Survey: Automatic Movie Plot and Script Generation

Prerak Gandhi and Pushpak Bhattacharyya
Center for Machine Intelligence and Data Science
Indian Institute of Technology Bombay
{prerakgandhi, pb}@cse.iitb.ac.in

Abstract

Automatic story generation has long been researched in the field of Machine Learning. Stories are used to engage and communicate with people. One of the essential roles of stories is entertainment. We can craft stories that have strong plots, interesting characters, multiple genres, and a plethora of emotions. The entertainment industry is an area where NLG has much potential. Scripts are written in a particular screenplay format which contains multiple elements. The different elements make it more challenging for the model to learn the text. Plots are different than stories with some subtle differences. Neural networks manage to generate creative stories but maintaining the coherence and reducing hallucination are few of the challenges to overcome.

1 Problem Statement

Automatic story generation using neural networks is not as expected compared to rule-based systems. This is rare in an era when we look at how computer science is advancing in every field with the help of neural networks. They are not only irreplaceable but also increasing in complexity and research areas. Template-based models in story generation are often used to convert unstructured data to structured text. One of the most prominent applications is posting news or making reports based on many data. These cases do not require the creativity of neural networks but rely on premeditated rules and cases.

Story generation in itself is not a big problem to overcome yet. Human beings are creative, especially in the field of storytelling; there have been an uncountable number of stories produced in any genre imaginable. From moral stories to science fiction books that are turned into blockbusters to religious books that were written thousands of years ago, the text is one of the fields where human beings are hard to replace with programs.

The majority of areas that require creativity in story generation can be said to belong in the entertainment industry. People consume movies and television shows online almost every day for hours. The requirement of new forms of humour- in terms of comedy shows, parodies and mainly online memes- due to the increase in stress caused by the fast and competitive world can be fulfilled by automatic text generation.

The problem statement can be divided into two tasks:

- Automatic movie scene generation, where the input is a small description of the scene required and the output generated is a scene, with four of its major elements-*scene headings, action lines, dialogues and character names*-retained.

- Automatic movie plot generation, in which a long plot (700 words) is generated given a short storyline.

2 Motivation

Automatic story generation (ASG) has many fundamental AI research problems about the requirements of a system for storytelling. Storytelling requires a system to have a great un-

derstanding of basic concepts and much knowledge. A weak story can be figured out quickly when the essential details are wrong or contradicting. The advancement of ASG leads the field of storytelling to improve as well. ASG can teach children to write creative stories from just a set of words or 1-2 lined sentences. Stories can be generated to focus on different cultures, people, places, traditions, insights and morals.

Multiple computer games feature stories or plots, which can be generated. Interactive stories are those in which the user assumes the role of a character in a story and can change the story with their actions. These games are called Role-Playing Games or RPGs and are becoming popular by the day. Creating big games such as Elder Scrolls, Final Fantasy, and more requires time and work to develop.

The visual entertainment industry has been redefined in the 20th century since the invention of motion pictures. The ability to create something and cater to the entire world makes it a trendy field. Although people are very interested in movies and television shows, the effort required to create something novel and exciting is also increasing. This need can be fulfilled by neural networks, unbounded by imagination and work on forces controlled by probability.

There are 3 Ds of automation- Dull, Dirty and Dangerous. All the tasks that catgorize under these 3 Ds should be handed over to AI. Dull tasks use up the valuable time that human beings can donate to some other productive areas. Dirty tasks are those which are unhygienic and can create new diseases. Finally, there are many dangerous tasks in the world due to the high ambitions of man. Some of them are bomb investigations, space exploration etc., which can be carried out by drones or other advanced capability robots. Movie plot and script generation falls under one of the 3 Ds, namely Dull. Movie scripts are around 30,000 words long which is equivalent to a 100-page book. On top of the large amount of words, scripts are written in a screenplay format. This requires the writer to take care of the scene headings, action lines, transitions, characters, dialogues, parenthesis and a few more minor elements. This increases the time taken to write a script even more.

Automatic scene generation has never been done before and is a new field to be explored whereas automatic plot generation is also looked at with a different perspective.

3 Structure of a Movie Script

A screenplay is a structured text used for television shows, movies and sometimes games.

A screenplay format is such that one page equals roughly one minute of screen time. Movie scripts range from roughly 60-180 pages and, on average, are around 110 pages long.

A screenplay is supposed to help the reader imagine the scene just by reading it. Hence, each scene has several different elements, which help visualize the scene. The main elements found in almost all screenplays are:

- **Slugline (Scene heading)–** This describes the scene's location and the time, day or night. It starts with either EXT. or INT. meaning external or internal, respectfully. It then contains the location name and the setting for either day or night. To give a more precise location, the added words are separated by a hyphen.
- **Action Lines–** These are usually placed right after the scene headings. They are used to describe the physical setup of the scene and provide a narrative to the non-spoken content. They convey almost everything about a scene except for the dialogues spoken by the characters.
- **Character Name–** Each dialogue is spoken by a character, and the name is centrally aligned and comes right before the dialogue.
- **Dialogue-** These are the lines that are spoken by the on or off screen characters. They are centrally aligned as well and come immediately under the character name.

- **Extensions–** They are written next to the character name to indicate where the character is when the dialogue is spoken by someone not visible on screen. Voice Over (V.O.) and Off-Screen (V.O.) are the most usual extensions.
- **Transitions–** They are placed at the extreme right-hand side of the page and indicate any major changes to the scene. FADE IN, FADE OUT and CUT TO are the main three commands used.
- **Parentheticals–** They are used within the dialogue to indicate small details that are extremely important. They provide stage direction and tone, which would otherwise be used in action lines — for example, speaking on the phone, opening a door.

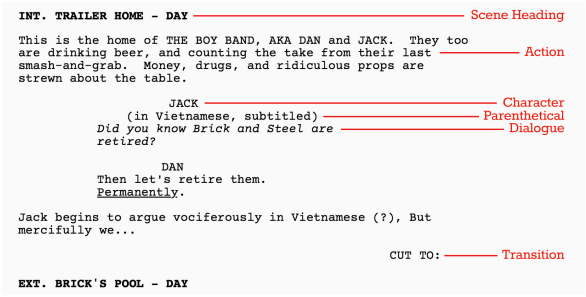

Figure 1: Structure of a Movie Script

4 Storytelling Frameworks

Storytelling has been one of the oldest means of communication for humans. Stories differ from other kinds of texts in many ways. Stories are meant to be coherent, and good stories are those which also do not have any loose ends. In order to ensure that the stories are good or worthwhile, people have created various templates that help in storytelling.

The most famous template of a story is the 3-act structure. The setup, the confrontation and the resolution make up the three acts. Generally, the second act is the largest and twice the size of the other two acts. Although this template has been used throughout the years, it was primarily popularized by Field (2005).

There have been a number of templates created to assist with story writing. Some of them are extensions of the 3 act structure while some of them are made with other themes deciding the structure. One of the famous themes to create structure in stories is to focus on the protagonist.

There are a lot of mythologies and religious texts from ancient civilisations and some of them revolve around a protagonist. Ramayana revolves around Lord Ram and Mahabharata mainly revolves around Lord Krishna. These myths or legends are examples of stories whose theme is built around the protagonist.

4.1 The Hero's Journey

In the 20th century, the rise of cinema led to a breakthrough in the storytelling themes. Campbell (1968) introduced *The Hero's Journey* which is a story template with 17 steps within a protagonist's journey. It explores the rollercoaster journey that the protagonist has with his world constantly changing. The template mainly consists of 3 divisions:

1. Departure
2. Initiation
3. Return

These 3 acts are further divided to a total of 17 steps. The *Departure* act shows the protagonist live in his ordinary world and due to some circumstances he receives a call to go on an adventure. The *Initiation* act shows the protagonist travel to the new world and face new challenges. He then reaches the lowest point of the journey and must overcome the main enemy or challenge. The *Return* act shows the hero return to his ordinary world again. This time he is a changed person and has a reward of kind with him. He finally gains spiritual power over both worlds.

4.2 The Beat Sheet

The Hero's Journey inspired many more themes that revolved around a protagonist. Blake Snyder (2005) introduced the concept of a Beatsheet to write scripts. The Beatsheet is

an ordered template that allows a scriptwriter to focus on particular twists and concepts of the protagonist's journey.

Snyder starts by explaining the Beatsheet in 15 steps and then expanding it from 15 to 40 to cover all the scenes of a script. Since a 2 hour movie has around 110 scenes, each beat needs to cover only 3 scenes. This makes it easier for the scriptwriter to work on different areas of the script and then combining it to complete the entire script. In the Table 1, is the 15-part Beat Sheet divided into 3 acts.

These storytelling frameworks are not only used in script-writing but can also be used in story generation tasks. We use a 4-act structure to annotate movie plots to create a dataset that improves the quality of plots generated.

5 Natural Language Generation

Natural Language Generation is considered as a subfield of Natural Language Processing. In academics, NLP includes everything that involves the interaction of computers with language. NLG under the same field, implies the generation or production of text made by computers. Reiter and Dale (2002) define NLG to be the subfield of artificial intelligence and computational linguistics that is concerned with the construction of computer systems that can produce understandable text in English or other human languages from some underlying non-linguistic representation of information. The part of the definition that mentions "other human languages" relates to another field of NLP known as Machine Translation. The final part of the definiton "underlying non-linguistic representation of information" implies that even though the output generated is text, the model may learn from non-text inputs such as graphs, charts, images, videos or sound.

Though we are using NLG for automatic script generation, it is used to perform multiple other tasks like, question-answering, summarization, chatbot response generation and multiple types of structured data generation.

For any NLG task, the target corpus should help it to generate all ranges of text that the system/task is expected. One of the methods to create the corpus involves studying and analysing the initial corpus. The information available from such corpora can be classified into one of the following classes:

Unchanging/fixed text– This is a part that remains unchanged within each output generated. Examples include the signature at the bottom of an email or a letter.

Directly-available data– This information consists of everything that the model/system has direct access to. It may be in terms of text, like the prompt of a movie in case of text-based input corpus.

Computable data– This is the type of data that the model can calculate or derivations. This includes mathematical deductions from the data, such as, the number of people travelling in a train daily, when given as input the number of tickets sold.

Unavailable data- Information that is not directly accessible to the model and depends on external data is called unavailable data. This type of data requires either a large knowledge about general things or is based on experience. For example, in unknown areas, maps may not be able to show the best routes available that local people would know well.

After the target text corpus has been created, we look at the task that the NLG system has to perform. The common subproblems to simplify the main task are presented below:

Content Determination– This is the step where the information to be used is determined. The best method to display the content also has to be chosen.

Discourse Planning– This problem is also called text structuring where the message that is set by the content determination task is made to be coherent. It should convey the relations between different sentences.

Sentence Aggregation– The sentences, once formed, have to be combined to make the text more fluent. Various grammatical methods are used that avoid and remove repetition in multiple sentences. Example, "I went to the shop. I went to the playground" can be aggregated to say "I went to the shop and the

Act 1	Act 2	Act 3
Opening Image	Break into Act 2	Break into Act 3
Theme Stated	B Story	Finale
Setup	Fun and Games	Final Image
Catalyst	Midpoint	
Debate	Bad Guys Close In	
	All is Lost	
	Dark Night of the Soul	

Table 1: The 15-part Beat Sheet under 3 Acts

playground."

Lexicalisation– Choosing the right phrase or words to describe the relation between different concepts to suit the message clearly. An example would be choosing a more appropriate synonym of a word.

Referring expression generation– This procedure involves naming or describing particular entities. This may include the usage of pronouns or proper nouns.

Linguistic Realisation– The final subproblem is linguistic realisation. This step makes sure the sentences are grammatically correct. It checks for correct punctuations, plurals, spellings as well as the syntactical structure.

6 Automatic Story Generation

Human beings have been telling stories as a form of education or information throughout history. Storytelling through a medium of writing also has historical origins which can be trace back to hieroglyphs in Ancient Egyptian civilization. Sacred books like The Bible, Quran and Bhagawad Gita have also been communicated through the years via manuscripts. Once the Gutenberg press was discovered in the 15th century, it started a Printing Revolution. Stories from then onwards could be easily shared and communicated with people. Automatic Story Generation is a revolution that changes the methods and volume in which stories are generated. We look at some of the most important terms used in story generation:

- **Narrative:** The recounting of a sequence of events that have a continuant subject and constitute a whole.
- **Event:** An event describes the change in the state of the world.
- **Story:** A story is a timeline or a sequence of events. Eg. The king died and the queen died.
- **Plot:** A plot provides reasoning for the order of the events. Eg, The king died, and then the queen died avenging him in battle.
- **Coherence:** It is a measure of whether sentences are logically consistent.
- **Hallucination:** It is a term used for when the generated text is non-factual or non-related to the source text.

6.1 Symbolic Approach

Symbolic systems dominated much of the history of automated story generation. These systems are capable of generating reasonably long and coherent stories. The defining feature of this approach is the reliance on knowledge bases containing hand-coded knowledge structures. Symbolic approaches can be further divided into two methods– case-based and event-based.

Event-Based Methods This method uses events to create the outline of a story. The relationship between different events is the scale with which models are built. A causal relation between two events indicates that the temporally latter event is made possible by the presence of the temporally former event. Some are goal events, and some are sub-goal events; both are combined to achieve a final goal.

Case-Based Methods This method is based on the idea that most reasoning is not done from principles but instead adapts memories of solutions to related problems to new contexts. Humans sometimes tell stories by adapting existing stories to new contexts. This method approaches story generation with the assumption that there is access to existing stories. When a problem is encountered, the agent retrieves a solution to an older problem, applies the old solution to the new problem, adapts the old solution to fit the current problem and stores the new solution. The following are the works done on text generation.

6.1.1 Symbolic Models

The Mexica system (PÉrez and Sharples, 2001) operates on a theory of creative writing. It alternates between two main processes. The first process (engagement) operates in a case-based method. It compares the generated story so far to existing stories in its library to produce continuations. These are not guaranteed to link up with the existing story elements. Hence, the second process (reflection) uses a backward chaining planner to revise the new snippets and add actions to link everything up.

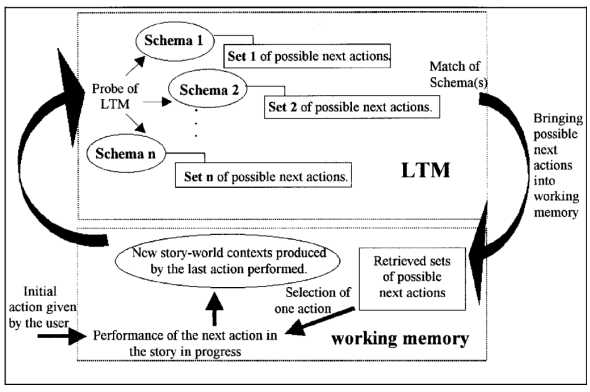

Figure 2: Action retrieval during engagement

The Minstrel system (Turner, 1993) is one of the most famous case-based story generation systems. In Minstrel, all the elements that comprise a story are represented as schemas. There are two types of schemas– author schemas which are author-level goals, and character schemas, which are character-level goals.

6.2 Neural Approach

Machine Learning-based story generation systems have the advantage of neural networks to acquire and use corpora knowledge. This helps build story generation systems that generate a larger space of stories about a bigger range of topics. The disadvantages are the decrease in story coherence which we obtained due to rich knowledge structures in symbolic methods. Some of the models to generate text are seq2seq models built with RNNs or LSTMs. Some of the related works in ASG using neural approaches are briefly explained below.

7 Story Generation using Language Models

A language model learns the probability of a token based on the previous tokens. Starting with a prompt, the language model will provide one or more tokens to continue the text. The combined prompt plus continuation can then be used as input to get the subsequent continuation, and so on. The main limitation of language models is that they create new tokens based on previous tokens. Since they are backward-looking, the text generated may not be coherent or reach a particular goal. As the story expands, more of the earlier context is forgotten. This brings the need to address forward-looking models. Latest language models like GPT-2 (Radford and Wu, 2019) and GPT-3 (Brown et al., 2020) are capable of long text generation tasks.

7.1 Hierarchical Neural Story Generation

Fan et al. (2018) created a model that takes a 1-2 sentence prompt to produce a paragraph. The model is divided between two levels of hierarchy–premise and story. The premise or prompt provides the outline of the story. The model then conditions on the prompt using a fusion (Sriram et al., 2017) of two seq2seq models to generate the story. This allows the story to preserve long-term dependencies. The model uses convolutional networks (Dauphin et al., 2016) instead of RNNs to parallelize the computation. It generates words based on random sampling, which is prone to errors. This

also leads to some repetition in the generated stories.

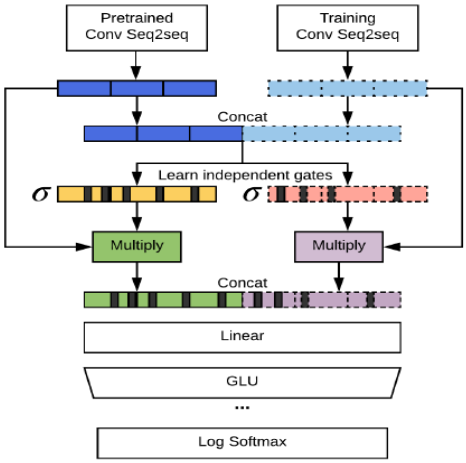

Figure 3: Seq2Seq fusion model for hierarchical story generation

7.2 Plan-and-Write: Automatic Storytelling

The plan-and-write technique (Yao et al., 2019) also uses a two-level approach as above. Instead of a single high-level sentence, the system learns to generate a sequence of keywords. Each of the keywords is used to condition a language model to produce content for that keyword. If the keywords present a coherent progression, then the rest of the content will show the same.

Title (Given)	The Bike Accident
Storyline (Extracted)	Carrie → bike → sneak → nervous → leg
Story (Human Written)	Carrie had just learned how to ride a bike. She didn't have a bike of her own. Carrie would sneak rides on her sister's bike. She got nervous on a hill and crashed into a wall. The bike frame bent and Carrie got a deep gash on her leg.

Figure 4: Title, storyline and story example

The model takes as input the title of the story and generates a storyline (Fig 4). These two are then used as input to a sequence-to-sequence model, which controls story generation. A few limitations of the model are incoherence and repetition.

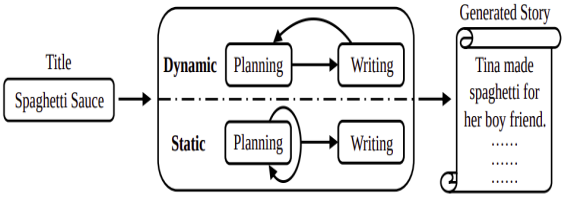

Figure 5: Overview of the Plan-and-write system

7.3 PlotMachines

PlotMachines (Rashkin et al., 2020) conditions a generator on a set of concept phrases given by a user. However, instead of using this set of concept phrases as an outline, the system decides the order to introduce the concepts (Fig 6). The system uses memory state tracking and discourse structures to generate the story plot and observe the text to maintain coherence.

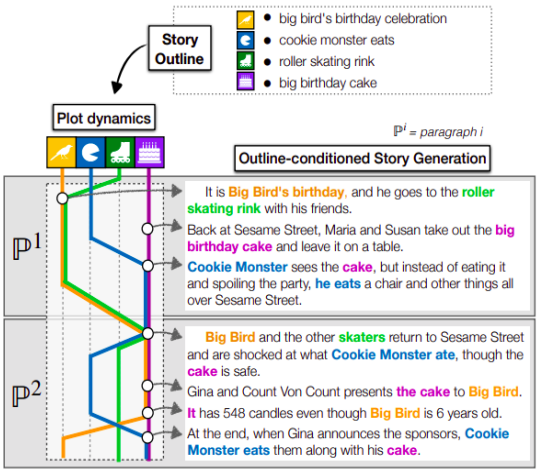

Figure 6: Non-linear connections of the key points from the outline to the story

The model uses the outline, high-level discourse representation, previous memory and preceding context to generate text at each step using their own end-to-end trainable transformer built on top of GPT (Radford et al., 2018). This model shows improvement in

coherence and narrative while creating multi-paragraph stories.

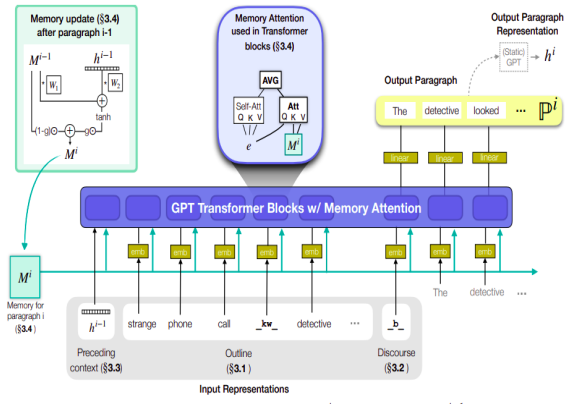

Figure 7: Memory attention built on top of GPT blocks

7.4 Controllable Neural Story Plot Generation via Reward Shaping

Reinforcement Learning is a technique in which the model can learn from its actions based on a predefined reward function or policy. Tambwekar et al. (2019) uses reinforcement learning to fine-tune a seq2seq language model to generate story continuations to reach a goal. The reinforcement learner is used as a non-differentiable loss function. The system extracts verbs from the corpus, clusters them based on distance from the goal and rewards the model when it generates a verb in a cluster closer to the goal. The limitation of this model is training it every time there is a new goal.

Goal	Model	Goal achievement rate	Average perplexity	Average story length
admire	Test Corpus	20.30%	n/a	7.59
	Seq2Seq	35.52%	48.06	7.11
	Unrestricted	15.82%	**5.73**	7.32
	Clustered	**94.29%**	7.61	4.90
marry	Test Corpus	24.64%	n/a	7.37
	Seq2Seq	39.92%	48.06	6.94
	Unrestricted	24.05%	9.78	7.38
	Clustered	**93.35%**	**7.05**	5.76

Figure 8: Reinforcement Learning model- automated experiment results

8 Summary

Automatic Story Generation has gone through many phase shifts, the main one going from symbolic approaches to neural approaches. Symbolic systems derived their work from well-known knowledge bases. However, these knowledge bases have to be structured by hand, which limit the range of what the systems can generate. Neural approaches gained the power of neural networks, which helped to use knowledge from a corpus. These systems can build a larger space of stories about a greater range of topics. The symbolic approaches are richer in story coherence, whereas the neural approach is rich in creativity and novelty. Storytelling frameworks are used extensively in script-writing and are also used in writing books. Similar frameworks can be applied to plots and scripts to improve the understanding of neural models for story generation.

References

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Joseph Campbell. 1968. *The hero with a thousand faces / by Joseph Campbell*, 2nd ed. edition. Princeton University Press Princeton, N.J.

Yann N. Dauphin, Angela Fan, Michael Auli, and David Grangier. 2016. Language modeling with gated convolutional networks. *CoRR*, abs/1612.08083.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.

Syd Field. *Screenplay: The Foundations of Screenwriting*. A Delta book.

Rafael PÉrez Ý PÉrez and Mike Sharples. 2001. Mexica: A computer model of a cognitive account of creative writing. *Journal of Experimental & Theoretical Artificial Intelligence*, 13(2):119–139.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford and Jeffrey Wu. 2019. Rewon child, david luan, dario amodei, and ilya sutskever. 2019. *Language models are unsupervised multitask learners. OpenAI Blog*, 1(8):9.

Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020. Plotmachines: Outline-conditioned generation with dynamic plot state tracking. *arXiv preprint arXiv:2004.14967*.

Ehud Reiter and Robert Dale. 2002. Building applied natural language generation systems. *Natural Language Engineering*, 3.

B. Snyder. 2005. *Save the Cat!: The Last Book on Screenwriting You'll Ever Need*. Cinema/Writing. M. Wiese Productions.

Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. 2017. Cold fusion: Training seq2seq models together with language models. *CoRR*, abs/1708.06426.

Pradyumna Tambwekar, Murtaza Dhuliawala, Lara J Martin, Animesh Mehta, Brent Harrison, and Mark O Riedl. 2019. Controllable neural story plot generation via reward shaping. In *IJCAI*, pages 5982–5988.

Scott R Turner. 1993. *Minstrel: a computer model of creativity and storytelling*. Ph.D. thesis, University of California, Los Angeles.

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7378–7385.