Contextual Summarization of Email Threads

Fozia Khan[1], Muhammad Hussain Mughal[2,*], Shaukat Wasi[1],

[1]Department of Computer Science, MAJU, Karachi, Pakistan
[2]Department of Computer Science, Sukkur IBA University, Sukkur, Pakistan
[*]Corresponding author: muhammad.hussain@iba-suk.edu.pk

Abstract

Email has become the primary medium for official communication, enabling the exchange of text, files, and attachments among recipients. A single email thread often encapsulates extensive information shared among multiple participants across diverse topics, resulting in complex and lengthy discussions. Effective summarization of email threads is crucial for generating concise and precise summaries without omitting critical details. This study presents a novel methodology for generating contextual summaries of email threads. The proposed framework begins with constructing a custom dataset comprising university event related email threads $(2 < thread\ length < 6)$ annotated for summarization tasks. The summarization process involves a three step approach: (1) clustering semantically similar sentences using K-Means and Agglomerative Hierarchical Clustering, (2) extracting contextual information from clusters using Latent Dirichlet Allocation (LDA) and Key-Phrase Extraction, and (3) generating abstractive summaries for each contextual cluster using pre-trained transformer models (BART and T5). The proposed approach was systematically evaluated at each stage using standard automatic evaluation metrics, demonstrating its effectiveness in identifying and condensing essential information from lengthy threads. The results highlight the potential of this methodology to streamline email thread summarization. Future work will explore advanced techniques and models to enhance the quality and applicability of contextual summarization further.

Keywords—Email Thread Summarization, Contextual Summary, Actions, Natural Language Processing, Machine Learning, Extractive Summarization, Abstractive Summarization, contexts, Clustering

1 Introduction

Email has become official communication globally, with standard components: metadata (subcomponents are To, From, cc, bcc, and date), subject line, and email body. An email thread is a chain of related emails that multiple participants reply to on a particular topic. It often includes 're' (reply) or 'fwd' (forward) in the subject line, indicating the nature of the communication. This work focuses on threads that involve replies within the chain. Text summarization shortens lengthy text or documents into concise summaries while retaining key information [1]. The two main approaches are extractive, which selects essential sentences, and abstractive, which generates new sentences maintaining the main idea [2]. Both have limitations, such as introducing bias or failing to detect their semantics [3]. Email thread summarization can save recipients' efforts and time by providing concise messages, allowing them to identify the essential details and main ideas quickly. Furthermore, context refers to the information or conditions surrounding the particular event. Contextual summarization provides a summary of group emails shared in a similar context. In particular, it identifies the action items performed and decisions taken concerning each context from an email within a thread. [4] The existing research had focused on email thread summarization. During our analysis, we identified generated summaries from various models that lack coherence and fail to establish a meaningful relationship with their context. Merely selecting important sentences does not result in a complete summary of the entire email thread. The system generated summaries often overlook crucial components such as action items, decisions made, and responsibilities assigned, which are essential elements of a practical summary.

We designed a system that analyzes the email content, extracts context from each email within a thread,

ISSN: 2523-0379 (Online), ISSN: 1605-8607 (Print)
DOI: https://doi.org/10.52584/QRJ.2302.04
This is an open access article published by Quaid-e-Awam University of Engineering Science & Technology, Nawabshah, Pakistan under CC BY 4.0 International License.

groups all the similar extracted contexts, and concisely summarizes them without losing any essential information. We collected university related emails because the standard datasets lacked multi contextual information, essential for effective summarization. We focused on event related emails, filtering out 50 threads with 3 to 6 emails each. These emails were manually annotated with contexts, grouped by similar topics, and then summarized. Our methodology involved a three step process: grouping similar emails, identifying their contexts, and summarizing the grouped emails. We used two techniques for context extraction: LDA and Key-phrase extraction with Transformer. We used K-Means and Hierarchical Agglomerative Clustering for grouping, and BART and T5 models for summarization.

We evaluated our techniques using metrics like For Clustering: RAND Score, Silhouette Score, and For Summary and context extraction: Rouge, and Meteor. The results showed that Hierarchical Agglomerative Clustering worked better for grouping, Key-phrase extraction was more effective for context extraction, and BART gave the best summarization results with a precision score of 0.861. The paper is structured as follows: Section 2 covers related work, Section 3 discusses the dataset and methodology, Section 4 presents experimental results, and Section 5 concludes with future directions

2 Literature Review

In existing literature, machine learning and deep learning techniques were applied to solve email thread summarization and involved different evaluation metrics to assess the accuracy of the models.

Email-Sum: Abstractive email thread summarization, focused on abstractive Email Thread Summarization using the EMAILSUM dataset containing annotated short and long summaries. They performed a detailed study to explore different summarization techniques such as abstractive and extractive methods, single document and hierarchical document models, and transfer and semi supervised learning. Both short and long summary creation tasks are conducted by human evaluation. They highlight the main difficulties that current abstractive summarization models have while performing this task, including figuring out who is sending the message and their intentions. Furthermore, they found that widely used automatic evaluation metrics like ROUGE and BERT-Score are weakly correlated with human evaluation on tasks like email thread summarization, indicating whether the Summary is generated in terms of its context [5].

Dialogpt: Large-scale generative pre-training for conversational response generation, focuses on improving online conversations. The authors created a system for analyzing online discussions, like email threads, news comments, forums, and Q&A sites. They used a framework to annotate data, which was then crowdsourced to build four large datasets. The authors compared their model's performance with existing conversation summarization models to establish a strong baseline. They also introduced argument mining, using graphs to organize topics, opinions, and claims while removing irrelevant information. Both automatic and human evaluations showed that their model performed well, sometimes even better than others [6].

This paper focused on generating an effective subject line for emails that accurately reflects the purpose of the message. They crowdsourced a dataset and used an abstractive summarization approach to do this. They explained that this task differs from summarizing a single document or creating news headlines. The authors developed a deep learning system with two parts: an extractor to select essential words for the subject line and an abstractor to generate concise subject lines. They compared their system with other recent models and found that while other systems only achieved a METEOR score, their model outperformed in other evaluation metrics [7].

Keyword extraction for conversation summarization tasks, introducing the Forum-Sum dataset, which comprises diverse, high quality conversation summaries from various online forums. They detailed the dataset creation process to encourage its use and expansion by other researchers. Experiments demonstrated that models trained on Forum-Sum exhibit superior zero-shot and few-shot transferability compared to datasets like SAMSum. Additionally, they found that pretraining on conversational corpora improves the performance of chat summarization models. While automatic evaluation showed strong results on ForumSum, human evaluation yielded consistent and effective outcomes [8].

A survey based study on dialogue summarization to create concise and informative summaries of dialogues by covering all salient information. They presented a comprehensive field review, systematically categorizing current works across domains such as email, online forums, meetings, customer service, medical dialogues, chats, TV shows, and interviews. The study also provided an overview of publicly available datasets, including Enron, W3C, Email-Sum, and Avocado, and introduced two leader-boards under combined metrics. The authors highlighted future research directions, such as ensuring faithfulness in dialogue summariza-

tion, incorporating multi-modal and multilingual approaches, and expanding to multiple domains. As the first extensive survey in this field, the work serves as a resource to guide researchers, offering a broad understanding and motivating further developments in dialogue summarization [9].

ConvoSumm, a conversation summarization benchmark that generates To-Do items from emails where the sender has committed to acting. This work is the first to tackle the problem of creating To-Do items from email conversations. They employed a commitment classifier to detect action intents within emails and used a two-stage framework: one for computing relevance scores and another for generating the To- Do items. The model was trained using 500k raw sentences from the Avocado email dataset on a seq-to-seq model. The evaluation metrics, BLEU and ROUGE, yielded scores of 0.23 and 0.63, respectively. Future work includes improving the architecture to utilize structured metadata better and replacing the two stage framework with a multitask generation model to identify relevant context more effectively [10].

This paper addresses the challenge of email overload by utilizing advanced natural language processing (NLP) techniques to generate concise summaries of lengthy email threads. It evaluates the performance of three large language models—GPT-3, Google's Bard, and LLaMA2—by comparing their generated summaries with human written ones using metrics like ROUGE and BERTScore. Results indicate that GPT-3 produces summaries most similar to those of humans, followed by Bard and LLaMA2. [11]

The application of ChatGPT for email summarization aims to address the challenge of managing large quantities of email. By using ChatGPT's conversational abilities and finetuning it on a large email dataset, the research demonstrates that ChatGPT produces clear, concise summaries that effectively capture key information. User studies indicate high satisfaction and suggest that ChatGPT-based summarization is a promising solution for reducing email overload and improving productivity. [12]

This paper proposes a medical text summarization framework based on LongFormer to address the challenges of processing lengthy medical documents. Using long-range self- attention, the model effectively captures key content and outperforms traditional models such as RNN, BERT, and T5 in both automatic metrics and expert evaluations, particularly in information retention and grammatical accuracy. However, the study notes issues of redundancy that affect conciseness, and future work aims to optimize the structure and fluency of the model to achieve a more efficient summary of medical text. [13]

Existing research on email thread summarization has apparent gaps, as referred to in Table 1, as most models focus on extracting meaningful sentences but fail to create coherent summaries that fit the context of the email thread. Key details like action items, decisions, and responsibilities are often overlooked, making the summaries less useful. To address this, we have developed a method that generates contextually meaningful and comprehensive summaries, effectively capturing crucial elements such as action items, decisions made, and assigned responsibilities.

3 Methodology

Our targeted task was to generate an email thread summary without losing contextual information. To achieve our goal, we followed some steps, including data creation, annotation, pre-processing, following our defined three-step approach, and then evaluation. The methodology we used is as follows, in Figure 1, along with an explanation of each step in detail, and covering all the information regarding models and evaluation techniques, which will show how an email thread was processed using our proposed approach.

3.1 Dataset Construction and Annotation

Figure 2 illustrates that the initial step of the research involved creating a dataset focused on university related emails. A collection of 200 email threads was gathered, from which irrelevant emails were removed, leaving 50 threads rich in contextual information about university events. Each email thread contained between three and six emails, and all were 'replying' threads. These 50 threads were stored as '.txt' files and uploaded to Google Drive alongside a read-me file containing instructions and descriptions for easy access and use of the dataset.

After collecting the dataset, the next step was data annotation, which is crucial for training machine learning models. The annotation process involved three main steps: identifying similar sentences within email threads, determining the context for each group of sentences, and manually crafting a summary for each contextual group. These annotations were stored in JSON format for easy accessibility.

3.2 Data Preprocessing

Figure 3 shows that after data annotation, we focused on data pre-processing, which is essential for text-related tasks. The process began by splitting email threads into individual emails. Each email was then

TABLE 1: Summary of research papers focusing on various summarizations techniques and datasets

Email Subject Line Gen. (Mukherjee, 2020)	**2020**	**Subject line creation (Email)**	**Crowdsourced**	**Extractor + Abstractor**	**METEOR + others**
Email-Sum (Zhang et al., 2021)	2021	Abstractive email summarization	EMAILSUM	Abstractive extractive, hierarchical, semi-supervised	ROUGE, BERTScore, Human
ConvoSumm (Fabbri et al., 2021)	2021	To-Do extraction from emails	Avocado (500k)	Commitment classifier + 2-stage seq2seq	BLEU, ROUGE
Forum-Sum (Khan et al., 2022)	2022	Forum summarization	Forum-Sum	Conversational pre-training	Auto + Human
ChatGPT for Emails (Bhuvaneswari et al., 2023)	2023	Email summarization	Gmail API	Fine-tuned ChatGPT	User studies
Khetarpaul (2025)	2025	LLM-based email summarization	2 Crowdsourced datasets + InboxEmail	GPT-3, Bard, LLaMA2	ROUGE, BERTScore
LongFormer (Medical) (Sun et al., 2025)	2025	Medical summarization	Medical texts	LongFormer vs. RNN, BERT, T5	ROUGE, Expert
Our Approach (2025)	2025	Contextual summarization of email threads	Custom dataset (university events, 2–6 thread length)	Clustering (K-Means, AHC) + LDA + Key-Phrase Extraction + BART/T5	ROUGE, BERTScore

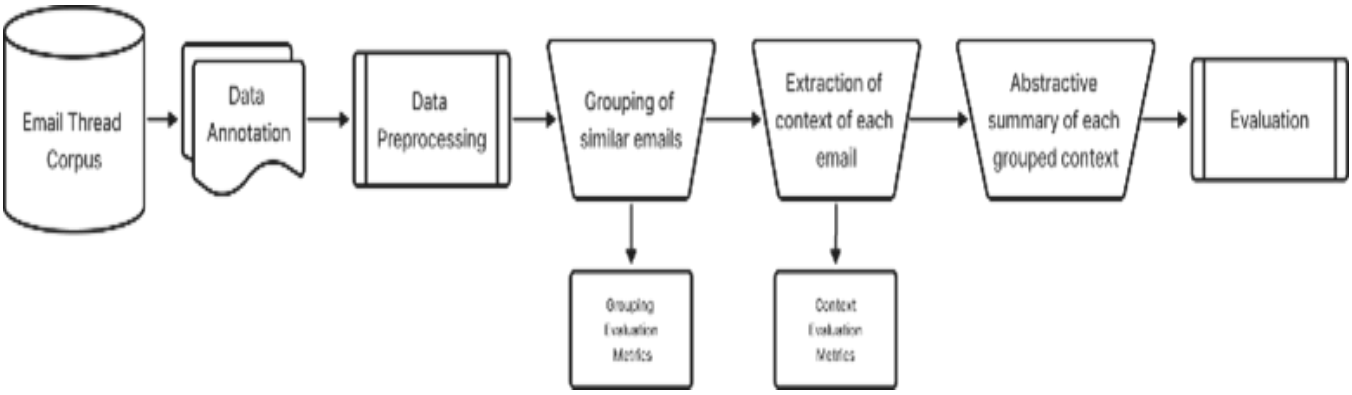

Fig. 1: Workflow used to preprocess emails, group sentences, and generate Contextual Summaries of Email Threads

divided into sub-components, including 'to', 'from', 'cc', 'bcc', 'date', 'subject', and the 'actual body' of the email. We applied several pre-processing steps: removing salutations and closing remarks, eliminating all punctuation except full stops, and converting text to lowercase (with exceptions for proper nouns). Stop words were removed, and stemming and lemmatization techniques were applied to transform the email body. The results were saved into separate files for lemmatized and stemmed text. Stemming truncates word suffixes to obtain the root form, while lemmatization reduces words to their base or dictionary form.

3.3 Grouping of Similar Sentences in an Email Thread

The pre-processed email bodies of the email thread, we employed grouping techniques to group similar sentences and text messages within each email thread.

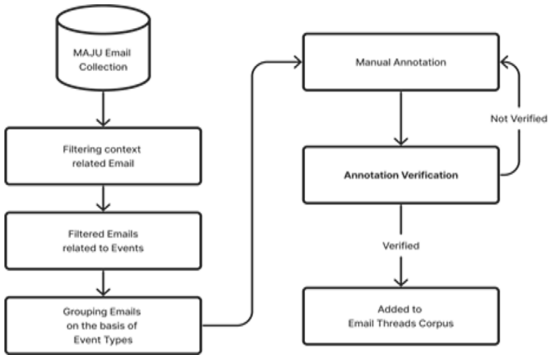

Fig. 2: Process of Dataset Construction and Annotation of University related Email Threads

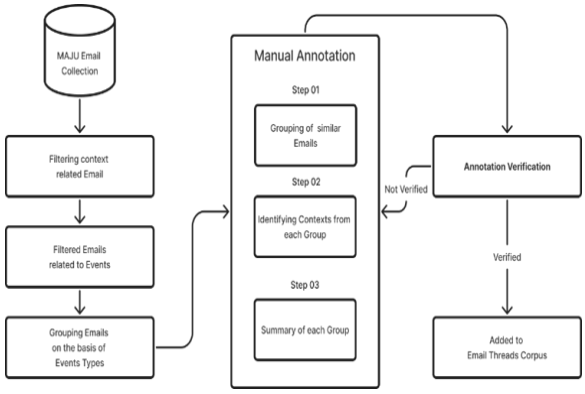

Fig. 3: Preprocessing Steps applied on University related Email Threads

Figure 4 illustrates an example of grouping similar emails. We used K-Means and Hierarchical Agglomerative Clustering for this purpose, resulting in grouped sentences that exhibited similarity. To assess the effectiveness of our grouping, we conducted evaluations using the Silhouette Score and RAND Score.

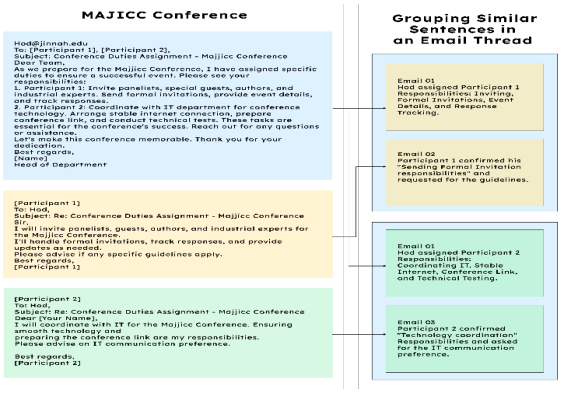

Fig. 4: Example showing how related sentences from different emails within a thread on the MAJICC Conference are grouped together

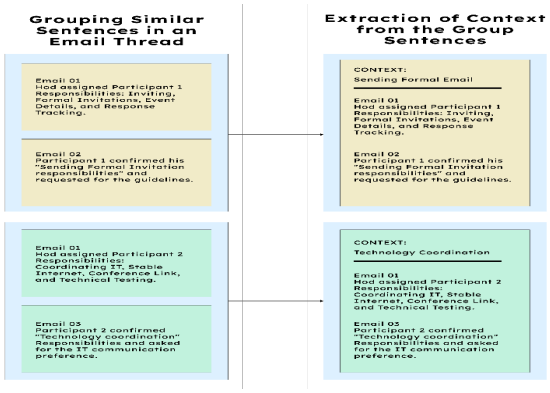

Fig. 5: Example showing extraction of contexts from each group of sentences of the MAJICC Conference Email Thread

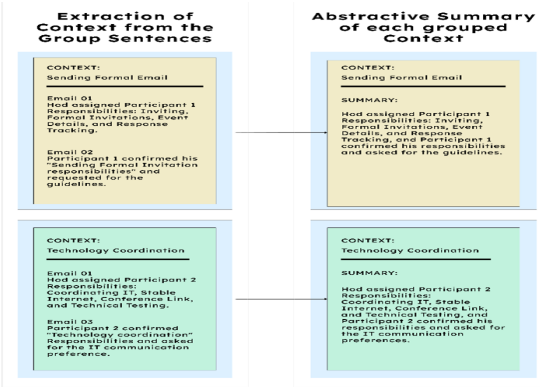

Fig. 6: Example showing Abstractive Summary of MAJICC Conference Email Thread

3.4 Extraction of Context from Each Sentence Group

As shown in Figure 5, the obtained grouped sentences extracted from each email within the email thread, we extracted the context from each group.

To accomplish this task, we employed Latent Dirichlet Allocation (LDA) and Key-phrase extraction methods (KBIR + Openkp) to extract the context from each group. Subsequently, we evaluated these approaches using Rouge and Meteor evaluation metrics.

3.5 Abstractive Summary of Contextual Sentence Groups

The contextually grouped sentences extracted from each email within the email thread, we proceeded to generate summaries for each contextual group, as shown in Figure 6.

We utilized BART (Bart-large-CNN) and T5 (text-to-text transformer) for generating abstractive summaries of these contextual, grouped email messages

TABLE 2: Clustering Results

Clustering Method	RAND Score (Supervised Technique)	Silhouette Score (Unsupervised Technique)
KMEAN Clustering	0.664	0.075
Agglomerative Hierarchical Clustering	**0.798**	0.080

within the email thread. After applying these techniques, we obtained summaries that contained contextual information. Subsequently, we evaluated our results using Rouge Score and Meteor Score.

3.6 Evaluation Metrics

We conducted evaluations for our model in three distinct steps. To assess the grouping of similar sentences within email threads, we employed the Silhouette Score and RAND Score, which are unsupervised and supervised techniques.

For evaluating the extracted contexts and abstractive summaries of contextual group summaries, we used Rouge1 (Precision, recall, and F-measure), Rouge2 (Precision, recall, and F-measure), Rouge3 (Precision, recall, and F-measure), and Meteor (Precision, recall, and F-measure). The outcomes varied across different models and techniques, and a detailed discussion of these results is presented in the subsequent chapter on Results.

4 Results

This research outlines a methodology for contextualization of email threads, achieving the best results through a combination of Agglomerative Hierarchical Clustering, Key-Phrase Extraction, and BART (Bart-large-CNN). Among clustering techniques, Agglomerative Hierarchical Clustering outperformed K-Mean Clustering with a superior RAND Score of 0.798 and a Silhouette Score of 0.080, indicating better sentence grouping, as mentioned in Table 2.

For context extraction, Key-Phrase Extraction demonstrated the best performance. Table 3 shows that, when applied to Agglomerative Clusters, it achieved a ROUGE1 Precision of 0.592, ROUGE-L Precision of 0.429, and a METEOR Precision of 0.361, significantly outperforming Latent Dirichlet Allocation (LDA).

In abstractive summarization, BART delivered the highest accuracy. When summarizing Agglomerative Clusters, it achieved a ROUGE1 Precision of 0.864, ROUGE-L Precision of 0.809, and a METEOR Precision of 0.981, surpassing both T5 and its performance on K-Mean Clusters, as referred to in Table 4.

The combination of Agglomerative Hierarchical Clustering, Key-Phrase Extraction, and BART generated the most effective contextual summaries, achieving superior precision and recall across all evaluation metrics.

5 Conclusion and Future Work

In this work, we propose an approach for generating contextual summarization of email threads on university related emails. The strategy involves the creation of university related email threads, followed by manual annotation of the dataset. After annotation, we formed clusters of similar sentences using K-Means and Agglomerative Hierarchical Clustering. We obtained clusters with shared similarities and extracted contexts from each cluster through Key-Phrase Extraction and LDA. Subsequently, we applied summarization techniques such as BART (Bart-large-CNN) and T5 (Text-to-Text Transformer) to generate contextual summaries for each cluster.

The evaluation results indicate that BART (Bart-large-CNN) outperforms in generating contextual summaries, particularly on clusters generated by Agglomerative Clustering and Key- Phrase extraction for context extraction. While other approaches also demonstrated good performance, BART stood out as a superior choice.

Future work will focus on integrating advanced clustering techniques and exploring hybrid models to enhance summarization quality. Approaches such as LongFormer can be explored for handling long text sequences, as its long-range self-attention mechanism has shown strong performance in summarization tasks. Incorporating generative techniques like GANs or reinforcement learning may further enhance summary quality. [13].

TABLE 3: Evaluation of context extraction techniques

Evaluation Matrices	Measure	LDA (Agglomerative)	Key-Phrase (Agglomerative)	LDA (KMEAN)	Key-Phrase (KMEAN)
Rough Score-R1	Precision	0.3074	**0.592**	0.274	0.54
	Recall	0.444	0.230	0.365	0.21
	F-Measure	0.352	0.328	0.298	0.30
Rough Score-R2	Precision	0.56	0.1995	0.058	0.16
	Recall	0.114	0.070	0.142	0.058
	F-Measure	0.075	0.102	0.083	0.084
Rough Score-RL	Precision	0.1777	**0.429**	0.218	0.437
	Recall	0.291	0.160	0.317	0.167
	F-Measure	0.215	0.234	0.246	0.237
Meteor Score	Precision	0.187	**0.3610**	0.138	0.285
	Recall	0.187	0.3610	0.138	0.285
	F-Measure	0.187	0.3610	0.138	0.285

TABLE 4: Comparison of different evaluation matrices between T5 and BART

Evaluation Metric	Measure	T5	BART
ROUGE-1	Precision	0.7962	**0.8641**
	Recall	0.615	0.743
	F-Measure	0.6644	0.7762
ROUGE-2	Precision	0.5997	0.7331
	Recall	0.5352	0.6901
	F-Measure	0.5514	0.7009
ROUGE-L	Precision	0.7148	**0.8098**
	Recall	0.5819	0.7212
	F-Measure	0.6174	0.7449
METEOR	Precision	0.981	**0.981**
	Recall	0.981	0.981
	F-Measure	0.981	0.981

References

[1] M. Kirmani, G. Kaur, and M. Mohd, "Shortmail: An email summarizer system," Software Impacts, vol. 17, 2023.

[2] A. Mahajani, V. Pandya, I. Maria, and D. Sharma, "A comprehensive survey on extractive and abstractive techniques for text summarization," pp. 339–351, Mar. 2019.

[3] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," 2021.

[4] P. N. Bennett and J. G. Carbonell, "Detecting action items in email," 2005.

[5] S. Zhang, A. Celikyilmaz, J. Gao, and M. Bansal, "EmailSum: Abstractive email thread summarization," 2021.

[6] Y. Zhang, S. Sun, M. Galley, Y.-C. Chen, C. Brockett, X. Gao, J. Gao, J. Liu, and B. Dolan, "DialoGPT: Large-scale generative pre-training for conversational response generation," 2019.

[7] S. Mukherjee, S. Mukherjee, M. Hasegawa, A. H. Awadallah, and R. White, "Smart To-Do: Automatic generation of to-do items from emails," 2020.

[8] O. A. Khan, S. Wasi, M. S. Siddiqui, and A. Karim, "Keyword extraction for medium sized documents using corpus-based contextual semantic smoothing," Complexity, vol. 2022, 2022.

[9] R. Zhang and J. Tetreault, "This email could save your life: Introducing the task of email subject line generation," 2019.

[10] A. R. Fabbri et al., "ConvoSumm: Conversation summarization benchmark and improved abstractive summarization with argument mining," 2021.

[11] S. Khetarpaul, D. Sharma, A. Barakoti, A. Sharma, and A. Jain, "Deciphering email threads: A comparative study of large language models," in Proc. Int. Conf. Innovation in Computing and Engineering (ICE), IEEE, 2025.

[12] R. Bhuvaneswari and T. P. R., "Exploring ChatGPT for email content compression and summarization," in Proc. Int. Conf. Communication, Computing and Industry 6.0 (C2I), Bangalore, India, IEEE, Dec. 2023.

[13] D. Sun et al., "A Longformer-based framework for accurate and efficient medical text summarization," 2025.