

# Story Representation in Analogy-Based Story Generation in Riu

Jichen Zhu and Santiago Ontañón

**Abstract**—Computational analogy offers a promising direction to algorithmically generating stories, a key challenge in computational narrative and AI. Since analogy methods are very sensitive with respect to the story representation being used, this paper focuses on story representation for analogy-based story generation. Specifically, we analyze existing story representation formalisms and propose a new approach based on the cognitive semantics theory of *force dynamics*. Finally, we present the results of our analogy-based interactive narrative system, Riu, to illustrate the utility of our proposal.

## I. INTRODUCTION

Computational narrative explores the age-old creative form of storytelling by algorithmically analyzing, understanding, and most importantly, generating stories. Because of its interdisciplinary nature, the field needs to address both technical and aesthetic challenges. On the one hand, the complex domain of story understanding and specially generation requires developments from various AI fields, such as knowledge representation, planning, learning, and natural language processing (NLP). On the other hand, computational narrative is identified as a potential art form of our time. In its full potential, computer generated stories should be able to depict a wide spectrum of human conditions and expressions with the same breadth and depth as traditional narratives [1].

Compared to these grand visions, computational narrative is still in its early stage. Although the field has made a lot of progress in the last decades, notably in planning-based approaches. Aesthetics has not been the focus of previous research, and as a consequence the stories that can be generated are still aesthetically limited. For example, the stories generated by planning often embody an unmistakable goal-driven, problem-solving aesthetics. Zhu & Ontañón [2] observe that different story generation techniques have specific built-in narrative affordances and constraints. In this paper, we explore a promising direction — computational analogy based story generation — to broaden the technical and aesthetic scope of the field.

As computational analogy methods are very sensitive to the knowledge representation they operate on, this paper focuses on story representation. We argue that existing story representations do not support the generation of many common analogy-based narratives in our culture. For example, the story of Seabiscuit in the 1930s is an important narrative

in the American history. Part of its popularity results from the clear analogy between the unimpressive racing horse, which eventually beat its formidable opponent, and the image of American people during the Depression. Another example is two parallel worlds, one real and the other fantastic, in the well-claimed film *Pan's Labyrinth* (2006). The analogical association between the two worlds intensifies their contrast and hence evokes audiences' emotional reactions to the main character. However, these widely-used analogies in traditional stories are difficult to achieve computationally using existing story representations.

To broaden the scope of analogy-based story generation, we categorize common story representations and analyze their affordances and limitations. Based on our observations, we propose a new representation built upon the cognitive semantics theory of *force dynamics* [3]. In the rest of the paper, we first provide the results of our survey of existing story representations. We then introduce force dynamics and why it can be useful for our purpose. Next, we evaluate its utility in our interactive narrative system Riu and compare our system with other related analogy-based systems. Finally we conclude our paper and propose several future directions.

## II. STORY REPRESENTATION IN NARRATIVE SYSTEMS

The choice of story representation has a profound impact on the effectiveness of the story generation techniques as well as the final story. This is particularly true for analogy-based story generation, as computational analogy techniques are very sensitive to the knowledge representation they operate on [4]. In this section we analyze a range of common story representations, emphasizing their affordances and limitations, especially when used with analogy. Although representation has to be considered in conjunction with generation techniques, our discussion here focuses on the built-in expressive affordances in these representations.

Common story representations can be classified into three broad categories. First, and most widely used, are *plans*. Whether manifested as total order plans (e.g. Tale-Spin [5] and Universe [6]) or the more recent use of partial order plans (e.g. Fabulist [7]), plan-based representations formalize stories as sequences of actions towards certain goals. For instance, Tale-Spin generates such sequences by decomposing a goal into more primitive ones using “delta-acts.” Universe does so by selecting predefined “plot fragments” that satisfy most of the currently open goals. More recently, Fabulist developed a more expressive representation based on partial order plans, which allows more flexibility in the sequence of actions. Certainly, all these systems include

Jichen Zhu is with The Department of Digital Media, University of Central Florida, 12461 Research Parkway Suite 500, Orlando, FL, USA 32826-3241 (email: jzh@mail.ucf.edu). Santiago Ontañón is with The Artificial Intelligence Research Institute (IIIA-CSIC), of the Spanish Council for Scientific Research (CSIC), Campus UAB, 08193 Bellaterra, Spain (santi@iiia.csic.es)

other information, such as character personalities and their relationships, but the key emphasis of plan-based representation is the *causal and temporal relationships* between events and actions, leading towards a certain *goal state*. However, a common story element such as conflict is difficult to represent with plans because they will be eliminated in the planning process [7].

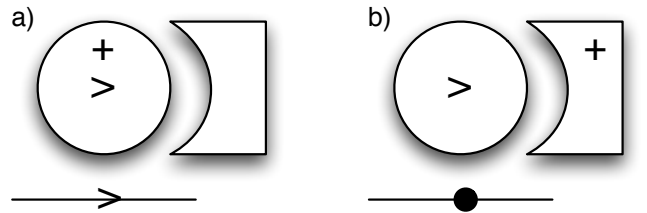
The second most common approaches are *frame-based* representations. Compared to plans, frame-based representations are not built around causal or temporal relationships. Instead, they focus on the properties of characters/setting and their relationship. For example, Minstrel [8] represents story fragments as graphs where the nodes are goals, states or actions whereas the links are their relations (e.g., “achieves”, “motivates”, “activates”, etc.). MEXICA [9] also uses a frame-based representation to formalize story states as *Story World Contexts*, where each node is a character, linked by their interpersonal relationships (e.g., “hates” and “loves”). Although this approach provides fine-grained control over these lower-level story elements and relationships, it is often used at the price of misrepresenting the higher-level story structure (although this is not an intrinsic limitation of frames themselves).

The third category consists of representations based on *plot-points* or *beats* common in interactive fiction systems. Plot-points [10] represent stories as a collection of events (plot-points), and their relationships (e.g., “plot-point 1” cannot occur unless “plot-points 2 and 3 have occurred but plot-point 4 has not.”). A branching story can be represented in a graph of plot-points, specifying the order in which plot-points can occur under different circumstances. In contrast, beats [11], [12] are self-contained story units with internal structure. More importantly, they are independent entities whose order is not specified until the beat sequencer (sometimes called the *drama manager* or the *director*) organizes them at runtime. Compared to the pre-authored graph of plot-points, beats provide better control over the desired content in the stories.

In summary, plans and plot-point graphs can easily represent temporal sequence and causality. But plans have troubles formalizing common story elements such as conflict [7] and plot-points are relatively inflexible. Frame-based representations are more flexible, but their past uses focus on the low levels, which makes it hard to find deeper analogies such as the ones found in the examples in the introduction. Finally, although we have focused on story generation, AI approaches to story understanding share most of these representations, such as Schank’s scripts (closely related to frames) [13] and plans. In the next section we will introduce *force dynamics*, which would fall into the *frame-based* category, but which focuses on representing a higher level interpretation of scenes in a story, rather than the low level details.

### III. FORCE DYNAMICS

Built upon his observation that a wide range of human linguistic and cognitive phenomena are structured based on how entities interact in regard to force, Leonard Talmy [3]



"Ales always wanted to be a painter, despite his long working hours. But his job got more demanding, and he eventually gave up his practice."

Fig. 1. Force dynamics in Talmy’s original notation.

defined the semantic category of *force dynamics*. Force dynamics captures fundamental structures such as “the exertion of force, resistance to such a force, the overcoming of such a resistance, blockage of the expression of force, removal of such blockage, and the like,” some of which are hard to represent under the traditional notions of causality.

A basic force dynamics pattern contains two entities, an *Agonist* (the focal entity) and an *Antagonist*, exerting force on each other. An entity has a *tendency* towards either motion/action or rest/inaction, and the stronger entity manifests its tendency at the expense of its opposer. To represent “The ball kept rolling because of the wind blowing on it,” for example, the Agonist *ball’s* intrinsic tendency towards rest is overcome by the Antagonist *wind’s* greater force, and hence the result is the motion of the Agonist. Force dynamics describes not only physical forces, but also *psychological and social interactions*. Conceiving such interactions as psychological “pressure,” force dynamics patterns can manifest themselves in various semantic configurations, such as the “divided self” (e.g., “He held himself from responding”) and complex social interactions (e.g., “She gets to go to the park.”) A *phase* in force dynamics describes the interaction between Agonist and Antagonist at a particular point in time.

Figure 1 illustrates a force dynamics diagram of a small episode in the Riu system: “Ales always wanted to be a painter, despite his long working hours. But his job got more demanding, and he eventually gave up his practice.” Each sentence here is represented in a different phase. In the first phase (left hand side of the figure), the Agonist (Ales, represented as a circle) has the tendency to move and is stronger than the Antagonist (his job). In the second phase (right hand side of the figure), their relative force strength shifts. The Antagonist strengthens and sets the Agonist at rest. In Talmy’s notation, the stronger of the entities is marked with a + sign. Moreover, the agonist is marked with a > when it has tendency to move, and with a black dot if it has tendency to rest. Under each phase there is a line, and in the line there is either a > or a black dot, representing that the result of the scene is movement or rest respectively.

Based on our previous discussion of existing story representations, force dynamics can enhance existing representations in several ways. First, causal relationships in story representations (e.g. plans) have been typically construed as “Event A causes Event B (not) to happen.” By contrast, force

dynamics captures more graded “causative” relationships, such as “letting,” “hindering,” and “helping.” For instance, in a plan-based representation, given that the necessary and sufficient conditions for an action to take place are satisfied, the rest of the state is irrelevant. However, there are situations where additional factors might “help”, or “push” a certain character towards doing some actions. These concepts are useful to various narrative scenarios, which would be cumbersome to represent otherwise. We’d like to remark that we are not claiming these ideas are impossible to represent using plans or predicate calculus, but that they would be cumbersome to represent.

Second, force dynamics’s level of abstraction affords deeper analogy. Most existing systems concentrate on characters, setting and actions. Without much knowledge of the overall plot, it is difficult for analogy to find structural similarities between stories. By abstracting out these surface information, force dynamics captures deeper force relationship, physical or psychological, between characters and settings. This leads to the deeper, structural analogies in Riu, as we will show later. Again, we’d like to remark that our claim is not that analogy using other representation formalisms cannot find deep analogies. Notice that actually force dynamics will be represented using a frame-based representation, thus, what force dynamics brings to the table is only a well founded set of terms (agonist, antagonist, stronger, tendencies, etc.) which can be used to represent a wide range of situations in a uniform way (and thus suitable for analogy). As pointed out by Hofstadter and Mitchell [4], systems like SME “rely on a precise and unambiguous representation of situations in the language of predicate logic.” For example, if we were to represent color as an object in our scene and as an attribute in another scene, SME would never find a mapping among them.

Third, force dynamics’s “psychological plausibility” lends itself greater expressive power in terms of character’s inner world: in a force dynamics diagram, the intentions of the agonist and antagonist, and their relation are explicit. The goal of our Riu system is to go beyond the goal-driven aesthetics and explore narrating characters’ psyche and other subjective experiences. Force dynamics’s built-in descriptive power of psychological and social interaction is a great asset towards this goal. Most importantly, force-dynamics-based representations offer a natural way to represent conflict with the concept of Agonist and Antagonist.

#### IV. THE RIU SYSTEM

Riu is a text-based interactive narrative system that uses analogy to generate stories in order to explore new narrative spaces. It creates stories about a robot character Ales, who has initially lost his memories. Similar to the character in *Pan’s Labyrinth*, Ales constantly oscillates between his recovering memory world and the main story world (real world). The two worlds not only share parts of the structure, but also influence each other. Events happening in the memory world may impact the development in the real world. Riu explores the same story world as Memory, Reverie Machine

One day, Ales was walking in an alley.  
when he saw a cat in front of him.  
*Ales used to play with a bird when he was young.*  
*Ales was very fond of it.*  
*But the bird died, leaving ALES really sad.*  
Ales hesitated for a second about what to do with the cat  
since he was late for work.  
(IGNORE PLAY FEED)  
**> play**  
Ales did not want to PLAY because he did not want  
CAT to be DEAD or ALES to be SAD  
(FEED IGNORE)  
**> ignore**  
Ales just kept walking.  
The cat walked away.  
...

Fig. 2. An interaction with Riu. Italics represents Ales’s past memories, and bold text represents user input.

(MRM) [14]. While MRM was developed on the GRIOT system’s conceptual blending framework [15], Riu focuses on computational analogy with a force-dynamics-based story representation.

Informed by stream of consciousness literature such as *Mrs. Dalloway*, our expressive goal is to depict characters’ inner memory world through its correlation with the real, physical world. In addition, we explicitly try to develop alternatives to the common goal-driven, problem-solving story aesthetics, which has been relatively unexplored in the computational narrative field as of today.

Figure 2 shows a sample interaction with Riu. The story starts with Ales’ encounter of a cat in the street while going to work, which triggers his memory of a previous pet bird in a “flashback”. There are three possible actions Ales can take at this point — “ignore,” “play” with, or “feed” the cat. In this example, the user first chooses “play.” However, the strong similarity between “playing with the cat” and “playing with his pet bird” leads to an analogical mapping and the subsequent (naive) inference that “if Ales plays with the cat, the cat will die and he will be very sad.” Hence Ales refuses to play with the cat and the system removes this action. The story continues after the user selects “ignore.”

Figure 3 shows Riu’s system architecture. Riu is composed of four main modules: a story engine, a memory retrieval component, the Ales module, and a computational analogy algorithm, SME. In addition to the previous four module, Riu contains two repositories of authored data: a *main story* (consisting of a graph where each node is a scene, and each link is a possible action that Ales may take) and a collection of *past memories*. This paper focuses on the memory retrieval and analogy components.

- The story engine is in charge of reading user input, executing the actions that Ales executes in the world, and presenting the story to the user in the form of English sentences, as it unfolds.
- The memory retrieval component is in charge of retrieving similar memories (from the past memories

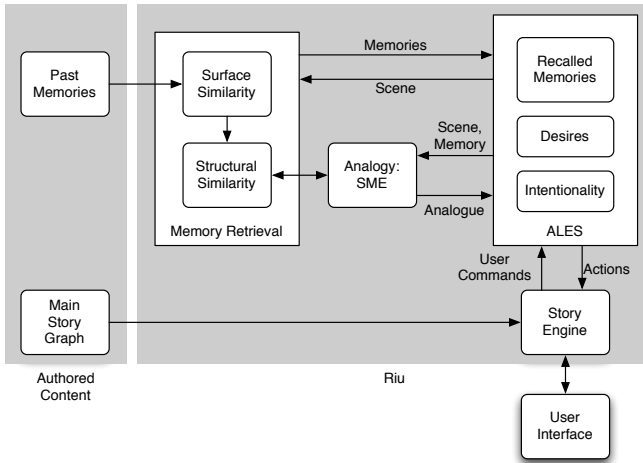


Fig. 3. Riu's system architecture.

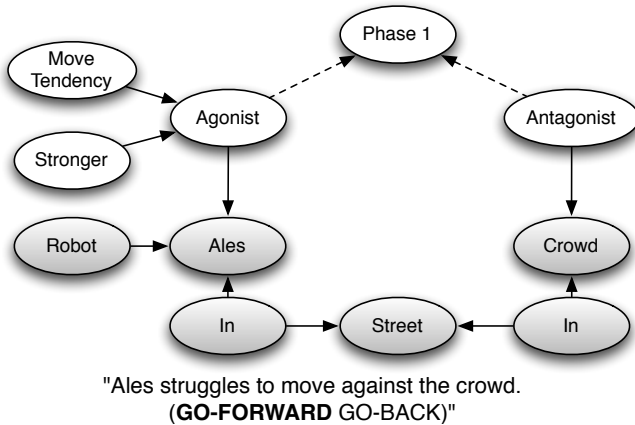


Fig. 4. A force-dynamics-based representation of a scene in the Riu system.

repository), which are similar to the current state of the story.

- The Ales module controls the main character, Ales. To do so, it has a list of desires of Ales (which in all our runs, just contains a single item: “being happy”), an intentionality degree, which is a number between 0 and 1, and the set of memories Ales has been able to recall (initially empty). When the intentionality is low (close to 0), Ales behaves like a mere avatar, just following the user commands; when the intentionality grows (approaches 1), Ales starts executing more and more actions on its own without waiting for user input, or even executing actions against user input. This aspect of Riu explores the *agency play*, presented elsewhere [16].
- Finally, the computational analogy module, SME, is used both by the memory retrieval and by the Ales module to find mappings between scenes.

The main story in Riu is represented as a graph, where each node corresponds to a scene, and each link is a possible action that Ales may take. Inside each node, there is a frame-based representation, as shown in Figure 4. They contain the

story elements in the scene (the grey nodes in Figure 4) as well as the force dynamics elements (white nodes). Figure 4 shows a scene where Ales is trying to make his way through a crowd of people in Phase 1. The user can choose whether Ales will keep moving forward or go back. Past memories are represented in the same ways as scenes. For example, the left hand side of Figure 5 shows a memory with two phases. In Phase 1 Ales wants to become a painter, and in Phase 2 he has to give up because of work. Natural language generation is not the focus of Riu. Each element in the frame-based representation is annotated with pre-authored texts, so that Riu can generate English output.

Computational analogy is used in two major aspects of Riu: *story matching* for memory retrieval, and *story generation* to establish the connection and influence between reality and the memory world. In both aspects, we use *Structure Mapping Engine* (SME) [17] as the core computational analogy component (although any other computational analogy system could have been used). SME is a symbolic analogy system that computes the similarity level of two domains (both surface and structural) based on Gentner's structure mapping theory [18].

Figure 2 shows two examples of how does Ales use analogy to influence its behavior (recalling memories and refusing to execute commands of the user). However, there is a number of other ways: Ales could select an action by himself without waiting for user command (if its intentionality level is very high); Or, he can express his opinions on the different actions before the player is to choose one (by narrating the small story snippets generated by analogy, and which correspond to what Ales imagines will happen if the different actions are executed).

The rest of this section describes Riu's analogy-based story matching and generation mechanisms.

#### A. Story Matching

Each time Ales faces a new scene, a certain piece of memory may be recalled if it shares enough similarity with the current scene. Each memory is defined as a sequence of phases, but for memory retrieval, Riu currently only considers the first phase of each memory. Similarity between a memory and the current scene is evaluated using a two step process:

*Surface Similarity:* Riu first extracts a series of keywords from the current scene and every unrecalled memory, and selects the 3 memories with the most overlapping keywords with the current scene. For example, the keywords of the memory in Figure 5 (left) are: robot, ales, works, job, learning, gives-up, and painter (i.e. the grey story-element-nodes in the graph).

*Structural Similarity:* Then, SME is used to compute analogical mappings and their strength, between each of 3 selected memories and the current scene. As indicated by the structural mapping theory, SME favors deeper (i.e., structural) similarity over surface (i.e., isolated nodes) similarity. Therefore, the memory that shares the largest structures with the current scene will receive the highest score from SME.

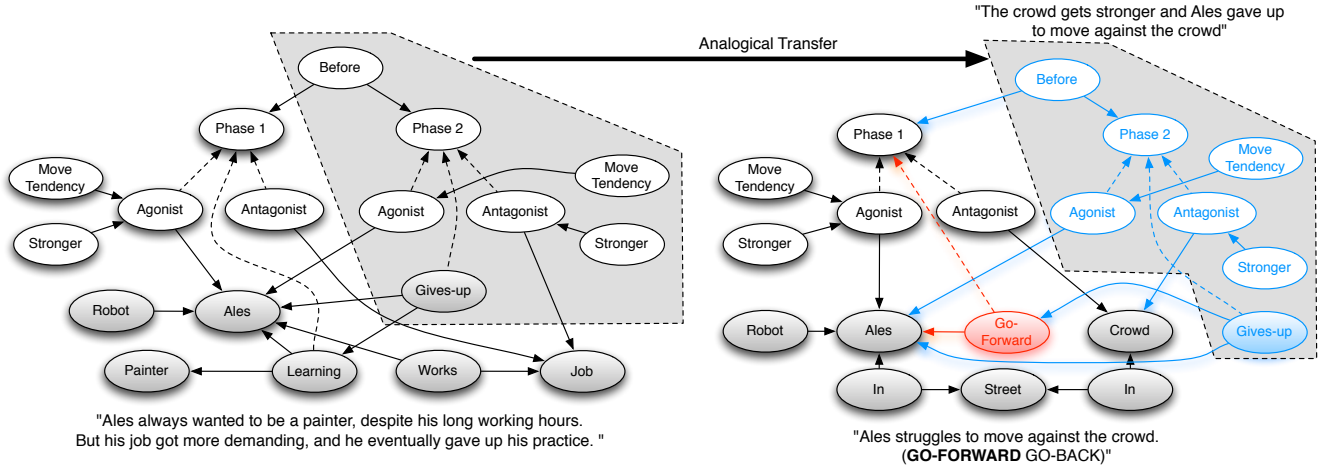


Fig. 5. Analogy-based story mapping and generation between a piece of memory (left) and a story scene (right).

If the score is above a certain threshold, the memory will be narrated in the “flash-back” and stored in the recalled memories repository.

The rationale behind this two step process is that structural similarity is a computationally expensive process (specially with large structures such as some of the scenes); thus, surface similarity is used to trim down the candidate memories to a small number, and only those selected ones will go through the structural similarity process. This is a well established procedure for analogy based memory retrieval systems such as MAC/FAC [19].

The utility of force dynamics here is that it provides a means to formalize the structure of the scene (with regard to forces and pressures) and not just the specific elements. Compared to the other frame-based story representation, force dynamics-based representation allows us to identify deeper analogies that are not apparent at the surface level. The memory’s Phase 1 in Figure 5, for instance, is represented as the Agonist (Ales) overcoming the Antagonist (job) and displaying its tendency of motion (becoming a painter). Thanks to this representation, any scene with similar force dynamic relations can be regarded as a match, such as the story scene of the crowd. Without a representation at this level of abstraction, SME can only rely on the similarity between specific story elements (e.g. Ales, job, crowd, etc) and may miss many higher-level similarities apparent to a human reader. The mapping between “crowd” and “job”, for instance, will be difficult to find because of their lack of surface similarities. In other words, the force-dynamics-based representation exploits SME’s original concept of “structural” similarity to find similarities at a higher level than the details of the story.

### B. Analogy-Based Story Generation

Every time the user faces a decision point, Ales “imagines” the consequences of the possible actions using analogical inference in the following way. Given a current scene  $S$  and an action  $A$ :

- 1) Riu first adds  $A$  to the current scene  $S$ , forming a *hypothetical scene*  $S'$ .
- 2) Then Riu uses the process specified in the previous section to choose, among all recalled memories, the most similar one,  $M$  to the hypothetical scene  $S'$ .
- 3) Then it establishes mappings of characters, settings, events and/or force dynamics structures between the first phase of the memory  $M$  and the current phase of the hypothetical scene  $S'$  using SME.
- 4) After that, it infers the consequences by transferring the subsequent phases of the memory  $M$  to the hypothetical scene  $S'$ .
- 5) Finally, Ales analyzes the resulting consequences seeing if they align with its desires (in our current implementation Ales has a unique desire to be happy).

Figure 5 illustrates the process of story generation by analogical inference from memory (on the left hand side) to hypothetical scene (right, and is same as Figure 4). Let us explain the process in detail with Figure 5, assuming the user chooses “go-forward”.

First, the action “go-forward” is added to the representation of the current scene as a node (in red color), which forms the hypothetical scene. Then the most similar memory among all the recalled memories is selected based on their Phase 1, using the same two-step story matching process above. SME then returns an analogical mapping between Phase 1 of the selected memory and of the current hypothetical scene.

Next, using analogical transfer, the hypothetical scene evolves into a sequence of phases *analogue* to the way the memory unfolded from Phase 1 to its respective subsequent phases. In Figure 5, the hypothetical scene (right) is extended with a second phase (in blue color), transferred by analogy from the selected phase (left). Specifically, based on the already established mapping, Riu transfers the unmapped nodes and links (i.e. those that do not have a match in the hypothetical scene) from Phase 2 of the memory to that of the target hypothetical scene. As a result, Riu infers that the Antagonist “crowd” in the current scene will also become

stronger and block Ales’s tendency to move.

Natural language generation is not the focus of Riu, and it only does very simple manipulations of the English text associated with each element of the scenes, resulting in the output text “the crowd gets stronger and Ales gave up to move against the crowd.”

Finally, Ales evaluates the consequence of the action based on the above inference. Currently, Ales has the desire to be happy. If the imagined consequence involves sad (non happy) elements, Ales may refuse to take the action. Conversely, he may take the action autonomously without the user’s consent. This process is currently implemented simplistically, just by checking if any “sad” or “happy” keywords appear in the resulting story. In the example of Figure 2, Ales refused to play with the cat because he infers that it may lead to its death (which is a sad keyword).

Force dynamics also plays a key role in Riu’s story snippet generation. Clearly, the memory of painting and the scene with the crowd in Figure 5 have very little in common (other than Ales) at the surface level. However, as said before, force dynamics allows SME to discover the analogy between “moving forward” and “wanting to be a painter”. Thanks to this analogy, Riu can generate a second phase for the scene at hand by analogical transfer. In the example in Figure 5, the whole Phase 2 force dynamics structure plus the *Gives-up* node are transferred from the memory to the hypothetical scene. In this particular case, the memory contained only two phases, and thus only Phase 2 was transferred. If the memory contains a larger number of phases, then the story generated by analogy will also increase its number of phases.

Finally, notice that in Riu, analogical mapping is only done among the first phase of the memory, and the current scene. Matching structures composed of multiple scenes is part of or future work. Also, the story generation in Riu works always by mapping the first phase, and transferring the subsequent phases. However, there is nothing preventing from mapping later phases, and transfer earlier phases.

## V. EMPIRICAL EVALUATION

Although a thorough evaluation of the contribution of force dynamics to the story generation components of Riu is still part of our future work, this section presents some evidence of how using force dynamics improves the quality of the mappings found by SME.

Figure 6 shows three pairs of scenes, and the mappings between the entities in the scenes that SME found. We show both the mappings found using force dynamics, and the mappings found if we remove all of the force dynamics elements in the scene definition. We only show the mappings between entities of the scenes (and not relations) for simplicity:

- The first pair scenes in the figure correspond to a memory of a pet bird and to a scene when Ales finds a cat in the street. In this case, both using and not using force dynamics achieves the same mapping. The bird is mapped to the cat, and Ales is mapped to Ales. This is so, since both the bird and the cat are animals, which

lets SME find the mapping. Ales is declared as a robot in both scenes, and thus SME can also find this mapping easily.

- The second pair of scenes correspond to a memory of when Ales had his first oil change, and the same scene of the cat. In this case, without using force dynamics, SME can find the mapping between the garage and the street (since both are locations), and between Ales and Ales. Using force dynamics, allows SME to find an additional mapping: owners to work. This relationship is adequate, since both the owners and the work are the antagonists in the scenes: they are what is causing trouble to Ales.
- The third pair of scenes correspond to even more different stories. One is a memory of a willow tree, and the other is the scene of the cat. In this case, there are no entities in the scenes with any apparent similarity nor shared predicates. So, without using force dynamics, SME cannot find any mapping. However, using force dynamics SME is able to see that the tree maps to Ales, and the work maps to the drought. This is so, since in both scenes there is an entity (Ales or the tree) fighting against another (work or the drought).

## VI. COMPARISON WITH RELATED WORK

Although planning is one of the most common techniques for story generation, there have been a number of systems that use computational analogy and related techniques (such as conceptual blending and case-based reasoning) to generate stories. A thorough overview can be found in [2]. Here we present a brief account of some of these systems and compare them to Riu.

Among the systems that adopt classic computational analogy, Riedl and León’s system [20] combines analogy and planning. It uses analogy as the main generative technique and uses planning to fill in the gaps in the analogy-generated content. The system uses the CAB computational analogy algorithm [21] for story generation and uses a representation consisting of planning operators. Moreover, the system applies analogy at the individual action level, not at the story structure level. Compared to Riu, its particular plan-based representation entails that analogy needs to operate at the individual action (i.e. plan operator) level. In contrast, Riu does focus not focus on actions; It applies analogy at a higher level, mapping a complete scene with another one, which may or may not contain actions. Moreover, since Riedl and León’s system uses a set of pre-authored stories, force dynamics annotations could be added to these stories to enhance the depth of potential analogies.

Several systems use techniques similar to analogy. For instance, GRIOT [15] and Memory, Reverie Machine (MRM) [14] use Harrell’s ALLOY conceptual blending algorithm to produce affective blends in the generated poetry (GRIOT) and narrative text (MRM). ALLOY creates mappings between the input spaces with operations similar to analogy and then generate generate blends. Other systems use case-based reasoning (CBR), such as Minstrel [8], ProtoPropp

<i>Scenes</i>	<i>Ales used to play with a bird when he was young. Ales was very fond of it.</i>	One day, Ales was walking in an alley. when he saw a cat in front of him. Ales hesitated for a second about what to do with the cat since he was late for work.
<i>Mapping without Force Dynamics</i>	Ales-----▶Ales Bird-----▶Cat	
<i>Mapping with Force Dynamics</i>	Ales-----▶Ales Bird-----▶Cat	
<i>Scenes</i>	Ales remembered the garage in which he had his first oil change, it was all red. His owners said he was rusty, and forced him to change his oil, he was a fool to accept.	One day, Ales was walking in an alley. when he saw a cat in front of him. Ales hesitated for a second about what to do with the cat since he was late for work.
<i>Mapping without Force Dynamics</i>	Garage-----▶Street Ales-----▶Ales	
<i>Mapping with Force Dynamics</i>	Garage-----▶Street Ales-----▶Ales Owners-----▶Work	
<i>Scenes</i>	The little willow tree behind the playground. grew into a big, silver tree despite the recent droughts.	One day, Ales was walking in an alley. when he saw a cat in front of him. Ales hesitated for a second about what to do with the cat since he was late for work.
<i>Mapping without Force Dynamics</i>	No mapping found	
<i>Mapping with Force Dynamics</i>	Tree-----▶Ales Drought-----▶Work	

Fig. 6. A comparison between the mappings found with and without force dynamics for some sample pairs of scenes in Riu.

[22] and the Virtual Storyteller [23]. These CBR systems also use analogy-like operations to establish mappings between the source cases and the target problem. Reminiscent of CBR, MEXICA [9] generates stories using an engagement/reflection cycle (also used in the Visual Daydreamer [24]). It maps the current state to the states in the pre-authored memories and retrieves the most similar one for the next action. Similar to these systems, the library of past memories in Riu can be seen as the case base of authored stories in the CBR-based systems. Furthermore, all these systems represent stories using frame-based representations, which could be easily strengthened by adding force dynamics structures.

Force dynamics by itself is not sufficient to represent the entire story, but it can effectively compliment other representations, which can capture additional story elements such as temporal relationship (e.g. plans) and descriptive features of story elements (frames).

## VII. CONCLUSIONS AND FUTURE WORK

This paper explores analogy-based story generation in order to broaden the aesthetic spectrum of computer-generated stories. Due to the strong impact of knowledge representation on analogy, we pay close attention to the story representation we use. After analyzing past uses of common story representations, we argue that most of them either emphasize too much the causal/temporal relationships, or operate on a level that is too detailed to capture deeper analogies among stories. We have shown that force dynamics extends the traditional notion of causality, as used in planning, by adding other crucial causal concepts such as “letting,” “hindering,” and “helping.” More importantly, it provides a means to represent story at a level of abstraction of force and pressure and beyond the details of particular story elements. As a result, we have shown in our system Riu that force-dynamics-based representations allow computational analogy techniques to find deep analogies between stories as well as to generate new stories using analogical inferences. These analogies may be hard to find if we only stay at the surface level.

Currently, Riu explores analogy-based story generation with small story snippets. As part of our future work, we plan to extend our system with larger stories, and also study the benefits of force dynamics at a higher plot level. We also want to perform user studies to compare the perceived quality of the generated stories. Finally, we plan to explore the usefulness of more generalized computational linguistics frameworks, such as conceptual semantics [25], and study their potential to enhance story generation.

## REFERENCES

- [1] J. H. Murray, *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. Cambridge: The MIT Press, 1998.
- [2] J. Zhu and S. Ontañón, "Towards analogy-based story generation," in *First International Conference on Computational Creativity (ICCC-X)*, Lisbon, Portugal, 2010.
- [3] L. Talmy, "Force dynamics in language and cognition," *Cognitive Science*, vol. 12, no. 1, pp. 49–100, 1988.
- [4] D. Hofstadter and M. Mitchell, "The copycat project: a model of mental fluidity and analogy-making," in *Fluid concepts and creative analogies: computer models of the fundamental mechanisms of thought*. New York, NY, USA: Basic Books, Inc., 1995, pp. 205–267.
- [5] J. Meehan, "The metanovel: Writing stories by computer," Ph.D., Yale University, 1976.
- [6] M. Lebowitz, "Creating characters in a story-telling universe," *Poetics*, vol. 13, pp. 171–194, 1984.
- [7] M. Riedl, "Narrative generation: Balancing plot and character," Ph.D. dissertation, North Carolina State University, 2004.
- [8] S. R. Turner, "Minstrel: a computer model of creativity and story-telling," Ph.D. dissertation, University of California at Los Angeles, Los Angeles, CA, USA, 1993.
- [9] R. P. y Pérez and M. Sharples, "Mexica: A computer model of a cognitive account of creative writing," *J. Exp. Theor. Artif. Intell.*, vol. 13, no. 2, pp. 119–139, 2001.
- [10] P. W. Weyhrauch, "Guiding interactive drama." Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, PA, USA, 1997.
- [11] R. McKee, *Story: Substance, Structure, Style and The Principles of Screenwriting*. It Books, 1997.
- [12] M. Mateas and A. Stern, "Structuring content in the façade interactive drama architecture," in *AIIDE*, 2005, pp. 93–98.
- [13] R. C. Schank and R. P. Abelson, *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*. Hillsdale, NJ: L. Erlbaum, 1977.
- [14] XXX, "Yyy," Ph.D. dissertation, ZZZ, 2009.
- [15] D. F. Harrell, "Theory and technology for computational narrative: An approach to generative and interactive narrative with bases in algebraic semiotics and cognitive linguistics," Dissertation, University of California, San Diego, 2007.
- [16] F. Harrel and J. Zhu, "Agency play: Dimensions of agency for interactive narrative design," in *AAAI Spring Symposium on Intelligent Narrative Technologies II*. Stanford CA: AAAI Press, 2009, pp. 44–52.
- [17] B. Falkenhainer, K. D. Forbus, and D. Gentner, "The structure-mapping engine: Algorithm and examples," *Artificial Intelligence*, vol. 41, pp. 1–63, 1989.
- [18] D. Gentner, "Structure-mapping: A theoretical framework for analogy," in *Readings in Cognitive Science: A Perspective from Psychology and Artificial Intelligence*, A. Collins and E. E. Smith, Eds. San Mateo, CA: Kaufmann, 1988, pp. 303–310.
- [19] D. Gentner and K. D. Forbus, "MAC/FAC: A model of similarity-based retrieval," *Cognitive Science*, vol. 19, pp. 141–205, 1991.
- [20] M. Riedl and C. León, "Generating story analogues," in *AIIDE 2009*. The AAAI Press, 2009.
- [21] L. B. Larkey and B. C. Love, "Cab: Connectionist analogy builder," *Cognitive Science*, vol. 27, no. 5, pp. 781–794, 2003.
- [22] P. Gervás, B. Díaz-Agudo, F. Peinado, and R. Hervás, "Story plot generation based on cbr," *Journal of Knowledge-Based Systems*, vol. 18, no. 4-5, pp. 235–242, 2005.
- [23] I. Swartjes, J. Vromen, and N. Bloom, "Narrative inspiration: Using case based problem solving for emergent story generation," in *Proceedings of the Fourth International Joint Workshop on Computational Creativity*, A. Cardoso and G. Wiggins, Eds. London: Goldsmiths, University of London, June 2007, pp. 21–28. [Online]. Available: <http://doc.utwente.nl/64164/>
- [24] R. P. y Pérez, R. Sosa, and C. Lemaitre, "A computer model for visual-daydreaming," in *AAAI Fall Symposium on Intelligent Narrative Technologies*. Washington D.C.: AAAI Press, 2007.
- [25] R. S. Jackendoff, *Semantic Structures*. MIT Press, 1992.