Automatic Summarization for Creative Writing: Denoising Auto-Encoder based Pipeline Method for Generating Summary of Movie Scripts

Aditya Upadhyay[&], **Nidhir Bhavsar**[$],**Aakash Bhatnagar**[1] **Muskaan Singh**[#] and **Petr Motlicek**[#]
[&] CSED, Thapar Institute of Engineering and Technology, India
[1] Boston University, Boston, Massachusetts
[$] University of Potsdam, Potsdam, Germany
[#] Speech and Audio Processing Group, IDIAP Research Institute, Martigny, Switzerland
adityaupadhyay1912@gmail.com, aakash07@bu.edu,
bhavsar@uni-potsdam.de, (msingh,petr.motlicek)@idiap.ch

Abstract

This paper documents our approach for the Creative-Summ 2022 shared task for Automatic Summarization of Creative Writing. For this purpose, we develop an automatic summarization pipeline where we leverage a denoising autoencoder for pretraining sequence-to-sequence models and fine-tune it on a large-scale abstractive screenplay summarization dataset to summarize TV transcripts from primetime shows. Our pipeline divides the input transcript into smaller conversational blocks, removes redundant text, summarises the conversational blocks, obtains the block-wise summaries, cleans, structures, and then integrates the summaries to create the meeting minutes. Our proposed system achieves first position with some of the best scores across multiple metrics (lexical, semantical) in the Creative-Summ shared task. We publicly release our proposed system here[1]

1 Introduction

Text summarization captures salient information by condensing long documents into short paragraphs. With the surge of online records, automatic text summarization makes it convenient for people to extract relevant information to their interests. There are several challenges with the current state-of-art summarization methods (i) processing long sequences spanning hundreds of pages of text, (ii) analyzing complex discourse structures such as narrative and multi-party dialogs (iii) interpreting figurative languages to understand and convey the salient points in the input. Most current works focus on the news, text, and scientific domain with limited input length, literal and technical language, hallucinations, positional biases, and constrained discourse structure. One under-explored text summarization domain is creative writing, which includes documents such as books, stories, scripts from plays, TV shows, and movies. The task of automatically summarizing the creative content is not straight forward. It involves, long length document, non-trival temporal dependency (parallel plot threads and non-linear plot development), complex structures with frequent context drifts combining narative creations and multi-party dialogs with variety of styles. In this paper we summarize these creative content of movie scripts with literary interpretations, conveying implicit information and heavily paraphrasing the input using state-of-art text summarization models.

[1]https://github.com/aditya-u/Script-Summarization

2 Related Work

Text summarization has been a topic of research since the mid 20th century. Most earlier methods relied on statistical analysis to score the importance of sentences and then extracting the sentences with the most importance. Christian et al. (2016) proposed using TF-IDF (Term Frequency-Inverse Document Frequency) to score sentences. They found TF-IDF to be effective at extractive summarization and found it to outperform other state-of-art system available. Nomoto (2005) proposed bayesian modeling as an approach to summarize text data. Kamal proposed using extractive key concept summarization and found it to match the performance and even outperform some of the existing models at the time. Qiang et al. (2016) proposed a novel pattern based summarization technique and found it to perform much better than standard text based methods.

Following statistical methods, deep learning methods were utilized to attempt to find a solution to the problem of text summarization. Rush et al.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 44–50
July 5–10, 2020. ©2020 Association for Computational Linguistics
https://doi.org/10.26615/978-954-452-056-4_007

(2015) proposed a novel sequence-to-sequence (seq2seq) model to perform abstractive summarization rather than extractive summarization. Tan et al. (2017) proposed a novel graph based attention mechanism in a hierarchical encoder-decoder framework, and propose a hierarchical beam search algorithm to generate multi-sentence summary. This architecture provided significant improvements over previously existing models. Jiang et al. (2018) introduced a feature enhanced seq2seq model. This model improved the encode and decoder performance using a 2-feature capture network which improves the models capability of storing long term features, this makes the generated summaries far more concise and accurate.

Following the introduction of the transformer model by Vaswani et al. (2017) most work in NLP shifted to models based on the transformer architecture. Zhang et al. (2019) proposed the PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization) model which demonstrated the effects of the pre-training corpora, gap-sentences ratios, vocabulary sizes and scaled up the best configuration to achieve state-of-the-art results on 12 diverse downstream datasets considered. Radford and Narasimhan (2018) introduced GPT unidirectional transformer encoder model, which improved the score of downstream NLP tasks by combining pre-training and fine tuning. Then, based on the two pre-training models, there are many fusion algorithm models to deal with the NLP task of automatic summarization. Song et al. (2019) introduced the novel MASS model, which allows the encoder and decoder to learn at the same time in the pre-training stage. It is the first time to realize the unification of the BERT plus generation model, and the rouge score is improved compared with the BERT and other models.

3 Proposed Methodology

As shown in figure 1, our suggested system is divided into three main modules, which include pre-processing input transcripts, a conventional sequence-to-sequence model for generating summaries, and post-processing, which further unifies summaries and eliminates redundant information.

3.1 Pre-Processing

The current summarization models lacks the ability to automatically ignore redundancies from a protracted dialogues. Additionally, for the models to produce accurate and high-quality text, the input must not exceed a token-length limit of $\{512, 1024\}$ tokens. This is why these models have trouble comprehending a longer string of multi-speaker utterances and the jumbled information that goes along with it. We employ an initial text processing procedure for utterance cleaning and redundancy elimination using some pre-engineered rules.

As described under the *pre-processing* section of figure 1, a raw transcript with Speaker-Utterance pairs of $X = \{(p_1, U_1), (p_2, U_2), ..., (p_L, U_L)\}$, where $p_i \subset P, 1 \geq i \geq L$ represents a participant and $U_i = \{w_1^i, w_2^i, ..., w_m^i\}$ denotes the i^{th} utterance in a tokenized format. As said earlier, we generate a cleaned sequence, $U_i^c = \{W_1^i, W_2^i, ..., W_m^i\}$, we do this by removing unnecessary words, non UTF-8 decoded words, narration statements. To make the input utterance more redundant and reliable to be processed by the summarization model, we develope a repository of stopwords $S = \{s_1, s_2, ..., s_l\}$, which is an extension of the nltk-stopwords[2] list. We develop this by appending carefully curated vocalsound and non-regular word expression like hmm, uhm, hellooo, byeee etc. Following this, we obtain a compressed utterance, $U' = U^c \cap S$

Next, Here, we use a brute-force method by breaking up the transcripts into blocks of segments with a preset token length. We do this to overcome the length restrictions of the current sequence-to-sequence summarization models. Our purpose is to retain the quality of the generated minutes while including all pertinent information in the transcripts, since we did want to be bound by the length constraint posed by the summarization model. We adopt of fluctuating set of threshold for each input-segement which might vary from $\{512, 768, 1024\}$.

3.2 Summarization

We use a finetuned BART-large model (Lewis et al., 2019) [3] for our primary summarization task. BART is a denoising autoencoder for pretraining sequence-to-sequence models. The model is trained by using arbitrary denoising functions to distort text and then instructing it to recreate the original content. Using BART provides the ability to use bi-directional attributes when operating on

[2]https://www.nltk.org/

[3]https://huggingface.co/lidiya/bart-large-xsum-samsum

sequence generation tasks which makes it useful for abstractive text summarization. While BERT cannot adopt a bidirectional mechanism for sequence generation, BART exploits the GPT-2 architecture for predicting the following words with the help of words encountered previously in the current sequence. Hence, we primarily test the pipeline with various BART-based setups. However, we majorly experiment with a fine-tuned version of BART initially trained on XSum (Narayan et al., 2018) & SAMSum (Gliwa et al., 2019) datasets. Finally, we extend the functionality of the model in sync with the proposed task by further finetuning the same using the publicly available SumScreen (Chen et al., 2021) abstractive screenplay summarization dataset.

The input sequence obtained from previous step is passed to the summarization module. For the k^{th} segment, constituting of role-utterance pairs $X^k = \{(p_1^k, U_1^k), (p_2^k, U_2^k), ..., (p_{L_k}^k, U_{L_k}^k)\}$, the model generates a summary $C^k = \{c_1^k, c_2^k, ..., c_{L'_k}^k\}$ where c_i^k is the i^{th} summary line of the k-th segment. Later we join all the generated summary segments to get raw aggregated summary text $Y^S = (C^1, C^2, ...C^K)$.

3.3 Post-Processing

The generated summaries contain a sufficient amount of information, although they are not entirely adequate. There might be an inclusion of casual discussion or other unnecessary information. This problem is addressed with TextRank. Based on our experimentations, we found out that from the whole report, the model typically catches 15% of trivial and unnecessary information. We rank the summary lines in increasing order of their importance and exclude out bottom 15% of the lines to obtain a "gold span" of the summary. To further compress the summaries, we add appropriate pronouns, eliminate grammatical inconsistencies wherever possible, and filter the final chain of conversation threads by excluding unnecessary words using stopwords set that we internally develop by observing the generated summaries.

4 Experimental setup

In this section, we describe dataset details in subsection 4.1, hyper-parameter setting in section 4.2 and evaluation strategy in section 4.3.

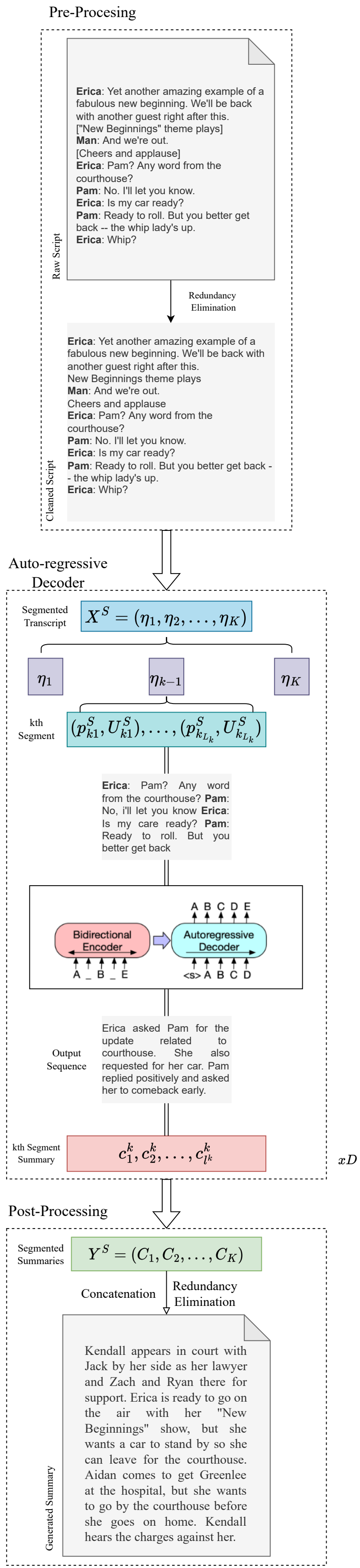

Figure 1: Proposed Architecture for BART

	value
number of shows	10
number of episodes	22503
min. episodes per show	168
max. episodes per show	3784
median episodes per show	1973.5
avg. episodes per show	2250.0

Table 1: Shows the general statistics of the SummScreen Dataset.

TV Transcript
Erica: Yet another amazing example of a fabulous new beginning. We'll be back with another guest right after this.
`"New Beginnings" theme plays`
Man: And we're out.
`Cheers and applause`
Erica: Pam? Any word from the courthouse?
Pam: No. I'll let you know.
Erica: Is my car ready?
Pam: Ready to roll. But you better get back – the whip lady's up.
Erica: Whip?
Pam: W-H-I-P? "We Help Women In Prison"? Remember?
Erica: Women in prison, Pam? With my daughter facing what she's facing? Cancel her.
Pam: The guest? She's already in the chair.
Erica: All right, all right, I will do that. But the minute I get word. I'm out of here. Thanks. Hello. Hi, nice to meet you.
`SCENE_BREAK`
Tad: Stop staring, Krystal.
Krystal: I'm not staring.
Tad: Yeah, you are – you're staring. Why don't you watch your daughter drool strained peaches for a while?
Krystal: Oh. Hey. Hey there. Somebody needs to talk to you about your table manners, little one. Now you're staring.
Tad: No, I'm not. It's more of a subtle glance.
Krystal: "Subtle"? Tad: Yeah. Krystal: Subtle, my eye. Come on, do we have to go?

Summary
Kendall appears in court with Jack by her side as her lawyer and Zach and Ryan there for support. Erica is ready to go on the air with her "New Beginnings" show, but she wants a car to stand by so she can leave for the courthouse. Aidan comes to get Greenlee at the hospital, but she wants to go by the courthouse before she goes on home. Kendall hears the charges against her. Kendall pleads guilty to all the charges. Kendall tells the judge everything that had led up to her stealing the chloroform from the hospital, and reporting to the police that Greenlee had stolen her little boy. Richie puts his plan in motion to place all the blame on Annie for stabbing him. Richie cons one of his inmates to pose as a doctor and call Annie to come to the hospital because his time is near. Adam and J.R. have breakfast together. Adam tries to con J.R. into moving back in with him at the tune of fifty million dollars. Krystal and Tad can't seem to keep their eyes off of Adam and J.R. Aidan and Greenlee appear in court. Greenlee asks to make a statement on Kendall's behalf. After Greenlee makes her statement, the judge calls for a recess. The judge agrees to put Kendall on probation for five years, pay fifteen thousand dollars and five hundred hours of community service. Hannah walks in just as the hearing is coming to a close. She hurries out before anyone can see her.

Figure 2: Illustrates an example derived directly from the SummScreen dataset. The first block contains the TV transcript followed by the associated summary.

4.1 Dataset

As stated, our summarization module utilizes a BART model that is initially fine-tuned on both XSum and SAMSum datasets. XSum dataset includes short summaries of articles and discussions, whereas SAMSum is a multi-party meeting conversation dataset usually comprising casual and friendly conversations. Training model on these two datasets allows it to grasp information both at the syntactic and morphological levels.

Next, to extend the model's summarization adaptability towards automatic summarization of creative writing we further train it on the SummScreen dataset. Table 1 shows the basic statistics of the SummScreen dataset. It comprises of TV transcripts and human-written recaps from primetime shows. The dataset is curated from two distinct data sources, i) TV MegaSite, Inc. (TMS) and ii) ForeverDreaming (FD). While the FD curated data contains more shows (88), spreading across 21 different genres, we train our proposed system using the TMS curated data, which includes data spanning across 4 genres with comparitavely lesser no. of shows. This is due to the fact that the FD curated data lacks original human-written recaps.

It contains 22503 transcripts taken from from TV series and their corresponding recap. The transcripts consist of dialogue utterances with speaker names, and descriptions of scenes or character actions. The recaps are human-written summaries of the corresponding transcripts. An example of snippet from the script is shown in figure 2.

4.2 Hyper-parameter Settings

For both training and inference we utilised the NVIDIA P100 GPU with 16 GB of primary memory and a hyper-threaded Intel Xeon processor with 2 cores operating at 2.3 GHz along with 52 GB of RAM. We train our summarization model for 3 Epochs with a train & evaluation batch-size of 4. During training we initialize the learning rate as 2×10^{-5} and set the max input length = 512, min target length = 128. we implement the AdaFactor optimizer, which internally adjusts the learning rate based on the scale parameter and relative/warmup steps.

4.3 Evaluation

The fine tuned and trained model was used to generate summaries for the test set provided for the shared task. This set was a previously unrevealed subset of the SummScreen test set. It contained scripts of various day time soap opera episodes. The summaries for these were generated and submitted to the shared task. They were compared to summaries of the episodes using various standard evaluation metrics such as stanza library to tokenize the summaries and then the summ_eval library to calculate ROUGE (Lin, 2004) its variants, pretrained metric such as BERTScore (Zhang et al., 2020) [4] , LitePyramid(Zhang and Bansal, 2021)

[4] `microsoft/deberta-xlarge-mnli_L40_no-idf_version=0.3.9(hug_trans=4.20.1)`

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Our Subm.	0.3921	0.0909	0.3794	0.5507	0.5550	0.5516	0.0740	0.0406	0.0625	0.0367	0.1133	316	1.9436	0.7618	0.2026	0.6688
Avg	0.1424	0.0222	0.1335	0.4426	0.4357	0.4354	0.0289	0.0059	0.0266	0.0044	0.0760	752	0.9139	0.6211	0.3943	0.8479

Table 2: It presents various scores obtained by our proposed summarization system on the SummScreen test dataset for the Creative-Summ 2022 automatic summarization for creative writing. It also compares our performance with average submission for the specific task. here (A)R-1 (B)R-2 (C)R-L (D)BERTScore-P (E)BERTScore-R (F)BERTScore-F1 (G)LitePyramid-p2c (H)LitePyramid-l2c (I)LitePyramid-p3c (J)LitePyramid-l3c (K)SummaCZS (L)Length (M)Density (N) Coverage (O)Novel 1-gram (P) Novel 2-grams

uses the NLI model[5], SummaCZS(Laban et al., 2022), zero-shot SummaC model on sentence-level granularity using the vitc NLI model, length of the model summaries, extractive density and coverage as (Grusky et al., 2018) and novel unigrams found in the model summaries w.r.t. the input. We have discussed our proposed system approach evaluation results further in section 5.

5 Results and Analysis

Our summarization model is evaluated on 16 unique metrics. These metrics tries to access model's performance both lexically and semantically against the human annotated summaries. Table **??** shows the evaluation results obtained on the test dataset provided under the CreativeSumm 2022 task. The test data comprises of 679 different scripts/transcripts extracted from various discrete episodes from SummScreen TMS dataset. As depicted in the table our system performs better overall in comparisons to other submission. We outperfrom their second best performing system by a margin of 0.05 across metrics like ROUGE-N, LitePyramid and SummaCZS. Our results are more adequate and fluent when referred against the original human generated annotations.

Figure 3 illustrates an actual example extracted from the SummScreen TMS dataset. It also portrays the outputs generated by passing the TV Transcripts via our proposed summarization pipeline. As it is clearly depicted our proposed system is able to extract and relate small details in the TV transcripts. These generated summaries are also grammatically tuned and adequate.

As described earlier we adopt a segmentation procedure to avoid the max-token-length conflicts posed by the current summarization models. However, a major drawback of this is that there is a loss of information which get ignored when we segement the input data. This can be solved by adopting an appropriate co-reference resolution procedure to seamlessly connect various integrative parts of the text such as noun-pronoun pairs verbs, associating adverbs which sometime due to segmentation gets disentangled. Even though, during pre-processing we try to precisely segment the text by using a floating max_threshold ranging between $\{512, 1024\}$, however sometimes these references get lost in during processing and thus might not be reflected precisely into the output text.

[5]`shiyue/roberta-large-tac08`

6 Conclusion

In this paper, we propose a unique architecture comprising of various segement working together seamlessly to produce good quality summaries of TV transcripts. We utilise a BART model initially fine-tuned on human text conversations and then on scripts specifically derived from the SummScreen TMS dataset. Our system submissions outperformed every other contribution across various evaluation metrics under the CreativeSumm 2022 task. However, the proposed pipeline still need some refinement in terms of the Language model and the inclusion of various pre-processing techniques. This could result in a significant improvement in the model's generation quality, making it more reliable and adequate.

7 Acknowledgements

This work was supported by the European Union's Horizon 2020 research and innovation program under grant agreement No. 833635 (project ROXANNE: Real-time network, text, and speaker analytics for combating organized crime, 2019-2022).

References

Mingda Chen, Zewei Chu, Sam Wiseman, and Kevin Gimpel. 2021. Summscreen: A dataset for abstractive screenplay summarization. *CoRR*, abs/2104.07091.

Hans Christian, Mikhael Agus, and Derwin Suhartono. 2016. Single document automatic text summarization using term frequency-inverse document frequency (tf-idf). *ComTech: Computer, Mathematics and Engineering Applications*, 7:285.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. *CoRR*, abs/1911.12237.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 708–719. Association for Computational Linguistics.

Xiaoping Jiang, Po Hu, Liwei Hou, and Xia Wang. 2018. Improving pointer-generator network with keywords information for chinese abstractive summarization. In *NLPCC*.

Sarkar Kamal. Automatic single document text summarization using key concepts in documents automatic single document text summarization using key concepts in documents. *Journal of Information Processing Systems* , 9(4):602–620.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Trans. Assoc. Comput. Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

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. *CoRR*, abs/1808.08745.

Tadashi Nomoto. 2005. Bayesian learning in text summarization. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 249–256, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Ji-Peng Qiang, Ping Chen, Wei Ding, Fei Xie, and Xindong Wu. 2016. Multi-document summarization using closed patterns. *Knowledge-Based Systems*, 99:28–38.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181, Vancouver, Canada. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.

Shiyue Zhang and Mohit Bansal. 2021. Finding a balanced degree of automation for summary evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6617–6632. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

A Appendix

TV Transcript

`Knock on door`

J.R.: Hey. Babe: Hey. J.R.: Can I come in? Since Iḿ no longer a suspected criminal, I figured we could do a more personal Christmas a little later on.
Babe: Yeah. Yeah, come on in.
J.R.: All right.
Babe: So, what happened, J.R.? I mean, did you finally remember where you were the night Zach was hit?
J.R.: With a little help from my mom. Itś a long story.
Babe: So –
J.R.: Well, a gentlemanś not supposed to kiss and tell, but I guess it depends on who the lady is. I was with Amanda, I was drunk as a skunk, and I told her every rotten thing that Iv́e ever done.
Babe: And she never said anything?
J.R.: Payback is sweet.
Babe: That bitch!
J.R.: Iḿ not going to argue with you on that one. But you want to know what the best part is? The cherry on top of the Christmas pie? Guess who else knew the truth and buried it like a bone in the backyard? Babe: Youŕe kidding.
J.R.: Nope. Thanks again, Dad, for everything.

`SCENE BREAK`

Adam: Hello, Stuart, itś your brother. ...

Human Recaps

Adam calls Stuart and tries to set up a time for them to have coffee together. Hannah comes in with a folder in her hand and tells Adam that she had what he wanted, Zach's head on a platter. Jack visits Greenlee and inquires if she trusts him. J.R. visits Babe, bearing gifts. He lets her know what had happened and how Amanda had set him up, and that Adam had known about it all along. Kendall is brought into an interrogation room. Zach walks up to the door and asks her why is she throwing everything away. Jack comes into the room and tells Kendall that he would like to represent her. Aidan visits Greenlee. Greenlee tells him that she thinks everyone had done a 180. Aidan was friends with Kendall and Greenlee was friends with Zach. Greenlee can't understand why Kendall is doing this. Hannah fills Adam in on what she had done to Zach and how everything now belongs to Adam. In putting the presents under the tree, J.R. finds one to Richie from Babe. Richie's doc from the prison visits him. Richie lets him know that he has chosen the day in which he is going to die, but he is leaving the insurance money to Babe instead of him

Generated Summary

J.R. was with Amanda the night Zach was hit. J.R was drunk and told Amanda everything he did wrong. Jack wants to clear his conscience by telling the truth about Greenlee and Kendall. Jack is doing it for Greenlee's sake. Kendall is turning herself in to the police. Hannah sold Chandler Enterprises in pieces to Adam. Adam will buy it back cheaper. Jack will prepare a strategy to knock some of the charges against her down. Kendall doesn't want to go to jail, but she's not going to take responsibility for all of that. Richie has changed the beneficiary of his insurance policy. Richie is going to die next week. Hannah took a job at Chandler Enterprises, but she doesn't want to work anymore. Adam offers her to be her concubine. Adam and Hannah argue. Adam thinks Hannah's talents are considerable, but he wouldn't overrate them. Hannah took the job because she was threatened by Alexander Cambias and she still loves Zach Slater. Kendall is in prison. Zach is angry at him because he broke his promise to God and to his family. Joe wants to have some additional blood work done on Greenlee Smythe. Little Adam drew Daddy a picture. Winnie helped Little Adam with it. Kendall and Zach kiss passionately. Annie and Richie are about to die. Adam and J.R. want to be father and son again. Kendall won't lie to the judge.

Figure 3: Illustrates an actual example derived from the SummScreen TMS dataset along with the output generated using our proposed summarization system.