DramaForge: AI-Assisted Tool for Theatrical Script Analysis and Adaptation Considering Production Constraints

Jaywoong Jeong
KAIST
Daejeon, Republic of Korea
jaywoong.jeong@kaist.ac.kr

Jae Young Choi
KAIST
Daejeon, Republic of Korea
jaeyoungchoi@kaist.ac.kr

Seon Gyeom Kim
KAIST
Daejeon, Republic of Korea
ksg_0320@kaist.ac.kr

Tak Yeon Lee
KAIST
Daejeon, Republic of Korea
reflect9@gmail.com

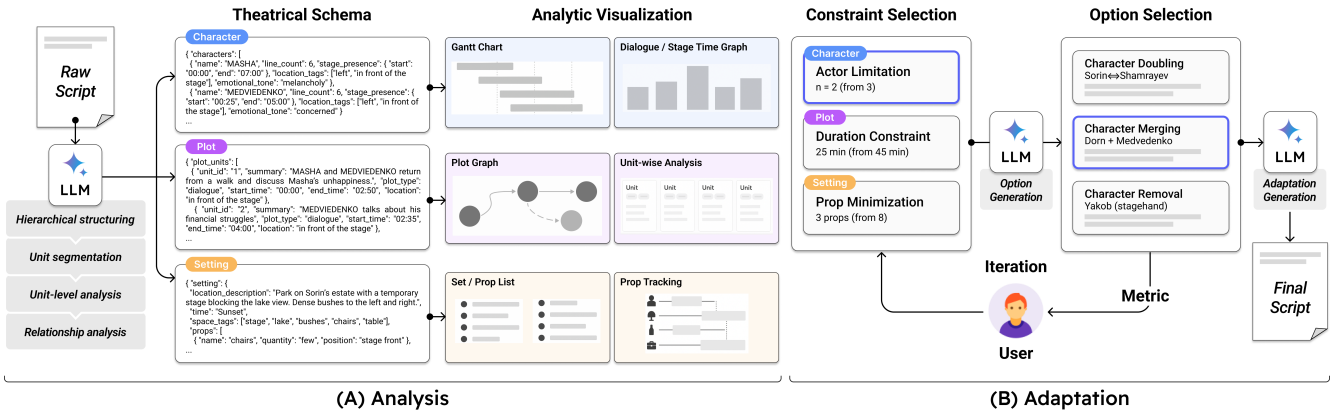

Figure 1: The DramaForge pipeline. (A) Analysis: A raw script is parsed into a structured schema and visualized. (B) Adaptation: Based on user constraints, the system generates adaptation options for user selection.

Abstract

Theatrical script adaptation is pervasive in theater production, but still requires complex manual effort. We investigate how Large Language Models (LLMs) can support this process. Our system, DramaForge utilizes an LLM to analyze and map a script's core structural dependencies. Based on this analysis, it generates targeted adapting options such as merging roles or cutting subplots to meet user-defined constraints. Our preliminary user study implies that theater practitioners find benefit in the system's visual analysis capabilities and iterative adaptation approach, suggesting strong potential for practical application in real-world theater productions.

CCS Concepts

• **Applied computing** → *Performing arts*; • **Computing methodologies** → *Artificial intelligence.*

Keywords

Natural Language Generation, HCI, Theater

ACM Reference Format:
Jaywoong Jeong, Jae Young Choi, Seon Gyeom Kim, and Tak Yeon Lee. 2018. DramaForge: AI-Assisted Tool for Theatrical Script Analysis and Adaptation Considering Production Constraints. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Theater is a performing art where a literary text is brought to the physical stage. In this process, the original script is often transformed to fit the production's unique context. For instance, a production may need to merge or remove roles to fit a smaller cast, cut scenes to meet a shorter runtime, or simplify the setting under a limited budget for props and sets. Since not all productions can leverage substantial resources, adapting the screenplay to accommodate specific constraints becomes crucial [1]. The goal of the adaptation is to refine the script while preserving its structural integrity, character consistency, and narrative causality [5, 11]. Maintaining these elements demands significant effort in practice: removing even a single scene may have unforeseen consequences for the story's coherence, making multiple table reads or rehearsals necessary

to assess the changes. Therefore, the adaptation is difficult and requires an iterative, experience-based process.

Recent HCI and Computational Linguistics research has explored Large Language Models (LLMs) as assistive tools in creative writing processes, particularly for generating content from scratch [3, 9]. These studies reveal LLMs' strengths in consistent plot construction, play creation, storytelling, and genre conversion [2, 7, 8, 10]. Their ability to capture subtext in lines and suggest creative transformations can be particularly valuable in adaptation work. However, to best our knowledge, the application of LLMs in theatrical script adaptation remains relatively unexplored.

To address this gap, we present DramaForge, an interactive system for structured analysis and constraint-based script adaptation. Using an LLM to deconstruct a script into a schema of characters and plot dependencies, DramaForge leverages this structure to suggest modifications. Our contribution lies in demonstrating the potential of LLMs for theatrical script adaptation through this structured approach.

2 DramaForge: System Overview

The core of DramaForge[1] lies in its framework for analyzing and adapting scripts using LLMs. The system transforms scripts from static text into a computationally tractable format designed specifically for the task of adaptation. This process is driven by two main components: first, a structured analysis that deconstructs the script into a rich schema as shown in Figure 1A, and second, a constraint-based adaptation interface that allows users to interact with this schema as illustrated in Figure 1B.

2.1 Use Case Scenario

To illustrate the utility of DramaForge, consider Jenny, the director of a university theater club. She needs to stage a production with only three actors and a runtime of under 20 minutes. When Jenny uploads her script and uses the **Analysis** mode, she gets results like those shown in Figure 1A: the Gantt chart reveals that two minor characters never appear on stage together, making them prime candidates for a double role, while the plot graph highlights a secondary plotline that is not causally linked to the main narrative's resolution. Next, switching to the **Adaptation** mode as shown in Figure 2, Jenny sets her constraints as a maximum cast of three actors and 20 minutes running time. The system, leveraging its analysis, suggests merging the two minor characters. The adaptation results are then presented through key metrics, showing how these changes affect the overall productions.

2.2 Structured Analysis

DramaForge creates a structured representation of scripts through a multi-stage pipeline that models narrative and production elements. First, the system performs **Hierarchical Structuring**. An LLM parses the raw script text, identifying fundamental dramatic structures like acts and scenes. It distinguishes between stage directions and spoken lines, and further classifies lines into dramaturgical subtypes such as dialogue, monologue, aside, or soliloquy [4]. Next, for a more granular analysis, the system does **Unit Segmentation**.

[1]A complete pipeline schema and the full LLM prompt are available at https://drama-forge.vercel.app.

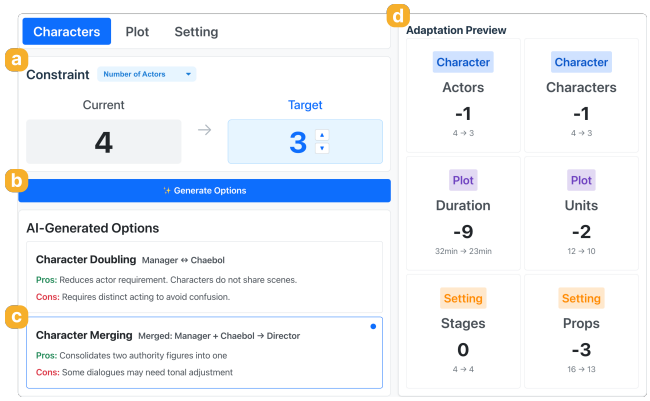

Figure 2: Adaptation Process of DramaForge (A) Users set constraints. (B) System generates strategies. (C) Users select adaptation options. (D) User review key metrics for the selected option.

Each scene is broken down into smaller narrative *units*, which represent distinct moments of action or shifts in conversation [12]. Then, **Unit-level Analysis** is conducted. For each unit, the system extracts entities like props and sets tracking their status (e.g., new, continuing, moved) by referencing the analysis of the preceding unit. It also infers both explicit and implicit character entrances and exits from the context. Finally, **Relationship Analysis** maps the causal and thematic links between all units to construct a plot graph, which allows for the distinction between the main plot and various subplots. The resulting schema plays a role as the data structure organized around the three fundamental dimensions of dramaturgy: *Character*, *Plot*, and *Setting* [4, 6].

Based on the schema extraction, the system presents the complex data through visualizations. These include Gantt charts illustrating character co-presence, bar graphs quantifying dialogue volume and stage time, and plot graphs mapping narrative dependencies. These visualizations empower creators with a holistic understanding of the script, allowing them to identify potential areas for adaptation before making any changes.

2.3 Constraint-based Adaptation

The adaptation step allows users to iteratively set their production constraints, modify and evaluate script. DramaForge operates on a unit-based level by leveraging the structured analysis of dramatic units and their causal and thematic relationships extracted from the preceding phase. This allows the system to propose targeted adaptation options. For instance, in response to a constraint such as reducing the number of actors, shown in Figure 2A, the LLM generates several adaptation options by utilizing the underlying analysis, as in Figure 2B. It might suggest character doubling options or propose character merging, as in Figure 2C. These suggestions are justified through co-presence data confirming characters never appear simultaneously, or causal analysis identifying subplots. For each option, the system outlines potential problems and consequences, allowing users to review adaptation options with their

trade-offs. The system then provides adaptation preview, as in Figure 2D, showing how the selected adaptation affects key production metrics. Once the user selects a preferred option, DramaForge executes the change by modifying the necessary units to produce a final, adapted script. This iterative loop of constraint-setting, AI-generated suggestion, and user-led selection ensures a transparent and user-centric process, where the AI serves as a powerful analytical assistant, not an autonomous creator.

3 Conclusion and Future Work

We present DramaForge, an LLM-based System that assists in theatrical script adaptation using a novel pipeline. By analyzing scripts across *Character*, *Plot*, and *Setting*, it provides structural insights and suggests adaptation options, empowering creators through an iterative process. We acknowledge that our approach currently excels at local adaptations over global narrative rewriting, and that maintaining a consistent character voice in automated edits remains a challenge. Building on our preliminary findings, we plan to conduct a comprehensive user study with theater directors to evaluate the system's usability and the quality of its suggestions in real-world production contexts.

References

[1] Jane Barnette. 2018. *Adapturgy: The Dramaturg's Art and Theatrical Adaptation.* SIU Press.

[2] Jing Chen, Xinyu Zhu, Cheng Yang, Chufan Shi, Yadong Xi, Yuxiang Zhang, Junjie Wang, Jiashu Pu, Rongsheng Zhang, Yujiu Yang, and Tian Feng. 2024. HoLLMwood: Unleashing the Creativity of Large Language Models in Screenwriting via Role Playing. arXiv:2406.11683 [cs.CL] https://arxiv.org/abs/2406.11683

[3] John Joon Young Chung, Wooseok Kim, Kang Min Yoo, Hwaran Lee, Eytan Adar, and Minsuk Chang. 2022. TaleBrush: Sketching stories with generative pretrained language models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–19.

[4] Keir Elam. 2003. *The semiotics of theatre and drama.* Routledge.

[5] Linda Hutcheon. 2006. *A theory of adaptation.* Routledge.

[6] Donald W Lucas. 1968. Aristotle poetics. *The Classical Review* 18, 02 (1968).

[7] Louis Mahon and Mirella Lapata. 2024. ScreenWriter: Automatic Screenplay Generation and Movie Summarisation. arXiv:2410.19809 [cs.AI] https://arxiv.org/abs/2410.19809

[8] Piotr Mirowski, Kory W Mathewson, Jaylen Pittman, and Richard Evans. 2023. Co-writing screenplays and theatre scripts with language models: Evaluation by industry professionals. In *Proceedings of the 2023 CHI conference on human factors in computing systems*. 1–34.

[9] Kyeongman Park, Nakyeong Yang, and Kyomin Jung. 2023. LongStory: Coherent, Complete and Length Controlled Long story Generation. arXiv:2311.15208 [cs.CL] https://arxiv.org/abs/2311.15208

[10] Tanzir Pial, Shahreen Aunti, Charuta Pethe, Allen Kim, and Steven Skiena. 2023. Analyzing Film Adaptation through Narrative Alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 15560–15579. doi:10.18653/v1/2023.emnlp-main.962

[11] Magda Romanska. 2015. *The Routledge companion to dramaturgy.* Routledge London and New York.

[12] Perviz Sawoski. 2010. The Stanislavski System: growth and methodology. *Santa Monica College* (2010).

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009