

On the Role of Domain Knowledge in Analogy-Based Story Generation

Santiago Ontañón

Artificial Intelligence Research Institute (IIIa - CSIC) School of Visual Arts and Design
Campus UAB, 08193 Bellaterra, Spain
santi@iii.csic.es

Jichen Zhu

University of Central Florida
Orlando, FL, USA 32826-3241
jzh@mail.ucf.edu

Abstract

Computational narrative is a complex and interesting domain for exploring AI techniques that algorithmically analyze, understand, and most importantly, generate stories. This paper studies the importance of domain knowledge in story generation, and particularly in *analogy-based story generation* (ASG). Based on the construct of knowledge container in case-based reasoning, we present a theoretical framework for incorporating domain knowledge in ASG. We complement the framework with empirical results in our existing system *Riu*.

1 Introduction

Computational narrative explores the age-old creative form of storytelling by algorithmically analyzing, understanding, and most importantly, generating stories. The various AI techniques developed in story generation can be extended to other forms of interactive entertainment and electronic literature, including computer games and interactive narrative. In this paper, we focus on *analogy-based story generation* (ASG), that is, story generation systems that use computational analogy or analogy-like algorithms (e.g. case-based reasoning) as their core generative mechanism. We present a theoretical framework of the role of domain knowledge in these systems and examine its effect through empirical evaluations.

Domain knowledge is key to both story generation and computational analogy. A good illustration is the first story generation system, *Tale-Spin* [Meehan, 1976], where the author reports having to add more and more domain knowledge to fix the stories his system created. In our previous work in ASG [Zhu and Ontañón, 2010a], we also observed semantic errors in the output of our system due to the lack of domain knowledge and semantic constraints. Domain knowledge is especially important to computational analogy because many analogy methods are very brittle and highly dependent on the way in which knowledge is represented.

Despite its crucial role, incorporating domain knowledge into story generation systems has mostly been dealt with in an ad-hoc way. In this paper we present an analysis of where and how domain knowledge can be used in ASG, and empirically analyze the benefits of each option in a specific ASG system.

The theoretical framework we present is based on Richter's work on *knowledge containers* [Richter, 1995], which analyzes the different roles of domain knowledge in CBR systems. We apply this construct to Hall's generic computational analogy framework [Hall, 1989]. We then report a set of experiments to evaluate the effect of domain knowledge in a particular ASG system, *Riu* [Zhu and Ontañón, 2010a]. Our conclusions can potentially be useful to other story generation and computational analogy systems.

The paper is organized as follows. The next section presents some background on story generation and ASG. Section 3 introduces our theoretical framework, introducing the idea of knowledge container and extending it to the ASG. Then, Section 4 briefly presents *Riu*, our ASG system, and an empirical evaluation of *Riu* is included in Section 5. Next, Section 6 represents related work. The paper ends with future work and concluding remarks.

2 Background

Story generation systems explore the ways in which stories (narratives) can be algorithmically generated. It is a highly complex domain as it involves problems such as natural language generation, character modeling, common sense reasoning, narrative aesthetics, etc. Different AI models have been proposed for story generation, from planning-based approaches [Meehan, 1976] to simulation and multiagent systems-based approaches [Theune *et al.*, 2003]. Among them, computational analogy is a promising approach. Although it has been used in several influential systems, ASG still remains relatively under-explored compared to the planning-based approach.

Computational models of analogy operate by identifying similarities and transferring knowledge between a source domain *S* and a target domain *T*. This process can be divided into four stages [Hall, 1989]: 1) *recognition* of a candidate analogous source, *S*, 2) *elaboration* of an analogical mapping between source domain *S* and target domain *T*, 3) *evaluation* of the mapping and inferences, and 4) *consolidation* of the outcome of the analogy for other contexts (i.e. learning). Existing analogy systems can be classified into three classes based on their underlying architecture [French, 2002]. First, *Symbolic* models, such as SME [Falkenhainer *et al.*, 1989]), heavily rely on the concepts of symbols, logics, search, etc. from the "symbolic AI paradigm." Second, *connectionist*

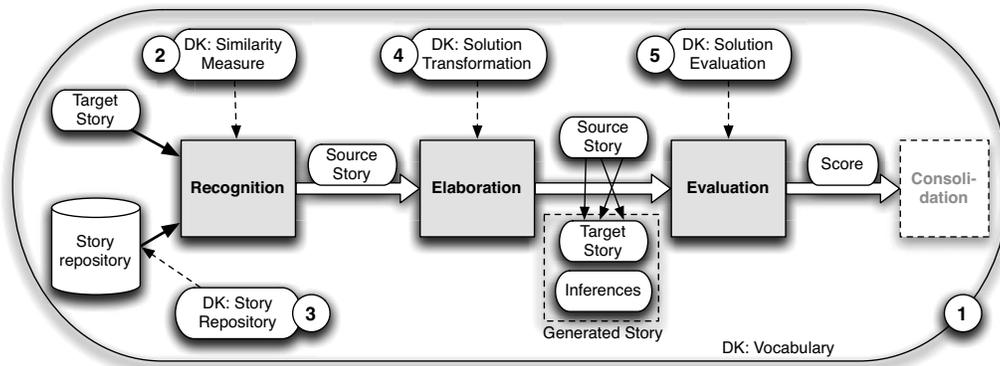


Figure 1: The five knowledge containers in computational analogy.

models, such as ACME [Holyoak and Thagard, 1989], adopt the connectionist framework of nodes, weights, etc. Finally, hybrid models, such as Copycat [Hofstadter and Mitchell, 1995], blend elements from the previous two classes.

A typical ASG system generates stories about a *target* domain T by identifying and making analogy with a *source* domain S . For instance, given an incomplete *target story* T (e.g. the initial state of the story) and a repository of complete *source stories* S , an ASG system completes the target story by analogy with one or more of the source stories.

Domain knowledge is essential to story generation and particularly ASG. However, it has not been sufficiently studied in the context of story generation or computational analogy. For instance, a prevailing understanding of domain knowledge in story generation typically refers to descriptions of the story domain (e.g., characters, locations, behaviors), exemplified by [Bringsjord and Ferrucci, 1999]. As we will show below, other (implicit) types of domain knowledge also exist in an ASG system and they play an equally important role in the final generated stories. The next section identifies different types of domain knowledge and where they can be incorporated in ASG systems, which is a necessary step to improve the performance of these systems.

3 Theoretical Framework

Our theoretical framework is based on Richter’s *knowledge containers* framework [Richter, 1995], which analyzes the knowledge contained in case-based reasoning (CBR) systems. Since CBR and computational analogy share similar underlying principles, this framework is useful for us to identify the locations where domain knowledge can be incorporated in ASG systems. The main idea behind Richter’s work is that, in addition to data repositories, other parts of a system, such as its procedures, also contain crucial (implicit) knowledge. Using that idea, Richter identified four containers of domain knowledge. Below, we will discuss them and extend this framework to the context of ASG:

1. *Vocabulary*: contains information about which data structures and which elements of the structures are used to represent primitive notions. In story generation systems, the vocabulary container captures knowledge such as “which are the important concepts in the stories”

and “how are they related.” For instance, the concepts of “character” and “action” and their relation are some common vocabulary for many story generation systems.

2. *Similarity measure*: assigns to each pair of problems (stories) a real number in the interval $[0, 1]$ corresponding to their similarity. In ASG, the knowledge in the similarity measure determines how to identify a source story for story generation in the Recognition stage of analogy.
3. *Case Base (Story Repository)*: contains the experiences, i.e. the cases, in CBR systems. In ASG, this refers to the story repository containing all available source stories. In addition to the domain knowledge regarding the story world, the story repository also captures concrete knowledge concerning the desired quality of outcome stories, since each stored story is assumed to be of high quality.
4. *Solution transformation*: takes care of fact that the solutions obtained from the case base using the nearest neighbor principle may need to be adapted. In ASG, this refers to the methods and knowledge used in the Elaboration process to map elements of the source story to the target story, and the way in which inferences are transferred to the target story.

Richter’s original formulation of knowledge containers was only concerned with the two first phases of the CBR process, *Retrieve* and *Reuse/Adaptation* (corresponding precisely to Recognition and Elaboration in analogy) and ignored the later two phases, *Revise* and *Retain* (Evaluation and Consolidation in analogy). As Evaluation plays a key role in the process of ASG, we propose a fifth knowledge container:

5. *Solution Evaluation*: contains knowledge about what constitutes a high quality solution (i.e. generated story). In both ASG and CBR, the quality of the solutions may be evaluated before being provided to the user.

Figure 1 illustrates the locations of the above five knowledge containers in Hall’s four processes of computational analogy. Notice that the *consolidation* (learning) phase is out of the scope of this paper and is hence not included here. The remainder of this paper analyzes the role of each of the 5 knowledge containers in ASG, using the example system Riu.

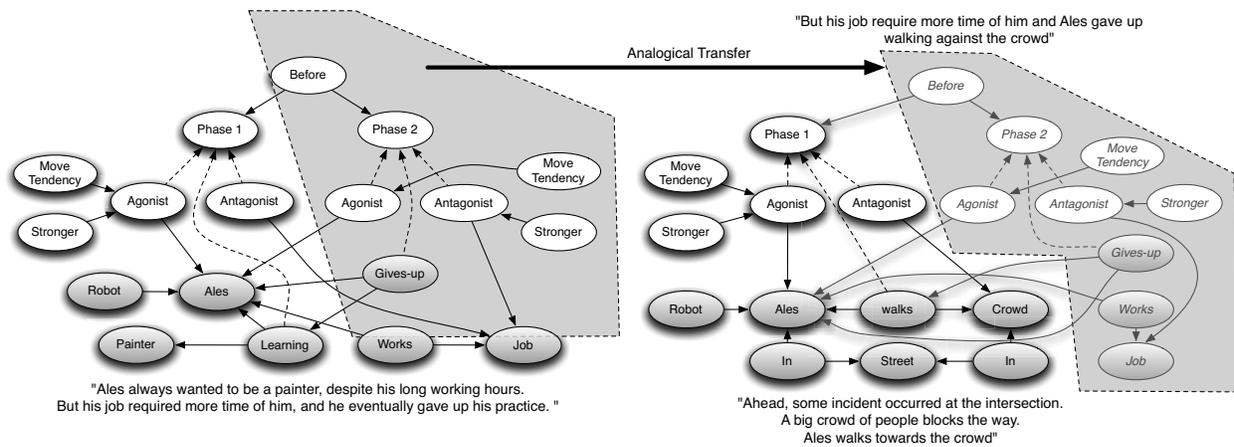


Figure 2: Analogy-based story generation from a source scene (left) and a target scene (right) in Riu.

4 The Riu System

Riu is a text-based interactive narrative system that uses analogy to generate stories about a robot character Ales, who has initially lost his memories. In the story, the user guides Ales through a series of events, some of which trigger Ales’s lost memories. When facing a decision point in the main story world, Ales “imagines” (through analogical inference) the consequences of each option using already recalled memories as the source domain. The result of his imagination influences Ales’s actions and the course of the story.

Riu uses the *Structure Mapping Engine* (SME) [Falkenhainer *et al.*, 1989] as the core analogy algorithm for story generation. A detailed description of *Riu* can be found in [Zhu and Ontañón, 2010a]. *Riu* uses a frame-based knowledge representation, enhanced by a *force dynamics*-based formalism to represent both a pre-authored graph of the main story and a collection of memory episodes. Force dynamics (FD) [Talmy, 1988] is a cognitive semantics framework based on the concept of *force* — physical, psychological or social. It captures the more abstract information about a scene, including its “agonist”, “antagonist” and their relative strength.

Figure 2 shows an example of the story generation process in *Riu* by analogy from a source scene (left) to a target scene (right). The source scene in this case is a scene where Ales had to give up his dream of becoming a painter because of his job, and the target scene represents a situation in which Ales walks against a big crowd. SME finds the following mappings: (ales ales), (painter crowd) (learning walks) (agonist agonist) (strong strong) (move-tendency move-tendency) and (phase1 phase1). The result is that *Riu* completes the target scene by analogically inferring that Ales’s job got also more demanding in time, and Ales had no time to walk against the crowd, eventually giving up.

In our previous work in *Riu*, we observed that *Riu* generates some extraneous scenes often due to the lack of enough domain knowledge. For instance, in the previous example, *Riu* has no knowledge of what is a “crowd” and may generate content inappropriate to these concepts. In order to reduce these mistakes, it is essential to incorporate more domain knowledge into the system.

5 Empirical Evaluation

This section discusses the effect of increasing domain knowledge in the 5 knowledge containers individually. We used *Riu* as our base-line system and WordNet as our source of additional generic domain knowledge.

5.1 Vocabulary

As shown earlier, the vocabulary of the story domain, including both ontology (e.g. “character”, “action”, “goal”) and representation formalism (e.g. frame-based) is a major container of domain knowledge for story generation systems. In our previous work [Zhu and Ontañón, 2010a], we compared *Riu*’s performance using two different story representation formalisms. First, we used a baseline representation consisting of standard narrative concepts such as entities and actions. In a second representation, we augmented the baseline representation with *force dynamics* annotations (hence more domain knowledge.)

The result of our experiment is that *Riu*’s SME component finds better mappings when using the FD-enhanced representation. In some situations, it found mappings where the baseline representation failed. In our latest experiments, *Riu* separately generated 55 stories with different combination of source and target scenes using both representations. In 72.7% of all the generated stories, FD-enhanced representation yields better mappings (i.e. more components are transferred from the source to target scene) than the baseline. In the rest of the situations, *Riu* found equivalent mappings. Under no circumstances, the FD-enhanced representation led to worst mappings.

Our conclusion from the above experiments is in accordance to previous evidences [Hofstadter and Mitchell, 1995] stating that computational analogy algorithms are very sensitive to the knowledge representation that they use. More domain knowledge at the vocabulary level substantially increases the quality of the analogy.

5.2 Similarity Measure

In the recognition phase, *Riu* identifies the source scene most relevant to the target scene using a similarity measure. It is

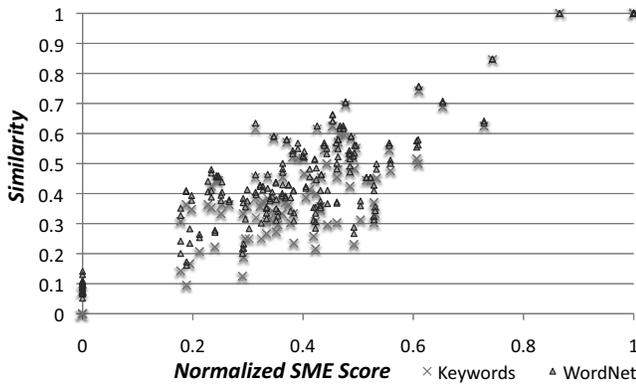


Figure 3: Correlation between two similarity measures and the SME score.

well known from the CBR literature that selecting the appropriate source is key to solving the problem. Below, we compare the results of two similarity measures used in Riu.

The base similarity measure is based on the MAC/FAC model [Forbus *et al.*, 1995]. Given two scenes, it computes a list of keywords appearing in both scenes and the percentage of shared keywords between them. For example, if a scene has the keywords (“animal,” “street,” “cute”) and another scene has the keywords (“young,” “animal,” “death”), they share 1 out of the 5 different keywords. The similarity here is $1/5 = 0.2$.

We extended this similarity measure with WordNet, using the latter’s hyponyms, hypernyms, synonyms, antonyms, meronym and holonym databases. Given two keywords, it measures how many related words do they share in WordNet’s databases. For instance, “bird” and “cat” are quite similar, since both of them are animals and vertebrates among other commonalities. Instead of only measuring the percentage of exact shared keywords, this new similarity measure also takes into account the semantic distance between keywords.

To evaluate these two similarity measures, we compared the quality of the respective source scenes they find using what we call the *SME score*. The score is assigned by SME internally to evaluate any given mapping between source and target based on structure-mapping theory. The larger the number of found mappings and the more related those mappings are to each other, the higher the SME score. We then evaluated Pearson’s correlation coefficient to see if the more informed WordNet similarity was better correlated with the SME score. We evaluated the similarity between the 289 possible pairs of stories in Riu’s repository. The keyword similarity measure has a correlation coefficient of 0.907, whereas WordNet-based similarity has a correlation coefficient of 0.913, only slightly higher. Figure 3 shows a scatter plot illustrating the correlation between the two similarity measures we considered and the SME score (which is normalized between 0 and 1).

This result shows that, despite common beliefs in CBR and computational analogy, a better similarity measure improves the resulting stories only marginally. We believe part of the

Sources	1	2	3	4	5	6	7
Score	5.71	7.36	7.00	7.53	7.94	8.47	8.50

Table 1: Average SME score of the stories generated using different story repository sizes.

reason is that the base keyword similarity, with a Pearson coefficient of 0.907, is already quite good. Another important factor is the difference between story generation and traditional problem solving, which most CBR and analogy systems are used for. As a *originality-driven task* [Díaz-Agudo *et al.*, 2008], story generation does not have clear-cut “right” or “wrong” solutions. Specifically to this experiment, while similar source story typically yields a better solution, a different source story scene may sometimes lead to a more surprising or interesting story.

5.3 Story Repository (Case Base)

Similar to CBR and other analogy systems, Riu requires a collection of *source* story scenes. Compared to the general domain knowledge stored in other knowledge containers such as vocabulary, the story repository contains a large amount of specific domain knowledge.

In principle, the more scenes in the story repository, the higher quality stories an ASG system can produce. However, as it is well known in the CBR community, a larger number of source problems decreases the system performance as they require more computational resources [Smyth and Cunningham, 1996]. Thus, there is a trade-of between quality and time. Moreover, in CBR systems, the quality of the the solution typically increases very slowly or even remains stalled after a certain threshold size of the case base [Smyth and Cunningham, 1996].

In Riu, we gradually changed the size of its story repository from 1 to 7 source stories and asked Riu to generate 11 different stories (with the 11 different target scenes available in the system). Like the previous experiment, we measured the quality of the generated stories by their SME score. The results are displayed in Table 1. As we can see, the quality of analogies clearly increases with the number of potential source story scenes.

It is worth noticing the high SME score when there is only one source scene (5.71), compared to that of the whole set of 7 scenes (8.50). A similar increase of cases in a typically CBR or machine learning system, by contrast, typically leads to a substantially larger improvement of performance. The reason is related to the argument in the previous subsection: the choice of source case has relatively little impact on the quality of the generated story. In the case of having only one source story in the repository, most of the generated scenes look alike, but they are nevertheless valid. Having more source stories increases the *variety* of the generated stories more than their *quality*. As part of our future work, we would like to further refine how we evaluate the generated stories in order to differentiate these two situations. For instance, we would like to take into account the quality of the source stories, in addition to their quantity.

5.4 Solution Transformation

Domain knowledge in the solution transformation container affects how mappings are established between the source and target scenes and how inferences are generated on the target scenes in different ways. In Riu, the SME algorithm implicitly contains knowledge based on the psychological theory of structure mapping. The only other knowledge that can be provided is a function which assigns a strength to each of the possible base mappings. The difficulty of adding more knowledge in solution transformation often makes SME subject to the criticism that it is too sensitive to the representation formalism (vocabulary) being used [French, 2002].

In comparison, other computational analogy algorithms accept additional domain knowledge in this container more easily. For instance, ACME takes into account (in addition to structure mapping constraints) semantic and pragmatic constraints and allows additional constraints (e.g. WordNet) to be added easily. In Copycat, additional domain knowledge is stored in its “SlipNet,” a network where concepts are related to each other based on their similarity.

In summary, different analogy methods impose different constraints on the amount and type of domain knowledge in the system’s solution transformation container. Choosing the appropriate method will have a big impact on the quality of generated stories.

5.5 Solution Evaluation

In the context of story generation, there has not been a lot of work on automatically evaluating story quality. An exception is Weyhrauch’s work [1997] on drama management, where he shows that with a carefully designed solution evaluation function (encoding the domain knowledge of narrative theory), the quality of generated stories can increase significantly.

In Riu, we examined the differences between the stories generated using two different solution evaluation routines. The baseline solution evaluation is the SME score. The second solution evaluation incorporates additional domain knowledge by using information from WordNet. The augmented solution evaluation judges the mapping generated by SME between the source and target story scenes. We used the same WordNet similarity measure described in Section 5.2 to assess the semantic similarity among the terms that get mapped together during story generation. The resulting evaluation function results from multiplying the SME score with the average WordNet similarity between the mapped terms.

Out of the 55 pairs of stories generated using the two evaluation routines respectively, only 5 are different from each other. Out of those 5, 3 stories generated by the WordNet-enhanced evaluation are better in the sense that they contain less statements that do not make sense. For instance, in one story generated using the baseline evaluation, Riu mapped “bird” to “food” and “play” to “buy”, and thus “play with the bird” to “buy food.” When later the “bird” “died”, Riu draws the inference that “food” also “died,” which is semantically incorrect. Using the WordNet-enhanced evaluation, Riu mapped “bird” to “cat” instead, since they share more semantical similarity in the WordNet database and eventually inferred that the “cat” “died.”

In summary, using a more informed solution evaluation criteria improved the quality of Riu’s generated stories in 3 situations out of 55 (i.e. in a 5.45%). Although not a very large improvement, solution evaluation is useful because knowledge in this container is rather easy to add.

6 Related Work

To the best of our knowledge, there have been no previous studies of the role of domain knowledge in ASG. However, the role of domain knowledge has been studied in some other areas of AI. For example, Anand et al. [1995] studied the role of domain knowledge in data mining. and Kautz and Selman [1998], did the same for automated planning.

Our work is relevant to analogy-based story generation systems in general. A more in-depth comparison between Riu and other ASG systems can be found at [Zhu and Ontañón, 2010b]. Here, we only include a brief overview. The PRINCE system [Hervás *et al.*, 2007] generates explanations of characters and objects in the target story by making the analogy of their equivalent in a different source story (e.g. “Excalibur was like the light saber of King Arthur”). Another example is Riedl and León’s system [2009], which combines analogy and planning. It uses analogy as the main generative technique and uses planning to fill in the gaps in the analogy-generated plot. Other systems use analogy-like algorithms for story generation. For instance, Minstrel [Turner, 1993] uses case-based reasoning, and Harrell’s GRIOT system [Harrell, 2006] implements the cognitive semantics theory of conceptual blending.

7 Conclusions

This paper has presented a study of the role of domain knowledge in analogy-based story generation (ASG) systems. Combining the Richter’s knowledge containers theoretical framework and Hall’s general computational analogy model, we have identified 5 independent roles that domain knowledge can play in ASG and reached the following conclusions based on our experiments in Riu:

- Vocabulary: high gains can be achieved with a proper vocabulary to represent stories, as demonstrated in our previous work [Zhu and Ontañón, 2010a].
- Similarity measure: a better similarity measure can improve quality, but very marginally. The selection of a source case has more impact on solution variety than in solution quality.
- Story repository: for the same reason, a large variety of source stories is useful to improve the variety, rather than the quality, of the stories the system can generate.
- Solution transformation: incorporating domain knowledge into the mapping algorithm is key for high quality story generation. However, different analogy algorithms have a different set of constraints on the type of domain knowledge that can be incorporated. Therefore, authors of ASG systems need to choose their analogy technique carefully.

- Solution evaluation: can improve solution quality by filtering out low-quality solutions. Although no large gains were observed in our experiments, it is a cost-efficient way to quickly increase the system's performance.

Engineering domain knowledge into ASG systems is a complex task. The contribution of our study is that it identifies the places where increasing domain knowledge can lead to substantial improvements. For example, the two areas that need to be specifically addressed for increasing the quality of ASG systems are the vocabulary and the solution transformation routine. Additionally, A better solution evaluation method is a cost-effective way to increase quality. Finally, the notoriously time-consuming tasks of authoring more story content (i.e., a larger story repository) can increase the variety of the stories, but does relatively little to increase the quality of the generated stories.

As part of our future work we would like to enhance our Riu system based on the results presented in this paper, concentrating on the knowledge containers that have a strong impact on story quality. Additionally, we plan to perform experiments in other ASG systems, in order to verify and expand the results reported in this paper. Finally, we would like to analyze other story generation paradigms (e.g. planning) to identify knowledge containers common to story generation in general.

References

- [Anand *et al.*, 1995] Sarabjot S. Anand, David A. Bell, and John G. Hughes. The role of domain knowledge in data mining. In *Proceedings of the fourth international conference on Information and knowledge management*, CIKM '95, pages 37–43, 1995.
- [Bringsjord and Ferrucci, 1999] Selmer Bringsjord and David Ferrucci. *Artificial Intelligence and Literary Creativity: Inside the Mind of BRUTUS, a Storytelling Machine*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1999.
- [Díaz-Agudo *et al.*, 2008] Belén Díaz-Agudo, Enric Plaza, Juan A. Recio-García, and Josep Lluís Arcos. Noticeably new: Case reuse in originality-driven tasks. In *ECCBR*, pages 165–179, 2008.
- [Falkenhainer *et al.*, 1989] Brian Falkenhainer, Kenneth D. Forbus, and Dedre Gentner. The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41:1–63, 1989.
- [Forbus *et al.*, 1995] Kenneth D. Forbus, Dedre Gentner, and Keith Law. Mac/fac: A model of similarity-based retrieval. *Cognitive Science*, 19(2):141–205, 1995.
- [French, 2002] Robert M. French. The computational modeling of analogy-making. *Trends in Cognitive Sciences*, 6(5):200–205, 2002.
- [Hall, 1989] Rogers P. Hall. Computational approaches to analogical reasoning: a comparative analysis. *Artificial Intelligence*, 39(1):39–120, 1989.
- [Harrell, 2006] D. Fox Harrell. Walking blues changes undersea: Imaginative narrative in interactive poetry generation with the griot system. In *AAAI 2006 Workshop in Computational Aesthetics*, pages 61–69, 2006.
- [Hervás *et al.*, 2007] Raquel Hervás, Rui P. Costa, Hugo Costa, Pablo Gervás, and Francisco C. Pereira. Enrichment of automatically generated texts using metaphor. In *MICAI*, pages 944–954, 2007.
- [Hofstadter and Mitchell, 1995] Douglas Hofstadter and Melanie 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*, pages 205–267. Basic Books, New York, 1995.
- [Holyoak and Thagard, 1989] Keith J. Holyoak and Paul Thagard. Analogical mapping by constraint satisfaction. *Cognitive Science*, 13:295–355, 1989.
- [Kautz and Selman, 1998] Henry A. Kautz and Bart Selman. The role of domain-specific knowledge in the planning as satisfiability framework. In *AIPS*, pages 181–189, 1998.
- [Meehan, 1976] James Meehan. *The Metanovel: Writing Stories by Computer*. Phd, Yale University, 1976.
- [Richter, 1995] Michael Richter. The knowledge contained in similarity measures. Invited talk at the International Conference on CaseBased Reasoning, Sesimbra, Portugal, 1995.
- [Riedl and León, 2009] Mark Riedl and Carlos León. Generating story analogues. In *Proceedings of AIIDE 2009*, 2009.
- [Smyth and Cunningham, 1996] B. Smyth and Páraig Cunningham. The utility problem analysed: A case-based reasoning perspective. In *Third European Workshop on Case-Based Reasoning EWCBR-96*, pages 234–248. Springer Verlag, 1996.
- [Talmy, 1988] Leonard Talmy. Force dynamics in language and cognition. *Cognitive Science*, 12(1):49–100, 1988.
- [Theune *et al.*, 2003] Mariët Theune, Er Faas, Anton Nijholt, and Dirk Heylen. The virtual storyteller: Story creation by intelligent agents. In *Proceedings of the TIDSE Conference*, pages 204–215, 2003.
- [Turner, 1993] Scott R. Turner. *Minstrel: a computer model of creativity and storytelling*. PhD thesis, University of California at Los Angeles, Los Angeles, CA, USA, 1993.
- [Weyhrauch, 1997] Peter W. Weyhrauch. *Guiding interactive drama*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 1997.
- [Zhu and Ontañón, 2010a] Jichen Zhu and Santiago Ontañón. Story Representation in Analogy-Based Story Generation in Riu. In *proceedings of IEEE-CIG*, pages 435–442, 2010.
- [Zhu and Ontañón, 2010b] Jichen Zhu and Santiago Ontañón. Towards Analogy-Based Story Generation. In *Proceedings of the First International Conference on Computational Creativity (ICCC-X)*, pages 75 – 84, Lisbon, Portugal, 2010.