup> to implement BART model. For PGNet and BART, we truncate the input text to 2,048 tokens, and remove the turns whose lengths are less than 5. All results reported in this paper are averages of three runs.

## 5.2 Effectiveness of Locator

First, we need to verify the effectiveness of the Locator to ensure that it can extract spans related to the query. Instead of the accuracy of capturing relevant text spans, we focus on the extent of overlap between the selected text spans and the gold relevant text spans. It is because whether the summarization process is built on similar contexts with references or not is essential for Summarizer. Therefore, we use ROUGE-L recall to evaluate the performance of different models under the setting of extracting the same number of turns.

We introduce two additional baselines: Random and Similarity. The former refers to randomly extracting a fixed number of turns from the meeting content, while the latter denotes that we obtain turn embedding and query embedding through a feature-based BERT, and then extract the most similar turns by cosine similarity. As shown in Table 2, because there are usually a large number of repeated conversations in the meetings, Random can get a good ROUGE-L recall score, which can be used as a baseline to measure the performance of the model. Similarity performs badly, even worse than Random, which may be due to the great difference in style between the BERT pre-trained corpus and meeting transcripts. Pointer Network is only slightly better than Random. We think this is because in the text of with an average of more than 500 turns, only three <start, end> pairs are given as supervision signals, which is not very informative and therefore is not conducive to model learning.

<sup>5</sup><https://github.com/pytorch/fairseq/tree/master/examples/bart>

Models	R-1	R-2	R-L
Random	12.03	1.32	11.76
Ext. Oracle	42.84	16.86	39.20
TextRank	16.27	2.69	15.41
PGNet	28.74	5.98	25.13
BART	29.20	6.37	25.49
PGNet*	31.37	8.47	27.08
BART*	31.74	8.53	<b>28.21</b>
HMNet*	<b>32.29</b>	<b>8.67</b>	28.17
PGNet <sup>†</sup>	31.52	8.69	27.63
BART <sup>†</sup>	32.18	8.48	28.56
HMNet <sup>†</sup>	<b>36.06</b>	<b>11.36</b>	<b>31.27</b>

Table 3: Experimental results on QMSum dataset. We use standard ROUGE F-1 score to evaluate different models. The models with \* denotes they use the spans extracted by our Locator as the input and <sup>†</sup> indicates this Summarizer uses gold spans as the input.

On the contrary, our hierarchical ranking-based Locator always greatly exceeds the random score, which demonstrates that it can indeed extract more relevant spans in the meeting. Even if 1/6 of the original text is extracted, it can reach a 72.51 ROUGE-L recall score, which significantly reduces the burden of subsequent Summarizer processing long text while ensuring the amount of information.

## 5.3 Experimental Results on QMSum

For comparison, we introduce two basic baselines: Random and Extractive Oracle. We randomly sample 10 turns of the original meeting for each query as an answer and this is the Random baseline in Table 3. Besides, we implement the Extractive Oracle, which is a greedy algorithm for extracting the highest-scoring sentences, usually regarded as the the upper bound of the extractive method (Nallapati et al., 2017). An unsupervised method, TextRank is also included in our experiment. We treat each turn as a node and add a query node to fully connect all nodes. Finally, the 10 turns with the highest scores are selected as the summary.

Table 3 shows that the performance of three typical neural network models is significantly better than Random and TextRank. When equipped with our Locator, both PGNet and BART have brought evident performance improvements (PGNet: 28.74 -> 31.37 R-1, BART: 29.20 -> 31.74 R-1). Compared to PGNet\*, the advantage of BART\* lies in the ROUGE-L score (1.13 improvement), which indicates that it can generate more fluent sentences.

Datasets	Product			Academic			Committee			All		
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
<b>Pro.</b>	35.43	10.99	31.37	22.59	3.41	19.82	24.48	3.84	21.94	30.02	7.58	26.62
<b>Aca.</b>	27.19	4.86	24.09	26.69	4.32	22.58	27.84	4.29	25.10	27.22	4.59	24.02
<b>Com.</b>	25.56	3.48	22.17	23.91	2.99	20.23	32.52	6.98	27.71	27.07	4.28	23.21
<b>All</b>	34.93	10.78	31.21	26.47	5.05	23.01	31.16	6.47	27.52	32.18	8.48	28.56

Table 4: Multi-domain and cross-domain summarization experiments. Each row represents the training set, and each column represents the test set. The gray cells denote the best result on the dataset in this column. We use BART<sup>†</sup> for these experiments and use standard ROUGE F-1 score to evaluate the model performance.

The current state-of-the-art meeting summarization model HMNet achieves the best performance, which may be attributed to its cross-domain pre-training process making HMNet more familiar with the style of meeting transcripts.

In addition, we also use the gold text spans as the input of different models to measure the performance loss caused by Locator. Surprisingly, for models (PGNet and BART) that need to truncate the input text, although Locator is an approximate solution, the models equipped with it can achieve comparable results with the models based on gold span inputs. Therefore, in this case, our two-stage pipeline is a simple but effective method in the meeting domain. However, for some models (HMNet) that use a hierarchical structure to process long text, inputting gold text spans can still bring huge performance improvements.

## 5.4 Experiments on Different Domains

In addition, we also conduct multi-domain and cross-domain experiments. First, we perform in-domain and out-domain tests in the three domains of QMSum dataset. In Table 4, we can conclude that there are obvious differences between these three domains. For instance, the models trained on the Academic and Committee domains perform poorly when tested directly on the Product domain, with only the ROUGE-L scores of 24.09 and 22.17 respectively. However, the model trained on the single domain of Product can achieve a ROUGE-L score of 31.37, which illustrates although these domains are all in the form of meeting transcript, they still have visible domain bias.

On the other hand, when we train all the domains together, we can obtain a robust summarization model. Compared with models trained on a single domain, models trained on QMSum can always achieve comparable results. In the Academic domain, the model with multi-domain train-

	Opin.	Inter.	Con./Dec.	Reason	Overall
Num.	22	40	19	7	12
Diff. 1	2.2	1.6	1.7	2.4	2.0
Diff. 2	1.9	2.1	2.3	2.2	1.6
R-L	27.0	30.1	26.1	24.9	30.9

Table 5: The number, human evaluation and model performance of different types of queries. Diff. 1 represents the difficulty of locating relevant information and Diff. 2 represents the difficulty of organizing content.

ing can even get higher ROUGE-2 (5.05 vs 4.32) and ROUGE-L (23.01 vs 22.58) scores. These results show that the multi-domain setting in meeting summarization task is apparently necessary and meaningful. Meeting transcripts cover various fields, making the transfer of models particularly difficult. Therefore, we need to introduce multi-domain training to make the model more robust, so it can be applied to more practical scenarios.

## 6 Analysis

In this section, we conduct comprehensive analysis of query types and errors in the model output.

### 6.1 Analysis of Query Types

We manually divide the query in QMSum into five aspects: personal opinion, multi-person interaction, conclusion or decision, reason, and overall content. For example, “Summarize the whole meeting.” requires a summary of the overall content and “Why did A disagree with B?” requires a summary of some reasons. The questions we are concerned about are: *what is the distribution of different types of queries in QMSum? Are there differences in the difficulty of different types of queries?*

To figure out the above issues, we randomly sample 100 queries from the test set, count the number of each type, and score the difficulty of each query. Table 5 illustrates that answering 40% of queries re-



quires summarizing the interaction of multiple people, and the queries that focus on personal opinions and different aspects of conclusions or decisions account for almost 20% each. Besides, queries about a specific reason are less frequent in the meetings.

We also perform a human evaluation of the difficulty of various query types. For each query, the relevant text spans and query-summary pair are shown to annotators. Annotators are asked to score the difficulty of this query in two dimensions: 1) the difficulty of locating relevant information in the original text; 2) the difficulty of organizing content to form a summary. For each dimension, they can choose an integer between 1 and 3 as the score, where 1 means easy and 3 means difficult.

As we can see from Table 5, query about reasons is the most difficult to locate key information in related paragraphs, and this type of query is also challenging to organize and summarize reasonably. Queries about multi-person interaction and overall content are relatively easy under human evaluation scores. The relevant paragraphs of the former contain multi-person conversations, which are usually redundant, so the effective information is easier to find; the latter only needs to organize the statements in the chronological order of the meeting to write a summary, so it has the lowest Diff. 2 score. The model performance also confirms this point, BART can get more than 30 R-L score on these two types of queries, but performs poorly on the rest. Therefore, the remaining three types of queries in QMSum are still very challenging even for powerful pre-trained models, and further research is urgently needed to change this situation.

## 6.2 Error Analysis

Although ROUGE score can measure the degree of overlap between the generated summary and the gold summary, it cannot reflect the factual consistency between them or the relevance between the predicted summary and the query. Therefore, in order to better understand the model performance and the difficulty of the proposed task, we sample 100 generated summaries for error analysis. Specifically, we ask 10 graduate students to do error analysis on the sampled summaries. Each summary is viewed by two people. They discuss and agree on whether the sample is consistent with the original facts and whether it is related to the query.

According to Cao et al. (2018), nearly 30% of summaries generated by strong neural models con-

tain factual errors. This problem is even more serious on QMSum: we find inconsistent facts in 74% of the samples, which may be because the existing models are not good at generating multi-granularity summaries. Although BART can achieve state-of-the-art performance in the single-document summarization task, it does not seem to be able to truly understand the different aspects of the meeting, thus create factual errors. What’s worse, 31% summaries are completely unrelated to the given query. This not only encourages us to design more powerful models or introduce more prior knowledge to overcome this challenge, but also shows better metrics are needed to evaluate model performance in generating multi-granularity summaries.

## 7 Conclusion

We propose a new benchmark, QMSum, for query-based meeting summarization task. We build a *locate-then-summarize* pipeline as a baseline and further investigate variants of our model with different Locators and Summarizers, adopt different training settings including cross-domain and multi-domain experiments to evaluate generalizability, and analyze the task difficulty with respect to query types. The new task and benchmark leave several open research directions to explore: 1) how to process the long meeting discourses; 2) how to make a meeting summarization model generalize well; 3) how to generate summaries consistent with both meeting transcripts and queries. 4) how to reduce the annotation cost for meeting summarization.

## Acknowledgements

The Language, Information, and Learning lab at Yale LILY) would like to acknowledge the research grant from Microsoft Research. We would also like to thank the annotators for their hard work and anonymous reviewers for their valuable comments.

## Ethics Consideration

We propose a novel query-based meeting summarization task, accompanying with a high-quality dataset QMSum. Since the paper involves a new dataset and NLP application, this section is further divided into the following two parts.

### 7.1 New Dataset

**Intellectual Property and Privacy Rights** Collecting user data touches on the intellectual prop-

erty and privacy rights of the original authors: both of the collected meeting transcripts and recruited annotators. We ensure that the dataset construction process is consistent with the intellectual property and privacy rights of the original authors of the meetings. All the meeting transcripts we collected are public and open to use according to the regulation<sup>6 7 8 9</sup>. The annotation process is consistent with the intellectual property and privacy rights of the recruited annotators as well.

**Compensation for Annotators** We estimated the time for annotating one meeting is around 1 - 2 hours. Therefore, we paid annotators around \$14 for each product and academic meeting and \$28 for each committee meeting. To further encourage annotators to work on annotations, we proposed bonus mechanism: the bonus of each of the 5th to 8th meetings would be \$4; the bonus of each of the 9th to 12th meetings would be \$5, and so on. Some of the authors also did annotations and they were paid as well.

**Steps Taken to Avoid Potential Problems** The most possible problems which may exist in the dataset is bias problem and the inconsistency among queries, annotated summaries and original meeting contents. With regard to bias problem, we find that product meeting dataset rarely contains any explicit gender information, but annotators still tended to use ‘he’ as pronoun. To avoid the gender bias caused by the usage of pronouns, we required annotators to replace pronouns with speaker information like ‘Project Manager’, ‘Marketing’ to avoid the problem. Also, when designing queries based on query schema list, we found that annotators usually used the same query schema, which might lead to bias towards a certain type of query. Therefore, we asked the annotators to use different schemas as much as possible. For the inconsistency problem, each annotation step was strictly under supervision by ‘experts’ which are good at annotation and could be responsible for reviewing.

---

<sup>6</sup><http://groups.inf.ed.ac.uk/ami/corpus/license.shtml>

<sup>7</sup><http://groups.inf.ed.ac.uk/ami/icsi/license.shtml>

<sup>8</sup><https://senedd.wales/en/help/our-information/Pages/Open-data.aspx>

<sup>9</sup><https://www.ourcommons.ca/en/important-notice>

## 7.2 NLP Applications

**Intended Use** The query-based meeting summarization application is aiming at summarizing meetings according to queries from users. We could foresee that the trained model could be applied in companies to further improve the efficiency of workers, and help the staff comprehensively understand the meeting contents. The annotated QM-Sum dataset could be used as a benchmark for researchers to study how to improve the performance of summarization on such long texts and how to make models more generalizable on the meetings of different domains.

**Failure Mode** The current baseline models still tend to generate ungrammatical and factually inconsistent summaries. If a trained baseline model was directly applied in companies, the misinformation would negatively affect comprehension and further decision making. Further efforts are needed to generate high-quality summaries which are fluent and faithful to the meeting transcripts and queries.

**Bias** Training and test data are often biased in ways that limit system accuracy on domains with small data size or new domains, potentially causing distribution mismatch issues. In the data collection process, we control for the gender bias caused by pronouns such as ‘he’ and ‘she’ as much as possible. Also, we attempt to control the bias towards a certain type of query schema by requiring annotators to use diverse schemas as much as possible. However, we admit that there might be other types of bias, such as political bias in committee meetings. Thus, the summarization models trained on the dataset might be biased as well. and We will include warnings in our dataset.

**Misuse Potential** We emphasize that the application should be used with careful consideration, since the generated summaries are not reliable enough. It is necessary for researchers to develop better models to improve the quality of summaries. Besides, if the model is trained on internal meeting data, with the consideration of intellectual property and privacy rights, the trained model should be used under strict supervision.

**Collecting Data from Users** Future projects have to be aware of the fact that some meeting transcripts are intended for internal use only. Thus, researchers should be aware of the privacy issues about meeting data before training the model.

## References

- Chenxin An, Ming Zhong, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2021. Enhancing scientific papers summarization with citation graph.
- Tal Baumel, Matan Eyal, and Michael Elhadad. 2018. Query focused abstractive summarization: Incorporating query relevance, multi-document coverage, and summary length constraints into seq2seq models. *arXiv preprint arXiv:1801.07704*.
- Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. **Faithful to the original: Fact aware neural abstractive summarization**. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4784–4791. AAAI Press.
- 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.
- Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1662–1675.
- Yen-Chun Chen and Mohit Bansal. 2018. **Fast abstractive summarization with reinforce-selected sentence rewriting**. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.
- Yiran Chen, Pengfei Liu, Ming Zhong, Zi-Yi Dou, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. **An empirical study of cross-dataset evaluation for neural summarization systems**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 3679–3691. Association for Computational Linguistics.
- Yun-Nung Chen and Florian Metze. 2012. Integrating intra-speaker topic modeling and temporal-based inter-speaker topic modeling in random walk for improved multi-party meeting summarization. In *Thirteenth Annual Conference of the International Speech Communication Association*.
- Sangwoo Cho, Logan Lebanoff, Hassan Foroosh, and Fei Liu. 2019. Improving the similarity measure of determinantal point processes for extractive multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1027–1038.
- Sumit Chopra, Michael Auli, and Alexander M Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.
- Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. 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 615–621.
- H. Dang. 2005. Overview of duc 2005.
- Hoa Trang Dang. 2006. **DUC 2005: Evaluation of question-focused summarization systems**. In *Proceedings of the Workshop on Task-Focused Summarization and Question Answering*, pages 48–55, Sydney, Australia. Association for Computational Linguistics.
- Hal Daumé III and Daniel Marcu. 2006. **Bayesian query-focused summarization**. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 305–312, Sydney, Australia. Association for Computational Linguistics.
- Max Grusky, Mor Naaman, and Yoav Artzi. 2018. **Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies**. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.
- Tatsuya Ishigaki, Hen-Hsen Huang, Hiroya Takamura, Hsin-Hsi Chen, and Manabu Okumura. 2020. Neural query-biased abstractive summarization using copying mechanism. In *European Conference on Information Retrieval*, pages 174–181. Springer.
- 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.



- Ruipeng Jia, Yanan Cao, Hengzhu Tang, Fang Fang, Cong Cao, and Shi Wang. 2020. Neural extractive summarization with hierarchical attentive heterogeneous graph network. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3622–3631.
- Chris Kedzie, Kathleen McKeown, and Hal Daume III. 2018. Content selection in deep learning models of summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1818–1828.
- Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.
- Jia Jin Koay, Alexander Roustai, Xiaojin Dai, Dillon Burns, Alec Kerrigan, and Fei Liu. 2020. [How domain terminology affects meeting summarization performance](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5689–5695, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Sayali Kulkarni, Sheide Chammas, Wan Zhu, Fei Sha, and Eugene Ie. 2020. Aquamuse: Automatically generating datasets for query-based multi-document summarization. *arXiv preprint arXiv:2010.12694*.
- Md Tahmid Rahman Laskar, Enamul Hoque, and Jimmy Huang. 2020. Query focused abstractive summarization via incorporating query relevance and transfer learning with transformer models. In *Canadian Conference on Artificial Intelligence*, pages 342–348. Springer.
- Logan Lebanoff, Kaiqiang Song, Franck Dernoncourt, Doo Soon Kim, Seokhwan Kim, Walter Chang, and Fei Liu. 2019. Scoring sentence singletons and pairs for abstractive summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2175–2189.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. [BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.
- 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.
- Marina Litvak and Natalia Vanetik. 2017. [Query-based summarization using MDL principle](#). In *Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres*, pages 22–31, Valencia, Spain. Association for Computational Linguistics.
- Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. 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 3721–3731.
- 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.
- Joseph E. Mroz, Joseph A. Allen, Dana C. Verhoeven, and Marissa L. Shuffler. 2018. [Do we really need another meeting? the science of workplace meetings](#). *Current Directions in Psychological Science*, 27(6):484–491.
- Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *CoNLL 2016*, page 280.
- Preksha Nema, Mitesh M. Khapra, Anirban Laha, and Balaraman Ravindran. 2017. [Diversity driven attention model for query-based abstractive summarization](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1063–1072, Vancouver, Canada. Association for Computational Linguistics.
- Jahna Otterbacher, Gunes Erkan, and Dragomir R. Radev. 2009. Biased lexrank: Passage retrieval using random walks with question-based priors. *Information Processing & Management*, 45(1):42–54.
- 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.
- Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.



- Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1073–1083.
- 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.
- Jared Spataro. 2020. The future of work—the good, the challenging the unknown.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.
- Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in neural information processing systems*, pages 2692–2700.
- Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. 2020. [Heterogeneous graph neural networks for extractive document summarization](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6209–6219. Association for Computational Linguistics.
- Danqing Wang, Pengfei Liu, Ming Zhong, Jie Fu, Xipeng Qiu, and Xuanjing Huang. 2019. Exploring domain shift in extractive text summarization. *arXiv preprint arXiv:1908.11664*.
- 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.
- Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. 2016. A sentence compression based framework to query-focused multi-document summarization. *arXiv preprint arXiv:1606.07548*.
- Shuohang Wang and Jing Jiang. 2017. [Machine comprehension using match- lstm and answer pointer](#). In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.
- Jiacheng Xu and Greg Durrett. 2019. Neural extractive text summarization with syntactic compression. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, Hong Kong, China. Association for Computational Linguistics.
- Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Discourse-aware neural extractive model for text summarization. *arXiv preprint arXiv:1910.14142*.
- Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander R Fabbri, Irene Li, Dan Friedman, and Dragomir R Radev. 2019. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7386–7393.
- Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. [Extractive summarization as text matching](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6197–6208. Association for Computational Linguistics.
- Ming Zhong, Pengfei Liu, Danqing Wang, Xipeng Qiu, and Xuan-Jing Huang. 2019a. Searching for effective neural extractive summarization: What works and what’s next. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1049–1058.
- Ming Zhong, Danqing Wang, Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2019b. A closer look at data bias in neural extractive summarization models. *EMNLP-IJCNLP 2019*, page 80.
- 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.
- Haichao Zhu, Li Dong, Furu Wei, Bing Qin, and Ting Liu. 2019. [Transforming wikipedia into augmented data for query-focused summarization](#). *CoRR*, abs/1911.03324.

## A Appendix

### A.1 Query Schema List

As mentioned in Section 3.2, we make query schema list to help annotators design queries. Towards general and specific meeting contents, we further divide the query schema list into *general query schema list* and *specific query schema list*. The detailed list is shown in Table 6.

### A.2 Other Details of Annotation Instruction

We show other details except those in Section 3.

#### A.2.1 Topic Segmentation

**Topics.** We require annotators to use noun phrases to represent the main topics. As we hope to select the most important topics, the number of annotated main topics should not be too much, and 3 to 8 is proper range.

**Distribution of Relevant Text Spans in Meeting Transcripts.** Relevant text spans of a main topic may be scattered in different parts of meetings, e.g., the main topic ‘scope of the project and team building’ in the leftmost part of Figure 2. So annotators are asked to label *all* the relevant spans, and these spans are allowed to be not contiguous.

**Whether Chatting Belongs to an Independent Main Topic or not.** Main topics should be objectively cover most of the meeting contents. In meetings, group members might take much time chatting. Though this is not close to the main theme of the meetings, it should also be counted as a main topic if the chat took lots of time.

#### A.2.2 Query Generation

**Diverse Query Types.** For the specific queries, we encourage the annotators to choose different schemas to compose queries, since we intend to diversify the query types and reduce the bias towards certain query types.

**Independent Query-answer Pairs.** Each query-answer pair should be independent, that is, not dependent on previous queries and answers. For example, for the query ‘How did they reach agreement afterwards?’, it seems that the query is dependent on the previous annotations, and it is ambiguous to know when they reached agreement and what the agreement referred to if we treat the query independently. Annotators should specify the information of when they reached an agreement and what the agreement is to make the query clear.

For example, annotators could rewrite the query as ‘How did the group agree on **the button design** when discussing **the design of remote control?**’.

#### A.2.3 Query-based Summarization

**Informative Writing.** When answering queries like ‘What did **A** think of **X**?’, annotators were required to not only summarize what **A** said, but also briefly mention the context which is relevant to what **A** said. This is designed to rich the contents of summaries and further challenge the model’s capability of summarizing relevant contexts.

**Relevant text spans’ annotation.** Since there might be multiple relevant text spans, we asked annotators to annotate all of them.

**The Usage of Tense.** Since all the meetings happened, we ask annotators to use past tense.

**How to Denote Speakers.** If the gender information is unclear, we would ask annotators not to use ‘he/she’ to denote speakers. We also asked them not to abbreviations (e.g., PM) to denote speakers, and use the full name like ‘Project Manager’ instead.

**Abbreviations.** In the raw meeting transcripts, some of abbreviations are along with character ‘\_’. If the annotators encountered abbreviations, e.g., ‘L\_C\_D\_’, ‘A\_A\_A\_’, etc., they would be required to rewrite them like LCD or AAA.

### A.3 Annotation Review Standards

We launched a ‘*pre-annotation*’ stage in which annotators were asked to try annotating one meeting and the *experts* who are good at our annotation task would review it and instruct the annotators by providing detailed feedback. Requirements include 1) faithfulness, 2) informativeness, 3) lengths of the relevant text spans of designed queries, 4) typo errors, etc. They could continue annotating only if they passed ‘*pre-annotation*’ stage. After ‘*pre-annotation*’, *experts* would keep carefully reviewing all the annotations according to the requirements. We write down the standards for the three annotation stages individually in annotation instruction. Details of annotation review standards could be referred to Appendix A.2.

### A.4 Examples of QMSum Dataset

We show two examples of our proposed QMSum dataset. One belongs to product meeting (Table 7), and the other one is about committee meeting (Table 8).

---

## General Query Schema List

**A: Speaker**

---

1. Summarize the whole meeting. / What did the group/committee discuss in the meeting? (Mandatory)
  2. What did **A** say in the meeting? / Summarize what **A** said.
  3. What was the conclusion / decision of the meeting?
  4. What was the purpose of the meeting?
  5. How did the group/committee split the work?
- 

## Specific Query Schema List

**A, B: Speakers, X: Annotated Main Topics, Y: Subtopics regarding to X**

---

1. Summarize the discussion about **X**. / What did the group/committee discuss **X**? (Mandatory)
  2. Why did the group/committee decide to do sth. when discussing **X**?
  3. What did **A** think of **Y** when discussing **X**? / Summarize **A**'s opinions towards **Y**.
  4. What did **A** think of **X**? / Summarize **A**'s opinions towards **X**. / What did **A** propose in the discussion about **X**?
  5. What was the advantage / disadvantage of sth. with regard to **X**?
  6. Why did **A** think regarding to **X**?
  7. Why did **A** agree / disagree with certain ideas? / Provide the reasons why **A** held certain opinions towards **X**.
  8. Why did **A** think of **Y** when discussing **X**?
  9. What was the decision / conclusion of the discussion about **X**? / Summarize the decision of the discussion about **X**.
  10. Why did **A** agree / disagree with **B** when discussing **X**?
  11. What did **A** recommend to do when discussing **X** and why?
  12. What did **A** learn about topic **X**?
  13. What did **A** and **B** discuss **X**?
- 

Table 6: General and specific query schema lists. **A, B** denote speaker names. **X** indicates one of the annotated main topics, and **Y** means the subtopics regarding to **X**.

---

**Product Meeting (IS1000a)**

Color: **Speakers**, **Main Topics**, **Subtopics**

---

**Turn 0:** **User Interface Designer:** Okay.

.....

**Turn 243:** **Project Manager:** Well, this uh this tool seemed to work.

.....

**Turn 257:** **Project Manager:** More interesting for our company of course, uh profit aim, about fifty million Euro. So we have to sell uh quite a lot of this uh things. ....

**Turn 258:** **User Interface Designer:** Ah yeah, the sale man, four million.

**Turn 259:** **User Interface Designer:** Maybe some uh Asian countries. Um also important for you all is um the production cost must be maximal uh twelve uh twelve Euro and fifty cents.

.....

**Turn 275:** **Project Manager:** So uh well I think when we are working on the international market , uh in principle it has enough customers.

**Turn 276:** **Industrial Designer:** Yeah.

**Turn 277:** **Project Manager:** Uh so when we have a good product we uh we could uh meet this this aim, I think. So, that about finance. And uh now just let have some discussion about what is a good remote control and uh well keep in mind this this first point, it has to be original, it has to be trendy, it has to be user friendly. ....

.....

**Turn 400:** **Project Manager:** Keep it in mind.

---

**Annotated Main Topics**

Scope of the project and team building (**Turn 41 - 245**)

Cost constraints and financial targets of the new remote control project (**Turn 246 - 277**)

Remote control style and use cases (**Turn 277 - 295**)

Prioritizing remote control features (**Turn 343 - 390**)

---

**Queries and Annotated Summaries**

**Query 1:** Summarize the whole meeting.

**Answer:** Project Manager introduced a new remote control project for television sets, and the team got acquainted with each other and technical devices. The remote control would be priced at 25 Euros and a production cost of 12.5 Euros. ....

.....

**Query 2:** What did the group discuss about **prioritizing remote control features**?

**Answer:** User Interface Designer and Industrial Designer expressed a desire to integrate cutting-edge features into the remote. Marketing pointed out that most of the market would buy it for standard use, like changing channels and adjusting volume .....

**Relevant Text Spans:** **Turn 343 - 390**

.....

**Query 4:** Why did **Marketing** disagree with **Industrial Designer** when discussing **prioritizing remote control features**?

**Answer:** Marketing believed that fancy features like IP would not be used by most people. The overwhelming majority of users would want convenient channel browsing and volume adjustment features .....

**Relevant Text Spans:** **Turn 358**

.....

**Query 7:** What did **Project Manager** think of **the cost constraints and financial targets of the new remote control project**?

**Answer:** Project Manager introduced the financial information: 25 Euro selling price and 12.5 Euro production cost. Project Manager then went on to elaborate that the target market would primarily consist of Europe and North America. ....

**Relevant Text Spans:** **Turn 248 - 277**

---

Table 7: A product meeting annotation example in QMSum dataset.



---

**Committee Meeting (Education 4)**

**Color:** **Speakers**, **Main Topics**, **Subtopics**

---

**Turn 0:** **Lynne Neagle AM:** Okay, good morning, everyone. Welcome to the Children, Young People and Education Committee this morning. I've received apologies for absence from .....

.....

**Turn 31:** **David Hopkins:** Yes, sure. The delegation levels are already very high in most authority areas, and we've got agreements in place with the Government to make sure that more money, or as much money as possible .....

**Turn 32:** **Sian Gwenllian AM:** Okay. But just the pressures coming in with the new Act et cetera could mean more expulsions.

**Turn 33:** **David Hopkins:** It shouldn't, but it could. It's difficult to know how headteachers and governing bodies will react. ....

.....

**Turn 44:** **Sharon Davies:** As Nick said, it does get more difficult at key stage 4, and it's working, then, with. It comes back to that team-around-the-family approach .....

.....

**Turn 47:** **David Hopkins:** I don't think I'm allowed to say at this point.

.....

**Turn 228:** **Lynne Neagle AM:** Item 4, then, is papers to note. Just one paper today, which is the Welsh Government's response to the committee's report on the scrutiny of the Welsh Government's draft budget 2020-1. ....

---

**Annotated Main Topics**

---

An increase of exclusions from school and current measures against it (**Turn 1 - 19, Turn 158 - 172**)

The funding issues (**Turn 20 - 38, Turn 177 - 179**)

The networking within the PRU and the transition arrangements (**Turn 39 - 56**)

.....

Schools' awareness of early trauma ACEs (**Turn 180 - 188**)

---

**Queries and Annotated Summaries**

---

**Query 1:** Summarize the whole meeting.

**Answer:** The meeting was mainly about the reasons behind and the measurements against the increasing exclusions from school. The increase brought more pressure to EOTAS in the aspects of finance, transition, curriculum arrangement and the recruitment of professional staff. Although much time and finance had been devoted to the PRU .....

.....

**Query 4:** What was considered by **David Hopkins** as **the factor that affected exclusions**?

**Answer:** David Hopkins did not think that the delegation levels were not high enough in most authority areas. Instead, he thought they had got agreements with the government to make sure that enough money was devolved to school. The true decisive factor was the narrow measure at the end of Stage 4 that drove the headmasters to exclude students or put them into another school.

**Relevant Text Spans:** **Turn 31 - 33**

.....

**Query 6:** What was the major challenge of **the transition of the excluded students**?

**Answer:** The students coming to the end of their statutory education were facing the biggest challenge, for it would be far more difficult for them to go back into the mainstream education process when they turned 15 or 16, not to mention the transition into further education, such as colleges.

**Relevant Text Spans:** **Turn 44 - 49**

.....

---

Table 8: A committee meeting annotation example in QMSum dataset.