

Story Generator Algorithms

Pablo Gervás

Created: 24. September 2012 Revised: 12. September 2013

1 Definition

The term story generator algorithms (SGAs) refers to computational procedures resulting in an artifact that can be considered a story. In the field of Artificial Intelligence (AI), the automated generation of stories has been a subject of research for over fifty years. An algorithm is understood as a set of instructions that, when applied to a given input, produces an output. In the present context, the desired output is a story. The underlying concept of “story” in SGAs is functional and does not imply any aesthetic notion. This is important because it sets the context for evaluation of generated stories, for which having a surface realization as a readable and appealing text is not necessarily a core issue.

2 Explication

Research on storytelling systems (computational systems capable of telling a story) has experienced considerable growth over the years. More recently, the number of systems developed has increased significantly. These systems initially arose as part of the general trend in AI to build computational solutions that could achieve the kind of tasks that are easy for humans and difficult for machines. Other examples of this trend include computer vision, speech processing and natural language understanding. Of all these examples, the first two have achieved success and given rise to commercial applications. Natural language understanding and story generation still remain at the exploratory research stage.

For story generation in particular, a large part of the problem is the fact that the task is not well defined in an AI / computational perspective. If an algorithm is to be devised for a given task, it should be very clear what the inputs must be and what characteristics are expected of the output. In the generation of stories, none of these are clearly defined. When humans produce stories, it is often not transparent what inputs they are bringing to bear on the process. Moreover, saying what makes a good story remains a question open for debate. As a consequence, existing story generation systems tend to be exploratory with regard not only to the algorithms they employ but also to the set of inputs they start from as well as the

characteristics their output stories are expected to produce. From the point of view of narratology, this is important, since different views on these fundamental decisions give rise to altogether different concepts of story generation. One of the main benefits of SGA-research is to under-defined narratological concepts.

3 History of the Term / the Concept

The term Story Generator Algorithms is of relatively recent coinage, first appearing in 2004 as the name of a project of the Hamburg Narratology Research Group ([<http://www1.uni-hamburg.de/story-generators//index_en.html [1]>]). The more abstract concept can be identified as an implicit assumption of the many storytelling systems developed since the 1950s.

3.1 Generating Systems

There is currently a large number of storytelling systems in existence. For this review, systems that generate classic sequential stories have been selected. Examples of story output are given for some systems where small enough significant fragments were available (for further detail, see Gervás 2009).

The first storytelling system on record is the Novel Writer system, developed by Sheldon Klein (Klein et al. 1973). Novel Writer created murder stories in a weekend party setting. This system is reputed to have generated “2100-word murder mystery stories, complete with deep structure, in less than 19 seconds.” A description of the world in which the story was to take place was provided as input, together with the characteristics of the participating characters (which included emotional links between them and their predisposition towards violence or sex). The particular murderer and victim depended on character traits specified as input (with an additional random ingredient). The motives arose as a function of the events in the course of the story. Possible motives for murder were restricted to greed, anger, jealousy or fear. The story was generated based on two different algorithms: 1) a set of rules that encodes possible changes from the current state of the world to the next; and 2) a sequence of scenes corresponding to the type of story to be told. The set of rules is highly constraining and allows for the construction of only one very specific type of story. Though more than one story could be built by the program, differences between them were restricted to who murders whom with what and why and who discovers them.

TALESPIN (Meehan 1977) was a system which generated stories about the lives of simple woodland creatures. To create a story, a character was given a goal, and then the plan was developed to solve the goal. TALESPIN introduced character goals as triggers for action. It also introduced the possibility of having more than one

problem-solving character in the story, introducing separate goal lists for each of them. Complex relations between characters were modeled (competition, dominance, familiarity, affection, trust, deceit and indebtedness). These relations acted as preconditions to some actions and as consequences of others, thus constituting a simple model of character motivation. The characters' personalities were modeled according to degrees of kindness, vanity, honesty and intelligence.

A sample TALESPIN story is given below: John Bear is given knowledge about the world and a goal to satisfy his hunger:

John Bear is somewhat hungry. John Bear wants to get some berries. John Bear wants to get near the blueberries. John Bear walks from a cave entrance to the bush by going through a pass through a valley through a meadow. John Bear takes the blueberries. John Bear eats the blueberries. The blueberries are gone. John Bear is not very hungry.

Dehn's AUTHOR (1981) was a program intended to simulate the author's mind as she makes up a story. Dehn's assumption is that story worlds are developed by authors as a post hoc justification for events that the author has already decided will be part of the story. An author may have particular goals in mind when he sets out to write a story, but even if he does not, it is accepted that a number of metalevel goals drive or constrain the storytelling process. These are issues such as ensuring that the story is consistent, that it is plausible, that the characters are believable, that the reader's attention is retained throughout the story, etc. These may translate at a lower level into subgoals concerning situations into which the author wants to lead particular characters, or the role that particular characters should play in the story. A story is understood as "the achievement of a complex web of author goals." These goals contribute to structuring the story, and to guiding the construction process. In the final story, however, these goals are no longer visible.

Lebowitz's UNIVERSE (1983) modeled the generation of scripts for a succession of TV soap opera episodes in which a large cast of characters plays out multiple, simultaneous, overlapping stories that never end. UNIVERSE was the first storytelling system to devote special attention to the creation of characters. Complex data structures were used to represent characters, and a simple algorithm was proposed to fill these in, partly in an automatic way. But the bulk of characterization was left for the user to provide.

UNIVERSE was aimed at exploring extended story generation, a continuing serial rather than a story with a beginning and an end. It was initially intended as a writer's aid, with additional hopes to later develop it into an autonomous storyteller.

UNIVERSE addressed a question of procedure by making up a story about a fictional world: whether the world should be built first with a plot added afterwards, or whether the plot should drive the construction of the world, with characters, locations and objects being created as needed. Lebowitz declared himself in favor of the first option, which is why UNIVERSE included facilities for creating characters independently of plot, in contrast to Dehn, who favored the second option.

An interesting point about UNIVERSE is that, being a story with no recognizable ending, the system alternated between generating a new episode to continue the story and telling the most recent episode it had generated.

MINSTREL (Turner 1993) was a computer program that told stories about King Arthur and his Knights of the Round Table. Each run of the program was based on a moral that was used as a seed to build the story, e.g.: "Deception is a weapon difficult to aim." MINSTREL created stories about one-half to one page in length. According to its author, MINSTREL could tell about ten stories of this length and it could also create a number of shorter story scenes.

MINSTREL used building units consisting of goals and plans to satisfy them. These operated at two different levels: author goals and character goals. Story construction in MINSTREL operated as a two-stage process involving a planning stage and a problem-solving stage which reused knowledge from previous stories.

Pérez y Pérez's MEXICA (1999) was a computer model whose purpose was to study the creative process. It was designed to generate short stories about the early inhabitants of Mexico. During the engagement phase, new story material was progressively generated, with no constraints imposed. During the reflection phase, the generated material was revised to ensure that generic constraints are met.

MEXICA was a pioneer in that it took into account emotional links and tensions between the characters as a means for driving and evaluating ongoing stories.

Jaguar knight was an inhabitant of the Great Tenochtitlan. Princess was an inhabitant of the Great Tenochtitlan. Jaguar knight was walking when Ehecatl (god of the wind) blew and an old tree collapsed, injuring badly Jaguar knight. Princess went in search of some medical plants and cured Jaguar knight. As a result, Jaguar knight was very grateful to Princess. Jaguar knight rewarded Princess with some cacauatl (cacao beans) and quetzalli (quetzal) feathers.

BRUTUS (Bringsjord & Ferrucci 1999) was a program that wrote short stories about

betrayal. BRUTUS was interesting because it based its storytelling ability on a logical model of betrayal. The richness of this model and the inferences that can be drawn from it enabled it to produce very rich stories. The system was also designed to take into account a large body of knowledge about literature and grammar.

BRUTUS was capable of creating a story of impressive quality, with most of the features (in terms of literary tropes, dialogue, identification with the characters, etc.) one would find in a human-authored story. However, the authors make it clear that BRUTUS is not creative at all but the result of reverse engineering a program out of a story in order to see whether it can build that particular story.

FABULIST (Riedl & Young 2010) was an architecture for automated story generation and presentation. The Fabulist architecture split the narrative generation process into three tiers: fabula generation, discourse generation, and media representation. The fabula generation process used a planning approach to narrative generation. AI planners are applications that, given a description of an initial state of the world and a specific goal, identify the optimal sequence of actions to reach the goal. They rely on detailed descriptions of the preconditions and postconditions of all the possible actions. The planning approach to narrative generation is based on the assumption that a sequence of actions leading from an initial state to a goal is a good approximation of a story. In the case of FABULIST, inputs provided included a domain model describing the initial state of the story world, possible operations that can be enacted by characters and an outcome.

3.1 Algorithm Types

The systems reviewed above include various types of algorithm to generate the stories they produce. Individual systems sometimes combine more than one type of algorithm. This section reviews the types of algorithm and explains in each case how a given algorithm is applied in each storytelling system.

A large number of existing storytelling systems rely on solutions based on planning. These solutions take as input an initial state of the world and a desired goal and then produce a sequence of actions that will lead from one to the other. Such solutions are applied to storytelling in different ways. Some systems use authorial goals to drive the story planning process while others simply consider character goals. The simplest versions just generate actions that may follow previous events, with no particular notion of goal. The Novel Writer system (Klein et al. 1973) relied on a micro-simulation model where the behavior of individual characters and events were governed by probabilistic rules that progressively changed the state of the simulated world (represented as a semantic network). The flow of the narrative arises from reports on the changing state of the world model. Because it had no

explicit notion of goal, this procedure required additional information to guide it (see below). TALESPIN combined forward-chaining (from events to their consequences) and backward-chaining (from desired outcomes expressed as goals that resulted from an event previous to the particular events that will lead to the outcome). A large part of the work of making up a story in AUTHOR is the perpetual reformulation of author goals. Both the UNIVERSE (Lebowitz 1983) and MINSTREL (Turner 1993) storytelling systems involve a planning stage that keeps track of a set of pending goals which drive the expansion of a partial draft of the story until a complete plot is obtained. The “Intent-Driven Partial Order Causal Link” (IPOCL) planning algorithm used by Fabulist (Riedl & Young 2010) simultaneously reasoned about causality and character intentionality and motivation in order to produce narrative sequences that are causally coherent (in the sense that they drive towards a conclusion) (Toolan → Coherence [2]) and have elements of character believability. Fabulist first generates a narrative plan that meets the outcome objective, ensuring that all character actions and goals are justified by events within the narrative itself.

Another typical solution is to rely on a set of resources that abstract key elements of story structure, similar to story grammars. Although the major driving mechanism of the Novel Writer system (Klein et al. 1973) was planning, the sequence of scenes used to build up the story was already spelt out and hard-wired in the code to correspond to the expected development of a weekend party, with the simulation only accounting for the interplay between the characters that fleshes out the plot. This sequence of scenes could be considered an instance of a primitive story grammar. The operation of BRUTUS (Bringsjord & Ferrucci 1999) involved both a simulation process (where characters attempt to achieve a set of pre-defined goals) and the application of a hierarchy of grammars (story grammars, paragraph grammars, sentence grammars) that define how the story is constructed as a sequence of paragraphs which are themselves sequences of sentences.

Other systems apply solutions that mine a set of previous stories to obtain material they can reuse in building new ones. The actual story generation process of UNIVERSE (Lebowitz 1983) uses snippets of plot that include information about goals and actions to generate plot outlines. The problem-solving stage of MINSTREL (Turner 1993) solved author-level goals by querying the system’s episodic memory (where instances of previous stories are stored) in order to instantiate a set of partially complete character schemas derived from the input. MEXICA (Pérez y Pérez 1999) searches a set of knowledge structures to find possible continuations to an ongoing plot, based on matching the set of emotions and tensions between one and the other (Emmott & Alexander → Schemata [3]).

3.2 Interactive and other Storytelling Applications

In addition to classic storytellers, there has been a very significant growth of *interactive storytelling applications*. These are interactive computer applications that allow the user to dictate the behavior of a given character involved in a simulated environment. The interactivity involved ranges from plain text interaction (as in Interactive Fiction, or IF, where the computer produces a rendition of the story as text, interspersed with the text commands that the user has written) to 3D simulated worlds similar to video games (where the story generation module is used to drive the behavior of virtual characters and the story is only rendered visually). These interactive storytelling applications are too numerous to be included in the present review, but they clearly deserve a separate study and deserve a specific reviewing effort.

Another related family of applications is that of systems designed to construct story text from a conceptually represented story discourse, as in Charles Callaway and James Lester's STORYBOOK (2002), or to construct story discourse from an underlying fabula, as in Nick Montfort's Narrator module in the nn system (Montfort 2007), now known as Curveship (Montfort 2011), or the Suspenser (Cheong 2007) and Prevoyant (Bae & Young 2008) systems, which aim for discourses showing evidence of specific characteristics such as suspense or surprise. All of these types constitute examples of computational algorithms for tasks that are clearly important in the construction of stories. However, they have been considered separately from the other systems reviewed above for the reason that they are not aimed at actually generating a story, but rather at telling it in particular ways. A related family of applications aims to develop cinematic visual discourse from an underlying fabula (Jhala & Young 2010). These applications also rely heavily on narratological concepts and constitute a significant research field that brings together computation and narratology.

Together with existing efforts at automating analysis of narrative in various ways (Mani 2010), this set of research lines is collectively becoming known as Computational Narratology (Mani → Computational Narratology [4]), which has recently experienced a very significant growth.

4 Relevance for Narratology

Story generator systems are clearly at an early stage of development. In their current state, they are obviously not producing the depth and richness of narrative that narratology is mostly concerned with. However, the engineering principles that must be applied when designing and constructing such systems force the consideration of issues that narratology has not focused on, but which may benefit

enormously from taking them into consideration. From a narratological point of view, the main relevance of this research lies in the “testing” of narratological concepts for their unambiguity and applicability, which are absolute criteria for an algorithm-driven system to function (Lönneker et al. 2005; Meister 2005). Whereas analysis of literary texts invites a broad array of concepts to ensure applicability over a large variety of texts and contexts, the design of algorithm-driven systems requires precise definitions on which story construction decisions are based. Storytelling systems may be used to identify prototypes of narratological concepts in actual use to support story building decisions. This may be helpful both in allowing identification of existing concepts that may be under-defined or ambiguous in their present formulation, and in putting forward additional concepts concerned with the process of composition of stories that may be worthy of further attention.

5 Topics for Further Investigation

Research on story generator algorithms is very much an ongoing effort. To date, the interaction between narratology and artificial intelligence on this subject has been limited and inconclusive. This is slowly changing, however, as evidenced by the rise of Computational Narratology. This frontier still needs to be explored, as each field could contribute significantly to the other. The algorithmic implementation of story generation systems requires not only a clearly defined set of concepts on which to base the implementation, but also a clear division of the overall effort of story generation into particular tasks that are easier to model. This may require a distinction between the process of constructing a story and the process of telling it once it has been constructed. These two processes are obviously interrelated in the case of fiction, but they are also conceptually different from the point of view of what their inputs and outputs are. A clear contribution from narratology to clarifying the relations and interactions between these two processes would constitute a significant contribution to storytelling. As mentioned above, story generation research may provide a very interesting benchmark for practical testing of the extent to which narratological concepts are clear and precise enough to be transformed into working implementations of storytellers.

6 Bibliography

6.1 Works Cited

Bae, Byung-Chull & R. Michael Young (2008). “A use of flashback and foreshadowing for surprise arousal in narrative using a plan-based approach.” U. Sperling & N. Szillars (eds.). *Interactive Storytelling. First Joint International Conference on Digital Storytelling, ICIDS 2008*, Erfurt, Germany, 26-29. Proceedings Series:

Lecture Notes in Computer Science, Vol. 5334. Berlin: Springer, 156–67.

Bringsjord, Selmer & David A. Ferrucci (1999). *Artificial Intelligence and Literary Creativity Inside the Mind of BRUTUS, a Storytelling Machine*. Hillsdale, N.J.: Lawrence Erlbaum.

Callaway, Charles B. & James C. Lester (2002). "Narrative prose generation." *Artificial Intelligence* 139.2, 213–52.

Cheong, Yun-Gyung (2007). *A Computational Model of Narrative Generation for Suspense*. PhD Thesis, North Carolina State University.

Dehn, Natalie (1981). "Story Generation after Tale-Spin." A. Drinan (ed.). *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, August 24–28, University of British Columbia, Vancouver, Canada. Los Altos: Kaufmann, vol. 116–18.

Gervás, Pablo (2009). "Computational approaches to storytelling and creativity." *AI Magazine* 30.3, 49–62.

Jhala, Arnav & R. Michael Young (2010). "Cinematic visual discourse: Representation, generation, and evaluation." *IEEE Transactions on Computational Intelligence and AI in Games* 2.2, 69–81.

Klein, Sheldon et al. (1973). *Automatic novel writing: A status report. Technical Report 186*, Computer Science Department, The University of Wisconsin, Madison.

Lebowitz, Michael (1983). "Creating a Story-Telling Universe." A. Nundy (ed.). *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, August 8–12, Karlsruhe, Germany. Los Altos: Kaufmann, vol. 1, 63–65.

Lönneker, Birte et al. (2005). "Story Generators: Models and Approaches for the Generation of Literary Artefacts." *ACH/ALLC 2005 Conference Abstracts. Proceedings of the 17th Joint International Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing*, Victoria, BC, Canada, June 15–18, 126–33.

Mani, Inderjeet (2010). *The Imagined Moment: Time, Narrative, and Computation*. Lincoln: U of Nebraska P.

Meehan, James R. (1977). "Tale-Spin, an interactive program that writes stories." *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, MIT, Cambridge, MA, August 22–25. Los Altos: Kaufmann, 91–98.

Meister, Jan Christoph (2005). "Computational approaches to narrative." D. Herman et al. (eds.). *Routledge Encyclopedia of Narratology*. London: Routledge, 78–80.

Montfort, Nick (2007). *Generating narrative variation in interactive fiction*. PhD Dissertation, University of Pennsylvania, Philadelphia.

Montfort, Nick (2011). "Curveship's automatic narrative variation." *Proceedings of the 6th International Conference on the Foundations of Digital Games (FDG '11)*, Bordeaux, France, June 29–July 1. New York: ACM, 211–18.

Pérez y Pérez, Rafael (1999). *MEXICA: A Computer Model of Creativity in Writing*.

PhD Dissertation, The University of Sussex.

Riedl, Mark O. & R. Michael Young (2010). "Narrative Planning: Balancing Plot and Character." *Journal of Artificial Intelligence Research* 39, 217–68.

Turner, Scott R. (1993). *Minstrel: a computer model of creativity and storytelling*. PhD Dissertation, University of California at Los Angeles, Los Angeles.

6.3 Further Reading

Bailey, Paul (1999). "Searching for storiness: Story generation from a reader's perspective." *Narrative Intelligence. Papers from the 1999 AAI Fall Symposium*. North Falmouth, MA, November 5–7. Menlo Park, CA: AAI Press, 157–63.

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

Meehan, James R. (1981). "Tale-Spin." R. Schank (ed.). *Inside Computer Understanding*. Hillsdale: Lawrence Erlbaum, 197–225.

Pérez y Pérez, Rafael & Mike Sharples (2004). "Three computer-based models of storytelling: BRUTUS, MINSTREL and MEXICA." *Knowledge-Based Systems* 17, 15–29.

Riedl, Mark O. & Neha Sugandh (2008). "Story planning with vignettes: Toward overcoming the content production bottleneck." U. Sperling & N. Szillas (eds). *Interactive Storytelling. First Joint International Conference on Digital Storytelling, ICIDS 2008*, Erfurt, Germany, 26–29. Proceedings Series: Lecture Notes in Computer Science, Vol. 5334. Berlin: Springer, 168–79.

Turner, Scott R. (1994). *The Creative Process: A Computer Model of Storytelling and Creativity*. Hillsdale: Lawrence Erlbaum.

To cite this entry, we recommend the following bibliographic format:

Gervás, Pablo: "Story Generator Algorithms". In: Hühn, Peter et al. (eds.): *the living handbook of narratology*. Hamburg: Hamburg University. URL = <http://www.lhn.uni-hamburg.de/article/story-generator-algorithms>
[view date:12 Feb 2019]