MovieSum: An Abstractive Summarization Dataset for Movie Screenplays

Rohit Saxena   **Frank Keller**
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburg
10 Crichton Street, Edinburgh EH8 9AB
rohit.saxena@ed.ac.uk   keller@inf.ed.ac.uk

Abstract

Movie screenplay summarization is challenging, as it requires an understanding of long input contexts and various elements unique to movies. Large language models have shown significant advancements in document summarization, but they often struggle with processing long input contexts. Furthermore, while television transcripts have received attention in recent studies, movie screenplay summarization remains underexplored. To stimulate research in this area, we present a new dataset, MovieSum,[1] for abstractive summarization of movie screenplays. This dataset comprises 2200 movie screenplays accompanied by their Wikipedia plot summaries. We manually formatted the movie screenplays to represent their structural elements. Compared to existing datasets, MovieSum possesses several distinctive features: (1) It includes movie screenplays, which are longer than scripts of TV episodes. (2) It is twice the size of previous movie screenplay datasets. (3) It provides metadata with IMDb IDs to facilitate access to additional external knowledge. We also show the results of recently released large language models applied to summarization on our dataset to provide a detailed baseline.

1   Introduction

Large language models have shown significant improvements in abstractive summarization in recent years (Zhong et al., 2022; Zhu et al., 2021; Zhong et al., 2021; Zhang et al., 2022), aiming to produce a concise and coherent summary of the input document. However, these models often struggle when the input context is long, particularly when the relevant information is distributed across the document (Liu et al., 2023). To better understand this phenomenon and to advance research, datasets

are needed that not only contain long-form documents but also have the property that important information is dispersed throughout the document. Movie screenplays have these characteristics: to generate a faithful summary, an understanding of characters and events across the entire length of the screenplay is required.

More recently, narrative summarization research has focused on TV shows and books (Kryściński et al., 2021; Moskvichev and Mai, 2023), with less attention given to movie screenplays (Gorinski and Lapata, 2015; Papalampidi et al., 2020). Notably, Chen et al. (2022) introduced a dataset of TV show transcripts which has gained considerable interest and was included in a long document summarization benchmark (Shaham et al., 2022). But unlike movie screenplays, TV episode transcripts tend to be relatively short and predominantly comprise spoken dialogue with minimal scene or character descriptions. Additionally, they are not self-contained, as the events or characters from previous episodes can be referred to. In contrast, movie screenplays are structured documents with various screenplay elements such as scene headings, locations, character names, dialogues and detailed scene descriptions. These are written by screenwriters and are characteristically formatted to denote each element.

The largest current movie screenplay dataset (Gorinski and Lapata, 2015, 2018) comprises 917 automatically formatted screenplays (ScriptBase-j), with the most recent movie from 2013. We built MovieSum, a new movie screenplay dataset for abstractive summarization, which consists of 2200 movies, more than twice the size ScriptBase-j. Importantly, our new dataset has been formatted using a professional script writing tool and paired with Wikipedia plot summaries. Each movie is also tagged with its IMDB IDs to facilitate the collection of other external knowledge in rhw future. The dataset consists of movies spanning a wide range

---

[1] Our dataset and code is available at `https://github.com/saxenarohit/MovieSum`.

of genres from 1930 to 2023.

We provide a detailed description of MovieSum, including the steps for collecting and filtering screenplays and statistics and comparison with other narrative datasets. We conduct extensive experiments to evaluate the performance of state-of-the-art summarization models on MovieSum, demonstrating its utility as a benchmark dataset for narrative summarization research. Experiments suggest that recent models struggle with long abstractive summarization, and we hope that our efforts will inspire further research in this area. Furthermore, we provide qualitative analyses of how the structure of the screenplay can be utilized in generating summaries.

2 The MovieSum Dataset

We present MovieSum, a movie screenplay abstractive summarization dataset that consists of 2200 movie screenplay-summary pairs. All movie screenplays in the dataset are in English.

2.1 Collection of Movie Screenplays

We collected movie screenplays from a range of movie screenplay websites.[2] In total, we assembled 5,639 movie screenplays documents in various text format along with metadata of movie name, IMDB identifier, and release year. If the IMDB identifier was missing, we extracted it using the IMDB database. We then manually removed movies based on two criteria. Firstly, we removed any duplicate movie screenplays by using the movie names and release years to identify the duplicates. Secondly, we filtered out screenplays which did not have text or were incomplete.

2.2 Screenplay Formatting

Movie screenplays are structured documents with various script elements such as scene headings (also known as slug lines), characters' names, dialogues, and scene descriptions (or actions). These elements have specific markers based on spacing. Most of this formatting is lost when extracting text from these movie screenplay documents, making it challenging to retrieve the elements using regular expressions. To ensure the quality of the dataset, after filtering, we manually corrected the movie screenplay and formatted each movie screenplay using Celtx,[3] a professional screenplay writing tool. This

process preserved the format of movie screenplay elements. We further filtered out screenplays with encoding or optical character recognition errors.

2.3 Collection of Wikipedia Plot Summaries

To build a robust summarization dataset, it is necessary to collect high-quality human-written summaries. Similar to previous work (Kočiský et al., 2018), we collected Wikipedia plot summaries, which we found to be of high quality, helped by the fact that Wikipedia summaries follow a consistent set of guidelines for movie plot summaries.[4]

To collect the Wikipedia plot summary, we first extracted the Wikipedia page of the movie using the movie name and year, then collected text under the Plot section. We filtered out movies where the Wikipedia page or the plot section was unavailable.

This process resulted in 2200 manually formatted movie screenplays with corresponding Wikipedia summaries.

% Novel n-grams in Summary			
1-grams	2-grams	3-grams	4-grams
31.69	68.88	93.12	98.6

Table 1: Statistics for percentage of novel n-grams in the MovieSum summaries.

3 Dataset Analysis

This results in a dataset consisting of 2200 manually formatted movie screenplays along with their corresponding summaries. The average length of the screenplays is 29k words, with an average summary length of 717 words. Importantly, this dataset is twice the size of the previously available movie screenplays dataset with formatted movie screenplays (Gorinski and Lapata, 2015). Figure 1a. illustrates the genre distribution of the movies within the dataset and showcases the broad range of genres. In Figure 1b, the distribution of release years is depicted, revealing that the movies span a wide range of years, with a substantial number of them originating in recent years.

To study the abstractiveness of the summary, we report the percentage of novel n-gram in Table 1 as reported by Fabbri et al. (2021); Zhao et al. (2022). It shows that a high number of 3-gram and 4-gram are novel in summary and not present in the movie screenplays implying high abstractiveness of the

---

[2]https://www.scriptslug.com/, https://imsdb.com/, https://www.dailyscript.com/

[3]https://www.celtx.com/

[4]https://en.wikipedia.org/wiki/Wikipedia:How_to_write_a_plot_summary

(a) Genres Distribution.

(b) Release years compared to ScripBase-j
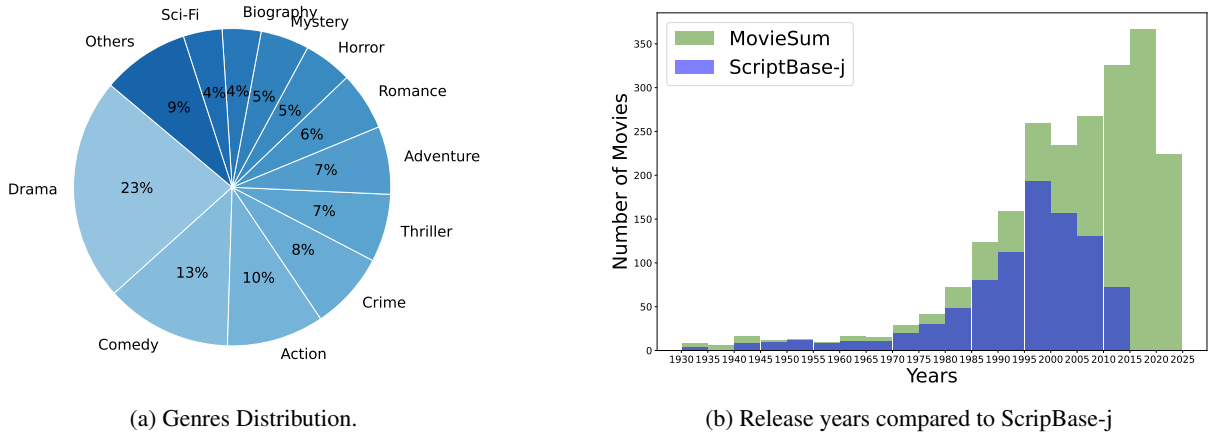
Figure 1: Distribution of movie genres and release years in the dataset.

Datasets	Domain	Type	Task	Size	Doc. Len.	Sum. Len
ScriptBase–j	Movies	Formatted Screenplays	Summarization	917	29K	753
ScriptBase–alpha	Movies	Formatted/Raw Screenplays	Summarization	917 / 359	29K	738
SummScreenFD	TV Episodes	Raw Transcripts	Summarization	4348	7605	113
NarraSum	Movies/TV	Plot Summary	Summarization	122K	786	147
NarrativeXL	Books	Raw Text	QA	1500	87K	–
NarrativeQA	Books/Movies	Raw Text/Screenplays	QA/Summarization	783 / 789	61K	650
BookSum	Books	Raw Text	Summarization	405	112K	1167
MovieSum	Movies	Formatted Screenplays	Summarization	2200	29K	717

Table 2: Comparison of MovieSum with different narrative datasets for movies and TV shows.

summary. See Appendix B for further analysis on the abstractiveness of the summary.

3.1 Comparison with Existing Datasets

We compare our dataset with various datasets in the narrative domain, and the statistics are reported in Table 2. These datasets include ScriptBase-j (Gorinski and Lapata, 2015), ScriptBase-alpha (Gorinski and Lapata, 2015), SummScreenFD (Chen et al., 2022), NarraSum (Zhao et al., 2022), NarrativeXL (Moskvichev and Mai, 2023), NarrativeQA (Kočiský et al., 2018), and BookSum (Kryściński et al., 2021). Notably, BookSum and NarrativeXL have a longer average document length but comprise books, not screenplays. SummScreenFD consists of TV show episode transcripts, which are much shorter in both document and summary length. Importantly, SummScreen consists of community-contributed transcripts and primarily comprises dialogues, unlike screenplays, which include detailed scene descriptions. Also, TV show episodes are not self-contained, as events or characters from previous episodes can be referenced. NarraSum contains plot summaries as documents rather than actual screenplays, and therefore has

the lowest average document length among the datasets we compare. NarrativeQA includes both books and movie screenplays, with books being notably lengthy, making the average document length comparable to book datasets. However, it consists of only 789 unformatted movie screenplays.

Both ScriptBase-j and ScriptBase-alpha datasets are close to our screenplay dataset. ScriptBase-j contains formatted screenplays, whereas ScriptBase-alpha comprises the unformatted raw text of screenplays. It is important to note that ScriptBase-j is a subset of ScriptBase-alpha, which consists of 917 formatted screenplays. On the other hand, ScriptBase-alpha includes an additional 359 movies. Our work can be considered as the extension of ScriptBase-j as it also consists of formatted screenplays. At the same time, we overcome two critical limitations of ScriptBase-j:

(1) The formatting of the movie screenplay was performed automatically. Although it is easy to detect the scene heading based on rules and string matching, it is challenging to distinguish dialogues, character names, and scene descriptions. The work does not provide any details regarding the auto-

	R-1	R-2	R-L	BS_p	BS_r	BS_{f1}
Lead-512	10.35	1.27	9.84	49.25	43.59	46.23
Lead-768	14.43	1.79	13.76	49.29	45.7	47.41
Lead-1024	17.93	2.24	17.15	49.12	46.91	47.98
TextRank	33.32	5.27	32.10	51.46	52.47	51.85
FLAN-UL2 8K (ZS)	23.62	4.29	22.01	52.9	49.57	50.87
Vicuna 13B 16K (ZS)	16.35	3.55	15.44	48.89	48.49	47.07
TextRank with Vicuna 13B 16K (ZS)	17.14	3.68	15.47	59.24	49.05	53.57
Moving Window Vicuna 13B 16K (ZS)	19.56	3.32	18.57	54.95	48.7	51.53
Pegasus-X 16K	42.42	8.16	40.63	58.81	50.56	54.36
LongT5 16K	41.49	8.54	39.78	56.09	55.36	55.68
Longformer (LED) 16K	44.85	9.83	43.12	59.11	58.43	58.73
Dialogues Only (LED)	44.68	10.02	42.94	59.30	58.29	58.74
Description Only (LED)	44.72	9.72	42.92	59.47	58.45	58.92
Heuristic Only (LED)	44.45	9.78	42.71	58.93	58.15	58.54

Table 3: Results of summarization models on the MovieSum dataset. We report ROUGE and BERTScores for the baselines and other summarization models.

matic formatting strategy. MovieSum, on the other hand, includes all the movies from ScriptBase-j formatted using professional screenplay tools.

(2) Both subsets of ScriptBase consist of movies released until 2013. In contrast, MovieSum also includes recently released movies. This is crucial for ensuring that summarization models remain robust to new movie narrative conventions. The movie release years are graphed in Figure 1b.

4 Experiments

We evaluate the MovieSum dataset using several baselines and state-of-the-art neural abstractive summarization models. We first report the **Lead-N** baseline, which simply outputs the first N tokens of the movie script as the movie summary. We varied the value N to understand the impact of summary length on performance and report results for **Lead-512**, **Lead-768**, and **Lead-1024**. For the extractive baseline, we used **TextRank** (Mihalcea and Tarau, 2004), a graph-based unsupervised extractive summarization method. For instruction-tuned large language models, we used **Vicuna 1.5 13B** 16K (Zheng et al., 2023), built on Llama-2 (Touvron et al., 2023), and **FLAN-UL2** (Tay et al., 2023; Wei et al., 2022) in a zero-shot setting. For fine-tuned models with long inputs, we utilized **LongT5** (Guo et al., 2022), **PEGASUS-X** (Phang et al., 2023), and the Longformer Encoder-Decoder (**LED**) model (Beltagy et al., 2020). We fully fine-tuned these models and report results on the test set. The implementation details of the models are

mentioned in Appendix A.

4.1 Results

Table 3 shows the summarization evaluation results using ROUGE F1 (1/2/L) scores (Lin, 2004) and BERTScore (Zhang et al., 2019) on MovieSum. The Lead baseline performs better with a higher number of words, achieving the best result with 1024 words. This is not surprising, as the ROUGE metric is known to give higher scores for longer summaries (Schluter, 2017). In the case of zero-shot, FLAN-UL2 8K performed substantially better than Vicuna 13B 16K. This confirms that a longer context does not necessarily lead to attention to the full context length. We also tested reducing the input context using TextRank, which only marginally improves the zero-shot performance for Vicuna 13 model. To further utilize the full context, we also report results for moving window chunk-based zero-shot summarization and concatenating the generated summaries. This performs better compared to using only the 16K context length. The best performance was achieved with fine-tuned models with longer context lengths. Pegasus-X, LongT5, and LED perform similarly well, with LED demonstrating superior performance.

5 Analysis of Screenplay Structure

We analyze the importance of screenplay elements, dialogue and scene description, and their impact on summarization performance. We selected the best model from the full fine-tuned experiment and stud-

ied the effects of fine-tuning solely on dialogues and descriptions. Additionally, we investigated the effects of selectively using both dialogue and scene description based on heuristics similar to Pu et al. (2022). Results in Table 3 show a marginal impact on summarization when dialogues or scene descriptions are removed. We also found the heuristic based removal of input context method results is comparable to the full-text results. This suggests that the current model does not fully utilize the inherent structure of the document, and new methods for content selection and summarization should consider both screenplay elements for movie script summarization.

6 Discussion and Conclusion

We introduce the MovieSum dataset, comprising formatted and recent movie screenplays paired with Wikipedia summaries. Our experiments demonstrate that it is a challenging dataset even for a large language model with a long input length. We hope that MovieSum will enable future research in the area of movie screenplay understanding and abstractive summarization.

Limitations

Limitations of the work include that the dataset consists of movie screenplays and their corresponding summaries only in English. Models trained on MovieSum may not generalize well to multilingual summarization tasks or applications requiring cross-lingual understanding.

Ethics Statement

Large Language Models: This paper uses pretrained large language models, which have been shown to be subject to a variety of biases, to occasionally generate toxic language, and to hallucinate content. Therefore, the summaries generated using our dataset should not be released without automatic filtering or manual checking.

Bias: Despite efforts to include a wide range of movies, the dataset may not fully represent the diversity of cinematic styles, languages, or cultural contexts. Models trained on MovieSum may therefore exhibit biases towards the types of movies included.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Mingda Chen, Zewei Chu, Sam Wiseman, and Kevin Gimpel. 2022. SummScreen: A dataset for abstractive screenplay summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8602–8615, Dublin, Ireland. Association for Computational Linguistics.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. SummEval: Re-evaluating Summarization Evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Philip John Gorinski and Mirella Lapata. 2015. Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1066–1076, Denver, Colorado. Association for Computational Linguistics.

Philip John Gorinski and Mirella Lapata. 2018. What's this movie about? a joint neural network architecture for movie content analysis. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1770–1781, New Orleans, Louisiana. Association for Computational Linguistics.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736, Seattle, United States. Association for Computational Linguistics.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Wojciech Kryściński, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong, and Dragomir Radev. 2021.

Booksum: A collection of datasets for long-form narrative summarization.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *arXiv preprint arXiv:2307.03172*.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Arsenii Moskvichev and Ky-Vinh Mai. 2023. NarrativeXL: a large-scale dataset for long-term memory models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15058–15072, Singapore. Association for Computational Linguistics.

Pinelopi Papalampidi, Frank Keller, Lea Frermann, and Mirella Lapata. 2020. Screenplay summarization using latent narrative structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1920–1933, Online. Association for Computational Linguistics.

Jason Phang, Yao Zhao, and Peter Liu. 2023. Investigating efficiently extending transformers for long input summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3946–3961, Singapore. Association for Computational Linguistics.

Dongqi Pu, Xudong Hong, Pin-Jie Lin, Ernie Chang, and Vera Demberg. 2022. Two-stage movie script summarization: An efficient method for low-resource long document summarization. In *Proceedings of The Workshop on Automatic Summarization for Creative Writing*, pages 57–66, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Natalie Schluter. 2017. The limits of automatic summarisation according to ROUGE. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 41–45, Valencia, Spain. Association for Computational Linguistics.

Uri Shaham, Elad Segal, Maor Ivgi, Avia Efrat, Ori Yoran, Adi Haviv, Ankit Gupta, Wenhan Xiong, Mor Geva, Jonathan Berant, and Omer Levy. 2022. SCROLLS: Standardized CompaRison over long language sequences. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 12007–12021, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. 2023. UL2: Unifying language learning paradigms. In *The Eleventh International Conference on Learning Representations*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yusen Zhang, Ansong Ni, Ziming Mao, Chen Henry Wu, Chenguang Zhu, Budhaditya Deb, Ahmed Awadallah, Dragomir Radev, and Rui Zhang. 2022. Summ^n: A multi-stage summarization framework for long input dialogues and documents. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1592–1604, Dublin, Ireland. Association for Computational Linguistics.

Chao Zhao, Faeze Brahman, Kaiqiang Song, Wenlin Yao, Dian Yu, and Snigdha Chaturvedi. 2022. NarraSum: A large-scale dataset for abstractive narrative summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 182–197, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

Ming Zhong, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2022. Dialoglm: Pre-trained model for long dialogue understanding and summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11765–11773.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev. 2021. QMSum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921, Online. Association for Computational Linguistics.

Chenguang Zhu, Ziyi Yang, Robert Gmyr, Michael Zeng, and Xuedong Huang. 2021. Leveraging lead bias for zero-shot abstractive news summarization. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 1462–1471, New York, NY, USA. Association for Computing Machinery.

A Implementation Details

For TextRank, we set the parameter $words = 1024$. We randomly split the dataset into 1800/200/200 as a train/val/test set to train the models. We used the base variants of Pegasus-X, LongT5, and LED for fine-tuning. Each input sequence for the movie is truncated to 16,384 tokens (including special tokens) to fit into the maximum input length of the model. We used AdamW as an optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.99$). For LED and LongT5, we used a learning rate of 2e-5 with a cosine scheduler and a warmup ratio of 0.01. We set the max_new_token to 1024 with greedy decoding for all the experiments. For Pegasus-X, we found that a learning rate of 5e-5 performed better with a linear warmup strategy and a warmup ratio of 0.01. All models were trained for 50 epochs, and the best model was selected using the ROUGE-1 on the validation set. The rest of the configurations for the models were kept as default. All the models were trained on A100 GPU with 80GB memory. We used the Huggingface $evaluate$ library for the implementation of the metrics.
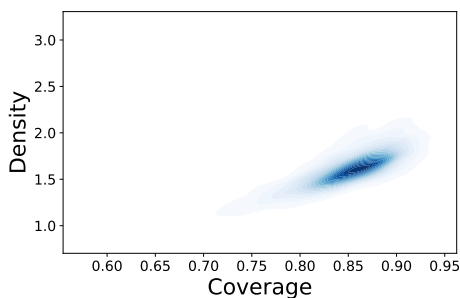
Figure 2: Coverage-Density plot of the summaries.

B Additional Statistics of Dataset

To further understand the abstractiveness of the summaries we computed the coverage and density of the summaries as discussed by Fabbri et al. (2021). The low density in Figure 2, indicates low overlap between the summary and the screenplays.

C Prompt Template

For the zero-shot experiments in Section 4, we used the following prompt template:

Prompt: Summarize the following movie script.
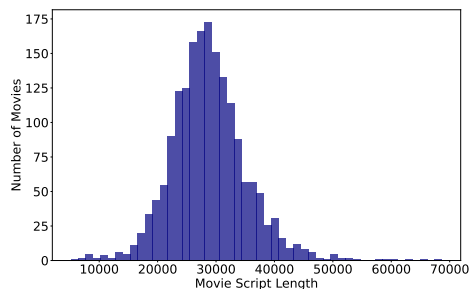Movie Script: {movie script text}
Summary:

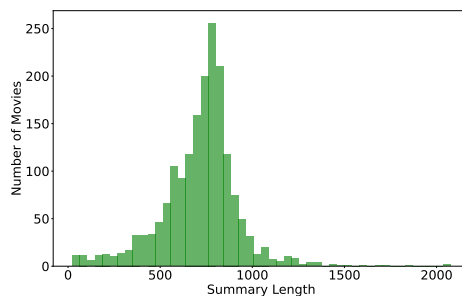Figure 3: Distribution of movie script length from the training set.

Figure 4: Distribution of summary length from the training set.

D Length Distribution

Figure 3 and 4 show the length distribution for movie scripts and their summaries across the training set. The mean length of movie scripts is 29K words, and the average length of summaries is 714 words.

E Example of a Movie Screenplay

Figure 5 shows an example of a cleanly formatted screenplay with distinct elements such as scene heading, characters, and dialogues. All the files are converted into XML using Celtx tool.

F Sample of Movie Summary

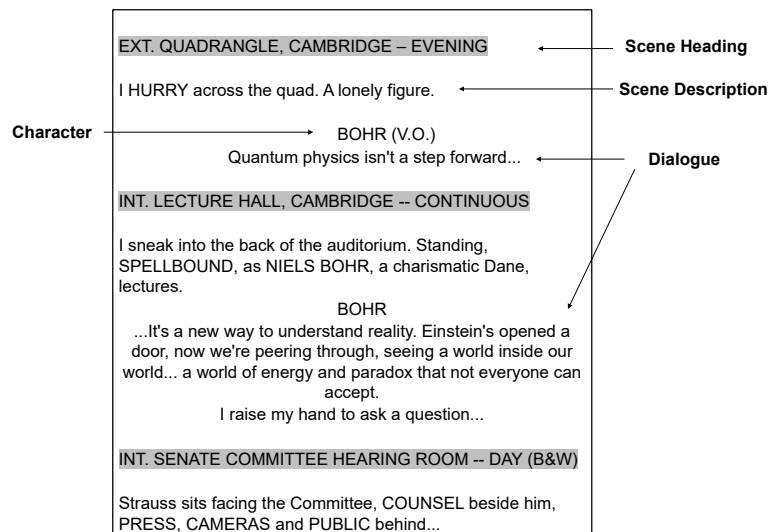Table 4 shows sample of generated summary of a movie using fine-tuned LED model (full-text).

Figure 5: Example of cleanly formatted scenes from a movie screenplay.

Movie: Oppenheimer

Generated Summary
In 1947, American physicist Dr. Robert Oppenheimer (Samuel L. Jackson) is appointed chairman of the American Enterprise Institute. After giving a lecture on quantum mechanics at Columbia University, Oppenheimer asks him, "How did you come to the place you are?" Angered by Oppenheimer's failure to present convincing proof of his theory, Oppenheimer dismisses him. Eltenton moves to California to start a school of his own, and he teaches himself the art of quantum mechanics. Lewis Strauss (Nicholson), a young scientist from New York City and a fellow associate at the institute, meets fellow physicist Dr. Robert Oppenheimer at a party meeting. They begin a relationship, and he introduces her to the Tolmans (Lerner) and their daughter, Peter. They form a secret organization, the Organization of American Unity, led by Dr. Robert Oppenheimer. Eventually, the Organization of American Unity is shut down. In 1958, Oppenheimer becomes the new chairman of the AEC, overseeing the construction of the Manhattan Engineer District. The AEC chairman, General Groves (Harvey Keitel), promotes Oppenheimer to general and orders him to head the school; he reluctantly does so. In 1964, Oppenheimer agrees to build a secret atomic weapon, the "Thermostellar Trigger" (Thermostellar Trigger) at Los Alamos, California. Oppenheimer asks the head of the school to help him construct a machine that could split the atom, but he is rebuffed. Angered that Oppenheimer would abandon the project, Oppenheimer urges him to leave the school and take a job in Washington, D.C. as a "secret member of the Communist Party". He argues that such a machine would be "destroyer of worlds". At the AEC meeting, he is overcome by guilt and despair, and Oppenheimer announces that he will start the school. In 1970, Oppenheimer died of a heart attack while trying to explain the concept of a supernova. Groves decides to start a secret organization, the Friends of the ABC, to support the project. Lawrence Bender (Edith Atwater), an old friend of Oppenheimer's, explains that in 1947 and 1954, he and several other scientists worked on a project to create a machine that could split the atom, but the team discovered that the machine could not get enough power to make the machine work. They sent the machine to Mexico, where it was secretly carried out. In 1969, Oppenheimer died of a heart attack while in the laboratory. Thirty years later, in 1997, Oppenheimer has died of a rare illness at the age of 73. Lawrence still works at the institute. In 2002, Oppenheimer loses his job because of the explosion. Thirty years later, in 2003, Oppenheimer, whose wife of over 25 years has died of a heart attack while working at the institute. Lawrence, Frank, and Jackie travel to Mexico to honor his friend and colleague, Dr. Hagenheimer. In a speech at his home following the death of his brother, Lawrence claims that he is ready to do what Oppenheimer asked of him. In a post-credits scene, as the screen fades to black.

Table 4: Model generated summary of the movie Oppenheimer. The summary consists of faithfulness errors.