Evaluating the Effectiveness of Large Language Models in Meeting Summarization with Transcript Segmentation Techniques

How well does *gpt-3.5-turbo* perform on meeting summarization with topic and context-length window segmentation?

Kristóf András Sándor [1]

Supervisor(s): Catholijn Jonker, Morita Tarvirdians

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 25, 2023

Name of the student: Kristóf András Sándor
Final project course: CSE3000 Research Project
Thesis committee: Catholijn Jonker, Morita Tarvirdians, Mathijs Molenaar

Abstract

Large Language Models (LLM) have brought significant performance increase on many Natural Language Processing tasks. However LLMs have not been tested for meeting summarization. This research paper examines the effectiveness of the *gpt-3.5-turbo* model on the meeting summarization domain. However due to input length limitations, it cannot be applied directly to this task. Thus the paper investigates two segmentation methods: a simple context-length window approach and topic segmentation using Latent Dirichlet Allocation (LDA). The context-length window approach's performance is close to the Pointer Generator framework. The topic segmentation gives worse results. Overall *gpt-3.5-turbo* performs worse with both approaches than state-of-the-art models which use a transformer architecture adapted for long documents.

1 Introduction

Meeting summaries enable participants to find the key takeaways and decisions of a meeting. This is valuable for both attendees and non-attendees alike. However, meeting summarization is a difficult and laborious task for people [1]. Automatic meeting summarization has emerged as a viable solution to improve the efficiency and accuracy of this process.

Research of meeting summarization can be divided into *extractive summarization* and *abstractive summarization*. Extractive summaries are a concatenation of important sentences. However abstractive summaries, which is a coherent text paraphrasing the transcript, are generally preferred by people over extractive summaries [2]. Previous research has produced a variety of abstractive summarization approaches including graph-, template- and query-based methods. The most notable are Longformer-BART [3] and DialogLM [4]. The former creates abstractive summaries by training a model with the longformer [5] architecture, which is based on the transformer architecutre but with modified attention for long documents. The latter is a window-based pre-trained sequence-to-sequence model focusing on dialogue summarization. A comprehensive overview of all the methods can be found in Kumar and Kabiri [1] and Rennard et al. [6].

Recent advances in Large Language Models (LLMs) have enabled automatic abstractive text summarization to reach a level comparable to that of freelance writers [7]. Moreover, news summaries from gpt-3 are generally preferred over existing fine-tuned models [8].

However, the task of dialogue summarization, particularly meeting summarization, has not yet been tested on state-of-the-art LLMs. It is also distinct from other forms of summarization tasks, as it presents unique challenges due to document length, low information density, multi-party aspect, and diverse roles and linguistic styles [1]. Consequently, this paper aims to address this research gap by examining the effectiveness of LLMs in meeting summarization. Moreover, this research examines transcript segmentation. The research question this paper aims to answer is: *What is the performance of the gpt-3.5-turbo model on meeting summarization with transcript segmentation?*

The context length of LLMs is a crucial factor to consider in meeting summarization. Most LLMs, such as LLaMa-based ones [9] and OpenAI's GPT-3 [10], have context length limitations ranging from 2 to 4 thousand tokens. This constraint poses a challenge when dealing with meeting transcripts, as they often exceed this. There is ongoing development of larger context length models, however these are either still being developed [11], or are in limited beta [1] [12; 13]. Besides the hard limitation of the models, segmentation, or community detection has been widely used in the literature to achieve better results. Zhang et al. [14] found that the *retrieve-then-summarize* pipeline works better than other approaches.

Two segmentation methods were used: a context-length window approach and topic segmentation. The context-length window approach is a simple way to generate segments. It takes as much of the meeting as the context length allows, and considers that a chunk. Then, if there is still text left, it moves on until the entire transcript is chunked.

Topic segmentation finds the topics in a text and uses the relevant utterances for each topic to construct a segment. In the text analysis literature, the most popular [15] and effective [16] non-neural topic modelling technique is latent Dirichlet allocation (LDA) [17]. Most of the topic modelling in summarization research uses LDA [18; 19; 20; 21; 22; 23; 24]. In meeting summarization, apart from LDA, LCSeg [25] is also used in Banerjee et al. [26] and Oya et al. [27]. LCSeg uses an analysis sliding window approach with cosine similarity and TF.IDF [28] to calculate a lexical cohesion score at each sentence break. Shang et al. [29] uses latent semantic analysis (LSA) as dimensionality reduction for the TF.IDF matrix, then uses k-means clustering to get the topics. In the case of neural topic modelling, in Liu et al. [30] the topics are trained on labels created based on rules from domain experts. Li et al. [31] uses a neural topic segmenter model called SegBot [32], which uses an encoder-decoder architecture to predict where the segment boundaries are. Since most of the research focuses on LDA, this is what was used here as well. It is further explained in Section 2.

The paper is structured as follows. In Section 2 the segmentation techniques and the prompt are explained. Then, in Section 3 the setup of the evaluation along with the results are described. Afterwards a discussion placing the research in a broader context can be read in Section 4. In Section 5 a reflection on the ethical aspects takes place. This is followed by limitations and future work in Section 6. The paper and the key findings are summarized in the Conclusion.

2 Methodology

This section deals with the choices which describe how the model was used to generate meeting summaries through

[1] While conducting this research, the gpt-3.5-turbo-16k model was released as a publicly available state-of-the-art LLM with 16k context length

meeting segmentation. It explains meeting segmentation methods and goes into detail about how the prompt is built.

2.1 Segmentation techniques

Context-Length Window

To demonstrate the performance of state-of-the-art LLMs on meeting summarization, the following approach was implemented. For preprocessing, when a speaker is talking and their speech is segmented into multiple turns, those turns are combined. This creates a more cohesive meeting and results in less tokens sent to the model. Tokens are words or parts of words which are frequently occurring sequences of characters. After combining the turns, the tokens are counted in each turn. The tiktoken tokenizer package [33] from OpenAI was used with the *cl100k_base* encoding, since gpt-3.5-turbo uses this. During chunking the prompt's token count can be checked against the context length.

Text from the turns are appended to the prompt until the total text in the prompt reaches 91% of the context length. Thus the summaries are at most 10% of the input, in line with guidelines [4]. This results in the meetings having on average 2.32 chunks for the AMI [34] and 4.57 chunks for the ICSI [35] corpus.

Topic Segmentation with Latent Dirichlet Allocation

To explore *gpt-3.5-turbo* on meeting summarization with topic segmentation, topic modelling with LDA is employed. LDA is a method for finding hidden topics in a document or collection of documents. Each document is assumed to be a mixture of topics, and each topic is a distribution of words. The model is a Bayesian generative model which infers the topic structure of the corpus and the topic composition of each document in an unsupervised way.

For LDA, the data is preprocessed in the following way. After merging the turns like before on the speaker, turns are tokenized, filtered and lemmatized. They are filtered based on frequency analysis, the top 10% of tokens are removed for each corpus, which appear too often in every topic to be meaningful, and are a result of the nature of speech. The NLTK [36] English stop words are also removed. Since preprocessing and removing words is an inherently domain specific process, custom meaningful words which were judged to be relevant in the domain like 'button' and 'controls' were kept in. These can be viewed in the codebase[2] constants. The LDA model is trained with the Gensim python package [37]. The turns are classified into the most likely topic, with *min_probability=0.5*, which gave large enough chunks that the summarization could be meaningful. In the AMI corpus, the average number of topics per document is eight, thus *num_topics=8*. A model is trained for each transcript, since each meeting can discuss different topics. The resulting utterances from each topic make up its own prompt.

2.2 Prompts

Modern LLMs require a natural language prompt as input to generate text. Prompt Engineering is a way to enable the full capabilities of the models. We are still in a very early phase of Prompt Engineering, and further research is needed to explain the reasons why some of these techniques work. The best practices from OpenAI [38] include writing clearly, providing reference context, splitting tasks, chain-of-thought prompting, using LLMs only when there is not something better available, and testing changes. We try to adhere to these principles. Since summarization is a common NLP task, the *summarize* keyword is very helpful on its own. The prompt separates the transcript to be summarized and the prompt itself with three double quotes, which is the recommendation. It also specifies the nature of the summary ('detailed') clearly, and gives an approximate length for the response.

For the different segmentation techniques, slightly different prompts are used. The length of the response text from the topic segmentation method needs to be shorter, since there are more segments. While in the context-length window method the phrase for the desired length is *paragraphs*, in topic segmentation *2-3 sentences* is used instead.

```
Summarize the meeting below in detailed
   paragraphs. Include all important
   information. Your answer should only
   include the summary.

Transcript: """
B: Okay Right well this is the kick-off
   meeting for our our project and this
   is just what we're gonna be doing
   over the next twenty five minutes so
   first of all just to kind of make
   sure that we all know each other I'm
   Laura and I'm the project manager Do
   you want to introduce yourself again
D: Great
A: Hi I'm David and I'm supposed to be
   an industrial designer
...
B: So thank you all for coming
A: Cool
"""

 Summary:
```

Figure 1: An example of the prompt given to *gpt-3.5-turbo* with the Context-Length Window approach to summarize a meeting transcript

The summaries are generated with zero-shot prompting, without giving examples of summaries. Few-shot prompting, that is, providing a few examples for the in-context learning of the LLM is not possible due to the issue of context length. Fine-tuning would be possible with the AMI, ICSI and ELITR datasets, however that is out of scope.

3 Experimental Setup and Results

3.1 Datasets

The two most popular meeting corpora is the AMI [34] and the ICSI [35] datasets. There is also the MEMO [42] and

[2] https://github.com/emherk/Recap

	AMI				ICSI			
	R-1	**R-2**	**R-L**	**BERTScore**	**R-1**	**R-2**	**R-L**	**BERTScore**
Our Approach - Topic Segmentation	40.92	11.30	19.31	59.77	37.68	7.35	16.10	56.89
PGNet [39]	42.60	14.01	22.62	-	35.89	6.92	15.67	-
Our Approach - Context-Length Window	43.22	12.31	21.42	61.45	39.04	7.98	16.45	58.13
TopicSeg [40]	53.29	13.51	26.90	-	-	-	-	-
DIALOGLM (l = 5, 120)[4]	53.72	19.61	51.83*	-	49.56	12.53	47.08*	-
Longformer-BART [3]	54.81	20.83	25.98	-	43.40	12.19	19.29	-

Table 1: ROUGE and BERTScore for the Context-Length Window and Topic Modelling approaches with gpt-3.5-turbo compared to other models in the literature. Scores taken from Feng et al. Our approach is highlighted in italics. [41].

the ELITR [43] corpora. The former includes videos as well as meeting transcripts, and the latter one is the first and only non-English meeting dataset [1]. ELITR focuses on minuting, providing bullet lists as summaries.

In general, meeting NLP tasks suffer from a lack of meeting datasets for training. However this research utilizes pretrained LLMs with zero-shot prompting, and thus there is no training of summarizer models involved. Thus without the need for large amounts of data, the AMI and ICSI datasets were used to keep in line with the literature and to make the results comparable to other methods. The AMI dataset contains approximately 65 hours of meetings where the participants act as members of a product team creating a remote control. The ICSI dataset includes 72 hours of naturally-occurring, unscripted meetings encompassing academic discussions in the field of NLP, as well as discussions specifically related to the ICSI dataset. They both include meeting transcripts with human annotated abstractive and extractive summaries. The preprocessed JSON version [3] [6] was used.

3.2 Evaluation Metrics

Since human evaluation of summaries is out of scope for this research, an automatic Natural Language Generation (NLG) metric is going to be used. This section is about concerns of these metrics.

In recent years there have been new NLG metrics proposed in the literature [44]. However these all face criticism [1; 6; 44; 45]. There are a few problems outlined. Firstly, the metrics are clearly described in terms of how they are calculated, however the way they should be interpreted are not well illustrated. Secondly, embedder-based metrics such as BERTScore [46], (which measures distance to the closest embedded word in the reference summary) and MoverScore [47] (which calculates based on weighted embedded n-grams) aim to measure semantic similarity between texts, however that is not representative of the aim of abstractive summarization [6; 45]. The reason is that having similar meaning does not mean presenting the same amount or similar information. Thirdly, the flaws of Natural Language Generation evaluation is well highlighted with the fact that all popular automatic evaluation metrics correlate closer to ROUGE-1 (percentage of 1-gram matches) than human judgement [45].

Deutsch and Roth [45] suggest that the important indicator of a quality summary is information instead of syntactic or semantic similarity. The authors suggest a metric to evaluate generated text called QAEval [48]. This is based on question answering and thus seeks to provide a closer correlation to the information in the summary rather than semantic similarity. It uses a question generation and question answering model. The questions and answers are generated on the gold summary, then the answering model is used to get the answers from the generated summary. The answer is then compared with ROUGE or BERTScore. All together, it is likely that QAEval is a more suitable metric for NLG. [4]

This means reporting the n-gram-based ROUGE scores are in line with the rest of the literature, however they do not necessarily indicate a quality summary. The BERTScore is also reported for future comparability. These scores are calculated for the provided gold summaries for the AMI and ICSI datasets. The evaluate package from huggingface [49] was used.

The BERT model used was *microsoft/deberta-xlarge-mnli* and the hashcode is *microsoft/deberta-xlarge-mnli_L40_no-idf_version=0.3.12(hug_trans=4.15.0)*. This model correlates the most with human judgement [46] and is in line with the reporting from the literature. All scores are averages of all meetings in the two datasets compared to the provided gold summaries.

3.3 Results and Model details

For the *gpt-3.5-turbo* model the *gpt-3.5-turbo-0301* checkpoint was used. The temperature was set to 0.3, making it more deterministic. Additionally, the presence penalty was set to 0.5 to slightly punish new words, making it more focused on the meeting and most likely increasing ROUGE scores. The maximum tokens allowed was 372 for each chunk, based on the calculation above in Section 2.1. Table 1 shows the ROUGE scores for the AMI and ICSI meeting datasets.

Due to the nature of these metrics explained in the previous section, these results must be interpreted with caution. The *gpt-3.5-turbo* model with the context-length window and LDA topic segmentation approaches perform worse on the ROUGE metrics compared to the state-of-the-art [1] meeting summarization approaches like DialogLM [4] and Longformer-BART [3]. The context-length window approach performs 2.30 points better than the topic segmentation approach and 0.62 points better than PGNet [39] on the

[3] https://github.com/guokan-shang/ami-and-icsi-corpora

[4] The author ran into setup problems both with the sacrebleu pip version and the repro docker version, thus the scores are not reported

ROUGE-1 metric. Further exploration on the reasons can be found in the following section.

4 Discussion

The state-of-the-art models DialogLM and ConvoSumm both use a modified attention compared to the transformer architecture [50] specifically for handling long contexts. These two models are currently the best at meeting summarization [1]. This indicates that enabling the model to directly process long documents is preferable to segmenting the documents.

The context-length window approach's performance is close to the Pointer Generator (PGNet) [39] framework. That is a sequence-to-sequence model which switches between abstractive and extractive summarization, thus it can copy from the input text while generating a normally abstractive summary.

Additionally, the ICSI scores are lower compared to AMI. This is the same with the rest of the state-of-the-art meeting summarization approaches. One possible reason for this is that ICSI meetings are longer, and are real meetings not role-play. This results in less structure in the meeting, which can be a reason for the worse summaries.

For both datasets, the topic segmentation approach performed worse. This could be due to the nature of LDA: The meeting is segmented based on topic, and thus the conversation might not be continuous. This can result in less structure and thus worse summaries. Additionally, the topic segmentation method also has worse ROUGE scores than TopicSeg, which uses a neural topic segmenter and the Pointer Generator Network, but with hierarchical attention on utterance to word, word to segment and segment to meeting level.

The BERTScore results are harder to compare to the literature. These scores are not reported in other state-of-the-art models [4; 3; 51; 52]. Most of the literature, including two surveys [1; 6] gave only ROUGE scores. This fact along with lack of interpretability of the scores and outlined problems with NLG metrics, make human evaluation ideal. However that is out of scope for this project.

5 Responsible Research

Reproducibility

The summaries can be found in the codebase under *output*. They can be generated by configuring an OpenAI secret key, then running *run.py* to run get all or using *main.py* in the console with the path to the preprocessed json transcript. The preprocessing can be replicated by running *preprocessing.py* with the json datasets under *data/ami-corpus* and *data/icsi-corpus* respectively. All scores can be found under *evaluation*, and can be replicated by running *evaluate_corpus* with the required corpus's name, metric, and method as parameters.

Models and Privacy

Due to the fast-changing field of NLP, model names can refer to different models over time. To counter this, the code and model snapshot was made available, enabling replication and verification of the results.

Meeting transcripts risk containing personally identifiable information. This risk is minimized by using public datasets with anonymized information.

6 Limitations and Future Work

It is important to recognize and address certain limitations that can impact the extent and applicability of the findings. Firstly, the datasets utilized in the study consist of project meetings and academic discussions, which implies that the model's performance has not been evaluated on other types of meetings. Moreover, it is necessary to consider the representativeness of these meetings in relation to real-world meetings that may exhibit greater linguistic diversity. Additionally, the study does not account for scenarios where meetings involve morally questionable topics or participants use offensive language, thus warranting further investigation into the model's behavior under such circumstances. Another limitation is that the generated summaries are only compared to one gold summary. This is the standard in the literature, however we risk overlooking alternative valid summarizations that capture different aspects of the meeting, or use different words. It may also be that the prompts can be built in a better way, which would produce better summaries.

Recent LLMs seem very capable at many NLP tasks. However for some downstream tasks, like meeting summarization, more research is needed to discover what the best way to perform them are. The *gpt-3.5-turbo* model has potential for zero-shot long-form meeting summarization. Recently the gpt-3.5-turbo-16k was released with a 16384 token context length. Early results confirm what others have found, namely that segmentation helps with more than just context length issues. Exploring longer context length models zero or even few-shot prompting would be a useful topic to research. Additionally, it would be compelling to explore the performance of a state-of-the-art large context length LLM that has been fine-tuned specifically for meeting summarization. Automatic evaluation metrics continue to be a limiting factor in summarization and NLG, however recent advances in QA techniques are promising. Using state-of-the-art LLMs for these techniques could provide a way to go forward.

7 Conclusion

In this paper, the performance of the *gpt-3.5-turbo* model was tested on the abstractive meeting summarization domain. To avoid issues with context length and to ensure focused attention on each part, the meetings are segmented. Context-length limited segmentation and topic segmentation with LDA were used. The context-length window method performs better than topic segmentation, and the former is comparable to Pointer-Generator Network. Both approaches perform worse on the ROUGE metrics than other state-of-the-art models.

References

[1] L. P. Kumar and A. Kabiri, "Meeting summarization: A survey of the state of the art,"

[2] G. Murray, G. Carenini, and R. Ng, "Generating and validating abstracts of meeting conversations: a user

study," in *Proceedings of the 6th International Natural Language Generation Conference*, Association for Computational Linguistics, July 2010.

[3] A. R. Fabbri, F. Rahman, I. Rizvi, B. Wang, H. Li, Y. Mehdad, and D. Radev, "Convosumm: Conversation summarization benchmark and improved abstractive summarization with argument mining," *ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, pp. 6866–6880, 6 2021.

[4] M. Zhong, Y. Liu, Y. Xu, C. Zhu, and M. Zeng, "Dialoglm: Pre-trained model for long dialogue understanding and summarization," *CoRR*, vol. abs/2109.02492, 2021.

[5] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *CoRR*, vol. abs/2004.05150, 2020.

[6] V. Rennard, G. Shang, J. Hunter, and M. Vazirgiannis, "Abstractive meeting summarization: A survey," 8 2022.

[7] tianyi zhang, faisal ladhak, esin durmus, percy liang, kathleen mckeown, and tatsunori b. hashimoto, "benchmarking large language models for news summarization," 1 2023.

[8] T. Goyal, J. J. Li, and G. Durrett, "News summarization and evaluation in the era of gpt-3,"

[9] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models,"

[10] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[11] google, "palm 2 technical report," 2023.

[12] openai, "gpt-4 technical report," 3 2023.

[13] Anthropic, "Introducing 100K Context Windows." https://www.anthropic.com/index/100k-context-windows, June 2023. Accessed 05/06/2023.

[14] Y. Zhang, A. Ni, T. Yu, R. Zhang, Chenguang, Z. Budhaditya, D. A. Celikyilmaz, A. H. Awadallah, and D. Radev, "An exploratory study on long dialogue summarization: What works and what's next,"

[15] H. Zhao, D. Phung, V. Huynh, Y. Jin, L. Du, and W. L. Buntine, "Topic modelling meets deep neural networks: A survey," *CoRR*, vol. abs/2103.00498, 2021.

[16] G. B. Mohan and R. P. Kumar, "A comprehensive survey on topic modeling in text summarization," in *Micro-Electronics and Telecommunication Engineering* (D. K. Sharma, S.-L. Peng, R. Sharma, and D. A. Zaitsev, eds.), (Singapore), pp. 231–240, Springer Nature Singapore, 2022.

[17] D. M. Blei, A. Y. Ng, and J. B. Edu, "Latent dirichlet allocation michael i. jordan," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[18] L. Zhao, W. Xu, and J. Guo, "Improving abstractive dialogue summarization with graph structures and topic words," in *Proceedings of the 28th International Conference on Computational Linguistics*, (Barcelona, Spain (Online)), pp. 437–449, International Committee on Computational Linguistics, Dec. 2020.

[19] K. A. R. Issam, S. Patel, and S. C. N, "Topic modeling based extractive text summarization," 6 2021.

[20] M. Ailem, B. Zhang, and F. Sha, "Topic augmented generator for abstractive summarization," 8 2019.

[21] N. Liu, X.-J. Tang, Y. Lu, M.-X. Li, H.-W. Wang, and P. Xiao, "Topic-Sensitive Multi-document Summarization Algorithm," in *2014 Sixth International Symposium on Parallel Architectures, Algorithms and Programming*, pp. 69–74, IEEE, July 2014.

[22] L. Wang and C. Cardie, "Unsupervised Topic Modeling Approaches to Decision Summarization in Spoken Meetings," June 2016. arXiv:1606.07829 [cs].

[23] DIT, NKUA, Grece, N. Gialitsis, N. Pittaras, DIT, NKUA, IIT, NCSR-D, Grece, P. Stamatopoulos, and DIT, NKUA, Grece, "A topic-based sentence representation for extractive text summarization," in *Proceedings of the Workshop MultiLing 2019: Summarization Across Languages, Genres and Sources associated with RANLP 2019*, pp. 26–34, Incoma Ltd., Shoumen, Bulgaria, Dec. 2019.

[24] Z. Wu, L. Lei, G. Li, H. Huang, C. Zheng, E. Chen, and G. Xu, "A topic modeling based approach to novel document automatic summarization," *Expert Systems with Applications*, vol. 84, pp. 12–23, Oct. 2017.

[25] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing, "Discourse segmentation of multi-party conversation," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - ACL '03*, vol. 1, (Sapporo, Japan), pp. 562–569, Association for Computational Linguistics, 2003.

[26] S. Banerjee, P. Mitra, and K. Sugiyama, "Abstractive meeting summarization using dependency graph fusion," 9 2016.

[27] T. Oya, Y. Mehdad, G. Carenini, and R. Ng, "A template-based abstractive meeting summarization: Leveraging summary and source text relationships," pp. 19–21, 2014.

[28] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing Management*, vol. 24, no. 5, pp. 513–523, 1988.

[29] G. Shang, W. Ding, Z. Zhang, A. Tixier, P. Meladianos, M. Vazirgiannis, and J.-P. Lorré, "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)*, (Melbourne, Australia), pp. 664–674, Association for Computational Linguistics, July 2018.

[30] C. Liu, P. Wang, J. Xu, Z. Li, and J. Ye, "Automatic dialogue summary generation for customer service," pp. 1957–1965, ACM, 7 2019.

[31] M. Li, L. Zhang, H. Ji, and R. J. Radke, "Keep meeting summaries on topic: Abstractive multi-modal meeting summarization," pp. 2190–2196, Association for Computational Linguistics, 7 2019.

[32] J. Li, A. Sun, and S. R. Joty, "Segbot: A generic neural text segmentation model with pointer network," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden* (J. Lang, ed.), pp. 4166–4172, ijcai.org, 2018.

[33] OpenAI, "tiktoken: A fast bpe tokeniser for use with openai's models." https://github.com/openai/tiktoken, 2023.

[34] W. Kraaij, T. Hain, M. Lincoln, I. Mccowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, W. Post, D. Reidsma, and P. Wellner, "The ami meeting corpus," 2005.

[35] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, "The icsi meeting corpus," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 1, pp. 364–367, 2003.

[36] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[37] R. Řehůřek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks*, (Valletta, Malta), pp. 46–50, University of Malta, 2010.

[38] "Gpt best practices." https://openai.com/docs/guides/gpt-best-practices. Accessed: 2023-06-20.

[39] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," *CoRR*, vol. abs/1704.04368, 2017.

[40] M. Li, L. Zhang, H. Ji, and R. J. Radke, "Keep meeting summaries on topic: Abstractive multi-modal meeting summarization," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), pp. 2190–2196, Association for Computational Linguistics, July 2019.

[41] X. Feng, X. Feng, and B. Qin, "A survey on dialogue summarization: Recent advances and new frontiers," *CoRR*, vol. abs/2107.03175, 2021.

[42] M. Tsfasman, K. Fenech, M. Tarvirdians, A. Lorincz, C. Jonker, and C. Oertel, "Towards creating a conversational memory for long-term meeting support: predicting memorable moments in multi-party conversations through eye-gaze," *ACM International Conference Proceeding Series*, pp. 94–104, 11 2022.

[43] A. Nedoluzhko, M. Singh, M. Hledíková, G. Tirthankar, and O. Bojar, "ELITR minuting corpus," 2022. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics, Charles University.

[44] A. Celikyilmaz, E. Clark, and J. Gao, "Evaluation of text generation: A survey,"

[45] D. Deutsch and D. Roth, "Understanding the extent to which content quality metrics measure the information quality of summaries," pp. 300–309.

[46] T. Zhang, V. Kishore, F. wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert,"

[47] W. Zhao, M. Peyrard, F. Liu, Y. Gao, C. M. Meyer, and S. Eger, "Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance,"

[48] D. Deutsch, T. Bedrax-Weiss, and D. Roth, "Towards question-answering as an automatic metric for evaluating the content quality of a summary," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 774–789, 8 2021.

[49] H. Face, "Evaluate." https://github.com/huggingface/evaluate, 2021. Accessed: 2023-06-19.

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017.

[51] C. An, M. Zhong, Z. Geng, J. Yang, and X. Qiu, "Retrievalsum: A retrieval enhanced framework for abstractive summarization," 9 2021.

[52] Y. Zhang, A. Ni, Z. Mao, C. H. Wu, C. Zhu, B. Deb, A. H. Awadallah, D. R. Radev, and R. Zhang, "Summˆn: A multi-stage summarization framework for long input dialogues and documents," *CoRR*, vol. abs/2110.10150, 2021.