INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 11, Issue 2 - V11I2-1273)
Available online at: https://www.ijariit.com

Narrative Intelligence @ Generative AI in Storytelling And Reshaping Creative Writing from Prompt Engineering to Co-Creation

Dr. M.K. Jayanthi Kannan
jayanthi.m@vitbhopal.ac.in
School of Computing Science, Engineering and Artificial Intelligence, VIT Bhopal University, Bhopal

Boda Sai Srujan
boda.sai2020@vitbhopal.ac.in
School of Computing Science, Engineering and Artificial Intelligence, VIT Bhopal University, Bhopal

ABSTRACT

In this research, we introduce a Generative AI-driven storytelling system that uses user prompts to produce a story with logic and without any grammar mistakes. The system uses DistilGPT2, GPT2 transformer-based models, and selects the best output using the BERTScore and coherence score. The selected story is further refined using a grammar correction model based on T5. The final output is well-structured with clear paragraphs and natural dialogues. This approach improves the quality of the story for educational, entertainment, and creative writing. The evolution of Generative AI has opened new avenues in creative writing and content generation. This project, titled "GenAI-driven Storyteller", presents an intelligent system that automatically generates engaging and grammatically correct short stories based on user-provided prompts. The system integrates the strengths of multiple language models—GPT2 and DistilGPT2—to produce diverse narrative outputs. These stories are evaluated using BERTScore for semantic relevance and a coherence scoring mechanism to measure logical flow. A weighted ensemble approach determines the most compelling version among the generated outputs. The selected story is then refined for grammar and fluency using a fine-tuned Google's T5-small model. This hybrid architecture ensures that the final story is not only contextually aligned with the user's prompt but also polished in structure and readability. The project highlights the potential of combining generation, evaluation, and correction models to enhance automated storytelling systems.

Keywords: *Generative AI, GPT2, Hybrid Model, Story Generation, BERTScore, Coherence Scoring, T5, Natural Language Processing*.

INTRODUCTION

Telling stories has long been one of the most natural and meaningful ways for people to pass on ideas, feelings, and life experiences. Whether through spoken word, books, or films, stories help us connect with each other, teach valuable lessons, and spark imagination. In today's online environment, the art of storytelling is taking on new dimensions, thanks to the advancements in Generative AI (Gen-AI). These systems can create full-length stories from just a small user input, making the process quicker and more accessible. Gen-AI storytelling is finding its way into various areas like classrooms, creative writing, gaming, and even mental wellness tools. It helps users come up with new ideas, saves time, and brings creative sup- port to those who may struggle with writing. This technology is not only changing how we tell stories—it's also opening the door for more people to become storytellers themselves. However, even though models like GPT2 and other Gen-AI tools are powerful, they still have some problems.

Many times, the stories they create are too short, repeat the same lines, or lose track of the plot. The flow can feel off, and some stories don't have a proper beginning, middle, or end, which makes them feel unfinished or confusing. Grammatical errors can sneak in, which can make the story tougher to follow. Most of these systems depend on just one model and don't have a way to choose the best version when multiple options are available, which limits the overall quality and flexibility of the output. To fix these problems, our paper introduces a new storytelling system that doesn't rely on just one model. Instead, it takes the best parts from two different language models—GPT2 and DistilGPT2—and compares the stories they generate. To choose the best story, it uses BERTScore (which checks how closely the story matches the user prompt) and coherence scoring (which checks if the story makes sense and flows well). The chosen story is then passed through a T5-small model that corrects grammar and improves the overall quality.

This system uses a multi-model evaluation approach, where outputs from more than one language model are generated and compared using automated scoring methods.
The best result is then selected, corrected for grammar, and formatted into a full story. The final story is organized into clear paragraphs and follows a natural flow with an engaging beginning, a strong middle, and a satisfying end. By combining generation, evaluation, and correction in one flow, this system helps create better and more meaningful stories. It works without needing manual editing and can be useful in many areas—from helping kids read creative stories, to supporting writers, or building fun educational tools. This paper is structured as follows: Section II reviews past work related to AI-based story generation and language models. Section III explains the proposed system, including the models used and the evaluation methods applied. Section IV presents the experiments, results, and analysis of story outputs. To wrap things up, Section V offers a summary of the study and suggests ways to enhance the system moving forward.

RELATED WORKS AND DOMAIN ANALYSIS

Automatic story generation has seen significant advancements with the rise of large language models (LLMs) and transformer-based architectures. Various researchers have explored different methods to improve creativity, coherence, and control in AI-generated narratives.

Bandara et al. (2022) introduced a children's story generator in Sinhala using trans-former models, showing that even low-resource languages can benefit from pre-trained models when fine-tuned appropriately for narrative generation [1]. Similarly, Bensaid et al. (2021) used GPT-2 fine-tuning for story generation in low-resource languages, emphasizing the adaptability of generative models to different linguistic settings [2]. Chalvatzaki et al. (2023) conducted a detailed analysis of the challenges in generating long-form stories using large language models, focusing on issues such as coherence drift, narrative inconsistency, and hallucination. Their work highlighted the importance of model alignment and memory mechanisms in improving long story generation [3]. In a different approach, Ammanabrolu et al. (2021) proposed a "story realization" model that expands abstract plot events into detailed story sentences, aiming to bridge the gap between plot planning and natural language realization. Several methods have also been proposed to make story generation more structured and user-controllable [4].

Yao et al. (2019) introduced the" Plan-And-Write" framework, where a storyline is first generated and then converted into a complete narrative, improving story consistency and plot progression [5]. Clark and Smith (2021) focused on inter-active fiction using a neural model that supports branching narratives, enabling users to influence story outcomes like in "Choose Your Own Adventure" books [6]. Keskar et al. (2019) developed the CTRL model—a conditional transformer language model that allows controllable generation through control codes. This model made it easier to guide the generated content according to desired themes or genres [7]. Guan and Huang (2020) provided a comprehensive survey on incorporating external knowledge into text generation models, underlining the importance of knowledge-aware storytelling for producing richer and more informative narratives [8]. Liu et al. (2020) used story intention graphs to guide generation, which helped maintain narrative purpose and coherence [9]. Meanwhile, Wang et al. (2022) proposed sentence-level co- herence evaluation using semantic similarity to assess and improve story flow—a technique relevant to our project [10]. Santhanam et al. (2021) introduced PLUG, a prompt-learning technique that steers the generation based on minimal exam- ples or keywords, making it a lightweight method to control narrative direction [11].

PROPOSED METHODOLOGY

The goal of our proposed work is to create a user-friendly and intelligent storytelling system powered by Generative AI. Instead of relying on a single model to generate stories, we use a multi-model evaluation approach to improve story quality, coherence, and grammar. The system starts with a simple prompt from the user and processes it through different stages to generate a meaningful, well-structured story.

The process begins when a user enters a prompt that sets the theme or opening of a story. This input is passed to two powerful pre-trained language models: GPT2 and DistilGPT2. Both models generate their own versions of the story based on the prompt. To pick the best one, we use an automatic scoring mechanism.

We apply BERTScore, which checks how closely the story follows the meaning of the prompt, and coherence scoring, which uses Sentence-BERT to check if the story flows logically from sentence to sentence. A final weighted score decides which story version is more appropriate. The selected story is then passed through a T5-small grammar correction model to fix any grammatical mistakes and improve overall fluency. Finally, the story is formatted into neat paragraphs using sentence tokenization so that the output is easy to read. Optional features like clearing the story flow and downloading the generated story as a .txt file make the system more practical for users. This method ensures we deliver complete, readable, and creative stories with minimal manual effort.

A. Dataset

We used the "GPT-Writing Prompts" dataset, originally collected from Reddit's Writing-prompts community and later published on Hugging Face. The dataset contains over 300,000 story samples, each paired with a creative prompt and a human-written continuation. This dataset is specifically de- signed for training and evaluating story generation models. It pairs thousands of creative writing prompts with human- written story continuations, making it a valuable resource for studying and developing AI-driven storytelling systems.

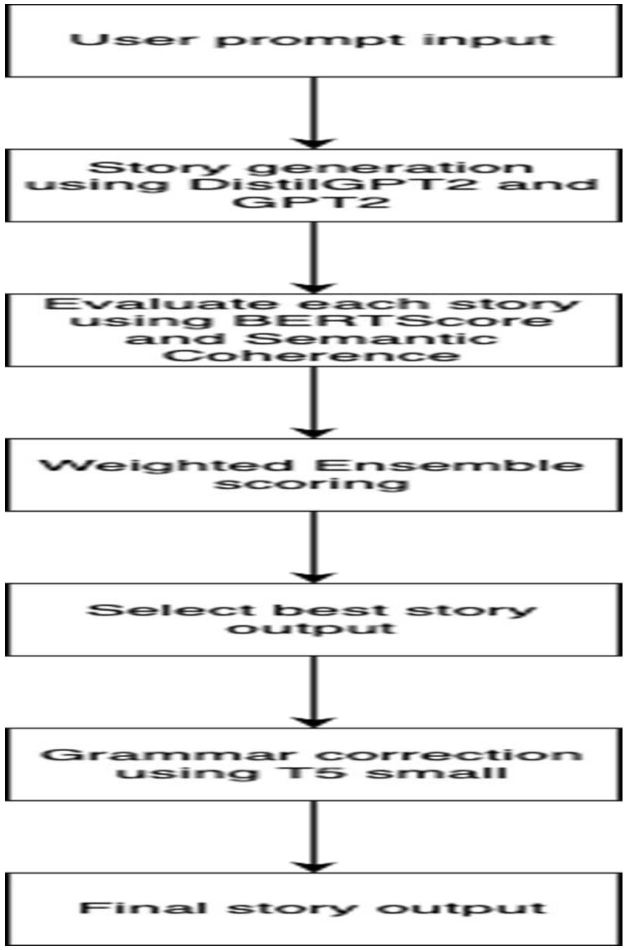

Fig. 1. Flowchart of Methodology Narrative Intelligence

A. Data Preprocessing

The dataset used in this study was stored in a com- pressed .csv.gzip format and contained over 27,000 rows. It was specifically designed for creative storytelling tasks, featuring a wide range of user-written prompts and their corresponding story completions. To begin, only the essen- tial columns—p_text for prompts and s_text for sto- ries—were extracted, while any rows with missing or null values were removed to maintain data quality. The columns were then renamed to prompt and story for consistency and clarity during processing. Due to hardware and time constraints, we chose to work with a sample of 5,000 rows instead of using the entire dataset. This sample size provided a manageable yet diverse set of examples for training and testing. The data was then converted into a format suitable for further processing by the models.

B. Tokenization

We used specific tokenizers for each pre-trained language model—GPT2, DistilGPT2, and T5—to handle the tokenization process. Making sure we turned the input text into numerical token IDs that these models can work with. Each prompt and story got tokenized on its own, and to keep everything neat, we added padding to the shorter sequences and trimmed down the longer ones, making sure everything was capped at a consistent length of 512 tokens. This standard setup made it easier for the models to work together during training and helped avoid any mismatches in input sizes, leading to more efficient and accurate learning for all the language models involved in the system.

C. Model Implementation

We created a Gen AI-based storytelling system by blending several pre-trained language models with modules for evaluation and grammar correction. Our main aim was to establish a smooth process that can produce, assess, and polish story outputs based on what users ask for. To kick things off, we brought in two popular generative models, GPT2 and Distil- GPT2, both of which we accessed through the Hugging Face transformers library. We chose these models because they're great at generating natural, imaginative text. When someone types in a prompt, each model generates story outputs on its own. Next, we evaluated these generated stories automatically using two methods: BERTScore (F1) and semantic coherence scoring. BERTScore looks at how well the generated text matches the user's prompt on a deeper meaning level, while coherence scoring (with sentence-BERT embeddings) ensures that the story flows nicely from one sentence to the next. We combine these scores using a weighted system and go with the output that performs the best. To polish the grammar of the selected story, we run it through a T5-small grammar correction model, which has been fine-tuned by Vennify.

This step makes sure that the final output is grammatically correct and easy to understand. At the end of the process, we organize the corrected story into well-structured paragraphs using NLTK sentence Tokenization, which boosts readability and gives the story a more refined feel. The whole system is wrapped up in a user-friendly interface where users can enter prompts, generate stories, clear previous content, and download the final story as a .txt file.

D. Fine -Tuning

To improve the quality and relevance of the story generation process, we fine-tuned the GPT2 and DistilGPT2 models on a custom dataset specifically curated for narrative genera- tion. The dataset, originally sourced from the Hugging Face repository (gpt_info.csv.gzip), contains thousands of prompt-to-story pairs designed for creative writing tasks. We fine-tuned the models to help them get a better grasp of storytelling elements like character arcs, plot development, and emotional shifts. With this targeted training on a special dataset, the models improved their ability to generate stories that are more coherent, engaging, and in line with the prompts we provided. We used the PyTorch framework along with the Hugging Face Trainer API to carry out the training. For the dataset, we tokenized it using the appropriate tokenizers for GPT2 and DistilGPT2, making sure to apply padding and truncation for a uniform approach. We also introduced energetic masking and a learning rate scheduler to keep the training process stable. The training took place on a local machine equipped with an NVIDIA GPU, using a batch size of 4 and a learning rate of 5e-5, and spanned 3 epochs. We kept an eye on evaluation loss throughout to avoid overfitting and to make sure that the models generalize well to new prompts. This fine-tuning really boosted the generative models' performance, making them much more responsive to what users were looking for and better at weaving together well-structured stories.

NARRATIVE INTELLIGENCE RESULT A N A L Y S I S AND EFFICIENCY

To evaluate the effectiveness of the proposed Gen AI-driven storytelling system, a series of experiments were conducted using a custom dataset containing approximately 5000 story prompts and corresponding human-written stories from the original dataset which consist of around 27,000 samples. Before training, the dataset underwent essential preprocessing steps, including removal of null entries, renaming of columns for clarity, and sampling. The processed data was then tokenized using the GPT2 tokenizer and formatted using the Hugging Face Datasets library. This ensured compatibility with language models and allowed for consistent training input. For story generation, we used two Pre-trained models—GPT2 and DistilGPT2—each known for their fluency and context retention. These models generated candidate stories based on user prompts. To ensure output quality, each candidate story was evaluated using two metrics: BERTScore (F1) to measure semantic similarity with the prompt, and Coherence Score to ensure logical sentence flow.A weighted scoring method was used to select the best output. The final score was computed using the formula:

$$\textbf{Combined Score} = w_b \times \textbf{BERTScore} + w_c \times \textbf{Coherence Score Where: } w_b = 0.6 \quad \textbf{and} \quad w_c = 0.4$$

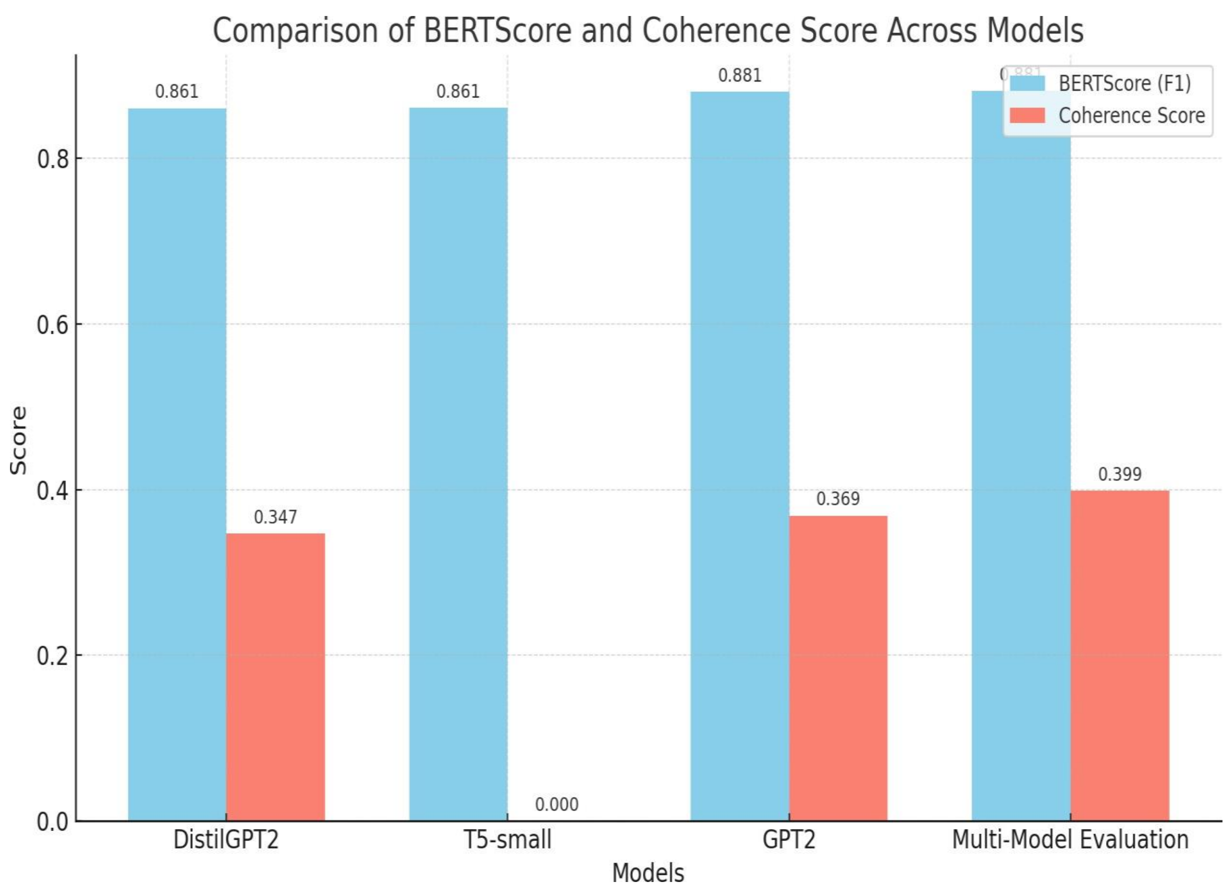

Figure. 2. Performance Comparison of Models Narrative Intelligence

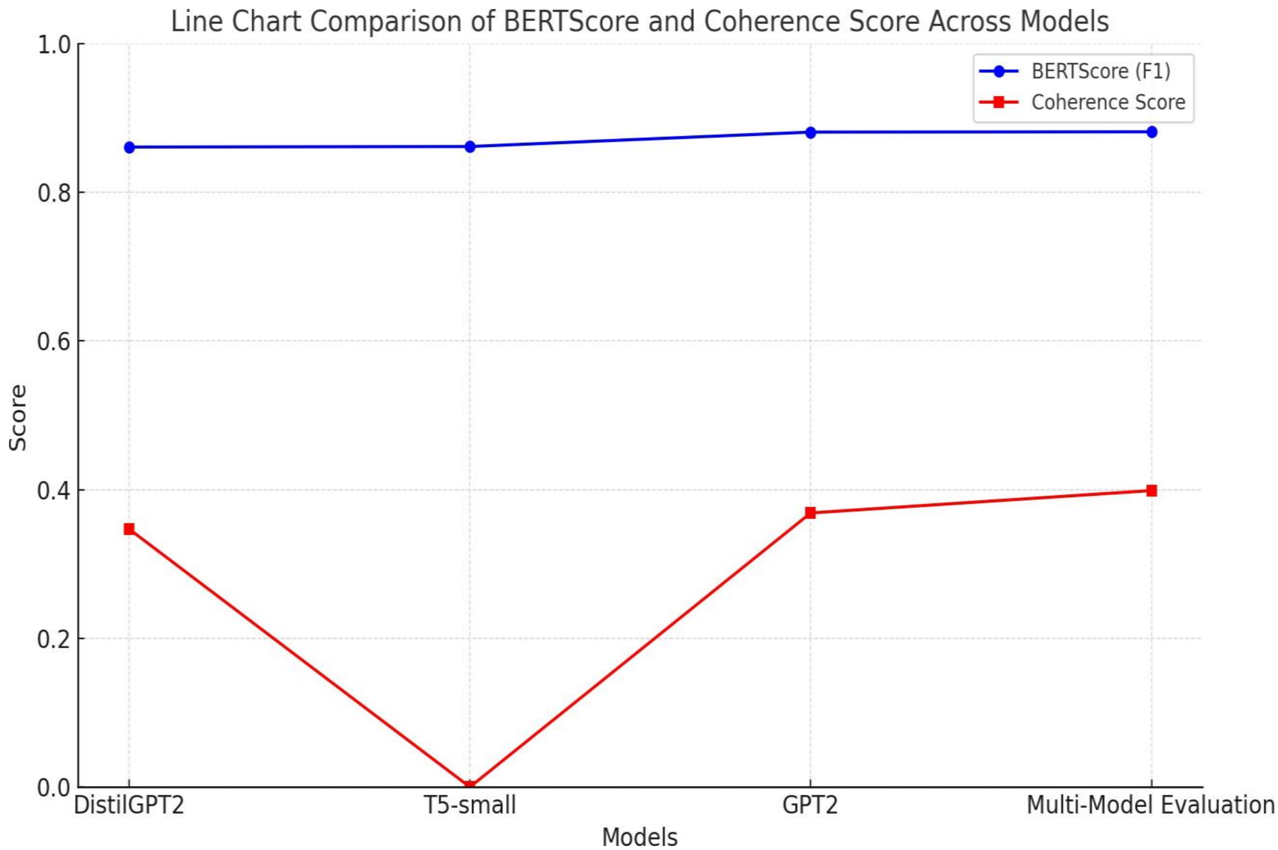

Figure 3. Comparison of BERTScore & Coherence Score Across Models Narrative Intelligence

The selected story was passed through a T5-small model for grammar correction. This step improved sentence fluency and fixed grammatical errors. The corrected story was then formatted into structured paragraphs for better readability. The models were evaluated across multiple prompts. GPT2 consistently produced more creative and structured narratives compared to DistilGPT2 and T5-small (which performed best as a grammar corrector rather than generator). By using a multi-model evaluation approach and scoring techniques to choose from the outputs of all the models, we ended up with stories that were not just higher in quality but also more engaging. You can see Table IV.1 for the BERTScore and coherence ratings for each model. To help illustrate how these models stack up against each other, Fig. IV.1 and Fig. IV.2 offer a bar graph and a line plot, respectively, showing performance trends and emphasizing the success of our combined evaluation method. The evaluation results highlight the strengths and limitations of each individual model and the proposed hybrid ensemble approach: GPT2 consistently outperformed DistilGPT2 in both BERTScore and coherence metrics. Its larger architecture and training capacity make it better suited for generating more coherent and contextually rich stories. DistilGPT2, being a smaller and faster version of GPT2, produced grammatically acceptable outputs but often lacked depth and narrative flow compared to GPT2. The hybrid ensemble model, which combines both BERTScore and semantic coherence for output selection, showed improved overall performance by intelligently selecting the most contextually appropriate and fluent story from multiple generated candidates. The grammar correction stage using a fine-tuned T5 model further enhanced readability and corrected minor syntactic errors, improving the final output's presentation quality. Despite good results, repetitiveness and limited creativity were observed occasionally, particularly in longer generations. This highlights the limitations of autoregressive models when not fine-tuned extensively for structure and narrative flow. The integration of semantic coherence scoring played a crucial role in mitigating incoherence and ensuring sentence-to-sentence flow, especially in GPT2 outputs. Overall, the results validate the effectiveness of the proposed hybrid approach for generating human-like, grammatically correct, and coherent short stories from user prompts. The system balances fluency and creativity while offering modularity for future improvements.

CONTRIBUTION AND USABILITY TEST

This research presents a Generative AI-based storytelling system that uses multiple language models and evaluation techniques to create better, more coherent stories. The system combines outputs from both GPT2 and DistilGPT2, and selects the best one using automatic evaluation methods like BERTScore and semantic coherence scoring. To improve readability and grammar, the selected story is further processed using a T5-small grammar correction model. The system is designed to be interactive, allowing users to generate stories based on their prompts in a natural, ongoing flow. It uses the GPT-WritingPrompts dataset, which was preprocessed and tokenized to better fit storytelling tasks. A detailed analysis comparing the individual models and the ensemble approach showed that the multi-model evaluation strategy improves overall story quality. This project also successfully integrates popular NLP tools and libraries such as Hugging Face Trans- formers, BERTScore, Sentence-BERT, HappyTransformer, and NLTK for smooth implementation. While the system produces good-quality stories, there is still room to grow.

Future im- improvements could involve training the models to generate con- tent based on specific genres like horror, romance, or science fiction.

TABLE I
PERFORMANCE OF INDIVIDUAL MODELS

Model	**BERTScore (F1)**	**Coherence Score**	**uality Observation**
DistilGPT2	0.84–0.88	0.34–0.45	enerates relatively ort and compact stories, sometimes lacking depth.
T5-small	0.81–0.86	0.00–0.30	ften produces in- omplete or template-like text, best suited for grammar rrection rather than story generation.
GPT2	0.85–0.89	0.55–0.61	enerates creative ories with better dialogue and logical flow.

Another possible direction is to build an integrated architecture that unifies both models at the structural level for better story flow. Human feedback could also be added to improve creativity and emotional impact. Additional features like plot control, character customization, and dialogue style preferences would enhance personalization.

CONCLUSION

The GenAI-driven storytelling system effectively combines multiple natural language generation models and evaluation techniques to produce high-quality, coherent stories. The main contributions of this work is Hybrid Model Design, which Developed a hybrid ensemble approach combining outputs from DistilGPT2 and GPT2 using BERTScore and semantic coherence to select the best generation. Grammar Correction Module: Integrated a grammar correction step using a T5-based model, enhancing the readability and grammatical accuracy of generated stories. Automatic Evaluation Metrics: Incorporated BERTScore (semantic similarity) and coherence scoring (sentence-level semantic flow) for objective output ranking and improved quality control. Interactive Storytelling Support: Enabled continuous story generation based on user prompts, simulating real-time creative writing with logical flow. Dataset Utilization: Utilized the GPT-Writing Prompts dataset and conducted preprocessing, tokenization, and fine-tuning experiments to align the system with storytelling needs. Comparative Analysis: Performed quantitative and qualitative evaluations of individual models (GPT2, DistilGPT2) and the ensemble system, supported by metrics and visualizations. Tool Integration: Employed advanced NLP libraries and tools, including Hugging Face Transformers, Happy Transformer, BERTScore, Sentence-BERT, and NLTK for system implementation.

REFERENCES

[1] Bandara, K., et al. (2022). ”AI Children Story Generator in Sinhala Using Transformers.” General Sir John Kotelawala Defence University.

[2] Ammanabrolu, P., et al. (2021). ”Story Realization: Expanding Plot Events into Sentences.”

[3] https://huggingface.co/datasets/vkpriya/GPT-WritingPrompts/viewer

[4] Balajee RM, Jayanthi Kannan MK, Murali Mohan V., "Image-Based Authentication Security Improvement by Randomized Selection Approach," in Inventive Computation and Information Technologies, Springer, Singapore, 2022, pp. 61-71

[5] Bensaid, M., et al. (2021). ”Story Generation with Fine-Tuned GPT-2 for Low-Resource Languages.” Proceedings of the International Conference on Artificial Intelligence.

[6] B. R M, S. Kallam and M. K. Jayanthi Kannan, "Network Intrusion Classifier with Optimized Clustering Algorithm for the Efficient Classification," 2024 5th International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 2024, pp. 439-446, doi: 10.1109/ICICV62344.2024.00075.

[7] See, A., et al. (2019). ”Massively Multilingual Neural Story Generation.”

[8] M. K. Jayanthi, "Strategic Planning for Information Security -DID Mechanism to befriend the Cyber Criminals to assure Cyber Freedom," 2017 2nd International Conference on Anti-Cyber Crimes (ICACC), Abha, Saudi Arabia, 2017, pp. 142-147, doi: 10.1109/Anti-Cybercrime.2017.7905280.

[9] Fan, A., et al. (2018). ”Hierarchical Neural Story Generation.” In Proceedings of the Association for Computational Linguistics (ACL).

[10] Kavitha, E., Tamilarasan, R., Baladhandapani, A., Kannan, M.K.J. (2022). A novel soft clustering approach for gene expression

data. Computer Systems Science and Engineering, 43(3), 871-886. https://doi.org/10.32604/csse.2022.021215

[11] Reimers, N., and Gurevich, I. (2019). ” Sentence-BERT: Sentence Em- Embeddings using Siamese BERT-Networks.” In Proceedings of EMNLP.

[12] G., D. K., Singh, M. K., & Jayanthi, M. (Eds.). (2016). Network Security Attacks and Countermeasures. IGI Global. https://doi.org/10.4018/978-1-4666-8761-5

[13] Raffel, C., et al. (2020). ” Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5).” Journal of Machine Learning Research.

[14] R M, B.; M K, J.K. Intrusion Detection on AWS Cloud through Hybrid Deep Learning Algorithm. Electronics 2023, 12, 1423. https://doi.org/10.3390/electronics12061423

[15] Brown, T. B., et al. (2020). ” Language Models are Few-Shot Learn- ers (GPT-3).” Advances in Neural Information Processing Systems (NeurIPS).

[16] Roemmele, M., and Gordon, A. S. (2018). ”An Encoder-Decoder Approach to Predicting Causal Relations in Stories.” In Proceedings of the NAACL Workshop on Storytelling.

[17] Naik, Harish and Kannan, M K Jayanthi, A Survey on Protecting Confidential Data over Distributed Storage in Cloud (December 1, 2020). Available at SSRN: https://ssrn.com/abstract=3740465 or http://dx.doi.org/10.2139/ssrn.3740465

[18] Radford, A., et al. (2019). ”Language Models are Unsupervised Multi- task Learners (GPT-2).” OpenAI.

[19] Kavitha, E., Tamilarasan, R., Poonguzhali, N., Kannan, M.K.J. (2022). Clustering gene expression data through modified agglomerative M-CURE hierarchical algorithm. Computer Systems Science and Engineering, 41(3), 1027-141. https://doi.org/10.32604/csse.2022.020634

[20] Kumar, K.L.S., Kannan, M.K.J. (2024). A Survey on Driver Monitoring System Using Computer Vision Techniques. In: Hassanien, A.E., Anand, S., Jaiswal, A., Kumar, P. (eds) Innovative Computing and Communications. ICICC 2024. Lecture Notes in Networks and Systems, vol 1021. Springer, Singapore. https://doi.org/10.1007/978-981-97-3591-4_21

[21] Santhanam, S., et al. (2021). ”PLUG: Prompt Learning for Story Generation.”

[22] M. K. J. Kannan, "A bird's eye view of Cyber Crimes and Free and Open Source Software's to Detoxify Cyber Crime Attacks - an End User Perspective," 2017 2nd International Conference on Anti-Cyber Crimes (ICACC), Abha, Saudi Arabia, 2017, pp. 232-237, doi: 10.1109/Anti-Cybercrime.2017.7905297.

[23] P. Jain, I. Rajvaidya, K. K. Sah and J. Kannan, "Machine Learning Techniques for Malware Detection- a Research Review," 2022 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), BHOPAL, India, 2022, pp. 1-6, doi: 10.1109/SCEECS54111.2022.9740918.

[24] Zhang, T., et al. (2019). ”BERTScore: Evaluating Text Generation with BERT.”

[25] B. R. M, M. M. V, and J. K. M. K, "Performance Analysis of Bag of Password Authentication using Python, Java and PHP Implementation," 2021 6th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2021, pp. 1032-1039, doi: 10.1109/ICCES51350.2021.9489233.

[26] Wang, L., et al. (2022). ”Evaluating Coherence in Story Generation Using Sentence Similarity.”

[27] Dr.M.K. Jayanthi and Sree Dharinya, V., (2013), Effective Retrieval of Text and Media Learning Objects using Automatic Annotation, World Applied Sciences Journal, Vol. 27 No.1, 2013, © IDOSI Publications,2013, DOI: 10.5829/idosi.wasj.2013.27.01.1614, pp.123-129. https://www.idosi.org/wasj/wasj27(1)13/20.pdf

[28] Dr. Naila Aaijaz, Dr. K. Grace Mani, Dr. M. K. Jayanthi Kannan and Dr. Veena Tewari (Feb 2025), The Future of Innovation and Technology in Education: Trends and Opportunities, ASIN : B0DW334PR9, S&M Publications, Mangalore, Haridwar, India-247667, ISBN-13 : 978-8198488824,https://www.amazon.in/gp/product/B0DW334PR9/ref=ox_sc_act_title_1?smid=A2DVPTOROMUBNE&psc=1#detailBullets_feature_div

[29] Keskar, N. S., et al. (2019). ”CTRL: A Conditional Transformer Language Model for Controllable Generation.”

[30] Python for Data Analytics: Practical Techniques and Applications, Dr. Surendra Kumar Shukla, Dr. Upendra Dwivedi, Dr. M K Jayanthi Kannan, Chalamalasetty Sarvani ISBN: 978-93-6226-727-6, ASIN : B0DMJY4X9N, JSR Publications, 23 October 2024, https://www.amazon.in/gp/product/B0DMJY4X9N/ref=ox_sc_act_title_1?smid=A29XE7SVTY6MCQ&psc=1

[31] Guan, J., and Huang, M. (2020). ”A Survey of Knowledge-Enhanced Text Generation.”

[32] Harish Naik and M K Jayanthi Kannan, A Research on Various Security Aware Mechanisms in Multi-Cloud Environment for Improving Data Security, ISBN:979-8-3503-4745-6, DOI: 10.1109/ICDCECE57866.2023.10151135, 2nd IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics ICDCECE 2023, https://ieeexplore.ieee.org/document/10151135

[33] Yao, L., et al. (2019). ”Plan-And-Write: Towards Better Automatic Storytelling.” In Proceedings of the AAAI Conference on Artificial Intelligence.

[34] Harish Naik Bheemanaik Manjyanaik, Rajanikanta, Jayanthi Mangayarkarasi Kannan, Preserving Confidential Data Using Improved Rivest-Shamir Adleman to Secure Multi-Cloud, International Journal of Intelligent Engineering and Systems, Vol.17, No.4, 2024 pp .162-171, DOI: 10.22266/ijies2024.0831.13, https://inass.org/wp-content/uploads/2024/02/2024083113-2.pdf,

[35] Zhou, H., et al. (2020). ”The Design and Implementation of XiaoIce, an Empathetic Social Chatbot.” Computational Linguistics, vol. 46, no. 1, pp. 53–93.