Research Proposal of A Story Generation System Driven by Deep Learning in Digital Game Development

Rui He

1 Overview

This proposal aims to propose a system incorporating story description, story generation and story visualization in the context of digital game development. Having reviewed relevant research work, we found those results could not be applied directly to generate story in digital game. Referring to conventional game development pipeline, we introduce a plausible scheme based on deep learning to address this issue and elaborate the rationale behind it. We believe our results will not only contribute to relevant work in game development but also to those in drama and movie production. A plan to implement this scheme is briefly formulated at the end of this proposal.

2 Intention

Story generation has been a long studied issue in the communities of computational narrative and interactive narrative. Nevertheless, most of the research results can not be directly applied to relevant work in digital game development. Based on the unique form in which stories are told in digital game and the conventional digital game production pipeline, this proposal aims to propose a new system to describe story, study story patten, generate story and visualize story. In particular, we intend to work on the following aspects.

1) Design a diagram language to describe story.

2) Abstract semantic information from the diagram description, encode the information and feed it to the recurrent neural network(RNN) to learn the story patten therein. As the trained network will generate new story in code sequence, and the system should transform it back to the diagram description.

3) Transform the diagram description into computer graphics(CG) animation script file. This file can be played in real-time in game, and also loaded into an animation editor, viewed and revised by artist.

4) Enable the system to support "conditional" story generation interactively: Human designer produces story starting, mid-event or ending event, and player and the system produce events in between to form a complete and coherent story line.

These work are supposed to contribute to the research areas of story generation, game development and movie production in the following aspects.

1) Comparing to text description, diagram description enables a simpler, while vivid and accurate way to depict story, facilitating relevant work for game designer, artist and programmer at the same time.

2) Unlike text description, diagram description will not rely on natural language analysis, thereby enabling an instant while accurate abstraction of those semantic information to train the RNN.

3) As diagram description could contain or infer those significant animation data like temporal and positional information, it can be transformed into CG script conveniently.

4) The system will help to build up next generation interactive storytelling game, enabling player to get customized experience by procedurally generating new story in game.

5) The system will cooperate with game designer, drama script writer and movie director to produce story work, improving their working productivity and efficiency.

3 Terminology

Before we review relevant research work, definitions of some commonly used terms in the story research community will be listed in this section to avoid ambiguity in their meanings.

1) Narrative is to tell the content of a story in a particular way.

2) Storytelling is the way in which a story is told.

3) Story is the main content of narrative; it includes story plot and story space where the narrative takes place.

4) Story plot consists of a set of events and a structure formulating the temporal and causal relationship between these events.

5) Story space is the setting of the background where the plot happens; it could include character setting, relationship among characters, environmental setting and etc.

6) Story plot planning refers to a process formulating plot structure in story.

Figure 1 organizes the relationship of these terms in diagram structure.

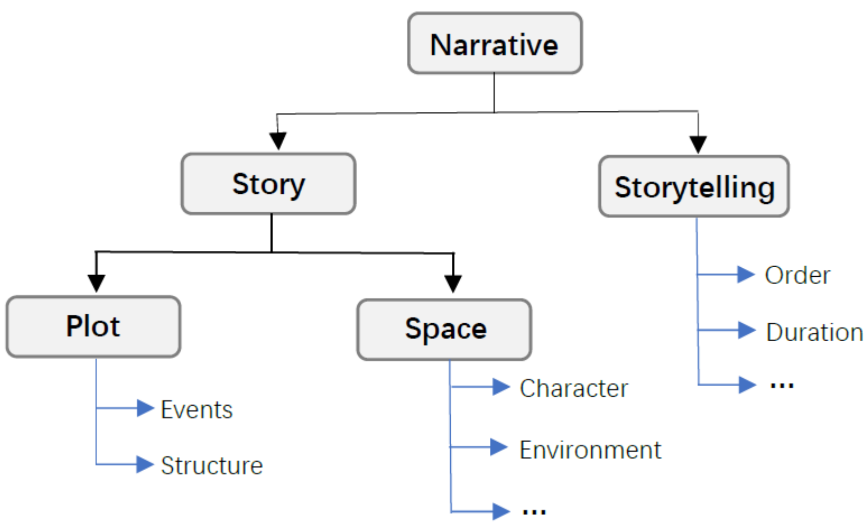

Figure 1: Diagram structure of narrative.

4 Background

Besides purpose for entertainment, storytelling has been seen as a driving force of the cultural development of humankind. Dating back, story, as the subject of the storytelling process, is structured and told by its exclusive form. This evolving form has always carried some indispensable elements to bring about dramatic effects, touching audience spiritually. In recent decades, besides text form narrative, storytelling is more being expressed via visual media: movie, video game and etc. Powered by the development of graphics technology, we have witnessed how these medias have evolved and reshaped audience's sensory experiences from time to time. Nevertheless, every time when people get used to a novel technical breakthrough, they will shift their focuses away from the technology and attach it back to the fundamental form of the pure story. In this sense, we can say it is the development of the story form that is determining the development of storytelling. Interactivity, is such a new element that numerous storytellers have dedicated to incorporate into the basic form of story to make storytelling more emotionally customized to audience. Among these efforts, a great many of them have reached substantial commercial successes.

Before 1990s, Infocom was one of the most influential producers of interactive storytelling game via a text-based interaction system: interactive fiction, where the story is told interactively in text on computer. Player acts by inputting sensible text command into the system to direct the movement of the story. As graphics technology progressed, in particular in this decade, interactive storytelling tends to be expressed through realtime animation. Quantic Dream produced a series of video games: *Heavy Rain*[6], *BEYOND: Two Souls*[7] and *Detroit: Become Human*[8], enabling player to choose favoured action to take from a list of options at every plot diverging point in game. These choices will further direct the story-line to move toward different plot branches. In recent booming big world games, like *The Legend of Zelda:Breath of the Wild*[22], player gets the experience of interactive storytelling by exploring the game world freely and collecting sub-stories or main story fragments.

Nevertheless, despite player's involvement in determining the story plot they have experienced in the above games, the plot they experienced are manually prepared but not generated. All of the plot branches(usually organized by plot tree), sub-plots or main plot fragments entail elaborate manual authoring. As player plays thoroughly through the whole plot space, they will get nothing new next. In this sense, the player has not experienced a real interactive story, but just interactively experienced a static story tree or story collections. Like all traditional storytelling games, these games still restrict players' inner desire for creating their unique customized story by a static story framework.

According to Chris Crawford on Interactive Storytelling[4], interactivity, which refers to audience's or player's control over the movement of story plot, always conflicts with the existed one controlled by the story author. Therefore, to generate a genuine interactive story, story author should not build up any concrete plot in advance, but turn to build those fundamental settings like story space and rules that govern the formation of plot, thereby enabling player to interact upon these settings and exclusively create a customized story plot.

In recent decades, numerous research work has devoted to study the formation, generation and visualization of story plot. We will review those most significant in section 3.1 and a potential new research direction in this field for a particular use in digital game development will be discussed in section 3.2.

4.1 Related Research

At an early stage of studying plot planning, Propp[12] built up a function model defining the grammatical structure of story plot by analyzing Russian folk tales. As is shown in Figure 2, each of the symbol in this function represents a unique element that often occurred in Russian folk tales. For instance, as W stands for wedding, this function indicates that most Russian folk tales ended up with the main character getting married. Lakoff[10] further incorporated rewriting rules into Propp's model, expanding every symbol in the function with a concrete event to generate a concrete plot. On the other hand, Grasbon and Braun[11] built an authoring tool for interactive fiction production based on Propp's model. Similar to Lakoff, this system rewrites a symbol in Propp's function by a manually

$$S \to ABC \uparrow DEFG \frac{HJIK \downarrow Pr - Rs^0 L}{LMJNK \downarrow Pr - Rs} QExTUW$$

Figure 2: Propp's grammatical function model summarizing the common pattern of Russian folk tales.

authored event according to players's input. This process is illustrated in Figure 3. Champagnat's

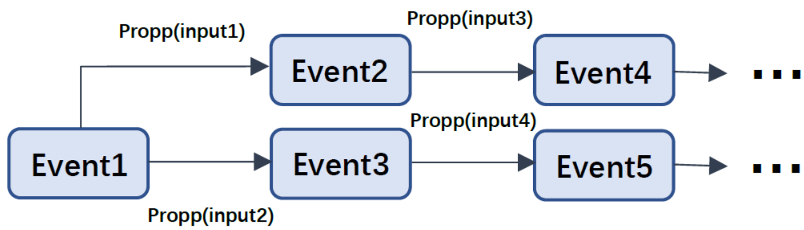

Figure 3: The plot structure of Grasbon and Braun's authoring tool for interactive fiction production.

system[3] is similar to Grasbon's, except that the function model was built based on Joseph Campbell's theory of Hero's Journey[2]. In addition to plot model and the rewriting events, Pemberton[23] added a relationship model among the characters in story, influencing a character's likelihood of involving in a plot event. REGEN[13] organized characters' relation via a graph structure and introduced character goals, which will interplay with the generated plot. Based on REGEN, UNIVERSE[14] further exerted an overall goal over the generation process. This goal was often expressed by a desirable ultimate state of characters' relationship at the end of the story plot. TALE-SPIN[20] built up more complex character model by introducing the personalty attribute which will influence the character's goals and actions. Facade[18] interactively built up and exhibited a story through real-time graphics animation and is often considered as the most successful attempt in building up commercial interactive story generation system. In this system, player acts as a close friend of a couple, whose personalities are modeled in depth. Player inputs text to interact with the couple, and this will lead to a consequent plot. To guarantee dramatic effects, this system exerted a dramatic arc over the generation process. The arc includes a series of transformations of character value(see Robert McKee's work[19] for in-depth definitions of these terms). Every event the system generates will reflect these transformations from time to time. Figure 4 is a sample scene of this system.

With the advent of artificial intelligence, the focus of relevant research work turned to study an automatic method for plot planning, thereby substituting previous tedious while complex manual work. Passage[27] regarded story planning as a process of patten recognition. Story plot in this system appear as a series of quests. Player's actions in every quest will be studied and summarized as a pattern. This pattern will be further classified to match a consecutive quest. This system made the deign of plot structure model into an automatic process, but those quests rewriting the model still entail manual authoring. Crowd source methods collects stories of a specific genre from different writers and builds up an according data set. To generate a new story of this genre, the plot planning process relies heavily on the rules learned from the set. SCHEHERAZADE[15] derived a story tree with different plot branches directly from the data set, while Swanson and Gordon's system[26] cooperated with human author to write every line of a new story in turn: human author writes a line, and the system writes the next line by selecting one from the data set that matches it most according to the learned rules. Instead of matching event or line, Matthew Paul Fay's work[9] in MIT applied matching strategies over characters' personalities. Based on a story corpus, Matthew's system classified characters in the corpus into distinct role type according to their personalities and similarities of their respective plots in story. A role type will hold a main plot line with a sequence of representative events. Before generating a new story, user of the system should define several characters with labels specifying their personalities, and the system will further match each character to an according type, and generates a plot line for

Figure 4: A sample scene of the Facade system.

the character based on those representative events of this type. Finally, these character lines will be woven into the final story line.

As machine learning develops, in particular in recent years, relevant advanced methods are introduced to drive the planning process. Rowe's work[24] incorporated reinforcement learning into the planning by choosing every consecutive event as a Markov decision process. During the training process, player's qualitative evaluation to his experience in an event determines the value of the Markov value function, and the planning model will be built up procedurally. Deep learning enables the modeling of nonlinear data. Among these methods, recurrent neural network(RNN)[21] differs from other network structures by the capability of memorizing, understanding and generating sequential data like text, speaking, music and etc. Therefore, numerous research devoted to study the generation of story text based on RNN, as it makes plot planning and event generation possible at the same time. Figure 5 summarizes the structure and working flow of RNN in handling story text. In this case, text of the fairy tale, Alice in Wonderland is used as the training set to train the RNN. As the training process completed, the RNN is able to generate new text similar to those in the original story text.

However, these generated results based on the original story corpus are not coherent and sensible enough. The reason lies in RNN's characteristic in understanding sequential character's interrelation: word or character level encoding makes the training text redundant to grasp the interrelation between sentence to sentence, and event to event inherently; Sentence level encoding, On the other hand, often produces distinct codes lacking semantic overlaps and the training results can hardly gasp these semantic relationships. Event representation method[17] developed by the school of interactive computing at Georgia Institute of Technology handled the above issues by transforming every sentence in the story text into a semantic tuple: (*subject, verb, object, modifier*). Encoding each element in the tuple independently, the tuple could grasp the semantic information of the original sentence by only four codes. What is more significant is that the abstracted tuples still contain rich semantic overlaps: every value of an element could occur in two or more tuples. In practice, the system collected movie plots from Wikipedia[1], represented the plots text into tuple representations, encoded them and train the

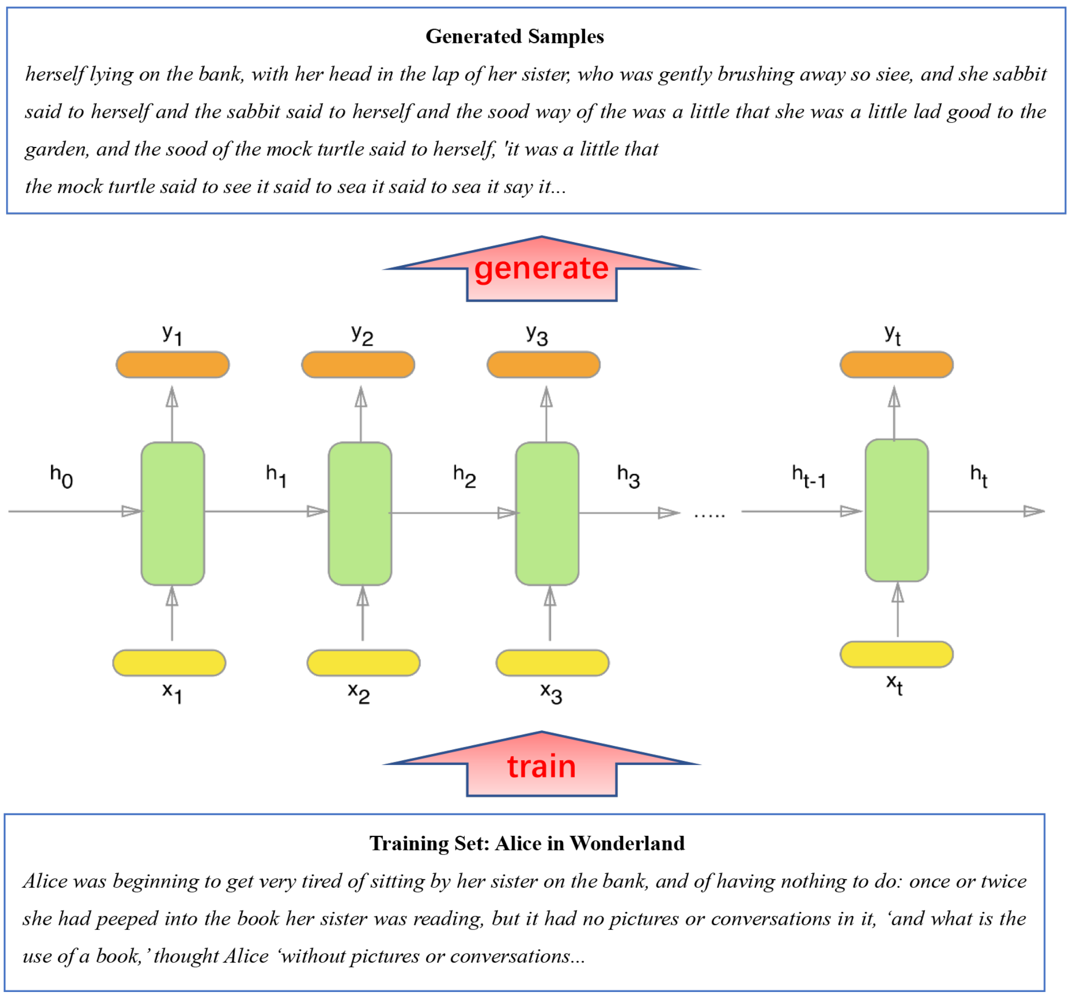

Figure 5: The structure of RNN and one of its working flows: learning the story pattern of Alice in Wonderland and further generating similar texts.

RNN. As the training completed, the RNN would be able to generate new story plot in tuple format. Finally, these results would be analyzed and transformed back to natural language text. The working flow of this system is illustrated in Figure 6. As is shown, despite the merits of this system, it relies heavily on natural language analysis and generation process which determine the semantic accuracy of an abstracted tuple and its corresponding natural language text.

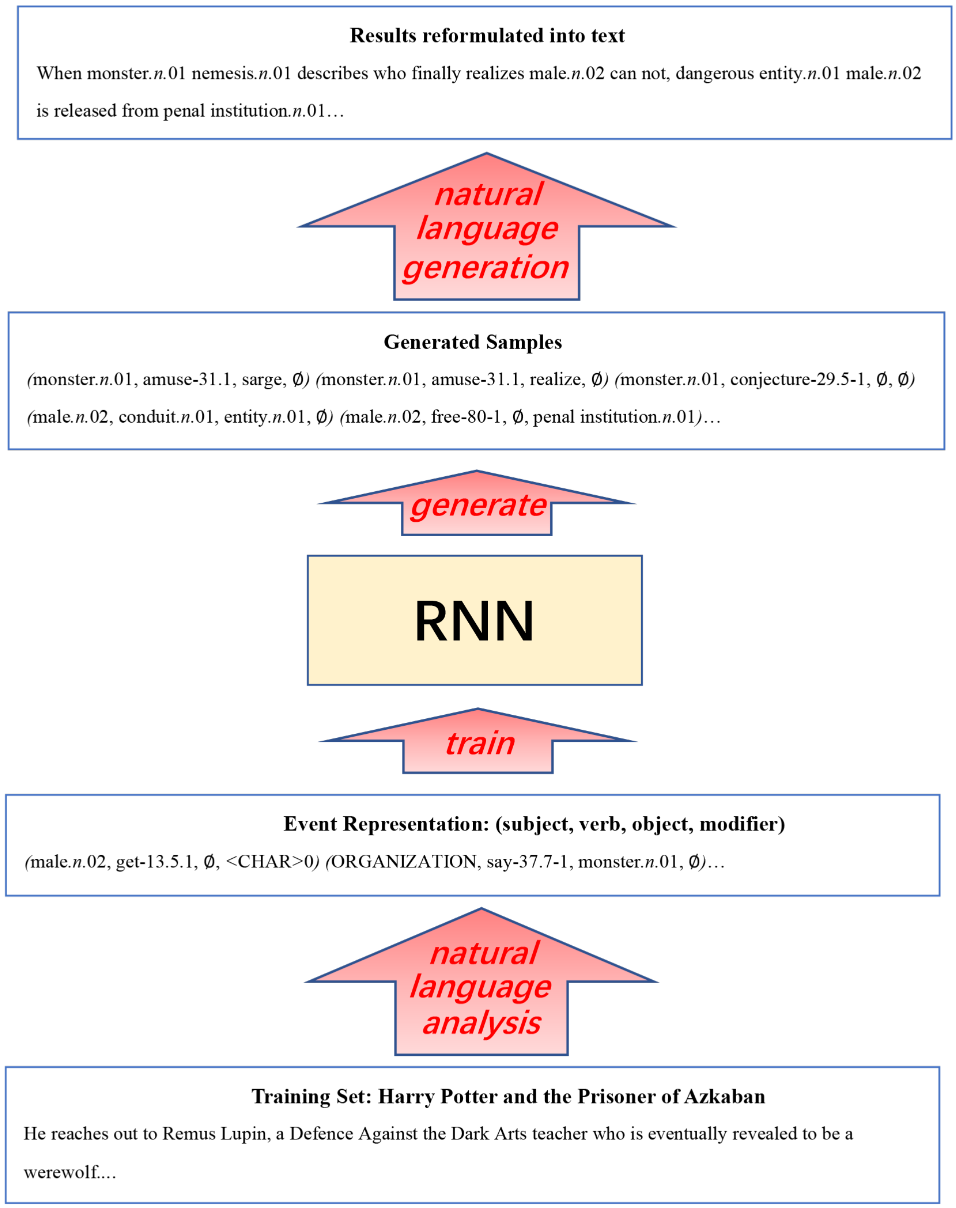

Figure 6: The working flow of RNN training and generating story plot based on event representation.

Besides the efforts in plot generation, many of others have turned to find ways to visualize plot text and some of them have shown plausible results. Delgado[5] searches suitable images or videos on the Internet to match specific plot in story. Nevertheless, the searching results turned to be accurate only for news text where relevant information of the time, place and people are actual and can be easily found on Internet. For fictional story plot however, in particular for those where metaphors and allusions thrive, the system can hardly produce plausible results. SCRIPTVIZ[16] enables story script writer to transform their text describing a scene into CG animation instantly and see the rehearsal performance. Specifically, a writer specifies a scene thoroughly via formalized written text, including the place, involved characters, their emotional states, behaviors and etc. The system further transforms the text into playable animation script with tracks combining those key frames specifying character behavior or conversation at the right time point. Figure 7 exhibits an animation instance produced by this system.

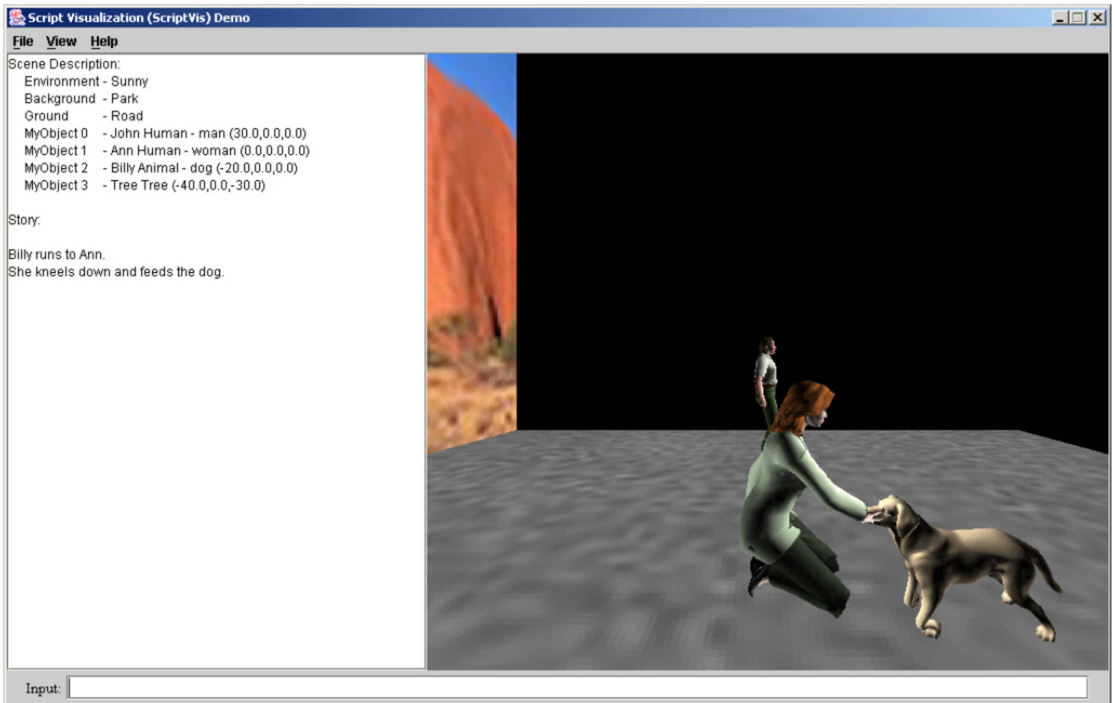

Figure 7: An animation instance produced by SCRIPTVIZ.

4.2 New Research Direction

We surveyed those most representative research work for generating story in the last section. However, We can discern that all of the work mentioned above are restricted to generate or understand story in text form. In nowadays digital game, as story is expressed through graphics animation, we can hardly apply these text-based work directly to generate story in animation. Although system like SCRIPTVIZ offers a way to transform text into animation script, it requires rigid text formulation and complex natural language analysis algorithm to abstract those inherent semantic information, and this could significantly reduce the accuracy and efficiency in understanding and generating story. Moreover, as story plot always entails large amount of text description by game designer, it is not vivid for game artist and programmer to read and implement relevant demands, in particular for those text with ambiguity. Therefore, we can say these text-based methods can hardly fit for generating story in digital game and its production pipeline.

What we should turn to address therefore, is to design a vivid, accurate while efficient form to describe story for human author and also for the system for training and generating story at the same time.

5 Methodology

As is introduced in the last section, it is essential to design a form to describe story in the context of digital game development. We define this form as a new story description language, and this language should fulfill the following conditions.

* Comparing to text description, our language should illustrate a story's grammatical structure and elements explicitly enough, thereby facilitating the abstraction and encoding of those semantic information, and the training of RNN.
* Our description language should be capable enough to describe most of the story scenario.
* To transform the description language into CG animation script, the description should contain or be able to infer those indispensable animation information like temporal and positional information of character action or conversation.
* In conventional digital game development pipeline, the production of CG animation involves the cooperation between game designer, artist and programmer. Therefore, our description language should be intuitive, while accurate enough for all of them to access.

Conforming to these conditions, we propose a diagram-based story description method in this proposal as the prototype of the language.

According to Robert McKee's specification of story structure, a story is composed by several acts. Each of the acts include a sequence of scenes, each scene encompasses a sequence of beats that procedurally change the inherent values in the story space and this type of scene thereby becomes an event. Our language will follow this structure and a scene will be the basic unit of our description. In Figure 8, we apply this diagram description language to describe the overall story structure of *The Legend of Zelda:Breath of the Wild* as an instance. Similar to event representation form, this description could be transformed into the semantic representations shown below.

scene1: (*Ganon, fight,* {*Link, Zelda, Sage*}), (*Ganon, defeat,* {*Link, Zelda, Sage*})
scene2: (*Link, resurge, Ø*), (*Link, awaken, Sage*), (*Sage, resurge, Ø*), (*Sage, empower, Link*)
scene3: (*Link, fight, Ganon*), (*Link, defeat, Ganon*), (*Zelda, resurge, Ø*)

However, unlike the original event representation process which entail natural language analysis over the written text to abstract value of each semantic element, the transformation from diagram description to the according semantic representation will be instant, as the grammatical structure and element value therein have been illustrated explicitly in the diagram. In addition, instead of representing a story sentence by sentence, we represent a story beat by beat to get rid of those redundant description in text, making our represented results much more compact.

In practice, given the story space setting and a specific story genre, we could build up the training set by collecting story samples written in diagram language by internal game designers or from external crowd source. After transforming story samples of the training set into semantic representations, we encode them by assigning a unique code to each value of the element in the tuple, and further input them into the RNN for training. The trained RNN will generate new story also in the form of semantic representation. For instance, training the RNN with samples like the given one in Figure 8, we are supposed to get some generated new stories shown below.

(*Ganon, fight, Link*), (*Link, defeat, Ganon*), (*Ganon, resurge, Ø*), (*Ganon, empower, Ganon*), (*Ganon, defeat, Link*), ({*Zelda, Sage*}, *awaken, Link*), (*Link, resurge, Ø*), (*Link, defeat, Ganon*)

(*Sage, empower, Ganon*), (*Ganon, defeat, Zelda*), (*Link, defeat, Sage*), (*Link, fight, Ganon*), (*Link, defeat, Ganon*), (*Link, awaken, Zelda*), (*Zelda, resurge, Ø*)

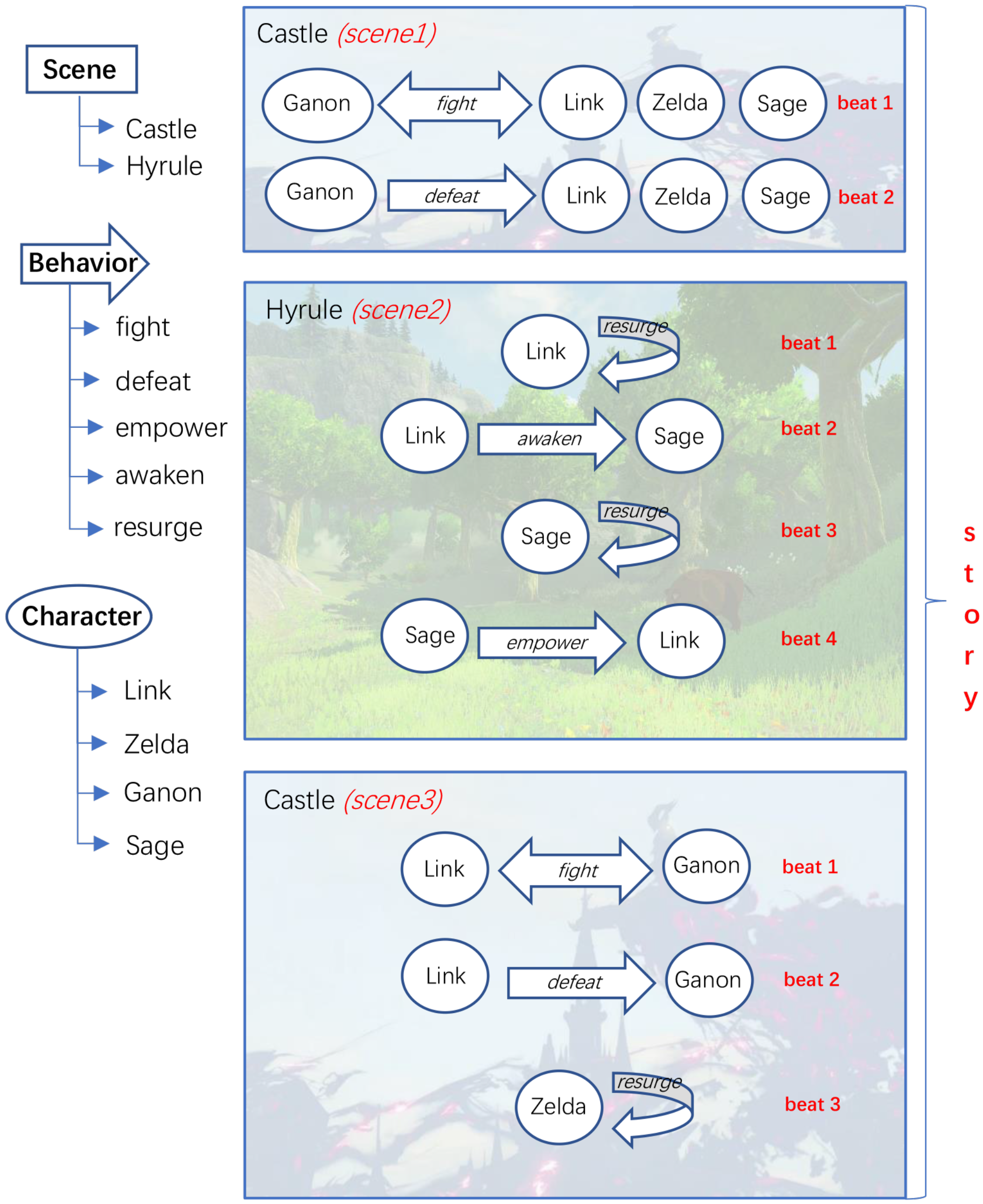

Figure 8: An instance of using diagram description language to describe a story.

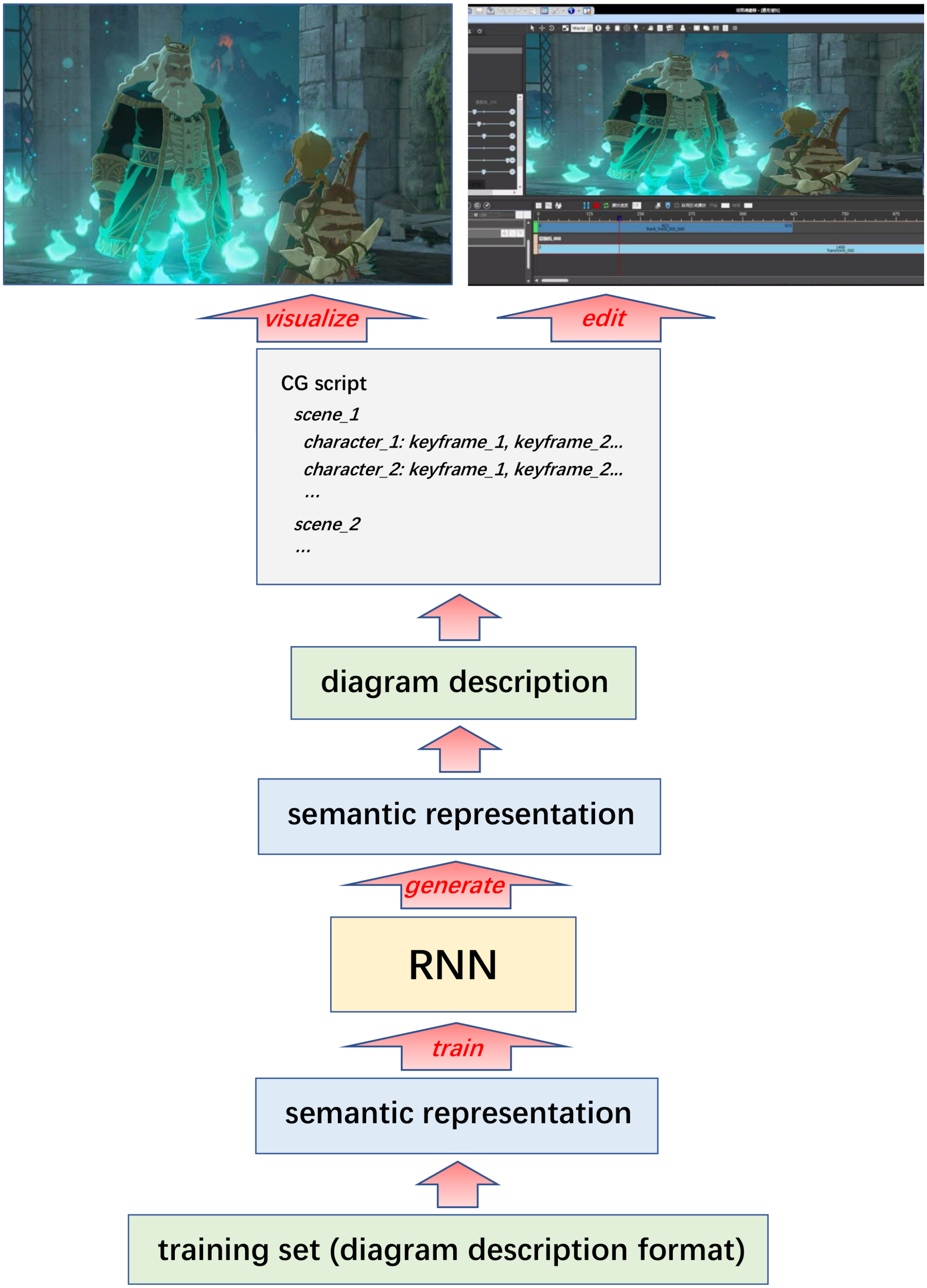

Figure 9: Pipeline of our whole story generation and visualization system.

(*Zelda, empower, Link*), (*Sage, empower, Ganon*), (*Link, fight, Ganon*), (*Link, defeat, Ganon*)

...

These results can be transformed back to diagram form for reading or editing. To actually visualize the generated story, we could further transform the generated results into CG animation script and play it through a cinematic system developed based on a game engine(Unity or Unreal) or revise them on an animation editor. Comparing to SCRIPTVIZ, our visualization process is based on explicit diagram elaboration, therefore will not depend on natural language analysis over the description text also. Moreover, we will evolve the visualization to make the style of the generated animation more close to movie narrative. In Figure 9, we depict the pipeline of our system encompassing all of the above components.

Lastly, to generate customized story plot in interactive storytelling game, player could cooperate with our system to create every event in turn and procedurally build up a completed while coherent story line. To control the overall plot movement, we will enable game designer to intervene in the generation process by inserting a manually designed story starting, ending or mid-event into the story line in advance as conditional event. All of other events created by player or the system will be restricted by these conditions. This feature will be built based on Bi-directional Recurrent Neural Network(BRNN)[25], and is briefly shown in Figure 10. Finally, we foresee several possible difficulties

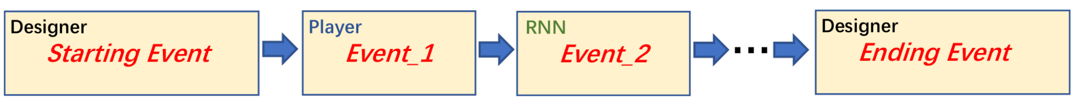

Figure 10: Conditional story generation process.

in developing this system.

1) The description language should be capable of representing complex story plot while still keeping the description compact, intuitive and editable.

2) The semantic representation form should include all of those indispensable semantic information, while be compact in its length to facilitate RNN's understanding.

3) How many samples are needed to build up the training set to let the RNN fully understand the inherent story pattern?

4) How to incorporate those rules governing movie narrative into the story visualization process?

References

[1] David Bamman, Brendan O'Connor, and Noah A Smith. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, 2013.

[2] Joseph Campbell. *The Hero With A Thousand Faces.* Pantheon Books, 1949.

[3] Ronan Champagnat, Guylain Delmas, and Michel Augeraud. A storytelling model for educational games: Hero's interactive journey. *International Journal of Technology Enhanced Learning*, 2(1-2):4–20, 2010.

[4] Chris Crawford. *Chris Crawford on Interactive Storytelling (2nd Edition).* New Riders, 2012.

[5] Diogo Delgado, Joao Magalhaes, and Nuno Correia. Assisted news reading with automated illustration. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1647–1650. ACM, 2010.

[6] Quantic Dream. Heavy rain. [PlayStation 3/4 game] Sony Interactive Entertainment, 2010.

[7] Quantic Dream. Beyond: Two souls. [PlayStation 3/4 game] Sony Interactive Entertainment, 2013.

[8] Quantic Dream. Detroit: Become human. [PlayStation 4 game] Sony Interactive Entertainment, 2018.

[9] Matthew Paul Fay. *Driving story generation with learnable character models.* PhD thesis, Massachusetts Institute of Technology, 2014.

[10] G.P.Lakoff. Structural complexity in fairy tales. *Study Man,*, 1:128–150, 1972.

[11] Dieter Grasbon and Norbert Braun. A morphological approach to interactive storytelling. In *Proc. CAST01, Living in Mixed Realities. Special issue of Netzspannung. org/journal, the Magazine for Media Production and Inter-media Research*, pages 337–340. Citeseer, 2001.

[12] Melville Jacobs. Morphology of the folktale by v. propp; laurence scott. *American Anthropologist (New Series)*, 73(4):897–898, 1971.

[13] Ben Kybartas and Clark Verbrugge. Analysis of regen as a graph-rewriting system for quest generation. *IEEE Transactions on Computational Intelligence and Ai in Games*, 6(2):228–242, 2014.

[14] Michael Lebowitz. Story-telling as planning and learning. *Poetics*, 14(6):483–502, 1985.

[15] Boyang Li, Stephen Lee-Urban, and Mark Riedl. Crowdsourcing interactive fiction games. In *FDG*, pages 431–432. Citeseer, 2013.

[16] Zhi-Qiang Liu and Ka-Ming Leung. Script visualization (scriptvis): a smart system that makes writing fun. In *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 03EX693)*, volume 5, pages 2990–2995. IEEE, 2003.

[17] Lara J Martin, Prithviraj Ammanabrolu, Xinyu Wang, William Hancock, Shruti Singh, Brent Harrison, and Mark O Riedl. Event representations for automated story generation with deep neural nets. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[18] Michael Mateas and Andrew Stern. Procedural authorship: A case-study of the interactive drama facade. *Digital Arts and Culture (DAC)*, 61, 2005.

[19] Robert McKee. *Story: Substance, Structure, Style and the Principles of Screenwriting.* ReganBooks, nov 1997.

[20] James R. Meehan. Tale-spin, an interactive program that writes stories. In *Proc of the International Joint Conference on Articial Intelligence*, 1977.

[21] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*, 2010.

[22] Nitendo. The legend of zelda:breath of the wild. [Switch/Wii U game] Nitendo, 2017.

[23] Lyn Pemberton and Bn Qn. A modular approach to story generation. In *Conference on European Chapter of the Association for Computational Linguistics*, 1989.

[24] Jonathan Rowe, Bradford Mott, and James Lester. Optimizing player experience in interactive narrative planning: a modular reinforcement learning approach. In *Tenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2014.

[25] M. Schuster and K.K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 2002.

[26] Reid Swanson and Andrew S Gordon. Say anything: A massively collaborative open domain story writing companion. In *Joint International Conference on Interactive Digital Storytelling*, pages 32–40. Springer, 2008.

[27] David Thue, Vadim Bulitko, Marcia Spetch, and Eric Wasylishen. Interactive storytelling: A player modelling approach. In *AIIDE*, pages 43–48, 2007.