Prototyping a Narrative Structure Interface as a Basis for Human-Computer Co-creation of Narratives

Taisuke Akimoto

***Abstract*—This paper describes a human interface, called a narrative structure (NS) interface, for the co-creation of narratives between generative computer programs and humans. This study forms part of work-in-progress research on a co-creative narrative generation system. This system produces diverse narratives through continual narrative-generating chain reactions involving various agents, including narrative generation programs and human narrative creators. Generally, computational narrative generation deals with the structural representation of a narrative, rather than the surface expression as with a natural language. The main role of the proposed interface is to connect a human narrative creation to the internal narrative structure level. In particular, this interface provides a visual diagram of a narrative structure and methods of editing the visualized narrative structure. In this paper, we present a prototyping version of the NS interface and discuss the achievements and future research directions thereof.**

***Index Terms*—Artificial intelligence, co-creative narrative generation system, human interface, knowledge creation, narrative as knowledge, narrative structure.**

I. Introduction

A narrative is a general mode of organically representing events in the real or a fictional world. The concept of narrativity or narrative ability is assumed to be one of the foundations of the human mind and knowledge, from the perspectives of psychology and artificial intelligence (AI) [1]-[3].

Narrative creativity, in particular, is a significant utility ability for humans, and creative AI technologies have the potential to aid, automate, or expand various fields of human activity, such as entertainment, communication, and artifact design. In many entertainment fields, narrative is a common element in composing content (for example, novels, films, comics, Web videos, and computer games). Interactive narrative is a major area of computer-based narrative technologies for entertainment purposes [4]. In the future, AI-based creative systems may automatically create original entertainment content, such as novels and comics. Narrative is also a familiar and effective mode for expressing various types of information. For this reason, generating a narrative text from data; for example, sensor data from human actions in a room, has been of major interest in recent studies on natural language generation [5], [6]. In addition, when designing an artifact, software, or a business plan, a quality user experience story is a crucial task for creating a better product, as well as its appeal to customers [7], [8].

The computational modeling of narrative creativity, including its computer implementation, has long been a challenging area in the field of AI. Many perspectives, methodologies, and systems have been proposed for computational narrative generation: simulation-based modeling of characters' actions in a modeled world [9]-[12], structural modeling of narrative generation based on theories of cognitive science and narratology [13]-[16], and narrative generation modeling using case-based reasoning [17]-[19]. Realizing full-fledged narrative generation systems requires integrating various methods and the vast amount of knowledge available [20]-[22].

The studies mentioned have largely focused on modeling an independent or closed computer program for generating narratives. However, [23] proposed the overall design and concepts of a co-creative (also known as a socially open) narrative generation system, based on a close relationship with humans. Although this long-term research plan provides a new approach to AI-based narrative generation systems, it involves major challenges.

In this paper, we describe the prototyping of one of the human interface components, the narrative structure (NS) interface, of the co-creative narrative generation system. An outline of the system and the position of the NS interface is explained in the next section. Section III introduces the model of narrative structure representation in this system. Section IV details the prototyping of the NS interface, and section V discusses the achievements of the study and future work. Section VI concludes the paper.

II. Co-creative Narrative Generation System and the Position of the NS Interface

The co-creative narrative generation system proposed in [23] is designed as a collective system that produces diverse narratives, known as a collection of narratives (CN) through continual narrative-generating chain reactions involving various agents, including narrative generation programs and human narrative creators. An outline of the overall structure of this system is shown in Fig. 1. The role of generative programs is to compose new narratives by using existing narratives in CN as a central knowledge base. In contrast, humans (users) access CN based on their values or objectives, and appreciate contained narratives or create new ones by recomposing those that already exist. Narratives created by both programs and humans are added to CN.

A. Basic Design of the Co-creative Interaction

In this system, the co-creative interaction among the

Manuscript received September 13, 2016; revised December 12, 2016.

T. Akimoto is with the Graduate School of Informatics and Engineering, the University of Electro-Communications, Tokyo, Japan (e-mail: t8akimo@yahoo.co.jp).

doi: 10.18178/ijke.2016.2.4.072

generative agents (that is, the programs and humans) is mediated by narratives produced in the CN. A computer-generated narrative may have an expansionary effect on human narrative creations; for example, by inspiring or providing new ideas. Conversely, computer programs can expand their generation ability through acquiring knowledge or collecting materials from human-created narratives.

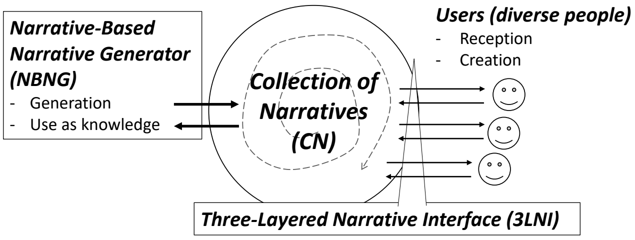


Fig. 1. Overall structure of the co-creative narrative generation system. (Source: Fig. 1 in [23])

Based on the above design, the task of generating narratives is formalized as structural recomposition (decomposition, transformation, and synthesizing) of one or more previous narratives, and is applied to both the programs and humans. This methodology is largely rooted in the concepts of "case-based reasoning" [24] and "intertextuality" [25]. Case-based reasoning is an AI model for solving a new problem by using previous cases or solutions from memory, and is also a major approach to computational narrative generation [17]-[19]. Intertextuality is a literary notion that any text is positioned as an organization and citation of related, preceding texts.

Furthermore, to generate diverse narratives, programs need a vast store of knowledge, such as linguistic, psychological, and common sense knowledge, as well as domain knowledge on narrative genres or styles. The described design includes a strategy for acquiring various types of knowledge from human-created narratives, based on the assumption that narrative itself is a comprehensive form of knowledge containing rich information. The knowledge aspect of narratives or stories from the perspective of AI is also discussed in [2], [26], [27].

B. Framework of the Human Interface Design

In the co-creative narrative generation system, the interface between the CN and humans is designed based on the three-layered narrative interface (3LNI) framework. The 3LNI consists of the following three functional layers:

1) **CN layer:** Collective view of the CN for exploring valuable narratives, in which the organization of CN is visualized in a two-dimensional space. Using this macroscopic perspective, a human explores and selects one or more narratives.
2) **NE layer:** Surface view of a narrative for appreciating it. This provides a surface expression of a narrative by using natural language or other expression media (for example, still/moving images).
3) **NS layer:** Internal view of a narrative structure for creating a new narrative. A human creates a new narrative by recomposing the visualized narrative structure.

The NS interface presented in this paper corresponds to the NS layer. The main role of the NS interface is to serve as a bridge between the internal representation of a narrative structure and its human-level representation, for creating narratives on the structural level. Narratives created by humans with this interface can be reused in the program side of the system. Fig. 2 illustrates the concept of the NS interface.

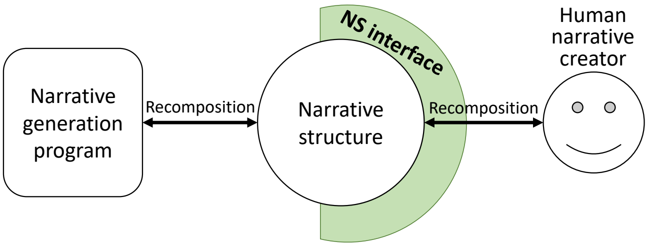


Fig. 2. Bridging generative actions of programs and humans via a narrative structure.

III. Narrative Structure Model

In this section, we define the structure of a narrative in the system. We make use of a hierarchical graph model [28], in which various types of narrative units are organized hierarchically, with network-like linkages. This model is outlined below.

A. The Three Structural Dimensions of a Narrative

A narrative is a complex object involving many structural elements. In the hierarchical graph model, narrative structure is divided into the following three dimensions:

- *StoryWorld*: The *background* world structure, containing information about entities (for example, characters, objects, and places).
- *Story:* The *structure* of chronologically organized events.
- *Discourse:* The structure of how a *Story* is organized into a surface *expression*.

The terms "story" and "discourse", as used above, have their roots in narratology [29], [30]. The former refers to the content plane (or *what* to narrate); the latter refers to the expression plane (or *how* to narrate) of a narrative.

In the computational modeling, each of the three dimensions is a symbolic representation. A surface expression with natural language, still or moving image, or other media is generated based on a *Discourse* structure. In other words, a *Discourse* is similar to the intermediate structure between content and expression. This type of intermediate structure is generally used in narrative generation systems as well as in natural language generation systems [31].

B. Structural Components of the Hierarchical Graph Model

The hierarchical graph model is applied to each of the three above-mentioned dimensions. A hierarchical graph consists of three types of structural component: a semantic element (*SE*), a *Group*, and a *Link*:

- The *SE* contains semantic information for the basic unit in each structural dimension of a narrative.
- The *Group integrates* any number of nodes (*SEs* and/or *Groups*) into a broader unit. This function is similar to the linguistic abstraction or categorization of the member

nodes. A *Group* itself also has semantic information, in the form of a comprehensive meaning that binds or abstracts the member nodes.

- The *Link* connects any two nodes with a relation or a set of connective *attributes*.

Fig. 3 shows an abstracted example of the structure with the three components. Although the components are commonly used in any structural dimension, the content of each type is different in every dimension. The subclasses of *SE*, *Group*, and *Link* for each dimension are defined in Table I.

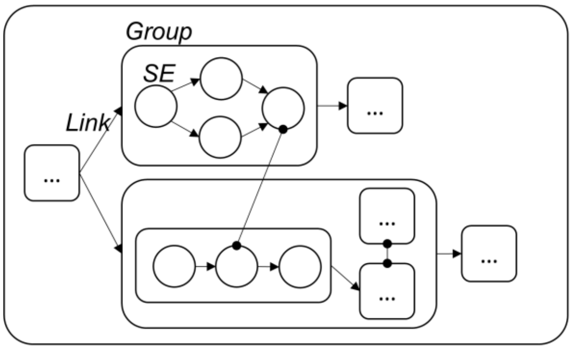


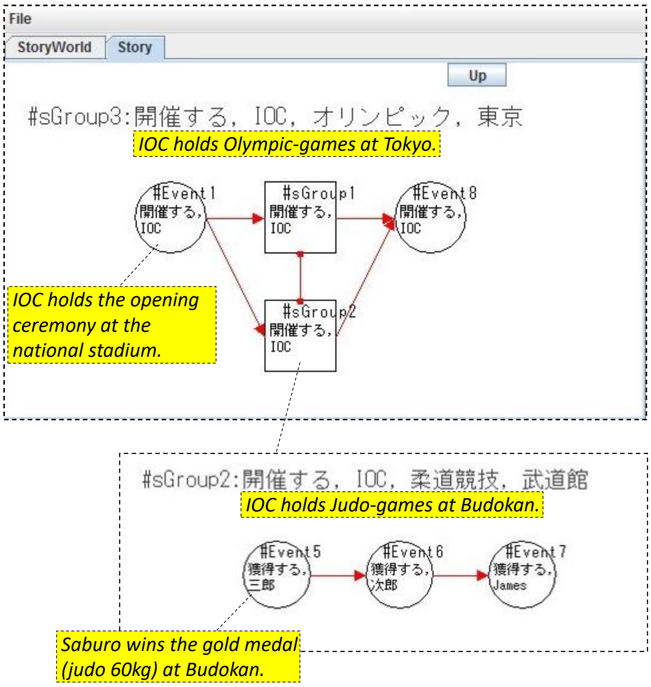
Fig. 3. Structural diagram of the hierarchical graph model. (Source: Fig. 3 in [28])

TABLE I: SUBCLASSES OF THE STRUCTURAL COMPONENTS OF THE HIERARCHICAL GRAPH MODEL

StoryWorld	*Entity (SE)*	An individual existence; for example, human, object, and place.
	wGroup	An aggregation of *Entities/wGroups*, such as a human group, a spatial unit, and functional kinship.
	wLink	A relation between two nodes; for example, interpersonal relations and spatial relations.
Story	*Event (SE)*	An action or happening.
	sGroup	An aggregation of *Events/sGroups*, such as a scene or a broader unit.
	tLink and cLink	A *tLink* is a relative temporal relation between two nodes for chronologically organizing a Story; for example, “before” and “equal.” A *cLink* is a causal relation between two nodes.
Discourse	*DU (SE)*	A minimum segment of a Discourse.
	dGroup	A broader unit of *DUs/dGroups* as a semantic or formal segment of a Discourse.
	dLink	A pathway between two nodes with connective logic. That is, the threads of a narrative are composed using *dLinks*, defining the order in which the DUs are expressed.

IV. FIRST PROTOTYPING OF THE NS INTERFACE

The NS interface includes two main functions: *visualization*, for providing visual expressions or diagrams of an internal narrative structure, and *editing*, for providing operational methods for a narrative structure. Using the hierarchical graph model [28], we prototyped an NS interface with Java. The interface includes a simple visualization function as well as several primitive editing functions for the *StoryWorld* and *Story* dimensions. Although usability—that is, the understandability and operability of narrative structures—is an important aspect of the NS interface, in this version of the system, the focus is on concretizing the systematic framework for its basic functions. Designing advanced functions for usability will form part of the next research stage.



IOC holds Olympic-games at Tokyo.

IOC holds the opening ceremony at the national stadium.

IOC holds Judo-games at Budokan.

Saburo wins the gold medal (judo 60kg) at Budokan.

Fig. 4. Visualization example of a *Story* structure.

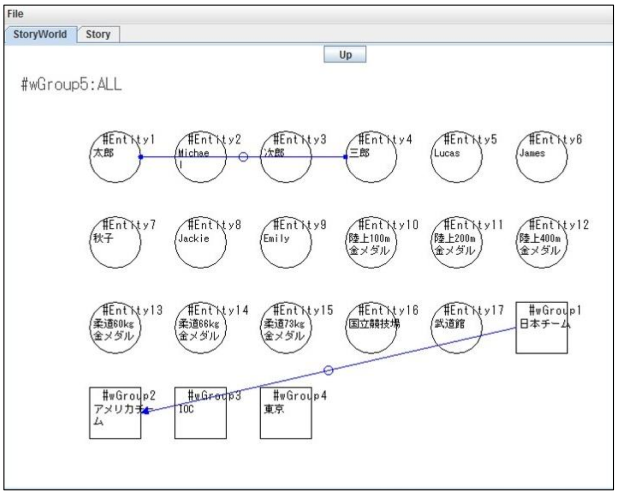


Fig. 5. Visualization example of a *StoryWorld* structure.

A. Visualizing the Hierarchical Graph Structure

The main interface window provides a diagram of the structure of a *StoryWorld or Story*, the tabs of which can be switched between. Fig. 4 shows the visualized structure of an example *Story*, based on the simplified fictional Olympic games story presented in [28]. Fig. 5 shows a *StoryWorld* visualization corresponding to this *Story*. In the visualized diagram, an *SE* and a *Group* are represented as a circle and square, respectively. Each node object has a label with its ID and keywords for semantic information: in particular, the verb and the agent for an *Event/sGroup* and the name or noun for an *Entity/wGroup*. A line between two nodes represents a *Link*; arrowed lines are directed links and those without arrows are undirected.

A window always depicts one layer in the grouping hierarchy at a time (in principle, multiple grouping of nodes is permitted in *StoryWorld* but not in *Story*). The user can move up and down the layer using the ‘Up’ button in the window and the ‘Down’ command in the pop-up menu for a *Group* object. In a *StoryWorld* visualization (Fig. 5), nodes are simply arranged from top to bottom and left to right. In a *Story* visualization (Fig. 4), the nodes are arranged in chronological order from left to right, based on *tLinks*.

In addition, the interface includes the functions of file input and output for created narrative structures: A user can save a created or edited narrative structure into a text file in the specified format, and a saved file can be opened at a later stage.

B. Editing Functions for Structural Elements

We implemented eight primitive operation types for the structural elements (see Table II). Possible target elements of each operation type are shown in Table III.

TABLE II: Operations for a Narrative Structure

Operation type	Explanation
Edit	Rewriting the semantic information of an object.
Remove	Removing an object.
New	Generating a new object.
Copy	Making a data copy of an object. The copied data is stored in an internal variable.
Cut	Copying and removing an object.
Paste	Generating a new object based on copied/cut data.
Elaboration	Transforming an SE into a Group that has the same semantic information as the original SE. The Group that is produced has no member nodes in its initial state.
Compression	Transforming a Group into an SE that has the same semantic information as the original Group. The members of the original Group are removed.

TABLE III: Possible Object Types of Each Structural Operation

	SE	Group	wLink	cLink	tLink
Edit	x	x	x	x	-
Remove	x	x	x	x	-
New	x	x	x	x	-
Copy	x	x	-	-	-
Cut	x	x	-	-	-
Paste	x	x	-	-	-
Elaboration	x	-	-	-	-
Compression	-	x	-	-	-

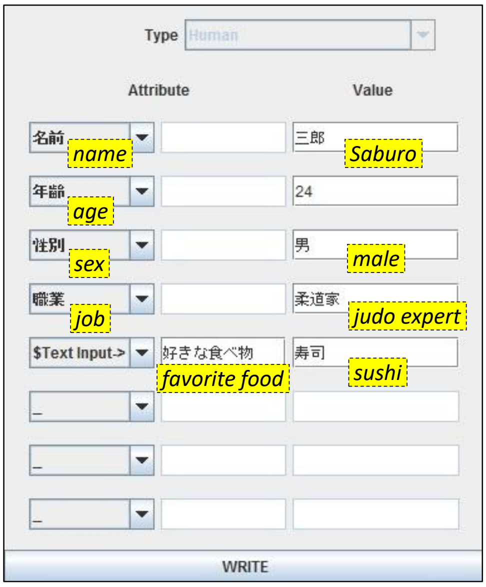
TABLE IV: Connecting Methods for a New Node in a Story (N: New Node; T: Target Node)

Type	Explanation
Add after	Adding N to the after position of T. When T originally has one or more after nodes, this action forms a *branched* structure.
Insert after	Inserting N into the after position of T. When T originally has one or more after nodes, the first one is moved into the after position of N.
Add before	Adding N to the before position of T. When T originally has one or more before nodes, this action forms a *branched* structure.
Insert before	Inserting N into the before position of T. When T originally has one or more before nodes, the first one is moved into the before position of N.
Equal	Adding N to the equal position of T.

A user can perform these operations via visualized structural objects. When a user clicks a visualized structural element, a pop-up menu with possible operations appears. ‘New’ and ‘Paste’ can be chosen from the pop-up menu when there is an empty space. A node newly generated by a ‘New’ or ‘Paste’ operation is added into the currently visualized *Group* (current *Group*). When adding a new node into a *Story* (using the ‘New’ or ‘Paste’ command), if there are one or more nodes in the current *Group*, the new node is connected to one of the existing nodes by a *tLink*. This target node is selected by the user by means of one of the five connecting methods listed in Table IV.

C. Editors for Narrative Structure

In principle, the semantic information of each element is represented in a frame format using Japanese words. The interface includes three types of editor: the *Entity/wGroup* editor, the *Event/sGroup* editor, and the *Link* editor. These are used in the ‘Edit’ and ‘New’ operations for each object.



Type Human

Attribute		Value
名前		三郎
年齢		24
性別		男
職業		柔道家
$Text Input->	好きな食べ物	寿司
_		
_		
_		

Fig. 6. *Entity/wGroup* editor.

1) Entity/wGroup editor

The semantic information of an *Entity/wGroup* in a *StoryWorld* is represented in a frame format that contains attribute slots and their values. Fig. 6 shows an example of a window for this editor. The *Entity* inputted in Fig. 6 corresponds to the information “(Human (name Saburo) (age 24) (sex male) (job judo_expert) (favorite_food sushi))”. This editor contains the following two fields:

1) *Entity type:* When carrying out the ‘New’ operation for generating an *Entity*, the user selects one of the following subtypes from the list: “Human,” “Animal,” “Plant,” “Artifact,” “NaturalObject,” “Food,” “Place,” and “Other.” This field is fixed when editing a previously created *Entity* (using the ‘Edit’ command).
2) *Attribute and Value:* An attribute consists of a slot name and its value. Eight lines are provided for setting attributes in this field. In each line, a slot name is selected from the candidate attributes in the left column or inputted as text into the center column. The value is inputted as text into the right column.

2) Event/sGroup editor

The semantic information of an *Event/sGroup* in a *Story* is represented as a case frame (verb-argument structure). Fig. 7 shows a window for this editor. The *Event* inputted in Fig. 7 example corresponds to "(Drink (agent Saburo) (object Sake) (location Budokan))" in the case frame format, which translates to "Saburo drinks sake at Budokan." This editor consists of the following three fields:

1) *Verb:* The user input a verb for the *Event* as text.
2) *Case and Value:* First, the user selects the required cases for the verb from "agent," "counter-agent," "object," "instrument," "location," and "time" (this is a tentative case list). Next, the user inputs the value of each case: basically, each case takes an *Entity* or *wGroup* from the *StoryWorld* as its value. The user can select one of these from the list in the center column, and if the value cannot be represented as an *Entity* or *wGroup*, the value can be input into the right column as text (for example, a word for an abstract concept).
3) *Time position:* As mentioned in the previous subsection, a new node is connected to another node by a *tLink*. For this field, the user selects a target node from the available nodes, and one of the connecting methods listed in Table IV. This field is not available when editing a previously created *Event* (with the 'Edit' command). (This field is also used in the 'Paste' operation for an *Event/sGroup*.)

3) Link editor for StoryWorld and Story

Fig. 8 shows a window for the *Link* editor, for editing a *wLink* in *StoryWorld* or a *cLink* in *Story*. This editor contains the following three input fields:

1) **Relation:** The user selects a relation type from the list of candidate relations or inputs as text.
2) **Node_1 and Node_2:** The user selects one of the available nodes for each of the starting and ending nodes.
3) **Direction:** The user chooses whether the Link has a direction or not.

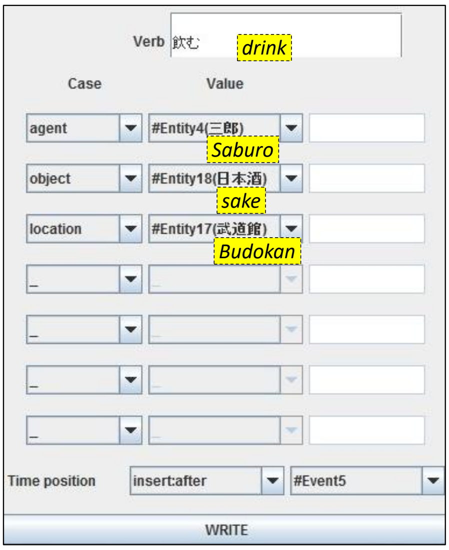
D. Elementary Mechanisms for Collecting Knowledge

The prototype interface includes two trial methods of collecting knowledge for composing narratives from human narrative creations:

- In the *Entity/wGroup* editor (Fig. 6), new attributes (slot names) inputted as text are stored in the "attribute base", which contains a list of candidate attributes for each entity type (for example, a "Human" *Entity* may have "age," "sex," "job," and "personality"). These lists are used in the attribute selection fields in the *Entity/wGroup* editor.
- In the *Event/sGroup* editor (Fig. 7), a combination of inputted verb and cases are generalized into a case frame, which consists of a verb and pairs of the case name and value type (for example, "Eat, agent=Human, object=Food"). These case frames are stored in the "verb base," and will provide generative programs with a type of semantic knowledge of verbs as a basis for composing new Events.

E. Example

Fig. 9 shows an example process of recomposing the *Story* in Fig. 4. In this example, detailed *Events* of the opening ceremony (corresponding to "#Event1" in Fig. 4) are constructed by using the 'Elaboration' and 'New' operations.

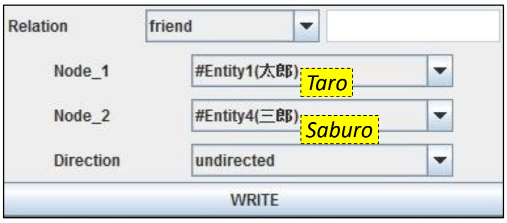


Verb 飲む drink

Case	Value	
agent	#Entity4(三郎) Saburo	
object	#Entity18(日本酒) sake	
location	#Entity17(武道館) Budokan	
_		
_		
_		
_		

Time position insert:after #Event5

Fig. 7. *Event/sGroup* editor.



Relation friend

Node_1 #Entity1(太郎) Taro

Node_2 #Entity4(三郎) Saburo

Direction undirected

Fig. 8. *Link* editor.

V. Discussion

In this section, we discuss the achievements of the study and outline future work on the basis of the design of the co-creative narrative generation system.

A. Concretization of the Basic Framework of the NS Interface

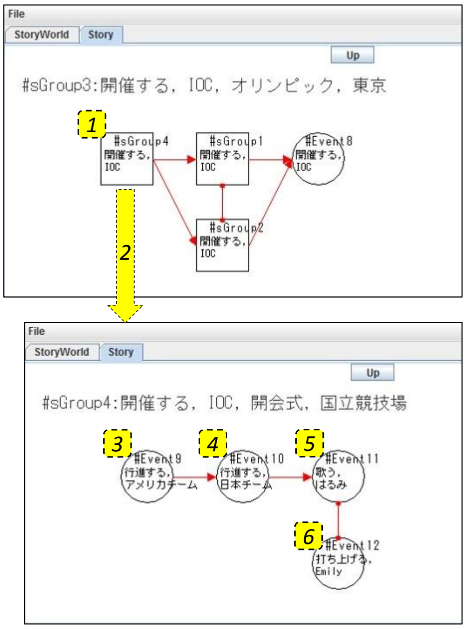
The benefits of this prototype are summarized as follows:

- We implemented a simple method of narrative structure visualization as one of the basic functions of the NS interface.
- The interface contains a set of primitive operations for editing a narrative structure.
- We implemented two simple knowledge collection mechanisms: "attribute base", containing lists of candidate attribute slots for characterizing *Entities*, and "verb base", with a list of case frames that serve as semantic knowledge of verbs for composing *Events*.

These functions are also important for the narrative generation programs for several reasons. Firstly, because the row data of a narrative structure is difficult for humans to understand, the visualization function will be useful for checking program-generated narratives at the structural level. Secondly, the functions of the primitive structural operators in a narrative structure can also be applied to the primitive methods in narrative generation programs. Sharing primitive structural operations among all generative agents is necessary for the formal unification of the holistic narrative generation process in the co-creative narrative generation system.

B. The Knowledge Collecting Mechanisms

Many researchers in AI have attempted to develop large-scale knowledge resources for providing a kind of common sense to computers; for example, WordNet (Japanese) [32], the predicate-argument structure thesaurus [33], ConceptNet [34], and the automatic learning of script-like event chains from corpora [35]. Such knowledge resources are also useful for various tasks in narrative generation. However, in the design of the co-creative narrative generation system, a key methodology is to position existing narratives as shared knowledge for generating new narratives, in order for the knowledge in the co-creative cycle to evolve continually. Hence, our central goal is to design a mechanism for systematically constructing various types of knowledge by reusing produced narratives.



1. Elaboration(#Event1)
→#sGroup4

2. Down (#sGroup4)

3. New (#Event9):
"American players march inside the national stadium."

4. New (#Event10,
Add after #Event9):
"Japanese players march inside the national stadium."

5. New (#Event11,
Add after #Event10):
"Harumi sings a song in the national stadium."

6. New (#Event12,
Equal #Event11):
"Emily shoots off fireworks in the national stadium."

Fig. 9. Example of a *Story* recomposition process.

Although we implemented simple mechanisms of knowledge collection in this prototype, designing a more systematic framework of knowledge collection is a key issue in the co-creative narrative generation system. Not only elemental knowledge, such as attribute lists and case frames, is necessary, but also more constructive knowledge or schema for composing a whole or part of a narrative structure. This constructive knowledge may include "thematic knowledge" for creating a dramatic story and "scripts" for providing detailed events in a specific situation, for example. The learning of these types of knowledge from narratives will be formalized as an abstraction or generalization of one or more concrete narrative structures.

C. Future Work for Enhancing Human Narrative Creations

Finally, we consider the developing of advanced-level functions for the NS interface. This prototype implemented the most basic visualization method, which provides plain diagrams of a hierarchical graph structure. Another possible method is target-oriented visualization, which provides a simplified expression of a specific structural aspect according to a human's creative goal. For example, when the goal is to recompose the human relationship among characters in a narrative, the interface exposes aspects of human relationships for supporting structural understanding and operations.

Next, we consider future extensions of the editors for narrative structure operations. Although the implemented editors use the internal frames of semantic representation of an *SE/Group* directly, transforming natural language input into semantic representation may be easier for humans. For example, when a human inputs an event as a simple sentence: "a dog with red eyes bites a child," the interface constructs an *Event* representation as "(bite (agent dog) (counter-agent child))". If "dog" and "child" are not contained in the *StoryWorld*, the interface constructs these *Entities*. In addition, the phrase "red eyes" is added as an attribute to the "dog" *Entity*. The techniques of recent natural language processing can be applied to this development.

VI. Conclusion

This paper described the concept of the NS interface with its prototype implementation, based on the design of a co-creative narrative generation system, as a basic component of human-computer co-creation of narratives. The key role of this interface is to provide a human-level representation of a narrative structure for sharing narratives between computers (narrative generation programs) and humans. The interface provides two main functions: narrative structure visualization and operational methods for editing a visualized narrative structure. In order to concretize the design of the NS interface, we developed a prototype version, including the basic methods for these two major functions. Our future work on the NS interface includes designing its advanced-level methods to enable human narrative creation to take place more smoothly and freely.

References

[1] J. Bruner, "The narrative construction of reality," *Critical Inquiry*, vol. 18, no. 1, pp. 1-21, 1991.

[2] R. C. Schank, *Tell Me a Story: Narrative and Intelligence*, Chicago, IL: Northwestern University Press, 1990.

[3] M. Mateas and P. Sengers, "Narrative intelligence," *Narrative Intelligence: Papers from the AAAI Fall Symposium, Technical Report FS-99-01*, 1999.

[4] M. O. Riedl and V. Bulitko, "Interactive narrative: An intelligent systems approach," *AI Magazine*, vol. 34, no. 1, pp. 67-77, 2013.

[5] A. Siddharthan, M. Green, K. Deemter, C. Mellish, and R. Wal, "Blogging birds: Generating narratives about reintroduced species to promote public engagement," in *Proc. the 7th International Natural Language Generation Conference*, pp. 120-124, 2013.

[6] P.-L. Vaudry and G. Lapalme, "Narrative generation from extracted associations," in *Proc. the 15th European Workshop on Natural Language Generation*, pp. 136-145, 2015.

[7] W. Quesenbery and K. Brooks, *Storytelling for User Experience: Crafting Stories for Better Design*, New York, NY: Rosenfeld Media, 2010.

[8] Y. Takeda, "Applying digital storytelling to business planning," in *Computational and Cognitive Approaches to Narratology*, T. Ogata & T. Akimoto Eds. Hershey, PA: IGI Global, 2016, ch. 5, pp. 140-163.

[9] J. R. Meehan, *The Metanovel: Writing Stories by Computer*, New York, NY: Garland Publishing, 1980.

[10] N. Okada and T. Endo, "Story generation based on dynamics of the mind," *Computational Intelligence*, vol. 8, no. 1, pp. 123-160, 1992.

[11] C. León, S. Hassan, P. Gervás, and J. Pavón, "Mixed narrative and dialog content planning based on BDI agents," *Current Topics in Artificial Intelligence: 12th Conference of the Spanish Association for Artificial Intelligence, LNAI 4788*, Berlin, Germany: Springer-Verlag, pp. 150-159, 2007.

[12] M. O. Riedl and R. M. Young, "Narrative planning: Balancing plot and character," *Journal of Artificial Intelligence Research*, vol. 39, pp. 217-267, 2010.

[13] L. Pemberton, "A modular approach to story generation," in *Proc. the Fourth Conference on European Chapter of the Association for Computational Linguistics*, pp. 217-224, 1989.

[14] T. Ogata and T. Terano, "Explanation-based narrative generation using semiotic theory," in *Proc. Natural Language Processing Pacific Rim Symposium 1991*, pp. 321-328, 1991.

[15] T. Akimoto and T. Ogata, "An information design of narratology: The use of three literary theories in a narrative generation system," *The International Journal of Visual Design*, vol. 7, no. 3, pp. 31-61, 2014.

[16] P. Gervás, C. León, and G. Méndez, "Schemas for narrative generation mined from existing descriptions of plot," in *Proc. 6th Workshop on Computational Models of Narrative*, pp. 54-70, 2015.

[17] S. R. Turner, *The Creative Process: A Computer Model of Storytelling and Creativity*, Mahwah, NJ: Lawrence Erlbaum, 1994.

[18] P. Gerv ás, B. D íaz-Agudo, F. Peinado, and R. Hervás, "Story plot generation based on CBR," *Knowledge-Based Systems*, vol. 18, no. 4-5, pp. 235-242, 2005.

[19] R. Swanson and A. S. Gordon, "Say anything: Using textual case-based reasoning to enable open-domain interactive storytelling," *ACM Transactions on Interactive Intelligent Systems*, vol. 2, no. 3, Article 16, 2012.

[20] S. Bringsjord and D. A. Ferrucci, *Artificial Intelligence and Literary Creativity: Inside the Mind of BRUTUS, a Storytelling Machine*, Mehwah, NJ: Lawrence Erlbaum, 1999.

[21] N. Montfort, R. P érez y P érez, D. F. Harrell, and A. Campana, "Slant: A blackboard system to generate plot, figuration, and narrative discourse aspects of stories," in *Proc. the Fourth International Conference on Computational Creativity*, pp. 168-175, 2013.

[22] T. Ogata, "Computational and cognitive approaches to narratology from the perspective of narrative generation," *Computational and Cognitive Approaches to Narratology*, PA: IGI Global, 2016, ch. 1, pp. 1-74.

[23] T. Akimoto and T. Ogata, "Designing a socially open narrative generation system," *Computational and Cognitive Approaches to Narratology*, Hershey, PA: IGI Global, 2016, ch. 3, pp. 91-117.

[24] C. K. Riesbeck and R. C. Schank, *Inside Case-Based Reasoning*, Mahwah, NJ: Lawrence Erlbaum, 1989.

[25] J. Kristeva, *Desire in Language: A Semiotic Approach to Literature and Art*, New York, NY: Columbia University Press, 1980.

[26] R. C. Schank and R. P. Abelson, *Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures*, Mahwah, NJ: Lawrence Erlbaum, 1977.

[27] C. León, "An architecture of narrative memory," *Biologically Inspired Cognitive Architectures*, vol. 16, pp. 19-33, 2016.

[28] T. Akimoto, "Toward a basic model of narrative structure representation for computer processing: Proposal of a hierarchical graph model," in *Proc. the International Workshop on Language Sense on Computer in IJCAI2016*, pp. 9-16, 2016.

[29] G. Genette, *Narrative Discourse: An Essay in Method*, NY: Cornell University Press, 1980.

[30] G. Prince, *A Dictionary of Narratology*, Lincoln, NE: University of Nebraska Press, 2003.

[31] E. Reiter and R. Dale, *Building Natural Language Generation Systems*, Cambridge, U.K.: Cambridge University Press, 2000.

[32] F. Bond, H. Isahara, S. Fujita, K. Uchimoto, T. Kuribayashi, and K. Kanzaki. "Enhancing the Japanese WordNet," in *Proc. the 7th Workshop on Asian Language Resources*, pp. 1-8, 2009.

[33] K. Takeuchi, "Thesaurus with predicate-argument structure to provide base framework to determine states, actions, and change-of-states," *Computational and Cognitive Approaches to Narratology*, Hershey, PA: IGI Global, 2016, ch. 7, pp. 192-212.

[34] H. Liu and P. Singh, "ConceptNet: A practical commonsense reasoning tool-kit," *BT Technology Journal*, vol. 22, no. 4, pp. 211-226, 2004.

[35] N. Chambers and D. Jurafsky, "A database of narrative schemas," in *Proc. Seventh International Conference on Language Resources and Evaluation*, pp. 1614-1618, 2010.

Taisuke Akimoto is a postdoctoral researcher in the Graduate School of Informatics and Engineering, the University of Electro-Communications, Tokyo, Japan. He received his Ph.D. from Iwate Prefectural University, Japan, in 2014. His research interests include artificial intelligence, informatics, and cognitive science. Especially, he interests automatic narrative generation system, computational modeling of narrative ability, and their applications. He is a member of the Japanese Society for Artificial Intelligence, Japanese Cognitive Science Society, and the Association for Natural Language Processing in Japan.