Moral-Driven Planning in Simple Automated Narrative Generator

Sanjay Garg
Indrashil University
gargsv@gmail.com

Purvish Khalpada
Nirma University
16ftphde19@nirmauni.ac.in

***Abstract*— A moral is a vital part of the story and is vital for both the author and the audience. However, moral-based story planning is relatively underexplored. This paper proposes a planning-based Simple Automated Narrative Generator framework that explicitly considers the moral input and plans the story events around it. To achieve that, we determine the character's emotional arc based on the polarity of the input moral and then plan the events that adhere to the emotional arc. In the experiment of 103 human evaluators and 35 generated narratives for 7 morals, we find encouraging normalized confidence of 0.86 for intended moral being conveyed by the narratives and reasonable confidence of 0.71 for selecting the narratives with morals versus without morals.**

Keywords—computational storytelling, narrative generation, computational narratology, story planning, automatic story generation, moral based narrative generation, moral driven planning, moral of the story

I. Introduction

Storytelling is an integral part of human civilization (Herman, 2013). The concept of storytelling is as old as the cave paintings, even older than language development itself. Storytelling, being with humanity for so long, has affected the evolution of the human brain to the extent that neuroscientists and evolutionary biologists believe that storytelling is what makes us human (Gottschall, 2012). With its roots so deep, storytelling can influence a person, causing the metamorphosis of Mister Gandhi to Gandhi, for example, and the society and the whole generation. The part of the story that makes the most impact is the story's message or moral (Turner, 1993).

Although computational storytelling has been a field of research for some time, the significant contributions have been towards generating and identifying the set of coherent events, addressing the creativity problem, and how the interaction of the audience can affect the development of the story. Explicitly incorporating morals in computationally generated narratives has been an underexposed element of computational storytelling. So, this paper proposes a moral-driven planner to lead event planning in Simple Automated Narrative Generator (SANG), a framework based on (Khalpada & Garg, 2021).

One factor that makes planning a story is the breadth of the exploration space of the possible events at each planning turn. To constrain the exploration, SANG uses the structure of an emotional arc, which depends on the polarity of the moral input. SANG uses a Directed Causal Graph (DCG) similar to (Khalpada & Garg, 2019) and a knowledge base to plan the events around the determined emotional arc. This flow is abstractly depicted in fig. 1.

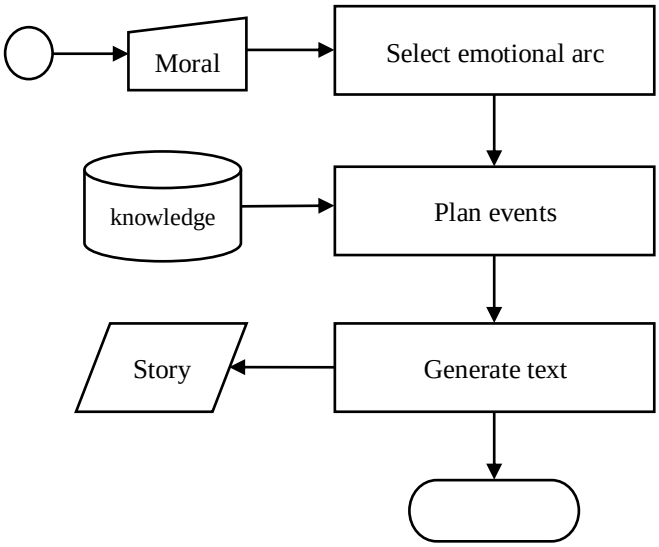


fig. 1 System overview

SANG takes moral input in higher-level language similar to predicate logic, with at least two atomic formulas connected with a causal relation. The last atomic formula determines the polarity of the moral. For example, the input $forgive(PersonX, PersonY) \rightarrow reward(PersonX)$ is encouraging the forgiving nature. On the other hand, the input $steal(PersonX, PersonY.car) \rightarrow punish(PersonX)$ discourages the theft. fig. 2 contains an example story for the moral 'be forgiving'.

Section V discusses the story generation and analysis in section VI. Before that discussion, Section II discusses the prevalent narrative generation techniques and why planning is relevant, more than ever, even with the rise of end-to-end narrative generative models. Section III explores plot structure and emotional arc, and section IV discusses character modelling. Finally, section VII concludes the paper with thoughts on future work.

II. Story Generation Strategies

Even with the advancement in artificial intelligence and natural language processing, narrative generation is still an open task. One of the earliest approaches was developed by Lakoff (Lakoff, 1972), who used rewrite rules to create a grammar of Russian folk tales based on Propp's (Propp, 2010) findings. The story grammar consists of expandable rules coded by the author in the system. In the story grammar, the non-terminals consisted of the chunk of text representing a respective story fragment, with variables for attributes like character names. As it was computationally less expensive, and it could incorporate a story structure, many approaches like (Bringsjord & Ferrucci, 2000)(Machado et al., 2001)(Spierling et al., 2002)(Onuczko et al., 2008)(Pemberton, 1989)(Kybartas & Verbrugge, 2014) used story grammar as a foundation to generate stories. Because the selection of expansion rules was random, and there was hardly any contextual information passing, often the stories turned out to be very incoherent (Black & Wilensky, 1979). Hence, the researchers considered generating stories as problem-solving, to an extent that Meehan (Meehan, 1977) argued that storytelling can be more, but it is at-least problem-solving.

Input: $\text{forgive}(x, y) \rightarrow \text{reward}(x)$

Output story: Once, there lived two kings Josh and Jacob, with neighbouring kingdoms. The kingdom ruled by Josh was one of the most prosperous kingdoms. So, Jacob envied Josh very much. He used to create troubles for Josh. One day, being tired of him, Josh declared war against him. Rich army of Josh badly defeated Jacob and Josh disarmed him. Josh had an opportunity to kill his troublemaker. However, he did not make the kill as Jacob apologised to him. He forgave Jacob, and they returned to their kingdom.
After a few years, a new bloodthirsty king, Lian, rose to power in a far foreign land. Lian was very greedy and wanted the prosperous kingdom of Josh. He was mighty and could beat Josh. So, he attacked Josh. Lian was almost winning the war. But Jacob came to help Josh. Lian could not defeat Josh and Jacob together. They together defeated Lian. Josh was delighted to get help from Jacob. He understood how forgiving Jacob had saved his life and his kingdom. He decided to be more forgiving towards others' mistakes.

fig. 2 Generated story for the moral 'be forgiving'

Translating the narrative generation into a problem-solving task requires an initial story world state, a goal state, and a knowledge base of events that could transform one state into another. Because the narrative generator had a goal state input from the human author and used a knowledge base crafted by the human author, the narratives are usually coherent and sensical. It is one of the most widely used techniques. Approaches like (Meehan, 1977)(Turner, 1993)(Skorupski & Mateas, 2010)(Gervás et al., 2005)(Lebowitz, 1985)(Pérez y Pérez, 1999)(Bae & Young, 2014)(Barros & Musse, 2008)(Charles et al., 2011)(Riedl & León, 2008)(Peinado et al., 2004) use problem-solving. As the selection of events primarily relied on the goal state and current state, the resultant stories often had inconsistent characters. Hence, researchers considered using autonomous agents as story characters and building stories from their actions.

Apart from proposing problem-solving, Meehan (Meehan, 1977) was one of the earliest to use agents to build story generation. With a rise in the research of autonomous agents in artificial intelligence and the better applicability of agent-based story generation in interactive environments like games, the story generation based on autonomous agents became popular soon. Approaches like (Meehan, 1977)(Theune et al., 2003)(Fairclough & Cunningham, 2003)(Cai et al., 2010)(Shim & Kim, 2002) used autonomous agents. Often the approaches based on autonomous agents assign a set of goals to the agents and use planning to build the series of actions to achieve the goal, requiring a similarly extensive knowledge base and high computational complexity.

One way to dodge those requirements is by searching in the corpus of stories with techniques like case-based reasoning. However, that often requires exhaustive conditioning on cases to avoid nonsensical generalization and adaptions. Due to advances in neural networks and machine learning, another approach is training and using a neural network to model narrative generation. One of the most intuitive networks for the task would be an already established sequence-to-sequence (seq2seq) neural network(Sutskever et al., 2014). It has attained a significant improvement in multiple language generation fields, like the generation of Wikipedia articles (P. J. Liu et al., 2018), poems (Zhang & Lapata, 2014), including generating narratives from a set of incoherent sentences (Jain et al., 2017). However, it is learning a language model, and it does not differentiate between story content and representation, unlike Chatman's taxonomy (Chatman, 1980). Hence, it has inherent difficulty in capturing the foundational characteristics (D. Liu et al., 2020) like character personalities (Bahamón & Young, 2017), story conflicts (Ware & Young, 2014), or action intents (Riedl & Young, 2010) like their planning counterparts. Also, as it does not keep track of the story world or pass contextual information, most of the stories become incoherent (Fan et al., 2018), have an unexplainable and irrelevant set of events (Yao et al., 2019).

To partially overcome these difficulties, Jian et al. (Jain et al., 2017) used incremental encoding and common-sense graph to generate a logical ending to a story of a few sentences. This strategy takes the previous set of events of the existing story as a context and predicts the next event. Although it only generated a single event, the idea of considering the set of events to make stories more consistent and coherent was intriguing. Later, Fan et al. (Fan et al., 2018) proposed to use a prompt to take a set of words or sentences as an input and generate stories in the context of the prompt. It demonstrated that having a set of plot points increases the probability of the narratives adhering to the original intention. Yao et al. (Yao et al., 2019) went further. They planned a storyline from the given title and used it to generate the story. This strategy provided even better coherence, as the events in the storyline shared context. Lara et al. (Martin et al., 2018) used a similar concept of adopting and integrating the event planning with the seq2seq based model, although they used network instead of planning. They proposed using one seq2seq based recurrent multi-layer encoder-decoder network Event2Event that learns and predicts the events and using another neural network Event2Sentence that generated natural language text for the event generated by Event2Event. These

approaches addressed the event dependency and coherence, while Liu et al. (D. Liu et al., 2020) addressed the issue of character consistency by learning character embeddings like (Oraby et al., 2019). Like Lara et al. (Martin et al., 2018), they built story generation in two phases, action selection, constrained by the learned character embedding, and sentence generation. However, they still suffer from repetition, inconsistencies, coherence problems, and the resultant stories usually do not have a structure and do not make sense as a whole story. Guan et al. (Guan et al., 2020) speculate that this is mainly because of a lack of common-sense knowledge, understanding of causal and temporal relationships between events. They proposed an extended GPT-2 based model that incorporates common-sense knowledge from ATOMIC (Sap et al., 2019) and ConceptNet (Speer et al., 2017) further blurring the line with the convectional story generation approach. The **Error! Reference source not found.** shows an example story generated by their model conditioned to ROCStories Corpus (Mostafazadeh et al., 2016).

Input: [MALE] has been married to his wife for 20 years.

Output story: he was very nervous about his upcoming reunion. he decided to go to an italian restaurant and get some food. the food was good but he could n't decide which dish he wanted. he decided to try a pork dish instead.

fig. 3 Example story from Guan et al. (Guan et al., 2020)

The story in fig. 3 is an excellent example for demonstrating the weakness of the discussed learning models. The story is logical and coherent, yet it lacks a purpose, central message, or moral. Also, the approaches that use neural networks adapt various aspects of planning or autonomous agents (or character profiles) to derive better content. This, with the rise of automatic knowledgebase construction (Bosselut et al., 2020), we believe we are in a situation to afford a more extensive knowledge base to use planning and address underexplored parts of the story, like the moral.

III. Background

A. Story Taxonomy

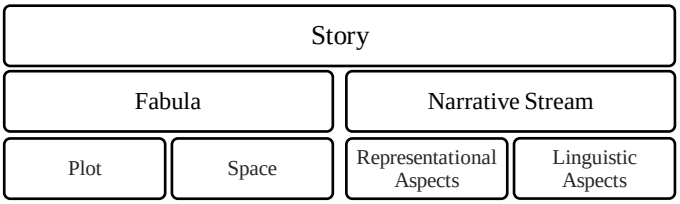


fig. 4 Story taxonomy

Inspired by Chatman's proposal (Chatman, 1980), we define story taxonomy as shown in fig. 4. In the taxonomy, space consists of every element of the story world. It includes all the characters, objects, locations, atmosphere, etcetera. The plot is the action part of the story, where all events exist. Together plot and space create the fabula, which is the main content of the story. On the other hand, the narrative stream defines how the fabula unfolds and is represented to the audience. It includes additional aspects of the story, like representational order of the events, length of the description of the space elements and events, which events are to be omitted, from whose perspective will the story be told, etcetera. One of the benefits of separating narrative stream from fabula is that by varying narrative streams, many stories can be generated for the same fabula (Callaway & Lester, 2002). Additionally, it also allows the process of fabula generation to be linguistically invariant. SANG considers the taxonomy and generates the fabula in internal representation form. Then, SANG converts fabula to the natural language story. The resultant stories are in simple English as of now. However, the language generation system can be replaced to generate the story in any other language without affecting other parts of the SANG.

B. Plot Structure

Aristotle was one of the first to formally study and identify the plot structure in a drama (Butcher & others, 1907). Since then, litterateurs have considered the plot structure an essential part of the plot (Theune et al., 2003). Many plot structures have been proposed and used in the literature. One of the famous plot structures is Freytag's pyramid (Freytag, 1908) as shown in fig. 5.

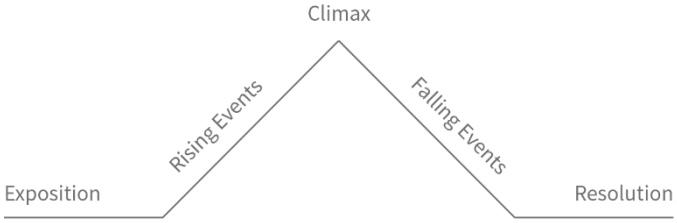


fig. 5 Freytag's plot structure

Irrespective of the shape and position of the components, most plot structures consist of five elements like Freytag's pyramid. The exposition holds the events related to the introduction of the space. It sets up the story world before the plot's main events. On the other hand, resolution pictures the story world after the plot's main events. The climax is where the main events are. It defines the essence of the story theme. Rising and falling events connect exposition with a climax and climax with resolution.

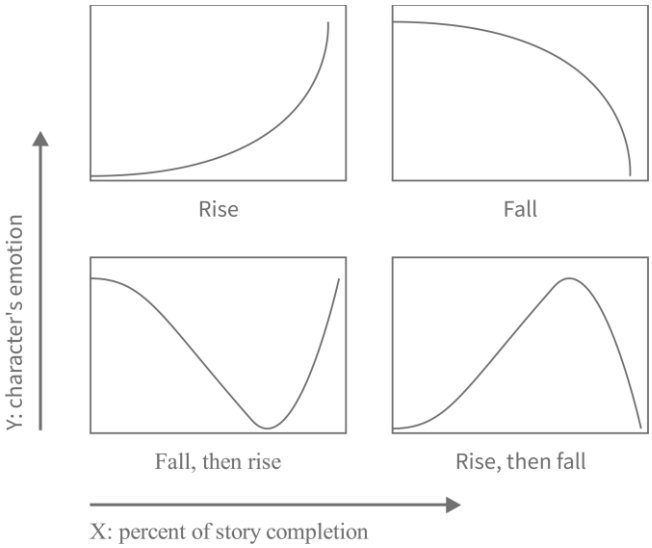


fig. 6 Emotion arcs

Freytag's pyramid may look like a graph, and we can interpret it better as a part of story completion versus tension

it builds in the audience's mind. So, the plot should first introduce the character and then build the tension up gradually, reaching the peak of climax and gradually relieving tension towards resolution. Now, predicting the audience's tension can be complex, as it requires modelling of the audience and their cognitive processes. Instead, we can reduce the problem of predicting the audience's tension into the problem of tracking the protagonist's struggle or emotions. The more the protagonist struggles, the more the tension in the audience's mind (Komeda & Kusumi, 2006). We call the resultant graph emotion arc, inspired by the proposals of (Reagan et al., 2016). Four emotion arcs are shown in fig. 6. At a given time τ in the story-world, we compute the character emotions as the signed summation of all the emotion the character is having at τ, where positive emotions like joy carry the positive sign, while negative emotions like sorrow carry the negative sign.

IV. Character Modelling

Characters are a vital part of the stories. The reason why character modelling is so crucial for story development is empathy. Characters in the story allow the audience to see the story world through their eyes. With empathy to one of the characters (usually the protagonist), the audience lives the story, and the events substantially impact the audience's mind. An author can convey the emotions, situations, perspectives, beliefs, etcetera, with the help of the characters, which would have been difficult otherwise.

SANG uses a library of pre-built characters. After planning the story events, it uses the strategy of building prime sets (Khalpada & Garg, 2019) and selecting the character that can satisfy the requirements of the prime trait set. If no such character exists in the library, SANG uses inverse planning to mutate the character from the closest match into a full match.

Often, characters are introduced by SANG to resolve a planning lock, a situation where the planner cannot plan the next event because the requirements of any event cannot be fulfilled with the current story world. Character aspects are maintained in the memory and updated as the story progresses. These aspects are as below.

A. Basic Character Data

It includes basic information about a character, like a name, age, health, etcetera. The majority of these parameters remain static during the narrative generation process. The only fields which may require updating during the story generation are the location and health of the character at a particular instance in the story world.

B. Emotions

A story with emotionless characters would be more of a report describing events, even if the events are perfectly arranged logically. Modelling emotions can be a complicated job. Rather than exploring the depths and breadths of the complex study of emotions, SANG follows a simple classification of emotions inspired by Plutchik's wheel (Plutchik, 2001). Emotions are stored in a way represented by the fig. 7. (Consider * as Kleene's star from regular expressions, meaning 0 or more occurrences of the entity.)

<id, emotion, subject, object*, strength, causal id>

fig. 7 Internal representation of character emotions

Emotions are not independent of each other. It is hard to imagine a person feeling rage and bliss simultaneously. As SANG has a common-sense knowledge base, SANG understands these dependencies. In SANG, the following scenarios can cause a change in the emotion of a character.

- Actions that the character performs. For example, if a person commits a crime, he may regret it and become sad later.
- Actions that other characters perform. For example, if a princess commits a crime, the king may become sad. It depends on the emotional change that action may cause and the emotional bond between the characters. For example, the king is happy with happiness value 2, his emotional bondage with the princess is 8, and the killing action causes sadness of 9. The king is now sad by the value 6. The strength of emotion varies in the range [0-10].
- State of the other character. For example, if a princess dies, the king may become sad. It depends on the property value and the emotional bonding between the characters. It is calculated similarly as in the previous case.
- Change in relationships. For example, a prince may become happier getting married to a princess. It depends on the emotional bond and his emotional value, that is, the emotional bonding of the prince towards the princess and the prince's happiness before the marriage.

C. Relationships

Like emotions, relationships between characters are also crucial for a narrative to interest the reader. The different types of relationships are defined in the database of relationship types in the knowledge base. SANG assigns the relationship to the characters and specialize them, i.e., converting them from gender neutralized form to gender-specific form based on the sex of the characters. For example, relation (parent, child) is converted to (father, daughter).

V. Story Generation

The overview of the system flow is shown in **Error! Reference source not found.Error! Reference source not found.**. As discussed in section II, we use planning to generate events. To use planning, we need an initial state of the story-world I and the final state of the story-world F. Instead of feeding in the real initial and final story world, we instead use the story-world at the beginning of rising events as I and end of falling events as F. This offers finer control over the generation of exposition and resolution and ensures that the resultant story has an exposition and resolution part. If we had opted for the other way, the planner would often omit these parts by considering the subsequent possible causal events.

To derive the pilot events, and hence the story-world, I and F, we first determine the emotion arc along with I and F from the moral. We then use

- planning to build fabula between them,
- backward chaining to build exposition, and
- forward chaining to build the resolution.

A. *Deriving Pilot Events*

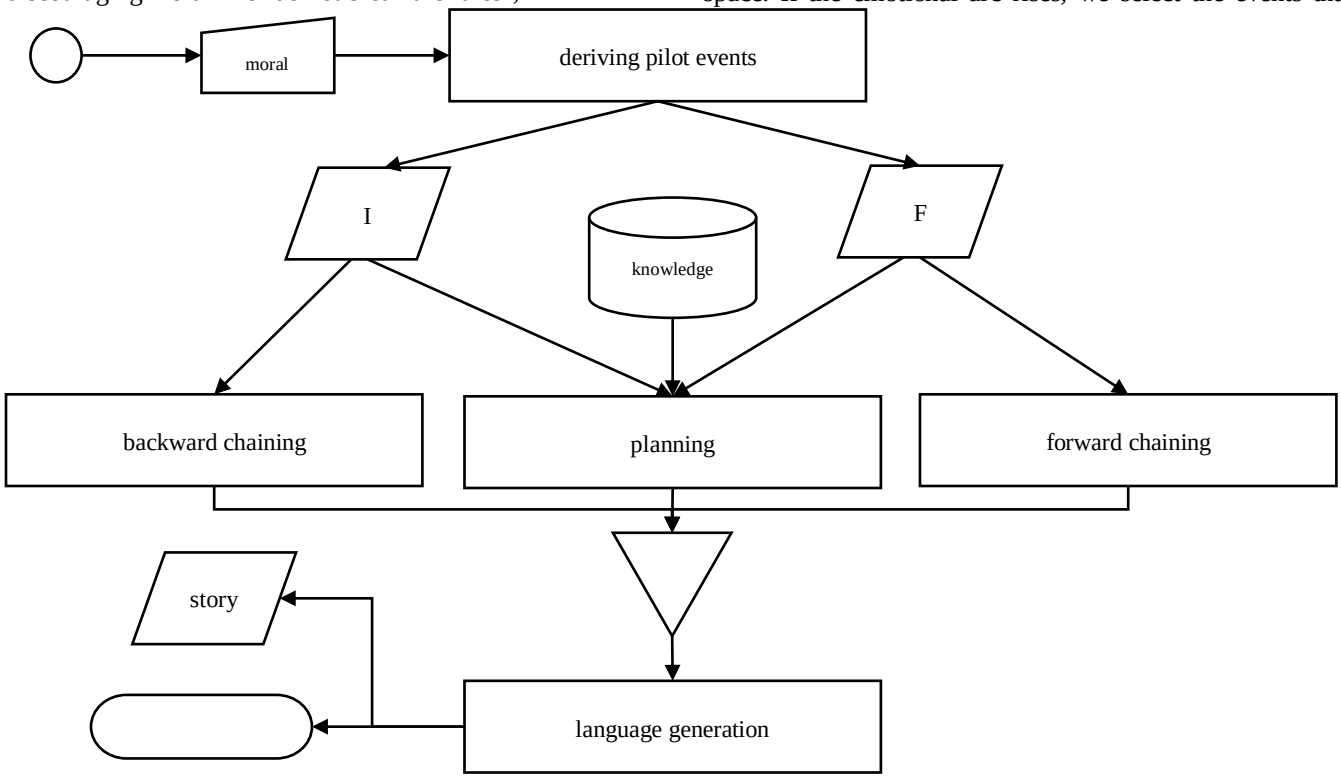
If we classify morals by polarity, we have encouraging and discouraging morals. Encouraging morals motivates positive actions and values, for example, "be forgiving," while discouraging morals demotivates negative actions and values, for example, "do not judge a book by its cover." Each moral can be conveyed in two ways:

- The character exhibiting that trait meets the respective consequence. For example, a person may judge another person based on appearance and lose something important because of such judgment.
- The character not exhibiting that trait misses the respective consequence. For example, a person does not judge another person based on appearance and is rewarded with something important because of his choice of not judging.

Psychologists suggest that appreciation and inspiration are the best way to encourage a person (Dweck, 2008) to act or behave in a certain way. Hence, we associated encouraging morals with the arcs ending on higher emotion while we associate discouraging morals with arcs that end on the lower arc. We then use the emotional arc to guide the planning of events.

The initial point of rising action, or I, in the story would be the first time the character shows the value or performs an action encouraged or discouraged by the moral. For example, if the user inputs a discouraging moral like "do not break the rules",

fig. 8 System flow



- the I would be the character's introduction to the situation where the character is tempted to break the rule.
- the F would be character meeting the consequences of breaking the rule.

On the other hand, if the user inputs an encouraging moral like "obey the rules", the I would be the same as above, but F would change to rewarding the character for obeying the rules. We then use the I, the F, and the character database to plan a set of events between exposition and resolution.

B. *Planning Events*

To plan the events, we use Directed Causal Graph (D.C.G.) like in (Khalpada & Garg, 2019), where nodes n represent the plot events and edges e represent the causal connections between the events. In addition to ConceptNet (Speer et al., 2017), we use CoMET (Bosselut et al., 2020) trained over ATOMIC2020 (Hwang et al., 2021) dataset to supplement the knowledge base crafted in first-order logic.

To ease the process of language generation, we represent the nodes in an internal representation form as shown in fig. 9. (Consider * as Kleene's star from regular expressions, meaning 0 or more occurrences of the entity.)

<id, event, subject*, object*, medium*, strength*, location, causal id>

fig. 9 Internal representation of events

While planning, we often have multiple choices for the next possible event that meet the story-world criteria at each node. We use the emotional arc to constraint the exploration space. If the emotional arc rises, we select the events that trigger or strengthen positive emotions in character. Similarly, if the emotional arc falls, we select the events that trigger or

strengthen negative emotions. Although planning with the emotional arc significantly reduces the search space, it often gets stuck in a planning lock.

A planning lock is a condition when none of the subsequent possible events meets the requirements (story-world criteria or emotion arc requirement). To overcome this, we use an inference mechanism to develop a subplot or prequel to change the current story-world context or the emotional arc to the matching one. For example, if we want Jill to date Jack but cannot meet the requirement of "Jill intending to date Jack", we can use the example inference rules from fig. 10. The result of the rules would be an introduction of a friend of Jill, who would then persuade Jill to date Jack.

Problem:
?y.intend(date, ?z)
Rules:
?y.be(friend,?x) → ?y.capable(persuade, ?x)
?y.persuade($activity, ?x) → ?x.intend($activity)

fig. 10 Example of an inference rule

C. Avoiding Repetitive Stories

Another issue that might arise from limited exploration space is repetitive story generation. To avoid that, we remember the edges we traversed for a given story-world context and events and try to avoid that edge next time when an analogous situation arises. Let us assume that, given a story-world context, we are at an event e_i and we have a set of subsequent possible events $\{n_1, n_2, n_3, n_4\}$ from which $\eta = \{n_1, n_2, n_3\}$ that meet the story-world criteria. Suppose planner selects n_3 as the next event, then we remember that for a given story-world context, on event e_i we selected n_3. We also maintain the times an edge was selected as x. For our example, we would increase the x for n_3 while decreasing the x for n_1 and n_2 reflecting they both were possible subsequent events that met the story-world criteria, but the planner used n_1 instead. Note that the x for n_4 would be unaffected because although it was a possible next event, it had not met the story-world criteria, so the planner could not have used it. We then update the likelihoods on the edges $e_{(n_i,n_k)}$ of all events $\forall n_k \in \eta$ connecting to n_i as

$$\rho_{e_{(n_i,n_k)}} = \rho_{e_{(n_i,n_k)}} - \left(\gamma \times tanh\left(\frac{x_{e_{(n_i,n_k)}}}{|\eta|}\right)\right)$$

where, $|\eta|$ is the cardinality of the set of subsequent possible events that meet the story-word criteria, γ is the user tunable effective rate, defining how much of the consecutive use of the edge $e_{(n_i,n_k)}$ would affect the weight. We use $tanh$ function to increase the penalty for subsequent edge uses. If any $n \in \eta$ that is not preferred for a while, x of it may become negative, resulting in increase of the likelihood of the edge.

D. Language Generation

SANG uses a rule-based system to generate English sentences from the internal representation format. It first uses context-sensitive grammar-based associations to generate simple sentences. Table 1 shows an example snippet of a fabula in internal representation form and simple story sentences derived from it.

Table 1 Example of simple sentence generation

Events in internal format	Simple sentence
<e1101, see, c1720, c1386.trait[2], l2710>	Jack was in a garden. Jill was in a garden. Jill was beautiful. Jack saw the beauty of Jill.
<f1602, love, c1720, c1386, 8, e1101 & c1720.prefTrait[0]>	Jack liked beauty. So, Jack strongly fell in love with Jill.
<e1102, propose, c1720, c1386, l2710, f1602>	Jack proposed to Jill in a garden because Jack fell in love with Jill.
<e1103, refuse, c1386, e1102, f1530>	Jill refused to Jack proposing to Jill because Jill liked Raj.
<f1603, sad, c1720, 5, e1103>	Jack became sad because Jill refused to Jack proposing to Jill.

SANG then uses language protocols defined in first-order logic from knowledgebase to generate compound and complex sentences. One such example is shown in figure 11.

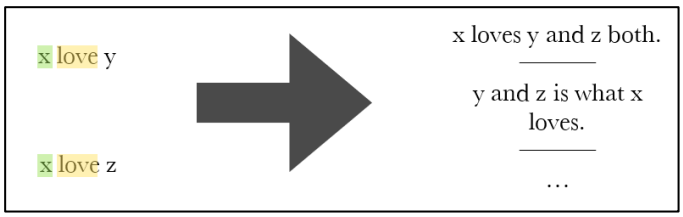


figure 11 An Example of Compound Rule

Table 2 shows an example of applying language protocols on simple sentences from Table 1.

Table 2 Example sentences after applying language protocols

Simple sentences	Rewritten surface sentences
Jack was in a garden. Jill was in a garden. Jill was beautiful. Jack saw the beauty of Jill.	Jack went to a garden, where he saw a beautiful girl, Jill.
Jack liked beauty. So, Jack strongly fell in love with Jill.	Because he always liked beauty, he deeply fell in love with Jill.
Jack proposed to Jill in a garden because Jack fell in love with Jill.	So, he proposed to her.
Jill refused to Jack proposing to Jill because Jill liked Raj.	But, she liked Raj, so she refused.
Jack became sad because Jill refused to Jack proposing to Jill.	Jack became sad.

VI. Experiment

To evaluate the narrative generation, we considered the following aspects.

- Does the narrative convey the expected moral?
- How well do the stories with moral fare against baseline stories without moral?

We generated 5 narratives for 7 different morals each to evaluate the narratives. We added 5 random stories generated using (Khalpada & Garg, 2019) without morals to eliminate outliers. We then asked 103 human evaluators to list a maximum of 3 morals they think the narrative conveys. To avoid bias and one-to-one relation, we provided a list of 21 morals and a none option to assign each narrative its closest moral.

Table 3 Human evaluation

Moral	Inter-evaluator agreement	Moral confidence in top 3	Moral as prime (Normalised)
Be forgiving	0.91	0.74	0.81
One should obey the rules.	0.83	0.85	0.89
It is okay to be different.	**0.71**	0.68	0.74
Do not judge a book by its cover.	0.86	**0.43**	**0.69**
One should not lie.	0.94	**0.96**	0.98
It is wrong to hurt others.	0.91	0.83	0.91
Hard work always pays off.	**0.97**	0.91	**0.99**
	0.87	**0.77**	**0.86**

Table 3 shows the result of manual evaluation. A strong agreement average of 0.87 between the evaluators imply that they strongly agree with the result. A good average of 0.77 for moral confidence shows that the generated narratives nicely convey the intended moral. This is a good result, even for cherrypicked morals, especially as there are hardly any baseline contenders. To better understand the confidence of the moral in the narrative, we filter the positive samples and examine how many of them have the intended moral with the highest confidence (on top of the list of 3 morals). An average of 0.86 shows strong confidence in the intended moral, suggesting that if an evaluator has associated the intended moral with the narrative, it is highly likely that the intended moral would have the highest confidence.

Inspecting the individual results, we realized that the narratives for the complex morals like 'do not judge a book by its cover' have lower confidence. We suspect this is mainly because of the higher order of exploration the planner may require. $observe(PersonX, PersonY.attribute1) \rightarrow assume(PersonX, PersonY.attribute2) \rightarrow punish(PersonX)$ requires exploration of the plot events and character attributes that convey the moral. In one generated narrative, because one character had a motorcycle, the protagonist assumed that the character would be short! Although being short and possessing a motorcycle are valid attributes for a character, they do not contrast enough to highlight a stereotype in society.

To understand how well the narratives with morals fare against those without morals, we cherrypicked 5 narratives from the proposed approach and paired them with the other 5 narratives generated using (Khalpada & Garg, 2019). We asked the same 103 evaluators to select a story from the pair they are likely to tell kids. Narratives with the proposed approach were selected with encouraging confidence of ≈ 0.71.

VII. Conclusion

We proposed a planning-based approach to generate narratives for a given moral. A moral is a vital part of the story. The second experiment shows that narratives with morals are highly likely to be shared with kids. The first experiment shows that the proposed system generates the narrative for intended morals with a high confidence.

References

Bae, B. C., & Young, R. M. (2014). A computational model of narrative generation for surprise arousal. *IEEE Transactions on Computational Intelligence and AI in Games*, *6*(2), 131–143. https://doi.org/10.1109/TCIAIG.2013.2290330

Bahamón, J. C., & Young, R. M. (2017). An empirical evaluation of a generative method for the expression of personality traits through action choice. *Proceedings of the 13th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2017*, 144–150.

Barros, L. M., & Musse, S. R. (2008). Towards consistency in interactive storytelling: Tension arcs and dead-ends. *Computers in Entertainment*, *6*(3), 7–9. https://doi.org/10.1145/1394021.1394036

Black, J. B., & Wilensky, R. (1979). An evaluation of story grammars. *Cognitive Science*, *3*(3), 213–229. https://doi.org/10.1016/S0364-0213(79)80007-5

Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Celikyilmaz, A., & Choi, Y. (2020). CoMET: Commonsense transformers for automatic knowledge graph construction. *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 4762–4779. https://doi.org/10.18653/v1/p19-1470

Bringsjord, S., & Ferrucci, D. a. (2000). Artificial Intelligence and Literary Creativity: Inside the Mind of BRUTUS, a Storytelling Machine. *Computational Linguistics*, *26*(4), 642–647. https://doi.org/10.1162/coli.2000.26.4.642

Butcher, S. H., & others. (1907). *The poetics of Aristotle*. Macmillan.

Cai, Y., Shen, Z., Miao, C., & Tan, A. H. (2010). DIRACT: Agent-based interactive storytelling. *Proceedings - 2010 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2010*, *2*(May 2016), 273–276. https://doi.org/10.1109/WI-IAT.2010.89

Callaway, C. B., & Lester, J. C. (2002). Narrative prose generation. *Artificial Intelligence*, *139*(2), 213–252. https://doi.org/10.1016/S0004-3702(02)00230-8

Charles, F., Porteous, J., Teutenberg, J., & Cavazza, M. (2011). Timeline-based Navigation for Interactive Narratives Categories and Subject Descriptors. *8th Conference on the Advances of Computer Entertainment Technology*. http://www.ace2011.org

Chatman, S. B. (1980). *Story and discourse: Narrative structure in fiction and film*. Cornell University Press.

Dweck, C. S. (2008). *Mindset: The new psychology of success*. Random House Digital, Inc.

Fairclough, C., & Cunningham, P. (2003). A Multiplayer Case Based Story Engine. *Game-On, Bremond*, 41-.

Fan, A., Lewis, M., & Dauphin, Y. (2018). Hierarchical neural story generation. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, *1*, 889–898. https://doi.org/10.18653/v1/p18-1082

Freytag, G. (1908). *Freytag's technique of the drama: an exposition of dramatic composition and art*. Scott, Foresman and Company.

Gervás, P., Díaz-Agudo, B., Peinado, F., & Hervás, R. (2005). Story plot generation based on CBR. *Knowledge-Based Systems*, *18*(4–5), 235–242. https://doi.org/10.1016/j.knosys.2004.10.011

Gottschall, J. (2012). *The storytelling animal: How stories make us human*. Houghton Mifflin Harcourt.

Guan, J., Huang, F., Zhao, Z., Zhu, X., & Huang, M. (2020). A Knowledge-Enhanced Pretraining Model for Commonsense Story Generation. *Transactions of the Association for Computational Linguistics*, *8*, 93–108. https://doi.org/10.1162/tacl_a_00302

Herman, D. (2013). *Storytelling and the Sciences of Mind*. MIT press.

Hwang, J. D., Bhagavatula, C., Le Bras, R., Da, J., Sakaguchi, K., Bosselut, A., & Choi, Y. (2021). (Comet-) Atomic 2020: On Symbolic and Neural Commonsense Knowledge Graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(7), 6384–6392.

Jain, P., Agrawal, P., Mishra, A., Sukhwani, M., Laha, A., & Sankaranarayanan, K. (2017). *Story Generation from Sequence of Independent Short Descriptions*. https://doi.org/10.475/123

Khalpada, P., & Garg, S. (2019). Balancing Consistency and Plot Structure in Computational Storytelling. *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2019*, 550–555. https://doi.org/10.1109/IEMCON.2019.8936138

Khalpada, P., & Garg, S. (2021). Simple Automated Narrative Generator (SANG). *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, 906–914.

Komeda, H., & Kusumi, T. (2006). The effect of a protagonist's emotional shift on situation model construction. *Memory & Cognition*, *34*(7), 1548–1556.

Kybartas, B., & Verbrugge, C. (2014). Analysis of ReGEN as a graph-rewriting system for quest generation. *IEEE Transactions on Computational Intelligence and AI in Games*, *6*(2), 228–242. https://doi.org/10.1109/TCIAIG.2013.2290088

Lakoff, G. P. (1972). Structural Complexity in Fairy Tales. In *The Study of Man* (Vol. 1, pp. 128–150).

Lebowitz, M. (1985). Story Telling as Planning and Learning. *Poetics*, *14*(6), 483–502.

Liu, D., Li, J., Yu, M.-H., Huang, Z., Liu, G., Zhao, D., & Yan, R. (2020). A Character-Centric Neural Model for Automated Story Generation. *Proceedings of the AAAI Conference on Artificial Intelligence*. www.aaai.org

Liu, P. J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, Ł., & Shazeer, N. (2018). Generating wikipedia by summarizing long sequences. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 1–18.

Machado, I., Paiva, A., & Brna, P. (2001). Real characters in virtual stories promoting interactive story-creation activities. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *2197*, 127–134. https://doi.org/10.1007/3-540-45420-9_14

Martin, L. J., Ammanabrolu, P., Wang, X., Hancock, W., Singh, S., Harrison, B., & Riedl, M. O. (2018). Event representations for automated story generation with deep neural nets. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 868–875.

Meehan, J. (1977). TALE-SPIN, An Interactive Program that Writes Stories. *IJCAI*, *77*, 91–98.

Mostafazadeh, N., Chambers, N., He, X., Parikh, D., Batra, D., Vanderwende, L., Kohli, P., & Allen, J. (2016). A corpus and cloze evaluation for deeper understanding of commonsense stories. *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*, 839–849. https://doi.org/10.18653/v1/n16-1098

Onuczko, C., Szafron, D., & Schaeffer, J. (2008). Stop Getting Side-Tracked. *AI Game Programming Wisdom*, *4*, 513–528.

Oraby, S., Reed, L., Tandon, S., T.S., S., Lukin, S., & Walker, M. (2019). *Controlling Personality-Based Stylistic Variation with Neural Natural Language Generators*. *July*, 180–190. https://doi.org/10.18653/v1/w18-5019

Peinado, F., Gervás, P., & Díaz-Agudo, B. (2004). A Description Logic Ontology for Fairy Tale Generation. *International Conference on Language Resources and Evaluation: Workshop on Language Resources for Linguistic Creativity*, 56–61.

Pemberton, L. (1989). A modular approach to story generation. *Fourth Conference of the European Chapter of the Association for Computational Linguistics*, 217–224. https://doi.org/10.3115/976815.976845

Pérez y Pérez, R. (1999). *MEXICA : A Computer Model of Creativity in Writing*. 245.

Plutchik, R. (2001). The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical

practice. *American Scientist*, *89*(4), 344–350.

Propp, V. (2010). *Morphology of the Folktale*. University of Texas Press.

Reagan, A. J., Mitchell, L., Kiley, D., Danforth, C. M., & Dodds, P. S. (2016). The emotional arcs of stories are dominated by six basic shapes. *EPJ Data Science*, *5*(1), 1–12.

Riedl, M. O., & León, C. (2008). Toward Vignette-Based Story Generation for Drama Management Systems. *Workshop on Integrating Technologies for Interactive Stories*, 23–28.

Riedl, M. O., & Young, R. M. (2010). Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*, *39*, 217–268. https://doi.org/10.1613/jair.2989

Sap, M., Le Bras, R., Allaway, E., Bhagavatula, C., Lourie, N., Rashkin, H., Roof, B., Smith, N. A., & Choi, Y. (2019). ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*, 3027–3035. https://doi.org/10.1609/aaai.v33i01.33013027

Shim, Y., & Kim, M. (2002). Automatic short story generator based on autonomous agents. *Pacific Rim International Workshop on Multi-Agents*, 151–162.

Skorupski, J., & Mateas, M. (2010). Novice-friendly authoring of plan-based interactive storyboards. *Proceedings of the 6th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2010*, 174–179.

Speer, R., Chin, J., & Havasi, C. (2017). Conceptnet 5.5: An open multilingual graph of general knowledge. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(1).

Spierling, U., Grasbon, D., Braun, N., & Iurgel, I. (2002). Setting the scene: Playing digital director in interactive storytelling and creation. *Computers and Graphics (Pergamon)*, *26*(1), 31–44. https://doi.org/10.1016/S0097-8493(01)00176-5

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, *4*(January), 3104–3112.

Theune, M., Faas, S., Faas, E., Nijholt, A., & Heylen, D. (2003). The Virtual Storyteller: Story Creation by Intelligent Agents. *Proceedings of the Technologies for Interactive Digital Storytelling and Entertainment (TIDSE) Conference*, 204–215. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.12.5222

Turner, S. R. (1993). MINSTREL: A computer model of creativity and storytelling. In *ProQuest Dissertations and Theses* (pp. 841-841 p.). http://search.proquest.com/docview/304049508?accountid=43623

Ware, S., & Young, R. M. (2014). Glaive: a state-space narrative planner supporting intentionality and conflict. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, *10*(1).

Yao, L., Peng, N., Weischedel, R., Knight, K., Zhao, D., & Yan, R. (2019). Plan-and-Write: Towards Better Automatic Storytelling. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*, 7378–7385. https://doi.org/10.1609/aaai.v33i01.33017378

Zhang, X., & Lapata, M. (2014). Chinese poetry generation with recurrent neural networks. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 670–680. https://doi.org/10.3115/v1/d14-1074