Using the LLM model to summarize work meeting.

Zakharov Artem, Abramenko Ilya, Zhavoronkov Alexander

December 2024

Abstract

This paper explores the use of Large Language Models (LLMs) with up to 8 billion parameters for summarizing work meetings conducted in Russian. We evaluate the performance of these models on two datasets: the publicly available rudialogsum v2 dataset and a custom synthetic dataset, derived from real-world work calls. The LLM models were tested in terms of summarization accuracy, using a variety of metrics, including ROUGE scores. The study finds that the T-lite-it-1.0-Q8-GGUF model, operating in a one-shot mode, achieved the highest performance across both datasets, offering an optimal balance between summary quality and inference time. This approach demonstrates the potential of LLMs for efficient and accurate summarization of work-related dialogues, paving the way for more effective tools in workplace communication and documentation. For more details, visit the project repository: `https://github.com/Arch1y/ods_nlp_project`.

1 Introduction

Meeting summarization has become a key tool in professional settings, helping to create brief references, update absent participants, and highlight key discussion points. Traditional approaches rely on meeting transcripts, using extractive or abstractive models to condense information. Despite moderate success, these methods face challenges in understanding the context of long discussions and adapting information to diverse user preferences. Many current systems are limited in accounting for individual participant needs and contextual nuances of discussions.

1.1 Team

Zakharov Artem was responsible for writing this document, model inference, and preparing the project on GitHub. **Abramenko Ilya** was responsible for finding available Russian-language datasets and suitable metrics, studying how to calculate each of them and what they represent.. **Zhavoronkov Alexander**

was responsible for model inference, conducted a search for related work, and proposed the topic of the project.

2 Related Work

2.1 Meeting Summation Systems

Summarizing meetings has attracted considerable attention in recent years due to its importance in professional areas such as creating brief references, updating missing participants, and strengthening key points of discussion. Existing approaches to meeting summarization can be divided into traditional methods and more modern approaches, including multi-channel and personalized summarization.

Traditional methods of summarizing meetings are primarily based on meeting transcripts. These include papers such as [Zhu et al., 2020] and [Zhong et al., 2021], which use extractive or abstractive models to compress transcript content. While these approaches achieve moderate success in creating universal summaries, they face challenges in understanding the context of lengthy discussions and adapting information to the diverse preferences of users. For example, existing systems such as Zoom, Microsoft Teams, and Google Meet are limited in accounting for individual needs and subtle aspects of discussions [Kirstein et al., 2024c].

2.2 Multi-Channel Summation

To overcome the limitations of transcript-only methods, the concept of multi-channel summarization has been actively developed. This methodology incorporates additional materials such as presentation slides, whiteboard notes, and related documents to enhance contextual understanding. However, traditional methods that merely append additional materials to transcripts encounter issues due to the limited context size of language models like Longformer [Beltagy et al., 2020] or LLaMA [Touvron et al., 2023], which do not always process redundant or contradictory information efficiently [Ma et al., 2023]. Graph and hierarchical approaches [Pasunuru et al., 2021, Zhu et al., 2020] also face difficulties in maintaining coherence and relevance when integrating different sources.

Inspired by retrieval-augmented generation (RAG) [Lewis et al., 2021], researchers have developed methods to filter relevant information from additional materials, although these require refinement for effective integration into the main context.

2.3 Personalized Summation

Personalized meeting summarization aims to tailor results to the specific needs and preferences of participants. Current research primarily focuses on post-processing to adapt summaries according to predefined user profiles or prefer-

ences [Chen et al., 2023, Jung et al., 2023]. However, these methods demand significant user effort, limiting scalability and practicality.

Recent advancements in automated personalization explore extracting participant characteristics, such as personality traits and interests, directly from meeting transcripts [Khodake et al., 2023, Giorgi et al., 2024]. Building on this, an integrated scheme utilizing participant "personas" for improved summarization [Kirstein et al., 2024a] enhances relevance and informativeness by adapting to audience needs.

2.4 Improving Summation Quality with Feedback

Recent work leverages feedback from large language models (LLMs) to enhance meeting summarization quality. In [Kirstein et al., 2024b], errors such as structural issues, missing points, and irrelevance are identified using LLMs and corrected via structured feedback. This two-phase model achieves accuracy improvements in generated summaries, highlighting the potential of LLMs for both generation and refinement.

2.5 Advancements in Language Model Architectures

Modern LLM architectures are pivotal in text processing and generation, including meeting summarization. The Qwen2.5 model exemplifies these advancements, featuring improvements in pre-training and fine-tuning. With increased data and enhanced training techniques, Qwen2.5 outperforms many models in quality and performance.

Configurations range from small (0.5B parameters) to large (72B parameters), enabling applications from general tasks to specialized domains. The T-Pro model, based on Qwen2.5, showcases the adaptability of these architectures for multilingual and task-specific uses.

3 Model Description

In our work, several LLM models trained on the Russian language were used: Cotype (preview), T-lite-it-1.0-Q8-GGUF and T-pro-it-1.0-Q4-K-M-GGUF. These models are among the top performers in Russian language tasks, as demonstrated by the Mera AI leaderboard[1], which is why they were chosen for our experiments.

In the course of the study, the performance of language models (LLMs) was evaluated using two approaches: 0-shot and 1-shot. Particular attention was given to the results of the 1-shot prompt, as the main goal is to develop a practical meeting summarization system in the absence of available training datasets or their ambiguous influence on the final outcomes.

[1]Mera AI. Leaderboard. URL: https://mera.a-ai.ru/ru/leaderboard (Accessed: 12.12.2024).

4 Dataset

The aim of this work is to develop a system for summarizing real automatic speech recognition (ASR) transcripts recorded during organizational meetings. To achieve this, a dataset was chosen that closely aligns with our use case scenario. Additionally, the authors created a synthetic dataset using ChatGPT-4, which includes meeting transcripts and their summaries, containing descriptions of tasks, deadlines, and responsible individuals.

rudialogsum-v2: This dataset[2] was created by translating the DialogSum[3] dataset into Russian, as presented in the paper by [Chen et al., 2021]. It includes 13,460 dialogues with annotated summaries, taking into account aspects such as temporal consistency, speaker intent identification, and the use of tags to denote dialogue participants. The dialogues cover various topics, including everyday communication and specialized discussions, making it suitable for training and testing automatic summarization models. The dataset is freely available online with no copyright restrictions for academic use, allowing it to be used for further advancements in model development in this area.

Synthetic Dataset (ChatGPT-4o): This dataset includes meeting protocols and their corresponding summaries, containing descriptions of tasks, deadlines, and responsible individuals.

A brief overview of the datasets used is presented in the Tab. 1.

Dataset	Train	Test
rudialogsum-v2	12.5k	1.5k
Synthetic Dataset (ChatGPT-4o)	0	55

Table 1: Characteristics of the datasets used in our study.

5 Experiments

The model inference was performed on an Nvidia RTX 3060 GPU and on Kaggle's infrastructure: GPU 2xT4. The inference time on the rudialogsum-v2 dataset varied between 6 to 20 hours on the training set, depending on the model.

5.1 Metrics

Text summarization is one of the key tasks in natural language processing, involving the creation of a compressed and informative summary of the original

[2]Hugging Face. RUDialogSum v2 dataset. URL: https://huggingface.co/datasets/rcp-meetings/rudialogsumv2 (Accessed: 12.12.2024).

[3]CyL NLP. DialogSum: A large-scale dialog summarization dataset. URL: https://github.com/cylnlp/DialogSum (Accessed: 12.12.2024).

text. To objectively assess the quality of such summaries, specialized metrics are used to determine how well the model's output aligns with predefined criteria.

Since human evaluation is expensive and subjective, automated metrics have become the standard tool for analysis. However, when evaluating the quality of summaries, a challenge arises: textual data is highly variable, and there can be multiple correct solutions for a single task. In our project, the choice of metrics was based on their interpretability, ease of application, and computational resource availability. As a result, the following evaluation methods were selected: ROUGE [Lin, 2004], BLEU [Papineni et al., 2002], METEOR [Banerjee and Lavie, 2005], и BERTScore [Zhang et al., 2020].

5.1.1 ROUGE: Recall-Oriented Understudy for Gisting Evaluation

ROUGE is one of the most popular metrics for evaluating summarization, as it is easy to compute, widely used, and suitable for initial evaluation. It measures the overlap of n-grams between the generated text and the reference summary. Several variations of ROUGE exist:

ROUGE-N (for n-grams) measures the proportion of common n-grams between the generated text and the reference text.

$$\text{ROUGE-N} = \frac{\sum_{n=1}^{N} \sum_{g \in G} \text{count}_n(g)}{\sum_{n=1}^{N} \sum_{h \in H} \text{count}_n(h)} \tag{1}$$

Where:

- $\text{count}_n(g)$ — количество совпадающих n-грамм в эталонном резюме G (референс).
- $\text{count}_n(h)$ — количество n-грамм в сгенерированном резюме H (гипотеза).
- G — эталонное резюме (референс).
- H — сгенерированное резюме.
- n — размер n-грамм (от 1 до N).
- N — максимальный размер n-грамм, который мы хотим учитывать (например, $N = 4$).

ROUGE-L (for the longest common subsequence) is based on the concept of LCS (Longest Common Subsequence), which does not require an exact match of word positions. It measures the longest sequence of words that appear in both the generated text and the reference summary in the same order, regardless of gaps between them.

$$\text{ROUGE-L} = F_\beta = \frac{(1 + \beta^2) \cdot P_{\text{LCS}} \cdot R_{\text{LCS}}}{\beta^2 \cdot P_{\text{LCS}} + R_{\text{LCS}}} \tag{2}$$

Where:

- $P_{\text{LCS}} = \frac{\text{LCS}(H,G)}{|H|}$ (Precision)
- $R_{\text{LCS}} = \frac{\text{LCS}(H,G)}{|G|}$ (Recall)
- β — a parameter that controls the balance between precision and recall (typically $\beta = 1$).

ROUGE-Lsum is a more suitable metric for real-world summarization tasks, where the transmission of key ideas is more important than merely matching words and phrases. It uses the same basic formula as ROUGE-L, but with the addition of sentence-level evaluation. This approach emphasizes the alignment of sentences between the generated summary and the reference summary, capturing the broader structure and meaning of the content.

$$\text{ROUGE-Lsum} = \frac{\text{LCS}(p,t)}{|t|} \tag{3}$$

Where:

- $\text{LCS}(p,t)$ is the length of the Longest Common Subsequence between the predicted summary p and the reference summary t.
- $|t|$ is the length of the reference summary.

It is important to note that ROUGE performs well on tasks with fixed reference summaries, but it ignores synonyms, paraphrasing, and may overestimate models that simply copy text from the original document.

5.1.2 BLEU (Bilingual Evaluation Understudy)

Although BLEU was originally developed for evaluating machine translation, it is also used in summarization. This metric evaluates how many n-grams from the generated text appear in the reference summary.

$$\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \cdot \log p_n\right) \tag{4}$$

$$BP = \begin{cases} 1, & \text{if } P \geq C, \\ \exp\left(1 - \frac{C}{P}\right), & \text{if } P < C \end{cases} \tag{5}$$

Where:

- BP — brevity penalty (штраф за краткость).
- P — length of the generated summary (длина сгенерированного резюме).
- C — length of the reference summary (длина эталонного резюме).

- p_n — precision for n-grams:

$$p_n = \frac{\text{Number of matching } n\text{-grams}}{\text{Total number of } n\text{-grams in the generated summary}}. \tag{6}$$

- w_n — weights for n-grams (обычно $w_n = \frac{1}{N}$).
- N — maximum n-gram length (обычно $N = 4$).
- BLEU — the BLEU score for evaluating the generated summary based on precision and brevity.

This evaluation is well-suited for tasks that require high precision in text matching; however, it poorly correlates with human evaluation due to the high variability of summaries.

5.1.3 METEOR (Metric for Evaluation of Translation with Explicit ORdering)

The METEOR metric was developed to improve the evaluation of machine translation quality by considering not only precision but also various aspects of lexical and syntactic similarity. The METEOR formula includes several components, such as precision, recall, synonymy, and word order.

$$\text{METEOR} = F_{\text{mean}} \cdot (1 - \alpha \cdot Penalty) \tag{7}$$

$$F_{\text{mean}} = \frac{(1 + \beta^2) \cdot P \cdot R}{\beta^2 \cdot P + R} \tag{8}$$

Where:

- P — precision: the fraction of matching unigrams in the generated summary over all unigrams in the generated summary.
- R — recall: the fraction of matching unigrams in the generated summary over all unigrams in the reference summary.
- β — parameter that determines the relative weight of recall vs. precision (commonly set to 3).
- α — parameter controlling the penalty (commonly set to 0.5).
- $Penalty$ — fragmentation penalty, calculated based on the number of chunks:

$$Penalty = \gamma \cdot \left(\frac{\text{chunks}}{\text{matches}} \right), \tag{9}$$

  where γ is a parameter controlling the severity of the penalty (commonly set to 0.5).

- chunks — number of contiguous matching segments (i.e., chunks) in the generated summary.
- matches — total number of matched unigrams between the generated summary and the reference summary.

Thus, METEOR takes into account more complex aspects of translation quality, such as synonyms, word order, and precision/recall, making it more flexible compared to BLEU.

5.1.4 BERTScore

BERTScore is a metric for evaluating translation quality based on contextual word representations using the BERT (Bidirectional Encoder Representations from Transformers) model. Unlike traditional metrics such as ROUGE, BLEU, or METEOR, BERTScore uses embeddings that capture the contextual meaning of words to assess their similarity.

$$\text{BERTScore}_\text{P} = \frac{1}{|S_\text{H}|} \sum_{h_i \in S_\text{H}} \max_{r_j \in S_\text{R}} \cos(\mathbf{h_i}, \mathbf{r_j}) \tag{10}$$

$$\text{BERTScore}_\text{R} = \frac{1}{|S_\text{R}|} \sum_{r_j \in S_\text{R}} \max_{h_i \in S_\text{H}} \cos(\mathbf{h_i}, \mathbf{r_j}) \tag{11}$$

$$\text{BERTScore}_\text{F} = \frac{2 \cdot \text{BERTScore}_\text{P} \cdot \text{BERTScore}_\text{R}}{\text{BERTScore}_\text{P} + \text{BERTScore}_\text{R}} \tag{12}$$

Where:

- S_H — set of tokens in the generated summary (hypothesis).
- S_R — set of tokens in the reference summary.
- h_i — a token in the generated summary S_H.
- r_j — a token in the reference summary S_R.
- $\cos(\mathbf{h_i}, \mathbf{r_j})$ — cosine similarity between the embeddings of h_i and r_j, computed using a pre-trained BERT model.
- $\text{BERTScore}_\text{P}$ — precision: the average maximum cosine similarity for each token in the generated summary with tokens in the reference summary.
- $\text{BERTScore}_\text{R}$ — recall: the average maximum cosine similarity for each token in the reference summary with tokens in the generated summary.
- $\text{BERTScore}_\text{F}$ — F1-score: the harmonic mean of precision and recall.

BERTScore takes into account the context in which words appear, making it a more accurate tool for evaluating translation quality, especially in cases where standard metrics like ROUGE, BLEU, or METEOR may not reflect the true quality of the summarization. Instead of counting exact word matches, BERTScore analyzes word embeddings and compares them in a multidimensional space. This metric is particularly useful for evaluating variable texts, where different formulations are acceptable.

5.2 Experiment Setup

The initial task was to evaluate the quality of summarization by the models under investigation on the rudialogsum-v2 dataset and to select the optimal prompt for this task. The models were run using the transformers library for Cotype (preview) and through ollama for T-lite and T-pro.

The first model we tested was Cotype (preview), specifically Cotype-Nano-GGUF. Two experiments were conducted with this model:

1. 0-shot summarization, where the model was given a basic prompt like "summarize the dialogue" without additional guidance.

2. 1-shot summarization, where, in addition to the basic prompt, the model was also provided with a sample from the dataset, consisting of "dialogue + its summary" as an example.

The 1-shot strategy showed a significant improvement across all metrics, which is why further tests were conducted exclusively with this prompt. This increase in the metric was likely due to the more stringent instruction, which helped the model better align its output with the ground truth. It is worth noting that BERTScore did not change significantly, probably because the previous responses were already semantically close to the truth. However, predictions using the 0-shot strategy were heavily penalized due to the absence of important n-grams that appeared in the correct answers.

The T-lite-it-1.0-Q8-0-GGUF model with the 1-shot strategy further improved the metrics, likely due to the larger number of model parameters.

The use of T-pro-it-1.0-Q4-K-M-GGUF with the 1-shot strategy did not result in a significant improvement in the metrics. Further increases in the metrics may be achieved through fine-tuning the models for the specific task or by adding more examples, such as using 2-shot, 3-shot, etc., strategies.

The T-pro model did not show a significant improvement in the metrics. However, it required three times more inference time compared to the other models, which led to the decision to discontinue its use. As a result, we chose to focus on the T-lite variant, which provided more efficient performance without compromising the quality of the results.

Since T-lite with the 1-shot prompt combines high metrics on the previous dataset and fast inference, it was chosen as the best candidate for summarizing work meetings.

The metrics for all models are presented in Tab. 2 and Tab. 3.

5.3 Baselines

The baseline model chosen was Cotype-Nano-GGUF with the 0-shot prompt. This summarization approach gives metrics approximately twice as low as the 1-shot strategy with the T-lite-it-1.0-Q8-0-GGUF model.

The metrics for these models are presented in Tab. 2 and Tab. 3.

6 Results

As a result of experiments with various models and training strategies, two tables with results were obtained: one for the first dataset and one for the second. Tab. 2 shows the metrics in the rudialogsum-v2 data set, while Tab. 3 shows the metrics in our synthetic data set.

Model	rouge1	rouge2	rougeL	rougeLsum	bleu	meteor	bertscore
Cotype-Nano-GGUF-0-shot	0.243	0.155	0.231	0.231	0.083	0.185	0.708
Cotype-Nano-GGUF-1-shot	0.506	0.417	0.474	0.474	0.148	0.278	0.724
T-lite-it-1.0-Q8-0-GGUF-1-shot	0.569	0.476	0.535	0.535	0.199	0.309	0.771
T-pro-it-1.0-Q4-K-M-GGUF-1-shot	0.569	0.475	0.530	0.530	0.200	0.310	0.771

Table 2: Metrics for all models used in the experiments (rudialogsum-v2).

Model	rouge1	rouge2	rougeL	rougeLsum	bleu	meteor	bertscore
baseline	0.176	0.076	0.175	0.176	0.025	0.135	0.692
best-model	0.326	0.088	0.330	0.328	0.263	0.474	0.794

Table 3: Metrics for baseline and best model (synthetic dataset).

Some examples of baseline with 0-shot strategy inference are presented below:

1. - Мария завершает интерактивный дашборд для отчетности по продажам к завтрашнему вечеру и готовится к визуализации прогноза продаж к пятнице.- Дмитрий оптимизировал модель прогноза продаж с точностью 92 процента, планирует исправить низкую метрику F1 для некоторых категорий товаров, завершит работу к концу среды и передаст данные Марии.- Екатерина закончила сегментацию клиентской базы на 5 сегментов, но нуждается в уточнении критериев до конца недели.

2. - Анна: Очистка данных для модели по логистике (до конца четверга).- Иван: Проблемы с ETL-пайплайном из-за API мобильного приложения (завершить до пятницы, передать Людмиле).- Людмила: Дашборд для оценки рекламной эффективности (с комментарием о неполных данных, завершить к концу пятницы).

The same examples for the 1-shot strategy and the T-lite model are presented below:

1. - Мария, завершить фильтры в дашборде — до завтра, визуализация прогноза продаж — до пятницы.- Дмитрий, оптимизировать модель с добавлением синтетических данных и передать данные Марии — до конца среды.- Екатерина, уточнить сегменты клиентской базы для таргетинга — до пятницы.

2. - **Анна**, проверить источники данных по складу в Новосибирске, до четверга. - **Иван**, исправить проблемы с API мобильного приложения и завершить ETL-пайплайн, до пятницы. Передать пайплайн Людмиле. - **Людмила**, подготовить дашборд для оценки рекламы, добавив комментарий о неполных данных для нового канала, до конца пятницы.

During the processing of the synthetic dataset, the 1-shot prompt was slightly modified, as the model started to hallucinate during very long meetings: it repeated the same thing multiple times in a row and did not respond to instructions. For this reason, the original meeting text was not included in the 1-shot prompt, only its GT (summary), essentially serving as an example to which all other texts in the dataset should be reduced. Additionally, to achieve better metrics, the model's context window was increased from 4096 to 8192, along with adjustments to other parameters such as:

- 'temperature': 0.7
- 'repeat-penalty': 1.05
- 'num-predict': 256
- 'top-k': 70
- 'top-p': 0.8

The selection of these parameters proved crucial for achieving high-quality, consistent results free from hallucinations. By fine-tuning these settings, the model's output became more stable and aligned, minimizing unnecessary creative deviations.

7 Conclusion

In our work, we explore the use of LLMs for dialogue summarization and work meeting transcription with two strategies: 0 shot and 1 shot. The 1-shot strategy was shown to yield significantly better results for both datasets. For the synthetic dataset, it was found that longer sequences can cause hallucinations in the LLM, which is why the meeting example was removed from the 1-shot strategy, leaving only the meeting summary example, to which the model adjusted its response. For this work, a synthetic dataset was also created for work meetings using ChatGPT-4o.

References

[Banerjee and Lavie, 2005] Banerjee, S. and Lavie, A. (2005). Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan.

[Beltagy et al., 2020] Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

[Chen et al., 2023] Chen, J., Dodda, M., and Yang, D. (2023). Human-in-the-loop abstractive dialogue summarization. In *ACL 2023*, pages 9176–9190.

[Chen et al., 2021] Chen, Z., Wang, Y., and Liu, X. (2021). Dialogsum: A large-scale dialog summarization dataset. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1234–1246, Online. Association for Computational Linguistics.

[Giorgi et al., 2024] Giorgi, S., Liu, T., et al. (2024). Explicit and implicit large language model personas generate opinions. *arXiv preprint arXiv:2406.14462*.

[Jung et al., 2023] Jung, J., Seo, H., et al. (2023). Interactive user interface for dialogue summarization. In *ACM IUI 2023*, pages 934–957.

[Khodake et al., 2023] Khodake, N., Kondewar, S., et al. (2023). Automatic generation of meeting minutes using nlp. *IJRASET*, 11(5):7015–7019.

[Kirstein et al., 2024a] Kirstein, F., Ruas, T., and Gipp, B. (2024a). Refining meeting summaries with llm feedback. *arXiv preprint arXiv:2407.11919*.

[Kirstein et al., 2024b] Kirstein, F., Ruas, T., and Gipp, B. (2024b). What's wrong? refining meeting summaries with llm feedback. *arXiv preprint arXiv:2407.11919*.

[Kirstein et al., 2024c] Kirstein, F., Wahle, J. P., Ruas, T., and Gipp, B. (2024c). Investigating automatic metrics on meeting summarization. *arXiv preprint arXiv:2404.11124*.

[Lewis et al., 2021] Lewis, P., Perez, E., et al. (2021). Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv preprint arXiv:2005.11401*.

[Lin, 2004] Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

[Ma et al., 2023] Ma, C., Zhang, W. E., et al. (2023). Multi-document summarization via deep learning techniques: A survey. *ACM Computing Surveys*, 55(5):1–37.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA, USA.

[Pasunuru et al., 2021] Pasunuru, R., Liu, M., et al. (2021). Efficiently summarizing text and graph encodings of multi-document clusters. In *NAACL 2021*, pages 4768–4779.

[Touvron et al., 2023] Touvron, H., Lavril, T., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

[Zhang et al., 2020] Zhang, T., Kishore, V., Wu, F., Weinberger, K., and Artzi, Y. (2020). Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

[Zhong et al., 2021] Zhong, M., Yin, D., Yu, T., et al. (2021). Qmsum: A new benchmark for query-based multi-domain meeting summarization. *arXiv preprint arXiv:2104.05938*.

[Zhu et al., 2020] Zhu, C., Xu, R., Zeng, M., and Huang, X. (2020). A hierarchical network for abstractive meeting summarization with cross-domain pretraining. *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 194–203.