TEAM UFAL @ CreativeSumm 2022: BART and SamSum based few-shot approach for creative Summarization

Rishu Kumar and **Rudolf Rosa**
ÚFAL, Charles University
lastname@ufal.mff.cuni.cz

Abstract

This system description paper details TEAM UFAL's approach for the SummScreen, TVMegasite subtask of the CreativeSumm shared task. The subtask deals with creating summaries for dialogues from TV Soap operas. We utilized BART based pre-trained model fine-tuned on SamSum dialouge summarization dataset. Few examples from AutoMin dataset and the dataset provided by the organizers were also inserted into the data as a few-shot learning objective. The additional data was manually broken into chunks based on different boundaries in summary and the dialogue file. For inference we choose a similar strategy as the top-performing team at AutoMin 2021, where the data is split into chunks, either on [SCENE_CHANGE] or exceeding a predefined token length, to accommodate the maximum token possible in the pre-trained model for one example. We implemented two different strategies as splits on [SCENE_CHANGE] did not necessarily mean having less than 1024 tokens in a segment.

1 Introduction

Creative Summarization as a field is rather novel, which neatly exists between document summarization and Conversation Summarization. The task of summarization focuses on extracting relevant information from the entire document where the task of minuting includes an additional objective of getting rid of redundancies in the dialogues as well as extracting relevant information based on different boundaries within the text i.e. topic switching. It is written, proof-read and mostly contains of coherent and grammatically correct sentences, and since it is also supposed to mimic how people speak, it also contains grammatically incorrect sentences as well as people speaking over each other. Thus, this track of research has a unique opportunity to leverage the recent advances in document summarization and Automatic Minuting. Unlike the dataset used for Automated Minuting, the dataset in this subtask carries a special property where the conversation changes are marked with special tokens such as [SCENE_CHANGE] with on average a transcript containing 20±14, Scene breaks across training, test and dev dataset splits. Since the dataset consists of transcripts from different shows, which presumably are written by different screenplay writers resulting in different writing style, which explains the very high standard deviation on how often SCENE_CHANGE is there in a transcript. While this approach makes it easier to split a transcript into multiple parts, it does not guarantee that the segments will not exceed the tokens limits of the pre-trained model, 1024 in our case.

The task for summarizing a TV show episode introduces a unique challenge compared to the task of summarizing a conversation. The transcripts for TV shows not only contain the dialogues, it also contains visual cue descriptions which are absent from the Minuting Summarization task as shown in . However, on a broad sense these tasks share some similarities as a summary can constructed from the perspective of one character in the show can be polar opposite to the summary generated from the perspective of a different character. Previous approaches to the multi-party summarization includes modeling intra-speaker and inter-speaker topics with random walk in a graph (Chen and Metze, 2012), leveraged word-embeddings (Shang et al., 2018) by using WordNet (Miller, 1995) to make summary more abstractive. The word-embedding based approach was further incorporated by (Zhao et al., 2019), and (Li et al., 2019) with different model architecture with (Li et al., 2019) incorporating information from visual modality into their summarization model.

Our approach in this subtask builds upon the ideas introduced in these papers, in training and inferences. We incorporate a pipeline with three components, Model, Data Cleaning and Inference

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 24–28

https://doi.org/10.26615/978-954-452-056-4_004

Figure 1: A snippet from the training dataset of SummScreen ForeverDreaming split

```
( ENTITY75 runs and climb@@ s up a sta@@ ir@@ well outside . Inside , a few doctors g
ather around Pat@@ el . )
DOCTOR : S@@ cal@@ p@@ el .
( Pat@@ el lays , unconscious . ENTITY75 breaks a cage outside and climb@@ s inside .
 At the operation , a doctor makes an inc@@ ision on Pat@@ el 's chest . Someone else
 wi@@ pes the blood away . ENTITY75 , climb@@ ing above where the operation is taking
 place , moves through an entr@@ y@@ way and gasps as her legs dang@@ le below throug
h a giant hole in the wal@@ k@@ way . She pulls herself up , grun@@ ting . On her han
ds and knees , she looks down . Coming up to another hole in the ceiling above where
the doctors are cur@@ r@@ ently cutting Pat@@ el open , she gets a better look with a
 small tele@@ scope that is about the size of a pen@@ cil . She see@@ es ENTITY69 bel
ow , ob@@ serving the operation . And ENTITY39 . She gets a better look . She sees Pa
t@@ el 's face . They 're making the inc@@ ision deeper . One of the doctors hol@@ d@
@ ds a small metal ca@@ sing that is for@@ med into a half moon . )
ENTITY69 : Uh , careful with that . That 's the equi@@ val@@ ent to three hundred pou
nds of T@@ N@@ T .
ENTITY39 : Yes . Do n't kill him .
( ENTITY75 watches in horror as the doctors put the bomb inside Pat@@ el 's chest . )
```

Table 1: A brief description of the SummScreen ForeverDreaming

		Transcript		Summaries	
Data Split	#Examples	Avg. word_count	Std. Dev	Avg. word_count	Std. Dev
train	18915	6360	±1612	380	±237
dev	1795	6336	±1591	380	±234
test	1793	6348	±1599	382	±247

as discussed in following sections.

2 Dataset

CreatievSumm shared task recommends using SummScreen (Chen et al., 2022) for training, evaluation and testing purposes. It is divided into two parts based on the source of collection, i.e. The TV MegaSite (TMS) and ForeverDreaming(FD). For our system, we chose to work with ForeverDreaming part of the SummScreen dataset. This split in turn is released into two forms, one is anonymized where character names are replaced by "ENTITY", and another is the normal transcript with character names present. For our submission, we chose to work with anonymized version of the dataset, which was in-line with AutoMin (Ghosal et al., 2021) dataset.

Figure 1 provides a glimpse of how data looked in its original phase. We first began by sanitizing the data and removing information which was not relevant to us. Our sanitization process included removing @@, and using MosesTokenizer (Koehn et al., 2007) to fix the tokenization in the dataset. We also implemented various regex expansion to convert *I'm* → *I am*, *..shouldn't* → *..should not*. After a qualitative overview of the dataset in all splits, we implemented more rule-based text processing such as removing the additional information which did not include a character's action. Among such rules, one was to remove the line if it did not start with "ENTITY" after cleaning all the punctuations and extra space. We also change dialogues such as "ENTITY1 Laughs" to "ENTITY1 : Laughs" to reduce the number of lines removed from the dataset for not being a conversational utterance.

We added 16 examples from the dataset with high number of [SCENE_BREAK] and sentence count in summaries. The mixture of dataset is visualized in Figure 3. We manually split the summaries based on the relevant splits of the transcript. We also implement similar measures for the AutoMin dataset, where we also sampled eight transcripts and their corresponding summaries. The relevant statistics for the SummScreen ForeverDreaming dataset is included in Table 1.

3 Result

Table 2 and 3 presents the official results on the test set as calculated by the organizers. Our submission performs on par with other teams and achieves a higher average number of words per summary while getting similar automated evaluation scores. It is also worth mentioning that since our model was trained on ananomized data from AutoMin

Figure 2: The training regime we used to introduce different conversational style from the training dataset and AutoMin dataset to the SamSum datset

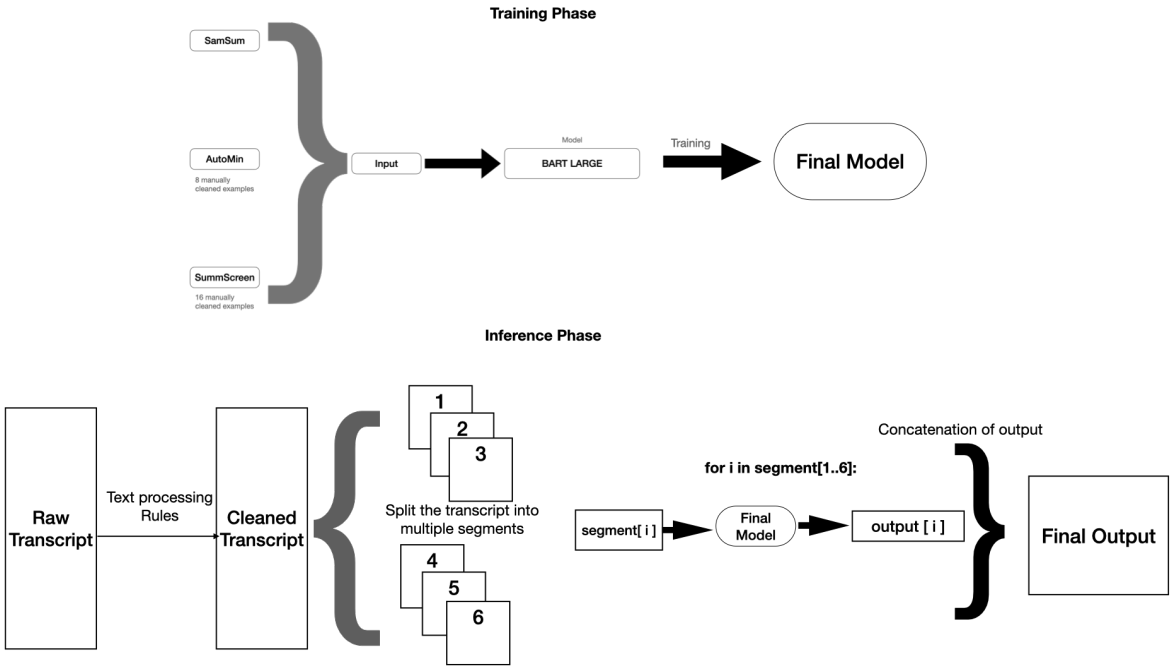

Table 2: Official Results – part I

	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore-P	BERTScore-R	BERTScore-F1	LitePyramid-p2c
LED_1024	0.1428	0.0154	0.1236	0.4100	0.4107	0.4052	0.1371
LED_4096	0.1694	0.0209	0.1501	0.4591	0.4752	0.4600	0.0337
LED_16384	0.1514	0.0170	0.1334	0.4485	0.4632	0.4489	0.0337
inotum_summscreen-fd.jsonl	0.2860	0.0624	0.2529	0.5934	0.5609	0.5750	0.0673
team_ufal_fd.json	0.2469	0.0408	0.2300	0.5038	0.5590	0.5285	0.0472
AMRTVSumm_summscreen-fd.jsonl	0.2307	0.0303	0.2106	0.4906	0.5344	0.5108	0.0116

shared task and the training files for this shared task, our output suffers from cases where the name resolution has not been perfect.

4 Methodology

For our experiment, we use pre-trained BART (Lewis et al., 2019) from Facebook research, released as a pre_trained model on XSum (Narayan et al., 2018) on HuggingFace (Wolf et al., 2019). We further fine-tuned the model on SamSum dataset. The exact implementation for our models is released publicly on GitHub[1]. We are also releasing the performance of different pre-trained models, such as T5 for zero-shot and few-shot learning on SummScreen ForeverDreaming dataset For inference, we use the same strategy as (Shinde et al., 2021)'s model for AutoMin2021. We split the transcript into multiple chunks of dialogues. This split is done either on [SCENE_BREAK] occurrence or when the token count exceeds a pre-defined limit. This helps us in extracting all the relevant information from the data without losing any information in truncation.

BART is a denoising autoencoder used to pre-trained seq-to-seq models for Natural Language Generation among other tasks. During the training of this model, a random denoising function is used to corrupt the text and then it learns how to recover the original text. It is also capable of operating bi-directionally on sequence generation unlike BERT (Devlin et al., 2018). We primarily focused on this training strategy because of its proven results on AutoMin shared task.

During inference as depicted in Figure 2, we split the conversation into chunks, and concatenate the output to construct final output.

5 Conclusion

In this system description paper, we explain the training regime we used to participate in the CreativeSumm shared task using SummScreen Forever-

[1]https://github.com/pyRis/creative_summ_subm

Figure 3: The training regime we used to introduce different conversational style from the training dataset and AutoMin dataset to the SamSum datset

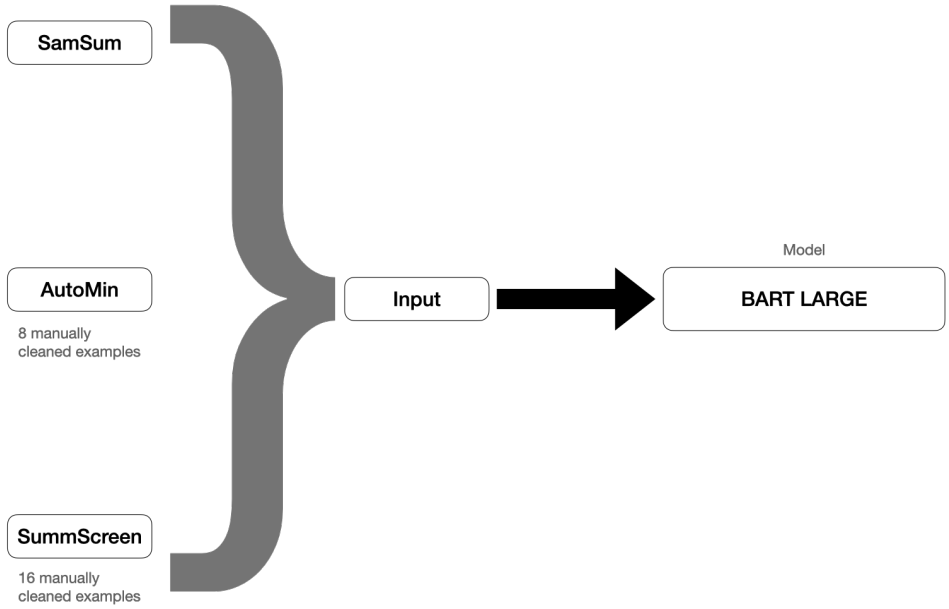

Table 3: Official results – part II

	LitePyramid-p2c	LitePyramid-l2c	LitePyramid-p3c	LitePyramid-l3c	SummaCZS	Length	Density	Coverage	Novel 1-grams	Novel 2-grams
LED_1024	0.1371	0.1200	0.0987	0.0878	0.0559	330	1.1440	0.7148	0.3060	0.7801
LED_4096	0.0337	0.0069	0.0304	0.0049	0.1052	188	1.4378	0.7343	0.2803	0.7314
LED_16384	0.0337	0.0069	0.0304	0.0049	0.1644	192	1.5474	0.7108	0.2904	0.7285
inotum_summscreen-fd.jsonl	0.0673	0.0560	0.0559	0.0534	0.0272	86	1.0321	0.6664	0.3715	0.8251
team_ufal_fd.json	0.0472	0.0229	0.0406	0.0191	0.1282	289	2.0821	0.7127	0.2484	0.6498
AMRTVSumm_summscreen-fd.jsonl	0.0116	0.0008	0.0138	0.0007	0.024	256	0.8789	0.6137	0.4924	0.8569

Dreaming dataset. Our system incorporates mixing of training samples from novel datasets into an existing dataset, which resembles a few-shot approach. In Section 6, we propose the direction we would like to take for incorporating the shared task training dataset into training.

6 Future Work

While we observe that zero-shot / few-shot learning creates coherent looking outputs, the problem to train on the complete data-set still remains open. Attributed to the lack of segment-wise summary data for longer transcripts, splitting text into segments for training does not work without loosing information or wrong alignment of text with corresponding line with summary. We plan to release our experiments on GitHub repository highlighting the issue, and the performance of the model on such training regime. This direction of research into a semi-supervised splitting of transcript and summary can help with the current problem of exceeding maximum length for tokenization.

Acknowledgements

Rishu Kumar has received financial support from the EMLCT programme[2] as part of his graduate study.

References

Mingda Chen, Zewei Chu, Sam Wiseman, and Kevin Gimpel. 2022. Summscreen: A dataset for abstractive screenplay summarization. In *ACL*.

Yun-Nung (Vivian) 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 *INTERSPEECH*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Tirthankar Ghosal, Ondřej Bojar, Muskaan Singh, and Anja Nedoluzhko. 2021. Overview of the First

[2]https://lct-master.org

Shared Task on Automatic Minuting (AutoMin) at Interspeech 2021. In *Proc. First Shared Task on Automatic Minuting at Interspeech 2021*, pages 1–25.

Philipp Koehn, Hieu T. Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

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.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium.

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.

Kartik Shinde, Nidhir Bhavsar, Aakash Bhatnagar, and Tirthankar Ghosal. 2021. Team abc@ automin 2021: Generating readable minutes with a bart-based automatic minuting approach. *Proceedings of the First Shared Task on Automatic Minuting at Interspeech*, 2021:1–8.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2019. Huggingface's transformers: State-of-the-art natural language processing.

Zhou Zhao, Haojie Pan, Changjie Fan, Yan Liu, Linlin Li, Min Yang, and Deng Cai. 2019. Abstractive meeting summarization via hierarchical adaptive segmental network learning. In *The World Wide Web Conference*, WWW '19, page 3455–3461, New York, NY, USA. Association for Computing Machinery.

A Example

This is an example of our model's output on the testset:
instance: The_Simpsons_388
summary: PERSON0, Bart and Lisa are making pancakes for Mom's birthday. They also got her a bottle of French perfume from gay Paree for her birthday. It's a surprise for Dad. It's Homer Simpson's 34th birthday. He's having dinner with his friends at the Singing Sirloin tonight. Marge's mother has not opened her birthday present yet. PERSON1 got a bowling ball as a present from PERSON3. She's not very good at it, so she's going to use it. PERSON4 gives her a paper with a score on it and a pair of shoes. Marge writes down the score on a PERSON1, Marge, Homer, Lisa, Bart and Maggie are learning how to play the game. It costs forty dollars for the lessons. PERSON1 and Marge are going to meet for brunch tomorrow. Marge is going bowling again tonight. PERSON10 is giving her a bowling lesson. Marge is in a restaurant. She is dancing with Jacques. PERSON13 and PERSON13 are angry at their father for not listening to their advice. Bart and Lisa are afraid something is wrong with their father. Marge made a peanut butter and jelly sandwich for PERSON3. He eats it. He is going to the backseat of
Mapping: "LISA": "PERSON0",
"MARGE": "PERSON1",
"WAITERS": "PERSON2",
"HOMER": "PERSON3",
"MANAGER": "PERSON4",
"LENNY": "PERSON5",
"SELMA": "PERSON6",
"HELEN": "PERSON7",
"VOICE": "PERSON8",
"COWORKER": "PERSON9",
"JACQUES": "PERSON10",
"PATTY": "PERSON11",
"MAN": "PERSON12",
"BART": "PERSON13",
"LISA + BART": "PERSON14",
"COP": "PERSON15"