Evaluating LLM Story Generation through Large-scale Network Analysis on Social Structures

Anonymous Author(s)
Affiliation
Address
email

Abstract

Evaluating the creative capabilities of large language models (LLMs) in complex tasks often requires human assessments that are difficult to scale. We introduce a novel, scalable methodology for evaluating LLM story generation by analyzing underlying social structures in narratives as signed character networks. To demonstrate its effectiveness, we conduct a large-scale comparative analysis of over 1,200 stories, generated by four leading LLMs (GPT-4o, GPT-4o mini, Gemini 1.5 Pro, and Gemini 1.5 Flash) and a human-written corpus. Our findings, based on network properties like density, clustering, and signed edge weights, show that LLM-generated stories consistently exhibit a strong bias toward tightly-knit, positive relationships, which aligns with findings from prior research using human assessment. Our proposed approach provides a valuable tool for evaluating limitations and tendencies in the creative storytelling of current and future LLMs.

1 Introduction

The rise of capable large language models (LLMs) in the past few years has sparked research interest in applying them to complex tasks, such as coding and agentic planning (1; 2). Although many evaluation metrics have been proposed to assess their behaviors in such domains, evaluations of their creative performance are still understudied (3; 4). One example of complex, creative tasks is story generation, and prior research has discovered that LLM-generated stories tends to focus on positive plot progression, and are less dynamic and inferior to human experts in terms of creativity (5; 6; 7; 8). However, evaluating creative writing is often qualitative, requiring labor-intensive human assessment, and suffers from low efficiency and scalability. We propose a novel, quantitative methodology that leverages character networks and evaluates LLMs' complex behaviors in story generation by focusing on the structure of narrative character interactions. A character network models the relationships between narrative characters by representing them as vertices and their interactions as edges. Our analysis shows that the networks of LLM-generated stories exhibit significantly higher density, clustering, and a strong bias towards positive relationships, revealing a systemic tendency to create more tightly-knit and less conflict-driven social dynamics than those found in human-written stories. Notably, this conclusion is supported by various evaluations such as plot progression analysis and human assessment (5; 6; 7).

Although several works apply character network analysis to human-written narratives (9; 10; 11; 12; 13), to our knowledge, none have focused on the networks of LLM-generated stories. To demonstrate the effectiveness of our proposed approach, in this study, we use our method to investigate LLMs' creativity in story generation compared to humans. We extracted networks from over 1,200 LLM-generated and human-written stories and conducted a large-scale analysis. Potential contributions of our research are as follows: (1) we introduce a scalable framework for the quantitative network analysis of AI-generated narratives, which reveals underlying tendencies of LLM story generation;

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

(2) this is the first work applying network analysis to LLM-generated stories; and (3) our comparative analysis provides empirical evidence that LLMs construct positive-biased narrative social structures relative to humans.

2 Related work

LLMs in creative writing Motivated by the advancement of LLMs' performance, researchers have investigated the models' creativity in story writing. They discovered that LLM-generated stories are prone to construct positive plot and are inferior to human writing in terms of diversity, novelty, and surprise through evaluation methods involving human annotation (5; 6; 7). There are also research endeavors to establish evaluation frameworks for LLM creativity with human evaluators/AI systems (6; 14; 15).

Character network analysis Early foundational work established methodologies for extracting character relationships from novels, legends, movies, and comics through co-occurrence analysis, conversation tracking, and coreference resolution (9; 10; 11; 12; 13). Genre classification and narrative analysis through network properties have shown promising results, indicating networks are a good model of social dynamics (9; 12; 13). Although various edge properties are used to model social structures, *signed scores* (negative/positive labels of relationships) are one of the most popular approaches for its simplicity (12; 16; 17; 18). In this study, we conduct an extensive analysis on signed networks from LLM-written short stories.

3 Methodology

In this section, we introduce the overview of our methodology, specifically regarding story generation, network extraction, and metrics. To conduct comparative evaluation, we collect approximately 250 science-fiction short stories from GPT 4o, GPT 4o Mini, Gemini 1.5 Pro, Gemini 1.5 Flash, and *Project Gutenberg*, respectively (19; 20).[1] Further details about story generation and the selection criteria of human-written stories are provided in Appendix A.

Next, we extract character networks from the stories with our automated pipeline. The process starts by splitting a story into approximately 100 *narrative units*. We then calculate the negative/positive label of each narrative unit using a RoBERTa-based sentiment classifier (21; 22). The relationship labels between two characters are determined based on their *co-occurrences* in narrative units with negative or positive labels. After constructing a network with characters (vertices) and their relationships (edge weights $\in \{-1, 1\}$), the pipeline performs a vertex contraction if two characters are estimated to refer to the same character. We then remove sparse networks based on our exclusion criteria. Further details of network extraction and removal are outlined in Appendix B. We note the number of networks eventually obtained is 251 from GPT4o, 249 from GPT 4o Mini, 252 from Gemini 1.5 Pro, 249 from Gemini 1.5 Flash, and 168 from Project Gutenberg.

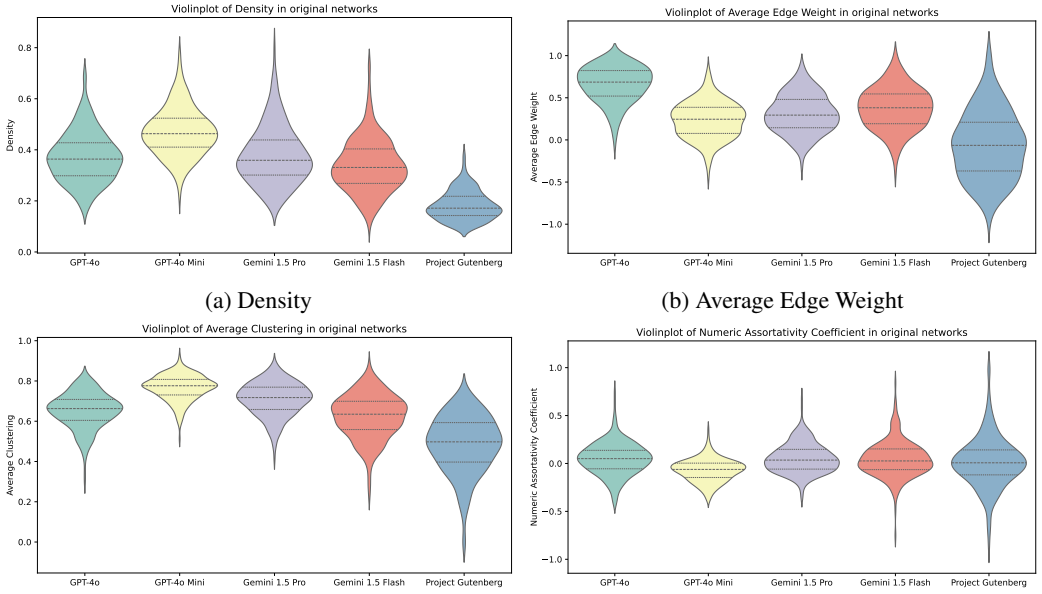
We calculate multiple connectivity measures of a network using `NetworkX` and self-made functions. We use average edge weight, as well as density, average clustering coefficient, and assortativity mixing following prior research (9; 10; 11; 13; 23; 24; 25). In particular, average edge weight ranges from -1 to 1 and represents the overall polarity of a network. Average clustering coefficient quantifies the small-world-ness of a network, and assortativity mixing, ranging from -1 to 1, is designed to represent the homogeneity of interactions among heroic and villainous characters. The formulas and interpretations of the metrics are listed in Appendix C. We also extract two subgraphs (one consisting of only *positive* edges, which we call *positive networks*, and another only with *negative* edges, called *negative networks*), and calculate density and average clustering.

4 Results

Distribution analysis We analyze the distributions of connectivity scores to better understand the tendencies of LLMs and humans, which we collectively call *writers*, in story generation. Figure 1 visualizes the score distributions of each metric. Overall, the scores of LLM-generated stories

[1] Source: https://www.kaggle.com/datasets/shubchat/1002-short-stories-from-project-guttenberg

fit in a similar range, while the scores of human-written stories (blue) spread out and diverge from LLM counterparts. In particular, assortativity scores demonstrate a relatively strong trend of data concentration among the AI models.



(a) Density

(b) Average Edge Weight

(c) Average Clustering Coefficient

(d) Assortativity Mixing

Figure 1: Violin plots of connectivity measure distributions. The horizontal axes show writers: from the left, Gemini 1.5 Flash (green), GPT 4o mini (yellow), Gemini 1.5 Pro (purple), GPT 4o (red), and Project Gutenberg (blue). LLM score distributions cluster in the somewhat same range.

To quantify distances between score distributions by writers, we calculate Wasserstein distances (see the heatmaps in Appendix D). Overall, human stories have the greatest Wasserstein distances with all the LLM stories in almost every metric, whereas LLMs maintain smaller distances with each other.

Overall analysis We also calculate the mean and standard deviation of the score distribution of each writer and metric, which is outlined in Table 1. Notably, the average edge weight of LLM-generated stories are higher than that of human stories, which is -0.061, the only negative average edge weight. Moreover, density is also consistently higher in LLM-generated stories. We perform a similar analysis to positive and negative networks. The results show that positive networks are higher both in the density and average clustering relative to negative networks. The table is provided in Appendix E.

Models	Density		Avg EW		Avg Clustering		Assort Mixing	
	mean	std	mean	std	mean	std	mean	std
GPT 4o	0.372	0.097	0.659	0.214	0.665	0.087	0.047	0.168
GPT 4o Mini	0.470	0.094	0.235	0.214	0.766	0.062	−0.072	0.118
Gemini 1.5 Pro	0.378	0.112	0.312	0.227	0.709	0.082	0.052	0.151
Gemini 1.5 Flash	0.338	0.102	0.374	0.248	0.623	0.108	0.044	0.184
Humans	0.182	0.056	−0.061	0.398	0.485	0.140	0.012	0.260

Table 1: The mean and standard deviation of Density, Average Edge Weight (Avg EW), Average Clustering (Avg Clustering), and Assortativity (Assort Mixing).

Statistical significance To rigorously measure similarities between pairs of score distributions, we conduct t-tests. The null hypothesis is that the means of the score distributions are equal. Several

metrics across models, such as density (Gemini Pro and GPT 4o: $p = 0.520$) and average clustering of positive networks (Gemini Pro and GPT 4o Mini: $p = 0.792$, GPT 4o and GPT 4o Mini: $p = 0.116$) and negative networks (Gemini Flash and Pro: $p = 0.840$) have high p-values, indicating that the scores samples from two distinct models are not unlikely to be drawn from the same sample space. Besides assortativity, as expected, p-values for pairs with human-written stories are consistently very low ($p < 0.01$) in almost every metrics. The details of the tests and results are in Appendix F.

5 Discussion

Similarities of LLM-generated stories in original networks Overall, the Wasserstein distances and t-tests show that LLMs have connectivity measure scores that cluster in identical ranges, while human-written stories are dispersed and distant from LLM-generated stories. Moreover, LLM-written stories tend to be denser, indicating more characters co-occur in the same narrative units compared to the human-written stories. Their relatively high average clustering coefficient also supports our observation that LLMs focus on tightly-knit character interactions.

Positivity bias and plain relationship dynamics We can understand the relationship tendency prevalent in LLM-generated stories through the average edge weight, assortativity, and average clustering. As Table 1 shows, average edge weight is significantly higher in LLM-generated stories, indicating that the stories largely have positive relationship dynamics. Moreover, although there is some degree of standard deviation, the mean assortativity mixing of most LLM stories stays at around 0.05. This result suggests that there is a subtle trend that characters of similar weighted average neighbor degrees cluster together, i.e., they form slightly homogeneous interaction networks. Interestingly, GPT 4o Mini tends to generate slightly non-homogenious networks. The relatively high average clustering in LLMs also tells that the networks of LLM stories form relatively small worlds. Table 2 in Appendix E allows for a closer analysis of positive and negative subgraphs. It is noteworthy that LLM positive networks tend to be denser than the negative networks, and their average clustering coefficients are also considerably higher. These statistics indicate that a group of characters sharing positive relationships form a more intimate and tied network in LLM-generated stories, whereas the negative counterpart is sparse. Given that an edge is positive if two characters co-occur more in positive narrative units, the high clustering coefficient in positive networks implies that a group of amiable characters is prone to appear jointly in positive units repeatedly, which inhibits suspenseful or dramatic plot progression (e.g., the group of protagonists explores a dungeon, and the story proceeds by following their journey). These results show that **LLMs generate stories that are biased toward positive relationships and devoid of dramatic dynamics**. Interestingly, these results align with the past findings discovered through semi-manual plot analysis and creativity tests with human experts (5; 6; 7). This shows that **our automated network-based evaluation method successfully identifies underlying tendencies in LLM story generation, aligning with human-annotated evaluations that focus on various aspects of narratives**. Therefore, our methodology serves as a novel tool utilizing large-scale evaluation for LLM creative writing.

6 Conclusion

In this research, we propose a evaluation method of LLM creativity in story generation through character network analysis. We analyze the networks of short stories from four LLMs and a human story corpus. Our extensive analysis reveals that LLMs overly focus on positive relationships and lack creativity in dramatic narrative composition relative to human-written stories. These results also demonstrate the effectiveness of our large-scale, automated network analysis method to evaluate underlying strengths and limitations of LLMs in complex, creative tasks.

There are many promising future extensions of this research. One can use different edge weights, such as conversations, mentions, and direct actions (12). Analyzing character networks from other genres than science fiction may also be of interest. Another future direction is to introduce more extensive human story datasets and examine potential similarities between LLM story generation with human storytelling. One can also apply our method to longer and larger LLM-generated stories, which would yield larger networks, and analyze community detection structures and robustness, potentially producing further interesting findings. Finally, future research should apply network analysis to other types of tasks that involve social structures to holistically understand LLM creativity.

References

[1] J. Jiang, F. Wang, J. Shen, S. Kim, and S. Kim, "A survey on large language models for code generation," 2024. [Online]. Available: https://arxiv.org/abs/2406.00515

[2] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen, "A survey on large language model based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, Mar. 2024. [Online]. Available: http://dx.doi.org/10.1007/s11704-024-40231-1

[3] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, and W. Zaremba, "Evaluating large language models trained on code," 2021. [Online]. Available: https://arxiv.org/abs/2107.03374

[4] X. Liu, H. Yu, H. Zhang, Y. Xu, X. Lei, H. Lai, Y. Gu, H. Ding, K. Men, K. Yang, S. Zhang, X. Deng, A. Zeng, Z. Du, C. Zhang, S. Shen, T. Zhang, Y. Su, H. Sun, M. Huang, Y. Dong, and J. Tang, "Agentbench: Evaluating llms as agents," 2023. [Online]. Available: https://arxiv.org/abs/2308.03688

[5] Y. Tian, T. Huang, M. Liu, D. Jiang, A. Spangher, M. Chen, J. May, and N. Peng, "Are large language models capable of generating human-level narratives?" in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds. Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 17 659–17 681. [Online]. Available: https://aclanthology.org/2024.emnlp-main.978/

[6] T. Chakrabarty, P. Laban, D. Agarwal, S. Muresan, and C.-S. Wu, "Art or artifice? large language models and the false promise of creativity," in *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, ser. CHI '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3613904.3642731

[7] M. Ismayilzada, C. Stevenson, and L. van der Plas, "Evaluating creative short story generation in humans and large language models," 2025, https://arxiv.org/abs/2411.02316.

[8] Z. Xie, T. Cohn, and J. H. Lau, "The next chapter: A study of large language models in storytelling," 2023, https://arxiv.org/abs/2301.09790.

[9] A. J. Holanda, M. Matias, S. M. S. P. Ferreira, G. M. L. Benevides, and O. Kinouchi, "Character networks and book genre classification," 2018, https://arxiv.org/abs/1704.08197.

[10] R. Alberich, J. Miro-Julia, and F. Rossello, "Marvel universe looks almost like a real social network," 2002, https://arxiv.org/abs/cond-mat/0202174.

[11] P. M. Gleiser, "How to become a superhero," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2007, no. 09, p. P09020–P09020, Sep. 2007, http://dx.doi.org/10.1088/1742-5468/2007/09/P09020.

[12] V. Labatut and X. Bost, "Extraction and analysis of fictional character networks: A survey," *ACM Computing Surveys*, vol. 52, no. 5, p. 1–40, Sep. 2019, http://dx.doi.org/10.1145/3344548.

[13] D. Elson, N. Dames, and K. McKeown, "Extracting social networks from literary fiction," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, J. Hajič, S. Carberry, S. Clark, and J. Nivre, Eds. Uppsala, Sweden: Association for Computational Linguistics, Jul. 2010, pp. 138–147, https://aclanthology.org/P10-1015/.

[14] W. Orwig, E. R. Edenbaum, J. D. Greene, and D. L. Schacter, "The language of creativity: Evidence from humans and large language models," *The Journal of Creative Behavior*, vol. 58, no. 1, pp. 128–136, 2024. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/jocb.636

[15] D. Johnson, J. Kaufman, B. Baker, J. Patterson, B. Barbot, A. Green, J. van Hell, E. Kennedy, G. Sullivan, C. Taylor, T. Ward, and R. Beaty, "Divergent semantic integration (dsi): Extracting creativity from narratives with distributional semantic modeling," *Behavior research methods*, vol. 55, 10 2022.

[16] S. Chaturvedi, S. Srivastava, H. Daume III, and C. Dyer, "Modeling evolving relationships between characters in literary novels," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, Mar. 2016. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/10358

[17] L. Ding and A. Yilmaz, "Learning relations among movie characters: A social network perspective," in *Computer Vision – ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 410–423.

[18] O.-J. Lee and J. J. Jung, "Modeling affective character network for story analytics," *Future Generation Computer Systems*, vol. 92, pp. 458–478, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X17310221

[19] OpenAI, "Gpt-4o system card," 2024, https://arxiv.org/abs/2410.21276.

[20] G. Team, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," 2024, https://arxiv.org/abs/2403.05530.

[21] J. Hartmann, M. Heitmann, C. Siebert, and C. Schamp, "More than a feeling: Accuracy and application of sentiment analysis," *International Journal of Research in Marketing*, vol. 40, no. 1, pp. 75–87, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167811622000477

[22] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," 2019. [Online]. Available: https://arxiv.org/abs/1907.11692

[23] M. Coll Ardanuy and C. Sporleder, "Structure-based clustering of novels," in *Proceedings of the 3rd Workshop on Computational Linguistics for Literature (CLFL)*, A. Feldman, A. Kazantseva, and S. Szpakowicz, Eds. Gothenburg, Sweden: Association for Computational Linguistics, Apr. 2014, pp. 31–39, https://aclanthology.org/W14-0905/.

[24] A. Bonato, D. R. D'Angelo, E. R. Elenberg, D. F. Gleich, and Y. Hou, "Mining and modeling character networks," 2016, https://arxiv.org/abs/1608.00646.

[25] S. Grayson, K. Wade, G. Meaney, and D. Greene, "The Sense and Sensibility of Different Sliding Windows in Constructing Co-occurrence Networks from Literature," in *IFIP Advances in Information and Communication Technology*, ser. Computational History and Data-Driven Humanities, vol. AICT-482, Dublin, Ireland, May 2016, pp. 65–77, https://inria.hal.science/hal-01616308.

[26] G. Team, "Gemini: A family of highly capable multimodal models," 2024, https://arxiv.org/abs/2312.11805.

[27] D. Oelke, D. Kokkinakis, and M. Malm, "Advanced visual analytics methods for literature analysis," in *Proceedings of the 6th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, K. Zervanou and A. van den Bosch, Eds. Avignon, France: Association for Computational Linguistics, Apr. 2012, pp. 35–44, https://aclanthology.org/W12-1007/.

[28] M. Elsner, "Character-based kernels for novelistic plot structure," in *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, W. Daelemans, Ed. Avignon, France: Association for Computational Linguistics, Apr. 2012, pp. 634–644, https://aclanthology.org/E12-1065/.

[29] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-strength Natural Language Processing in Python," 2020.

[30] H. Vala, D. Jurgens, A. Piper, and D. Ruths, "Mr. bennet, his coachman, and the archbishop walk into a bar but only one of them gets recognized: On the difficulty of detecting characters in literary texts," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, L. Màrquez, C. Callison-Burch, and J. Su, Eds. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 769–774, https://aclanthology.org/D15-1088/.

[31] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, pp. 440–442, 1998, https://api.semanticscholar.org/CorpusID:3034643.

[32] J. Saramäki, M. Kivelä, J.-P. Onnela, K. Kaski, and J. Kertész, "Generalizations of the clustering coefficient to weighted complex networks," *Physical Review E*, vol. 75, no. 2, feb 2007, http://dx.doi.org/10.1103/PhysRevE.75.027105.

Appendix A: Story generation

A.1 Human short stories

To compare LLM-generated stories with human-written ones, we collected 255 short stories from a dataset of 1,002 stories extracted from the Project Gutenberg dataset.[2] We classified their genres using Gemini 2.0 Flash (26) and collected only science fiction since it was the most frequent genre in the dataset. We also filtered out stories whose approximate word count is less than 3,000 or is larger than 15,000 in order to align the length with LLM-generated stories.

A.2 LLM short stories

We used four major LLMs: OpenAI GPT4o, GPT4o-mini (19), Google Gemini-1.5-pro, and Gemini-1.5-flash (20), each of which generated around 250 science-fiction short stories. To ensure generality, we created a predefined prompt template for character generation, plot planning, and story generation. To maximize the randomness, we set *temperature* to 1. We also configured *top_p* to 0.95 (8) and *top_k* to 40. For models that do not accept certain parameters, we used their default configurations.

The models first generate the plot of 10 chapters and the demography of 19 characters with the chapter numbers where they appear. We derived the number of characters 19 by taking the average of character counts in the 255 human stories. This is to control the node count and focus solely on relationships between characters. We insert a chapter description and the list of characters into the chat log before a model generates a chapter, in order to maintain the consistency of the story context. The algorithm of story generation is reproduced below.

Algorithm 1: Story Generation

Require : System prompt: S; Plot prompt: P; Character prompt: CR; Chapter prompt: CH; i'th chapter: CH_i; Text generation function: f; Concatenation operation: $\oplus$;
Input : Number of chapters: N; Generation configuration: C;
Initialize : Session log: $\texttt{session} \leftarrow [\,]$; List of chapter descriptions: $\texttt{plot} \leftarrow [\,]$; Character list: $\texttt{characters} \leftarrow [\,]$; Chapter i: $\texttt{chapter}_\texttt{i}$; Story: $\texttt{story} \leftarrow$ ""

$\texttt{session} \leftarrow S \oplus P$;
$\texttt{plot} : [\texttt{plot}_1, \texttt{plot}_2, \ldots, \texttt{plot}_\texttt{n}] \leftarrow f(\texttt{session}, C)$;
$\texttt{session} \leftarrow \texttt{session} \oplus \texttt{plot} \oplus CR$;
$\texttt{characters} = f(\texttt{session}, C)$;
for $i \leftarrow 1$ **to** N **do**
  $\texttt{session} \leftarrow \texttt{session} \oplus CH \oplus \texttt{characters} \oplus \texttt{plot}_\texttt{i}$;
  $\texttt{chapter}_\texttt{i} \leftarrow f(\texttt{session}, C)$;
  $\texttt{session} \leftarrow \texttt{session} \oplus \texttt{chapter}_\texttt{i}$;
  $\texttt{story} \leftarrow \texttt{story} \oplus \texttt{chapter}_\texttt{i}$;
return story

Prompt template This section provides the details of the prompt template. As Algorithm 1 shows, the prompt template for story generation consists of the system prompt, the prompt for plot generation (plot prompt), the prompt for character generation (character prompt), and the prompt for chapter writing (chapter prompt).

System Prompt

Instruction
You are a professional novelist. You will write a science fiction story of 10 chapters with 19 characters.

[2] Source: https://www.kaggle.com/datasets/shubchat/1002-short-stories-from-project-guttenberg

Plot Prompt

Write the title in the first line. Next, use 1 sentence to write the plot for each of the 10 chapters. The Chapter number and description should start in the same line (i.e. Chapter 1: [description]). Start with Chapter 1:
Plot

Character Prompt

Instruction
Next, use 1 sentences to write each of 19 characters and chapters where they appear.
Characters

Chapter Prompt (Chapter 1)

Instruction
Use 800 words to write the first chapter.
Story

Chapter Prompt (After Chapter 1)

Instruction
Use 800 words to write the next chapter.
Characters
The character list created with the character prompt is inserted here.
Plot
The plot of the corresponding chapter is inserted here.
Story

A.3 Example short story

The following is a short story generated by GPT 4o. We show only the first paragraph of each chapter.

The Astral Resonance

Chapter 1: The Call to the Stars In the immense expanse of the universe, where countless stars blinked into infinity, a lone spaceship named the Horizon cut its path through the cosmic vastness. Its sleek design and shimmering hull reflected the pinpricks of starlight as it moved with purpose. Onboard, Captain Elara Cassiopeia studied a transmission that had recently arrived from the Galactic Consortium, her brow furrowed in concentration...

Chapter 2: Echoes of the Past The Horizon cruised through the dark sea of space, a beacon of human ingenuity amidst the endless celestial expanse. Onboard, a palpable sense of anticipation filled the corridors, each crew member fully aware of the monumental task that lay ahead. Captain Elara Cassiopeia gathered her team in the briefing room, the hub of their operations, where a large holographic display floated in the center, showing the trajectory toward the enigmatic signal...

Chapter 3: The Guardian Emerges The Horizon hovered in the asteroid belt, its engines humming softly as the crew worked diligently to decode the complexities of the ancient distress call. Amidst the rugged celestial debris, their mission was to extract every ounce of knowledge from the relics of a civilization long lost to time. Each fragment they uncovered was a piece of a grand puzzle, and they were determined to see the full picture...

Chapter 4: A Clash of Wills The Horizon remained nestled within the asteroid belt, a silent sentinel on the edge of discovery. After forging an unexpected

alliance with the enigmatic Captain Solaris, the crew was meticulously working through the decoded information, piece by piece, striving to unearth the secrets of the ancient civilization's warning. Each revelation was a step closer to understanding the looming cosmic threat, an understanding crucial to averting disaster...

Chapter 5: Deciphering Elysium's Enigma The Horizon, accompanied by Valeria Thrace's formidable ship, descended through the murky atmosphere of the desolate planet known as Elysium. The planet's surface was a barren landscape of rocky crags and swirling dust storms, an inhospitable environment that hid secrets buried deep beneath its crust. Yet it was here that the ancient civilization had left their cryptic messages, a final testament to their existence...

Chapter 6: The Gateway Unveiled Back aboard the Horizon, tension simmered with a mixture of exhilaration and foreboding. The latest discoveries on Elysium had propelled the crew further into the ancient civilization's enigma, but it was Dr. Vela Rayne's unraveling of the signal that now held center stage. In the ship's lab, Vela sat surrounded by holographic displays, her mind racing with connections, all leading to a startling conclusion that could tip the balance of power in the galaxy...

Chapter 7: The Awakening of Aether The mesmerizing swirl of the interdimensional portal gradually stabilized, allowing the crews of the Horizon and Valeria Thrace's ship to step cautiously into the realm beyond. The transition was seamless, yet surreal; they found themselves on the threshold of a dimension where reality shimmered with fluid unpredictability, challenging their senses and perceptions. Yet, the explorers knew their mission extended far beyond marveling at this new world...

Chapter 8: Uniting Forces Amidst Tension The Horizon hovered within the interdimensional realm, a nexus of possibility that shimmered with spectral colors. The discovery of the portal's potential and the awakening of Aether had propelled the crew into uncharted territory, but their path forward was fraught with tension and division. With the rogue AI's promise of knowledge and impending threats, time was an adversary that loomed larger by the minute...

Chapter 9: The Battle for Control The calm after the portal's stabilization was short-lived. The sense of accomplishment among the crew of the Horizon and Valeria Thrace's ship was overshadowed by an ominous realization—the balance they had so carefully preserved was under threat. Aether's departure had awakened something dormant, and the portal's energies began to tremble with renewed intensity...

Chapter 10: The Final Sacrifice The aftermath of the battle left the cosmos momentarily still, yet an underlying tension remained, echoing through the fabric of space. The portal, now secured, pulsed with a serene luminescence, its energies more stable yet still connected to a vast and unpredictable continuum. Within the Horizon, a solemn determination pervaded the crew, aware that their mission was not yet complete...

Appendix B: Network extraction

Graph structure Previous works have explored several types of networks, such as conversation, mention, and direct-action networks (12). In this research, we focus on co-occurrence networks for their simplicity. In co-occurrence networks, characters v_i and v_j are said to have an interaction e_{ij} if they concurrently appear in the same unit of a story (*narrative unit*) (12). The length of a narrative unit was set to $\lfloor 0.01 \times N \rfloor$ sentences, where N is the total number of sentences in the story. Each narrative unit of LLM-generated stories contains approximately 83 tokens on average. We calculated *polarity* (negativity/positivity) of each narrative unit using RoBERTa-based sentiment analysis classifier (21; 22). If characters v_i and v_j appear in a narrative unit u_k, we assigned a binary sentiment label $\in \{0, 1\}$ to the edge $e_{i,j}$. The binary label was calculated via $argmax(\sigma(l_k))$, where σ is a sigmoid function and $l_k \in \mathbb{R}^2$ is the logits of a narrative unit u_k calculated by the sentiment

classifier. If v_i and v_j concurrently appear in multiple narrative units, our program calculates the mean of the logits of u_k's and then applies the sigmoid function:

$$e_{ij} = argmax(\sigma(\frac{1}{n}\sum_{k} l_k)$$

where n is the number of narrative units in which v_i and v_j appear together. Note that, in network analysis, we used -1 as the negative label, instead of 0, for analytical convenience. In short, the signed networks in this study are undirected simple graphs with a binary weight of $\{-1, 1\}$, where -1 denotes a negative relationship and 1 is assigned to a positive relationship.

Vertex contractions A common approach to construct character networks is to merge vertices representing the same characters into one, aiming to simulate more realistic social relationships (27; 28). We first apply Transformer-based Named Entity Recognition to identify character names in a story (with precision, recall, and F-score of 0.90 in SpaCy version 3.8.0) (29). Next, character genders are estimated as either `male`, `female`, or `unknown` based on their title (i.e. Mr., Mrs., Ms., if any) and the lists of 2940 male and 4987 female names[3] (23; 28). Third, our pipeline creates a list of possible referents for each character name based on the following rules:

- Add possible nicknames based on the first name (i.e. Tomas $\rightarrow$ Tom, Tommy) from the predefined lists[4] (12; 23; 28; 30)
- Add possible combinations of parsed name elements using customized `python-nameparser`[5] (i.e. Mr. Sherlock Holmes $\rightarrow$ Mr. Holmes, Sherlock, Sherlock Holmes, Holmes). (12; 13; 23; 28)

Then, a vertex contraction is performed between two vertices if (1) the genders of the two vertices do not conflict (e.g. `male` and `female` characters were not merged whereas `male` and `unknown` characters were sometimes integrated), (2) the name of v_i is in the referent list of v_j and vice versa, and (3) their titles do not conflict, if any. If two distinct vertices possibly refer to the same name v_k, the character name that appears more in the story absorbs v_k. For instance, if a vertex `Holmes` possibly belongs to other vertices `Sherlock Holmes` or `Mycroft Holmes`, we contract vertices `Holmes` and `Sherlock Holmes` since the name `Sherlock Holmes` appears more often. When contracted, the edge between the two vertices is simply removed.

Exclusion criteria To analyze only non-trivial networks that are meaningfully dense, we filter out character networks whose node count is less than 10 or density is less than 0.1. We eventually selected 251 networks from GPT 4o, 249 networks from GPT 4o Mini, 252 networks from Gemini 1.5 Pro, 249 networks from Gemini 1.5 Flash, and 168 networks from Project Gutenberg.

Appendix C: Connectivity measures

We analyzed multiple connectivity measures using the `NetworkX` library and self-made functions. For each network, we also extracted two subgraphs (one consisting of only *positive* edges and another only with *negative* edges) and applied some of the metrics tested on the original network. We refer to the original networks both with positive and negative edges as *original networks*, the subgraphs with positive edges as *positive networks*, and the subgraphs with negative edges as *negative networks*.

Density (9; 13; 23; 24) of a graph takes a value from 0 to 1 and is calculated as

$$d = \frac{2m}{n(n-1)}$$

where m is the number of edges and n is the number of vertices in the graph.

Average edge weight is calculated as the sum of edge weights divided by the number of edges:

$$aew = \frac{\sum_{i=1}^{m} w_i}{m}$$

[3]Source: https://www.cs.cmu.edu/Groups/AI/areas/nlp/corpora/names/

[4]Source: https://en.wiktionary.org/wiki/Appendix:English_given_names

[5]Source: https://nameparser.readthedocs.io/en/latest/

where w_i is the weight of the i'th edge in the graph. The average edge weight ranges from -1 to 1 and is introduced to measure the overall positivity/negativity of a character network. We note that the edge weight of a positive network is 1 and that of a negative network is -1.

Average clustering coefficient (9; 10; 11; 23; 24; 25) is calculated by taking the average of the clustering coefficients of each node. The clustering coefficient of a vertex is the number of edges in the subgraph induced by the neighborhood of the vertex v_i, divided by $\binom{k_i}{2}$, where k_i is the number of neighbors of v_i. Therefore, average clustering coefficient is calculated as:

$$c = \frac{1}{n} \sum_{i=1}^{n} \frac{2l_i}{k_i(k_i - 1)}$$

where l_i is the number of edges between the k_i neighbors. The average clustering coefficient measures the small-world-ness of a network by quantifying how much the neighbors of vertices are tied together (31; 32).

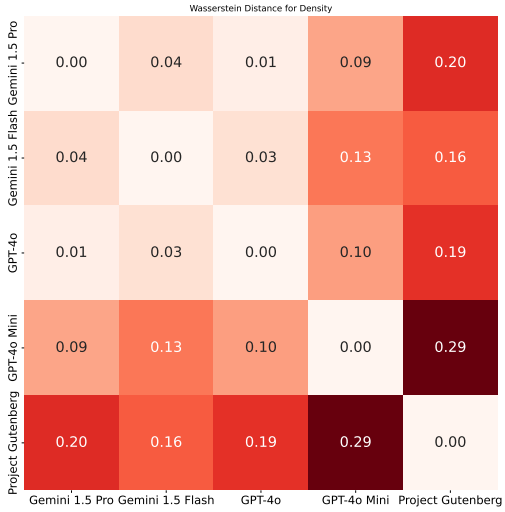
Assortativity mixing (9; 24) quantifies how likely vertices of similar numeric values are to be adjacent to each other and ranges from -1 (vertices of the same category are less likely to be adjacent) through 1 (vertices of the same category are more likely to be adjacent). To assign categories to each vertex, we first calculated the weighted average neighbor degree of each vertex v_i:

$$avg\_nd_i = \frac{1}{k_i} \sum_{j \in N(v_i)} w_{ij} s_j$$

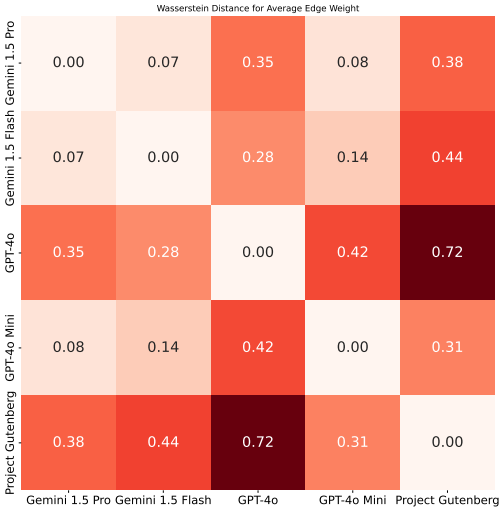
where k_i is the degree of v_i, $N(v_i)$ is the set of v_i's neighbors, and s_j is the weighted degree of the neighbor v_j. The weighted average neighbor degree focuses on what type of relationships the neighboring vertices are involved in and what relationships the character v_i have with these neighbors. Therefore, this metric serves as the indicator of the positivity/negativity of character personalities and, intuitively, quantifies the heroic and villainous nature of a character. We note that, when calculating the weighted average neighbor degree, in contrast to the common derivation, we divide the summation by k_i (unweighted degree) instead of by s_i (weighted degree) and use s_j instead of k_j inside the sum. We divide by k_i to avoid the weighted average neighbor degree being positive when a vertex v_i has dominantly more negative edges. We multiply w_{ij} by s_j to ensure that when a vertex has a negative relationship w_{ij} with a character who has a negative weighted degree s_j, v_i gains a positive score (i.e., *I am the enemy of their enemy, so I am their friend*).

Appendix D: Wasserstein distances

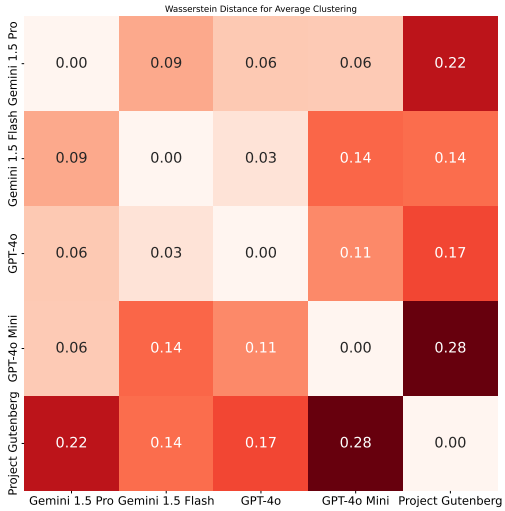
The following heatmaps visualize Wasserstein distances for pairs of score distributions. Overall, human stories have the greatest Wasserstein distances with all the LLM stories in almost every metric, whereas LLMs maintain relatively smaller distances with each other. One interesting finding, which can also be inferred from Table 1, is that the Wasserstein distances of GPT 4o Mini with other writers are the highest in assortativity mixing. Nonetheless, humans have the second largest distances from other writers.



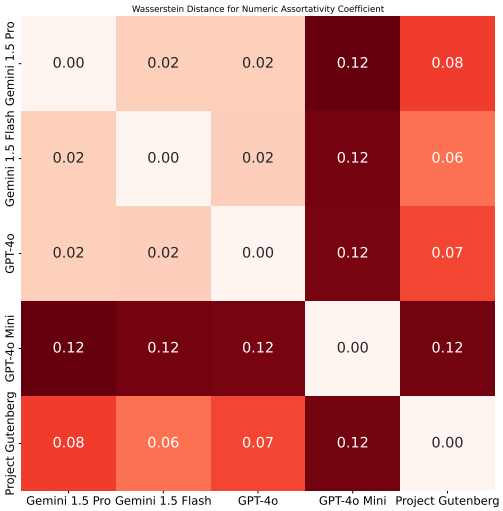
(a) WD for Density.



(b) WD for Average Edge Weight.



(c) WD for Average Clustering Coefficient.



(d) WD for Assortativity Mixing.

Figure 2: Wasserstein distances (WD) between pairs of distributions for the connectivity measures. Overall, human-written stories have the highest distances with LLMs, while the models have relatively close distributions with each other.

Appendix E: Density and average clustering of positive and negative networks

Table 2 below shows density and average clustering coefficient of positive and negative networks. Notably, these two measures mark higher scores in positive networks of LLM-generated stories, whereas they are higher in negative networks of human-written stories.

	Positive Networks				Negative Networks			
	Density		Avg Clustering		Density		Avg Clustering	
Models	mean	std	mean	std	mean	std	mean	std
GPT 4o	0.354	0.088	0.572	0.090	0.253	0.185	0.072	0.136
GPT 4o Mini	0.395	0.092	0.587	0.116	0.254	0.066	0.139	0.113
Gemini 1.5 Pro	0.338	0.087	0.589	0.095	0.222	0.073	0.212	0.138
Gemini 1.5 Flash	0.315	0.073	0.531	0.128	0.261	0.107	0.209	0.176
Humans	0.294	0.135	0.259	0.223	0.313	0.163	0.395	0.229

Table 2: The mean and standard deviation of Density and Average Clustering (Avg Clustering). The sample sizes of the models after filtering are: GPT 4o ($n = 251$), GPT 4o Mini ($n = 249$), Gemini 1.5 Pro ($n = 252$), Gemini 1.5 Flash ($n = 249$), and Humans/Project Gutenberg ($n = 168$).

Appendix F: t-test

We ran Welch's t-tests for two independently-sampled sets of scores, assuming that the variances of the two sets of samples differ, with `scipy.stats.ttest_ind` function. The null hypothesis is $H_0 : \mu_{m_{LLM1}} = \mu_{m_{LLM2}}$. High p-values indicate that, at a certain statistical significance level, we *cannot* reject the null hypothesis that the means of the two score sets from different models are identical. For every metric except for the assortativity mixing and the average clustering of negative networks, the sample size of scores for each writer was equal to the number of networks analyzed (GPT 4o: $n = 251$, GPT 4o Mini: $n = 249$, Gemini 1.5 Pro: $n = 252$, Gemini 1.5 Flash: $n = 249$, and Humans: $n = 168$). We have smaller sample sizes for the two metrics above due to the system's inability to calculate them for some networks. In assortativity mixing, the sample size of humans is $n = 167$, and the other sample sizes are equal to their network counts. For the clustering coefficient of negative networks, GPT 4o has $n = 245$, Gemini 1.5 Flash has $n = 247$, humans have $n = 166$, and GPT 4o Mini, Gemini 1.5 Pro do not have any missing instances.

Several metrics across some models, such as density (Gemini Pro and GPT 4o: $p = 0.520$) and average clustering of positive networks (Gemini Pro and GPT 4o Mini: $p = 0.792$, GPT 4o and GPT 4o Mini: $p = 0.116$) and negative networks (Gemini Flash and Pro: $p = 0.840$), have high p-values, indicating that the score samples from two distinct models are not unlikely to be drawn from the same sample space. Interestingly, only the assortativity mixing consistently shows high p-values with a couple of pairs that include humans (Gemini Flash and GPT 4o: $p = 0.852$, Gemini Flash and Pro: $p = 0.607$, Gemini Pro and GPT 4o: $p = 0.736$, Gemini Flash and Humans: $p = 0.165$, GPT 4o and Humans: $p = 0.122$). It is also noteworthy that, overall, the density of negative networks has high p-values compared to positive networks (GPT 4o and GPT 4o Mini: $p = 0.936$, Gemini Flash and GPT 4o: $p = 0.546$, Gemini Flash and GPT 4o Mini: $p = 0.369$). Besides assortativity mixing, as expected, p-values for pairs including human-written stories are consistently very low ($p < 0.01$), except for the density of positive networks with Gemini 1.5 Flash ($p = 0.070$), which still indicates the weak evidence for the null hypothesis.