Evolutionary Combination of Subplot Patterns into Meaningful Plots

Pablo Gervás (pgervas@ucm.es)
Complutense University of Madrid
Gonzalo Méndez
Complutense University of Madrid
Eugenio Concepción
Complutense University of Madrid

Research Article

Keywords: story generation, subplot patterns, evolutionary approach, metrics on pattern compatibility, character instantiation

Posted Date: October 27th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2169837/v1

Evolutionary Combination of Subplot Patterns into Meaningful Plots

Pablo Gervás[1,2*], Gonzalo Méndez[1,2] and Eugenio Concepción[1]

[1*]Facultad de Informática, Universidad Complutense de Madrid, c/ Profesor García Santesmases, 9, Madrid, 28040, Spain.
[2*]Instituto de Tecnología del Conocimiento, Universidad Complutense de Madrid, c/ Profesor García Santesmases, 9, Madrid, 28040, Spain.

*Corresponding author(s). E-mail(s): pgervas@ucm.es;
Contributing authors: gmendez@ucm.es; econcepc@ucm.es;

Abstract

Purpose: Many of the stories we are exposed to are built from small patterns of connected events involving a set of characters – *boy meets girl* leads to a *relationship* or *crime* leads to *revenge*. The present paper studies the computational constraints that apply to the task of putting together a story by combining a set of such patterns. This approach presents three challenges: how to mix up the elements in the different patterns, how to instantiate the characters across the patterns and how to tell acceptable combinations from the rest. **Methods:** The present paper applies an evolutionary solution that relies on a genetic representation for these combinations of patterns, and applies as fitness functions a set of metrics on compatibility constraints across pattern combinations. Outputs of this procedure are evaluated by human judges in comparison with baseline solutions. **Results:** The proposed solution generates a population of story drafts that resemble plot descriptions for simple stories. A comparative evaluation by human judges against baselines based on random gene assignment yields positive results. **Conclusion:** The genetic representation of pattern combinations and the metrics on compatibility across pattern pairs provide a valid evolutionary solution for constructing simple plots.

Keywords: story generation, subplot patterns, evolutionary approach, metrics on pattern compatibility, character instantiation

Contents

1 Introduction

Most of us consume stories regularly in our current existence, whether in the form of movies, novels, TV series or podcasts. Many of these stories are built from small patterns of connected events involving a set of characters – *boy meets girl* leads to a *relationship* or *crime* leads to *revenge*. As we read or watch the stories we identify such patterns in the stories and remain on the lookout for the events that complete them. A great part of our enjoyment of stories arises from this process – sometimes from seeing the patterns completed, sometimes from seeing them transgressed. Most people can identify this type of pattern, and yet there is no consensus on what the set of such patterns might be. Some non-academic efforts have been made to compile instances of these patterns, such as the *TVTropes*[1] web site, a very impressive crow-sourced

[1]https://tvtropes.org/

compilation (Brehob, 2013). But even if these building blocks are identified very little is known about the process by which they are combined to yield full stories. The present paper studies the computational constraints that apply to the task of putting together a story by combining a set of such patterns. This approach presents three challenges: how to mix up the elements in the different patterns, how to instantiate the characters across the patterns and how to tell acceptable combinations from the rest.

The present paper considers some existing work on potential means of representing plot at different levels of granularity and applies an evolutionary solution to explore the resulting search space. The evolutionary solution relies on a genetic representation for these combinations of patterns, and on fitness functions informed by a set of metrics on compatibility constraints across pattern combinations. Outputs of this procedure are evaluated by human judges in comparison with baseline solutions.

2 Related Work

Three topics are considered relevant for this paper: prior solutions for the representation of plot, approaches to constructing stories by combining small units of representation, and evolutionary approaches to creation of narratives of some kind.

2.1 Representing Subplot Patterns

The understanding of narrative as a form of communication has been a major subject of study in the field of humanities and became a challenge for computational approaches since the early advent of artificial intelligence.Some relevant approaches are reviewed here.

Russian formalist Vladimir Propp studies a corpus of Russian folk tales and proposed a formal representation for the basic units that made up their plot structures (Propp, 1968) . Propp postulated the concept of a *character function* as a relatively abstract representation of the meaning of an event involving some characters that is relevant to the plot of the story. These events represent the structural elements in a story at a very low level of granularity, because they involve individual actions such as characters meeting, misbehaving, fighting, or travelling. However, Propp identified that these events were connected to other events in the story by virtue of specific characters that necessarily took part in the same set of events. In this way, the victim of an abduction event at the start of the story establishes a connection with the rescue event that happens – to the same character – later in the story. Propp postulates a set of *spheres of action* that define certain specific roles that characters may play in a story: hero, villain, victim, donor, helper... Because they are quite simple and yet provide a semblance of formal structuring, character functions have been used often as basic representation in attempts to generate stories automatically. But, being limited to representing individual events, they fall short of providing a usable representation of the types of patterns that we want to consider.

Attempts to capture the structure of plot from beginning to end do consider sequences of events that correspond to observed plot archetypes. Existing efforts postulates different numbers of such archetypes as basic structures to understand narrative: one for *the hero's journey* (Campbell et al, 1990), seven in (Booker, 2004), twenty in (Tobias, 2012) or 1,462 in Plotto (Cook, 2011). Such efforts go to the other extreme, because they represent very complex units that completely define the plot of the story. They are therefore too large to represent the type of patterns that we require.

An intermediate degree of granularity has been defined in the *axes of interest* postulated in the PlotAssembler system (Gervás, 2019). These axes of interest (or AoIs) are small sets of events that do not necessarily occur contiguously in the discourse for a story but which are connected by shared characters that give them meaning – like the victim of a kidnapping being rescued later. As axes of interest are chosen to represent subplot patterns in the present paper, they are reviewed below.

2.2 Story Construction by Combination of Plot Relevant Units

The use of planning technologies for story generation (Young et al, 2013) may be considered an instance of processes of construction of stories by combining partially structured fragments of story material. In this case, the basic units used for construction are *planning operators*, which include a story action that represents the main event of the operator – usually in the form of a predicate-argument structure –, and a number of preconditions and postconditions – also represented by similar predicates. When building a plan structure to represent the outcome story, preconditions may be unified with predicates already in the plan – and not necessarily at positions in the story discourse contiguous to the event being added at that stage. Arguments shared across preconditions, main action and postconditions represent connections between different events. Used in this way, planning operators could be seen as possible representations for the type of pattern we want to represent. However, for planning techniques to be applicable the relations between the preconditions and the main action of a planning operator need to imply a certain causal relation. This is not necessarily true in many of the patterns we want to represent.

A different approach that also builds stories by combining predefined fragments of material that are partial representations of plot can be found in recent attempts to build stories with more than one plot line. Stories beyond the simpler instances are known to involve often more than one plot line. A *plot line* when used in this context refers to a sequence of plot-relevant elements or *scenes* that make sense in the order in which they appear in the story, and linked by at least a shared set of protagonist and secondary characters. The patterns that we want to represent may indeed by considered very small instances of plot lines, though in general, the concept of plot line has a connotation of slightly more complex sets of events and of the relations between them.

Closely related to the planning approach described above, Porteous et al (2016) present a plan-based procedure for creating multi-plotline stories for an interactive storytelling system. The complete plan is built incrementally as partial selections from the plans that result from attempting to lead the draft at that point towards a predefined goal. At each point, only the next action in the given plan is added to the draft before the initiative is passed to the user. The user intervention usually results in a need to rebuild the plan. In their approach, the different plot lines are represented by different spans of the overall draft involving specific sets of characters.

This concept of subset of a story involving a particular character is sometimes referred to as a *narrative thread*. The work by Fay (2014) relies on narrative threads of this type as building units for constructing complex stories. The approach starts from a set of narrative threads for particular types of characters – extracted from a corpus of existing stories – , and, for a given story request that mentions specific types of character, constructs multi-plotline stories by first selecting threads matching the type of characters in the request, finding a combination of the elements from each thread into a consistent timeline, and identifying valid bindings between characters in different threads that make the story consistent.

The PlotAssembler system (Gervás, 2019) – which introduces the concept of axes of interest mentioned above – takes as input a set of axes of interest provided and interweaves the scenes in these in an order designed to maximise the probabilities of character continuity across scenes – as mined from a corpus of prior stories.

At the furthest level of granularity in the representation of plot, the work of Concepción et al (2020) operates on a set of plot templates for complete stories, and it proposes procedures for weaving them together into multi-plotline stories. Some of these procedures are drawn from know techniques used in existing narrative but they also include simple computational approaches that are presented as baselines to compare with.

2.3 Evolutionary Construction of Narratives

Evolutionary solutions have been used in the past to construct stories from smaller units. McIntyre and Lapata (2010) use genetic algorithms to explore the search space of possible merges between plot lines previously extracted from a set of stories. Each plot line is represented as a partially ordered graph of events associated with a given entity. A set of entities is received as input and the process is driven by a fitness function designed to maximise story coherence and story interest. Gómez de Silva Garza and Pérez y Pérez (2014) build stories by using the GENCAD evolutionary approach for the adaptation stage in case-based solutions to architectural problems (Gómez de Silva Garza, 2000) to refine an initial population built using the knowledge-based heuristics of the MEXICA knowledge-based story generator (Pérez y Pérez and Sharples, 2001). Fredericks and DeVries (2021) present a generator of small fragments of narratives – to be used in text-based games – that applies an evolutionary solution driven by novelty search (Lehman and Stanley, 2004). Kartal et al (2014) generate narratives using a plan-based approach supported by a Monte Carlo tree search driven by a combination of measures of how believable the resulting story is and how many of the goals defined by the user are accomplished by the story. de Lima et al (2019) generate quests for games by combining a planner that constructs candidate quests as linear sequences of tasks for the user with an evolutionary search strategy that selects from them those that best match a a target curve provided by the user of how tensions should evolve in the quest.

The work of Gervás et al (2022) explores an evolutionary solution for the combination of plot templates for complete stories as described in (Concepción et al, 2020). This approach proposes a genetic representation for a combination of fragments of plot – such as plot templates or plot lines – that includes genes that govern the order in which elements from different fragments appear in the final discourse and genes that govern how character variables from different fragments may be instantiated to the same character in the final story. This division corresponds to the two main tasks that make up the process: *discourse planning* – decisions about in what order to present the elements of the story as a sequential discourse – and *character fusion* – decisions about how characters from the different fragments being combined are themselves fused into a single character in the final story. This provides a reliable mechanism for finding combinations of fragments of plot like that ones we want to consider. The fitness functions used in this approach relied on metrics that measured how consistent the final story was in terms of basic semantics such as characters being alive in the story at points of the story where they are active. Such constraints had been identified as relevant to human judges in the formative evaluation carried out in (Concepción et al, 2020).

3 Evolutionary Combination of Plot Units Driven by Consistency Metrics

The solution described in this paper explores how well the task of combining subplot patterns into a simple story can be addressed by a combination of: the *representation of plot* as axes of interest (Gervás, 2019), the application of the *genetic representation* presented in (Gervás et al, 2022) for combining spans of partially ordered plot elements, a set of new *metrics on compatibility of patterns of combination* of plot element for pairs of axes of interest and a *preprocesing stage* that checks a given set of axes of interest for mutual compatibility – in terms of that constraints on relative ordering that arise between the elements involved.

3.1 Knowledge Representation for Plot

To achieve the goal of the paper a representation is needed for small patterns of plot relevant events that can be considered building blocks for larger patterns of plot such as plot lines. For lack of a better word we will refer to them as *subplot patterns*. These building blocks need to be themselves constructed from plot relevant elements that allign with the concept of event, and they need to allow representation of the characters that take part in them. In this paper we will use *axes of interest* to represent the concept of subplot patterns, and the axes of interest will be ordered groupings of *plot atoms*.

3.1.1 Plot Atoms as Basic Units of Plot

As smallest unit of plot relevant element we will consider the concept of *plot atom*. A plot atom is conceptually similar to a character function in that it represents an action by one or more characters that is relevant to some aspect of the plot of the story. In contrast to character functions, each plot atom explicitly holds additional information to indicate how the roles specific to the plot atom (*kidnapper*, *kidnapped*) are filled in by characters playing roles that are relevant to the plot (**villain**, **victim**). This refinement allows for interesting articulation between roles specific to a plot atom and roles more general to the plot at large. The variables employed in a plot atom to represent the participating entities are separated into three different sorts: characters, objects and locations. In this way, objects and locations may play relevant roles in the plot as well as characters. The use of sorts to separate these types of entities ensures that during evolution there are no instances of characters replaced with objects or objects mistaken for locations.

3.1.2 Axes of Interest (AoIs) as Representation of Subplot Patterns

The type of small pattern of related and not necessarily contiguous plot atoms that we want to operate with are represented by *axes of interest* (AoIs). An axis of interest is a sequence of plot atoms related by a conceptual dependency. For example, a pattern representing a JOURNEY would include a plot atom for an event of `Departure` – usually somewhere towards the start of the story – and plot atom for an event of `Return` – again often somewhere towards the end of the story –, but these two plot atoms are structurally connected. The conceptual dependency may operate over a long range – as in the example of a journey – or at very short range – such as in a CONFLICT, where a `Struggle` is closely followed by a `Victory`. An axis of interest has a set of *narrative roles* – those of its constituent plot atoms – that are initially free variables but which can be instantiated to specific constants representing entities when the axis of interest is combined into larger structures. When a variable in this set is instantiated to a particular entity name, all the appearances of it in the associated plot atoms are instantiated as well. In the example above, for the AoI to make sense the traveller in a JOURNEY needs to be the same in the `Departure` and the `Return`, and the *origin* location for the `Departure` needs to match the *end* location of the `Return`.

Three different examples of axes of interest are shown in Table 1.

AXISofINTEREST =	Rags2Riches
`Poverty`	sufferer = **x**
`Aspiration`	aspirer = **x**
`Transformation`	transformed = **x**
`AspirationAchieved`	achiever = **x**
AXISofINTEREST =	HappyLove
`BoyMeetsGirl`	boy = **x**, girl = **y**
`FallInLove`	lover = **x**, beloved = **y**
`HappyEverAfter`	lover = **x**, beloved = **y**
AXISofINTEREST =	ShiftingLove
`GirlMeetsOtherBoy`	previous-love = **x**, girl = **y**, boy = **z**
`LoveShift`	lover = **z**, beloved = **y**, rival = **x**
`Reconciliation`	lover= **x**, beloved = **y**

Table 1 Three examples of Axes of Interest, with one – Rags2Riches –, two – HappyLove – and three – ShiftingLove – linked participating characters. Co-occurrence in variable names is relevant, so that, in the final story, the character that plays **x** in any of the plot atoms for a given AoI must be the same as the one that plays **x** in all the other plot atoms for the same AoI.

The set of axes of interest – and the corresponding set of plot atoms – used in the present paper resulted from the knowledge engineering effort described in (Gervás, 2019). In this effort, a number of sources in the literature were consulted – including Propp's character functions (Propp, 1968), Booker's seven basic plots (Booker, 2004) and Polti's situations (Polti and Ray, 1916) – and a process of abstraction and condensation was applied. As a result a set of 34 basic plot atoms was obtained, together with a set of 19 axes of interest that provide possible patterns of structuring for particular subsets of plot atoms. Interested readers are referred to the original paper for details.

3.2 Combining AoIs into Story Drafts

As representations of the kind of small patterns of related events that occur in a plot, we want these axes of interest to be combined together, interleaving the various sequences of atoms of the AoIs involved in an order that makes sense as description of the plot of a story. We consider such a description of the plot of a story as a *story draft*. In a story draft, the ordered sequence of plot atoms from the axes of interest is referred to as the *discourse* for the story draft. In this discourse, each plot atom carries an additional label to indicate the axis of interest that it comes from.

An example of story draft is presented in Table 2, which shows how the HappyLove, UnrelentingGuardian and Task axes of interest are interleaved to form the basic story draft. It also shows how the narrative roles for the story draft (**hero**, **love-interest**, **guardian**) are mapped to the roles specific to the plot atoms of the constituent axes of interest (*boy*, *girl*, *lover* and *beloved* for the HappyLove axis of interest; *lover*, *beloved* and *guardian* for

HappyLove	BoyMeetsGirl	(*boy* = **hero**, *girl* = **love-interest**)
HappyLove	FallInLove	(*lover* = **hero**, *beloved* = **love-interest**)
UnrelentingGuardian	CoupleWantsToMarry	(*lover* = **hero**, *beloved* = **love-interest**)
UnrelentingGuardian	UnrelentingGuardian	(*lover* = **hero**, *beloved* = **love-interest**, *guardian* = **parent**)
Task	DifficultTask	(*setter* = **parent**, *solver* = **hero**)
Task	Solution	(*solver* = **hero**)
UnrelentingGuardian	GuardianRelents	(*lover* = **hero**, *beloved* = **love-interest**, *guardian* = **parent**)
UnrelentingGuardian	Wedding	(*lover* = **hero**, *beloved* = **love-interest**)
HappyLove	HappyEverAfter	(*lover* = **hero**, *beloved* = **love-interest**)

Table 2 Example of story draft for a basic plot combining axes of interest for HappyLove, UnrelentingGuardian and Task. The first column shows the interweaving of the axes of interest. The co-occurrence of constants in the final column – shown in **bold** – provides the argumental connections between the three AoIs.

the UnrelentingGuardian axis of interest; and *setter* and *solver* for the Task axis of interest). This ensures that the various plot atoms in the plot are instantiated in a manner coherent with the narrative roles that the characters play in the overall story draft.

The inclusion of this type of connection in terms of shared characters between the constituent AoIs in a story constitutes an instance of *character fusion*. These connections that relate plot atoms across the different axes of interest being combined are going to be used to build the metrics that will be used as fitness functions in the evolutionary procedure.

3.3 Metrics for Acceptability of Stories

Any process of computational construction of stories is likely to yield a large number of potential stories, so there is a need for some means of measuring the quality of drafts that can help identify valuable candidates among this search space. There have been numerous efforts to develop valid metrics for story quality (León et al, 2020; Gervás et al, 2021), but they all show that the perception of quality for narratives is greatly influenced by many subjective matters such as emotion, attribution of motivation, empathy. . . that present two serious difficulties: very little is known about them and they are extremely complex to represent. Furthermore, none of them is captured in any way in the type of representation that we are proposing for this effort. This is why we are restricting the evaluation of quality of candidates stories to considering them acceptable in terms of the two aspects that the procedure is designed to consider: whether the relative order in the sequence of plot atoms in the discourse makes sense, and whether the co-occurrence of entities across the different AoIs in the story is coherent.

To this end, we develop a set of metrics that measure correct sequencing and correct co-instantiation of variables over each potential pairwise combination of two AoIs, designed to cover the following aspects:

- *role-sharing constraints* on a particular character playing a role X in one of the AoIs and a role Y in the other (say, the traveller in a journey becoming the victim of a kidnapping)
- particular *sequencing constraints* on the atoms for the AoIs involved, possibly arising from a particular shared role (for instance, a kidnapped traveller should return only after he has been released from his kidnapping)

An example of the way these constraints are represented in the entries for a particular combination of a pair of AoIs is:

```
CallToActionReward+Conflict =
hero + hero =
CallToAction + Struggle / Victory + Reward
```

This expresses that fact that, for a combination of a hero being called to action (CallToActionReward) and involved in a fight (Conflict) – first line –, the `hero` of one should be the `hero` of the other – second line –, the fight should take place after the hero has been called and the reward should be obtained after the victory – third line.

Not all pairwise combinations of AoIs allow the formulation of this type of constraints. This is actually helpful, because the constraints arising from different pairs may be incompatible with one another, and too many constraints make it difficult to produce acceptable solutions. A specific solution is required to handle this profusion of constraints when these metrics are applied as fitness functions for our evolutionary process (see Section 3.5).

A particular pairwise combination of AoIs is assigned a numerical score over a total of 100. Of that score, 50 points are assigned based on role-sharing constraints. Each role-sharing constraint present is scored 100 if met and 0 otherwise, and the average value of all role-sharing constraints taken as the total role-sharing score (normalised to 50). The remaining 50 points are computed by:

- assigning 100 points to any precedence constraint that is met (for A + B, A appears before B in the discourse sequence)
- if a required precedence constraint is not met, a partial score over 100 is assigned corresponding to the number of positions that one of the elements would need to shift for the constraint to hold (normalised over the length of the sequence)
- the average of all sequencing constraints is taken as the total sequencing score (normalised to 50).

These metrics provide a progressive scoring, so that drafts where the sequencing constraints are not met are scored relative to how far they need to be modified for the constraints to be met. This allows mutations that modify the sequence in the right direction to be scored progresively higher, allowing evolution to converge towards optimal solutions.

3.4 Representing AoI-based Stories for Evolutionary Construction

As mentioned in section 2.2, the difference between a small pattern for plot and a plot line as defined in (Gervás et al, 2022) is that the small pattern does not usually cover a complete story on its own. But their structural properties – in terms of being a sequence of plot atoms connected by restrictions on shared characters – hold just the same. We can therefore use a very similar formalism for representing the building blocks to construct stories and very much the same evolutionary mechanism for the task of combining them together into stories. The genetic representation presented in (Gervás et al, 2022) for combining templates for complete plots can be adapted to the combination of AoIs. To ensure that the present paper is understandable as a self-contained unit, this adaptation is described here in detail.

Because the task of combining the AoIs into stories requires decisions at two different levels – *discourse planning* decisions concerning the relative order of presentation of plot atoms and *character fusion* decisions concerning instantiations of shared characters across AoIs – the representation will require separate features to deal with each level.

One important difference with the original representation in (Gervás et al, 2022) is that the plot templates used there only considered characters as elements to be instantiated, whereas AoIs consider three sorts of elements to be instantiated: characters, objects and locations.

The operational details of the problem of constructing a story for a given a combination of AoIs is greatly determined by the particular set of AoIs taken as input, because the required genetic representation will differ based on the specific lengths of the AoIs involved and the number of character roles that each AoI contributes to the story. For a particular problem of combining N AoIs, the length of the final discourse is determined by the total number of scenes in the AoIs being considered, and the maximum number of possible entities featuring in the story is determined by the union of the sets of entities in the AoIs being considered.

Refined instances of storytelling often rely on advanced mechanisms for presenting a story – flashbacks, flashforwards – that involve presenting the scenes in the story in an order that differs from the chronological order in which they are supposed to have happened. Such instances of altered chronology are indeed very powerful tools and very interesting to explore as additional computational challenges, but we consider them beyond the scope of the present paper and we leave them for further work. We will therefore assume that the relative chronological order of the scenes in each AoI is respected in the final discourse.

A genetic representation of the discourse plan for a given story candidate must represent the following information: which AoI the discourse starts on, at what point in an AoI the discourse switches to a different AoI, and to which of the other AoIs in the draft does the discourse jump when it abandons the AoI it was on. To capture this information we use vectors that define the answer to these questions as follows:

- a single digit (0 or 1) defines which AoI the final discourse starts with
- a sequence of digits (0 or 1) defines for the total number of scenes in the final discourse whether the next scene follows on with the prior AoI or it switches to a different AoI

- a sequence of digits (ranging between 1 and N-1, where N is the total number of plots being combined) defines how many of the available AoIs are skipped whenever the discourse switches to a different AoI

A genetic representation of the way in which characters from different AoIs are combined would need to represent which of the terms used for entities in the story is assigned to each variable for entities that participates in each of the AoIs involved. We represent this information in terms of three different vectors – one for each different sort of entity: characters, objects or locations – that define how the entity roles for the different AoIs are instantiated by the entity names that appear in the final story draft. Within the restricted set of variables of a given sort, the procedure for instantiation, initial assignment, mutation and cross over is the same, but applied to the corresponding set of entities.

The set of possible entities for the complete story is defined by the unions of the sets of variable names for the each of the three sorts that appear in each AoI. These variable names need to be distinct across the different AoIs to avoid confusion. This is a challenge because, for instance, many of the definitions for AoIs identify a particular entity as "the hero", and most of the AoIs dealing with romantic liaisons include variables for "lover" and "beloved". To avoid this problem, the name of the AoI is assigned as a prefix to all the variable names that feature in that AoI.

The caracterisation of the choices for entity fusion for a given story candidate requires an assignment of entity names to each of the variables in the joint set of variables for the story. This is applied separately for each sort of entity.

For simplicity, the set of potential entity names of each sort for the story is defined to be the set of integers from 0 to N, with N being the cardinality of the joint set of variables of that sort for the story. To avoid confusion across sorts, entity names for a particular sort are assigned distinguishing prefix: `C` for characters, `O` for objects, `L` for locations. This is sufficient to represent any choices made in terms of entity fusion (with variables in two different positions in the name-assignment vector being assigned to the same (prefixed) integer). The form of the resulting stories would be significantly improved by a later stage of transforming these integer-based names for the characters into strings representing realistic names.

An example of representation is shown in Figure 1.

3.5 Fitness Functions

In order to apply the metrics, the genetic representation described needs to be applied to inform a process of construction of story drafts. Then the set of metrics available for all possible combinations of the AoIs in the draft is applied to the resulting story draft. The overall score for a given individual in the population is computed as the average of the scores assigned to the corresponding story draft by the metrics for each of the possible pairwise combinations of the AoIs included in it.

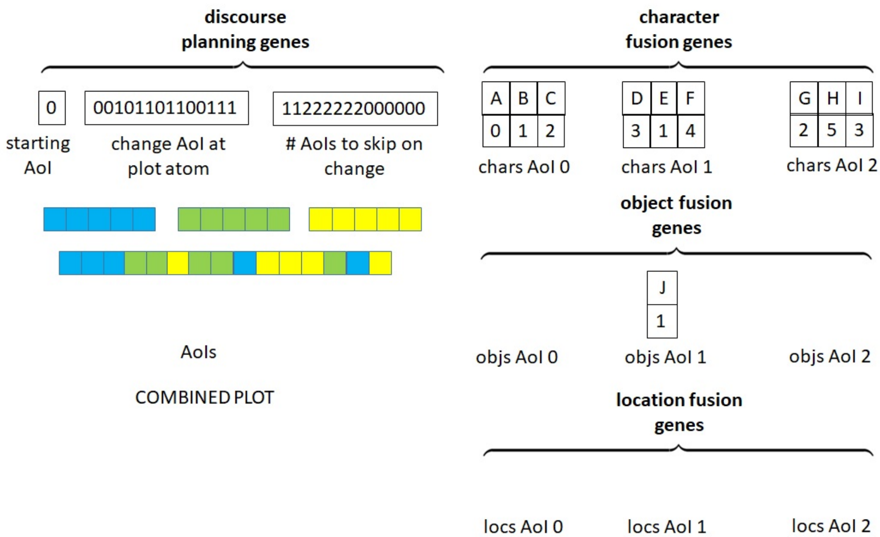

Fig. 1 Genetic representation for a combination of three AoIs of length 5, each with 3 characters, only one has an object, and none have locations. Fuses characters B (AoI0) / E (AoI1), C (AoI0) / G (AoI2) and D (AoI1) / I (AoI3).

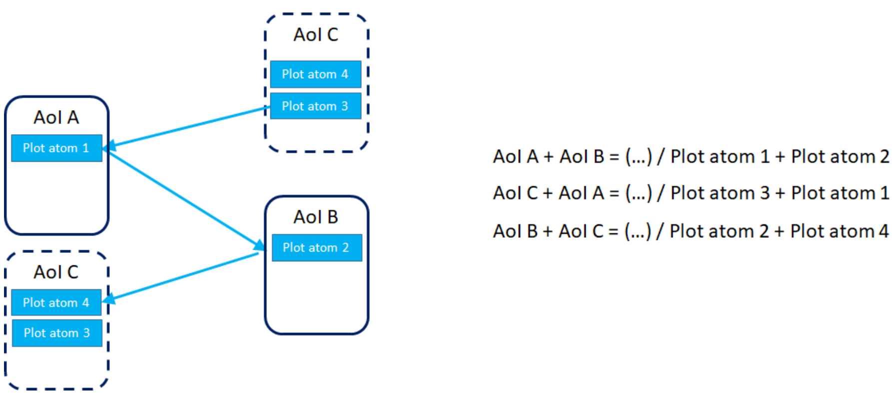

Fig. 2 Combination of three AoIs with incompatible sequencing constraints. Constraints on character fusion are ommitted from the entries shown for the metrics for clarity.

3.6 Selecting Sets of Compatible AoIs

Because the sequencing constraints for pairs of AoIs force particular positions of atoms in one AoI with respect to another, it is possible that a combination of more than two AoIs prove to be incompatible. This happens for instance if one constraint fixes the position between AoIs A and B, then another constraints forces elements from AoI C to be beyond A and a further constraint forces elements from AoI C to be before B. This type of situation is illustrated graphically in Figure 2.

HappyLove	RelentingGuardian	Task
{BoyMeetsGirl, FallInLove}	{}	{}
{}	[CoupleWantsToMarry, UnrelentingGuardian]	{}
{}	{}	{DifficultTask, Solution}
{}	{RelentingGuardian, Wedding}	{}
{HappyEverAfter}	{}	{}

Table 3 Example of data structure of constrained levels for axes of interest for HappyLove, UnrelentingGuardian and Task, showing relative levels of compatibility between their plot atoms.

This is not a big problem for the procedure we are proposing, because the fitness function will take the average of the metrics assigned to each pairing and an acceptable score will be assigned. However, in these situations the averaging procedure will result in a score with a relatively low upper threshold, which corresponds to combinations of AoIs that do not quite make sense. In order to improve the quality of the set of outputs as a whole, we will apply a preprocessing stage to filter the sets of AoIs to be used as input down to sets that are known to be reasonably compatible.

To this end, we have developed a process that constructs from a given set of AoIs a data structure that represents the relative orderings among the plot atoms in the combination as imposed by the corresponding set of constraints. This data structure is built incrementally by progressively adding the AoIs in the set, and for each addition applying the constraints for the combination of the new AoI with all the AoIs already in the structure. The structure is built of ordered lists of lists of plot atoms. Each of these lists of lists represents a *constrained level*, which contains plot atoms from the different AoIs – one constituent list for the contributions from a particular AoI – on which there is no relative ordering constraint. When the application of a new sequencing constraint establishes a relative ordering between elements from different AoIs that appear at the same constrained level, a new constrained level is built, and the corresponding contributions from the related AoIs are split into two and separated into the resulting two levels.

An example of the data structure of constrainted levels for the combination of axes of interest for HappyLove, UnrelentingGuardian and Task, as used in the example shown in Table 2 above, is shown in Table 3.

The data structure of constrained levels has the advantage that any instances of incompatibility for a particular combination of AoIs result in the plot atoms for one of the AoIs appearing in the data structure out of sequence. This is easy to spot and it allows such problematic combinations to be filtered out as potential inputs.

A set of AoIs needs to be connected for the process to make sense because otherwise the metrics will not be able to score the resulting drafts. The minimum requirement is that every AoI in the starting set be at least connected to another AoI in the set, and that all other AoIs in the set can be reached by traversing the connections from the given AoI.

The basic procedure for building a populations of drafts for a given combination of AoIs is as follows:

1. a given AoI to act as seed is provided as input, together with the number of AoIs that the combination should have,
2. a set of AoIs that are connected - at least indirectly – to the seed AoI is compiled,
3. the combination is validated in terms of compatibility – if incompatible the compilation is redone to try with a new set of connected AoIs –,
4. an evolutionary process is launched on the given combination.

3.7 Constructing an Initial Population

An initial population of story candidates is built by assigning values to the representation described in Section 3.4. For each of the different parts of the representation the process of assignment of values needs to be treated differently.

For the initial digit that defines which AoI to start on, and for the vector of decisions on whether to switch, random choice between 0 and 1 is suitable.

For the vector of decisions on skip size at each switch, random choice between 1 and N-1 (with N the total number of plots being combined) is suitable.

For the vector of decisions of which entity to assign to each variable, the choice is more complex. This is because variables from the same AoI should not be assigned to the same entity, at the risk of confusing the relations between entities in the corresponding subplot. The process of assignment is carried out separately for the set of variables of each sort for each thread. For such a set of variables, the process decides at random whether to assign to each variable either an entity name chosen at random from those of the same sort already used in some of the AoIs already processed, or an entirely new entity name chosen at random from the entity names that remain free. This ensures the required constraints are satisfied.

3.8 Evolutionary Operators

Once a population has been constructed, mutation and cross over operators are applied to it.

Because of the different nature of the various parts of the representation, specific operators of each kind are applied to the different parts.

For the mutation operators:

- for the starting point gene, the value is mutated at random
- for the switch point vector, values at a single point chosen at random are mutated
- for the skip size vector, values at a single point chosen at random are mutated to a value chosen at random within the required range
- for the entity assignment vectors, entity names at each point are either mutated or not depending on a threshold parameter, and, if required, mutated to an entity name chosen at random within the required range

For the cross over operators:

- for the starting point gene, the value of the two individuals being considered is swapped
- for the switch point vector, a point in the vector is chosen at random and the corresponding halves of the vectors for the two individuals are swapped over
- for the skip size vector, a point in the vector is chosen at random and the corresponding halves of the vectors for the two individuals are swapped over
- for the entity assignment vector, the assignments of entities for the two different individuals are swapped over (specific operators are defined for each sort of entity)

3.9 Textual Rendering for Story Drafts

The data structures on which the system relies for representing stories – as shown in the examples above – are appropriate for capturing the features that have been considered relevant, but they are not necessarily very user friendly as means of conveying the stories to readers not familiar with the formalism. To facilitate the task of human volunteers charged with providing an evaluation of the acceptability of the stories, a module has been added to the system to produce textual renderings of the resulting story drafts.

The textual rendering module performs four basic tasks: it compiles the set of constants used to refer to the entities that appear in the plot atoms of the final discourse, it assigns to each constant a proper name applicable to a person, it assigns to each plot atom in a story draft a String template that conveys the meaning of the plot atom as a natural language sentence – with place holder tokens for the constants used in the plot atom – and it replaces the place holder tokens for constants in these String templates with the corresponding proper name. The result of this process is a sequence of pseudo-sentences that provide a textual rendering of the discourse for the story draft. The sentences in this textual rendering are repetitive because they refer to all character by a proper name at all mentions, but they are easier for the untrained eye to read than the raw data structures.

4 Results and Discussion

The results of the proposed system are presented and the relation of the proposed approach with previous work is discussed.

The proposed system is run in each case with an initial population of 100 individuals generated at random, with the described operators for mutation (probability of mutation set to 0.2) and crossover (probability of cross over set to 0.05), for 100 generations. At each generation populations are culled by selecting the next generation using a best scoring criterion.

The ability of the system as described to generate acceptable stories is tested in different set ups to explore the impact of the choice of seed AoI and the number AoIs employed in the input. The system has therefore been run with combinations of 3, 4 and 5 AoIs, starting in each case from a different AoI. Because exhaustive testing over the set of 19 AoIs yields a substantial volume of outputs, the initial tests have been run over a selection of AoIs. The selected AoIs are: ABDUCTION, DONOR, RIVALRY and SHIFTINGLOVE.

These AoIs have been selected attempting to cover the different kinds of AoI that are present in the set. ABDUCTION represents a classic villainy often used to trigger traditional stories, DONOR represents the donor sequence in Propp's formalism – namely, the hero meets someone who tests him and, on a successful outcome, provides him with a magic object that will help him to achieve his goals, RIVALRY represents a different mechanism for introducing conflict in a story, and SHIFTINGLOVE introduces a specific plot elements dealing with romance – an existing love affair goes through difficult times but eventually succeeds.

Over this general set up, two types of evaluation are presented: a qualitative evaluation that analysis some examples of output, and a quantitative evaluation that compares outputs of the evolutionary solution with a randomly instantiated version of the genetic representation.

4.1 Qualitative Evaluation of Selected Examples

Some examples of results for the three different lengths of combination are shown below. An attempt has been made to use different AoIs as seed in each case. The examples constitute random samples from the potential search space in the sense that they are the result of the first successful run for each input configuration. The only exception is where a later run produced a combination too similar to those in the examples generated for different seeds used earlier, in which case a different result was generated to ensure broader coverage of the spectrum of possible stories in the selected examples.

Table 4 shows the result for a combination of three AoIs using ABDUCTION as seed AoI. The two AoIs chosen at random for combination are UNRELENTINGGUARDIAN and HAPPYLOVE. This example shows that, in spite of having tailored the metrics to capture basic compatibility constraints as found in traditional stories, the combinations produced by the system do not always match traditional expectations. In this example, the guardian opposing the union of his protegé with a suitor kidnaps the suitor, who then fights him successfully before being rescued by someone else; and this leads to the guardian relenting and allowing the proposed union.

Table 5 shows the result for a combination of four AoIs using SHIFTINGLOVE as seed AoI. The three AoIs chosen at random for combination are RIVALRY, VALIDATOR and RAGS2RICHES. It is interesting to see in this case, that, although the score (86) is not 100 %, the result is quite acceptable. In fact, some of the constraints that are not satisfied – like not having the protagonist of the SHIFTINGLOVE AoI also be the protagonist of the VALIDATOR AoI – yield interesting results – in this case, having the validation of the former lover play a role in the following reconciliation.

Discourse Structure

UnrelentingGuardian	`CoupleWantsToMarry`	(lover = **Krull**, beloved = **Alan**)
UnrelentingGuardian	`UnrelentingGuardian`	(lover = **Krull**, beloved = **Alan**, guardian = **West**)
Abduction	`Abduction`	(abducted = **Krull**, abductor = **West**)
Conflict	`Struggle`	(attacker = **Krull**, defender = **West**)
Conflict	`Victory`	(winner = **Krull**, looser = **West**)
Abduction	`Rescue`	(abducted = **Krull**, rescuer = **Mina**, abductor = **West**)
UnrelentingGuardian	`GuardianRelents`	(lover = **Krull**, beloved = **Alan**, guardian = **West**)
UnrelentingGuardian	`Wedding`	(lover = **Krull**, beloved = **Alan**)
HappyLove	`HappyEverAfter`	(lover = **Krull**, beloved = **Alan**)

Fitness Scoring

	Abduction+RelentingGuardian		
Role-sharing	Abduction victim is also RelentingGuardian hero	100 %	50
	Abduction villain is also RelentingGuardian obstacle	100 %	
Sequencing	UnrelentingGuardian precedes Abduction	100 %	50
	Rescue precedes RelentingGuardian	100 %	
	Abduction+Conflict		
Role-sharing	Abduction villain is also Conflict villain	100 %	50
Sequencing	Abduction precedes Struggle	100 %	50
	Victory precedes Rescue	100 %	
	Conflict+RelentingGuardian		
Role-sharing	Conflict hero is also RelentingGuardian hero	100 %	50
Sequencing	UnrelentingGuardian precedes Struggle	100 %	50
	Victory precedes RelentingGuardian	100 %	

Textual Rendering
Krull wants to marry Alan
Krull finds proposed union with Alan opposed by guardian West
West kidnaps Krull
Krull fights with West
Krull achieves victory over West
Mina rescues Krull from West
Krull vanquishes the opposition of West to the proposed union with Alan
Krull marries Alan

Table 4 Example of story draft for a basic plot combining three axes of interest, using Abduction as input seed and adding UnrelentingGuardian and Conflict as random – connected – extensions. The top part of the table shows the structure of the discourse following the conventions used in Table 2. The middle part of the table shows the scores assigned by the metrics to each pair of AoIs in the combination. The bottom part of the table presents the textual rendering for the story draft.

Table 6 shows the result for a combination of four AoIs using CrossDressing as seed AoI. The four AoIs chosen as random – connected – extensions are Rags2Riches, Task, Abduction and ShiftingLove. In this case the score is even lower (76) and the result is still acceptable. Transgressions of the expected combination patterns include: the protagonist of the story, Lilly, is the person who sets the task rather than the person trying to solve it – the solving of the task becomes the context in which Lilly's adventures take place –, the protagnist's partner shifts their romantic interest to the person that Lilly has charged with solving the task, the person in charge of solving the task commits a villainy – a kidnapping –, Lilly disguises herself as a man to rescue the victim, she also achieves her aims, the task gets solved and Lilly recovers her lover. It is interesting to note that the increase in the number of AoIs involved in the combination increases very significantly the number of constraints that need to be considered. This in its turn leads to a lower overall score, as it becomes more difficult for all the constraints to be satisfied at the same time. However, the satisfaction of those constraints that do hold contributes to the overall appearance of coherence of the final story.

Discourse Structure

ShiftingLove	`GirlMeetsOtherBoy`	(previous-love= **Alan**, girl = **Benson**, boy = **Hans**)
Rivalry	`Rivalry`	(rival1 = **Alan**, rival2 = **Benson**)
ShiftingLove	`LoveShift`	(lover = **Benson**, beloved = **Hans**, rival = **Alan**)
Rags2Riches	`Poverty`	(sufferer = **Alan**)
Rags2Riches	`Aspiration`	(aspirer = **Alan**)
Rags2Riches	`Transformation`	(transformed = **Alan**)
Validator	`Tested`	(tested = **Alan**, tester = **Benson**)
Rivalry	`Cooperation`	(rival1 = **Alan**, rival2 = **Benson**)
Validator	`Character'sReaction`	(tested = **Alan**, tester = **Benson**)
Rivalry	`RivalReconciliation`	(rival1 = **Alan**, rival2 = **love-interest**)
Validator	`Validation`	(validated = **Alan**, validator = **Benson**)
Validator	`ValidationRecognised`	(validated = **Alan**)
Rags2Riches	`AspirationAchieved`	(achiever = **Alan**)
ShiftingLove	`Reconciliation`	(lover = **Benson**, beloved = **Alan**)

Fitness Scoring

Rivalry+ShiftingLove			
Role-sharing	Rivalry shadow is also ShiftingLove love-interest	100 %	50
	Rivalry hero is also ShiftingLove hero	0 %	
Sequencing	Rivalry precedes LoveShift	100 %	50
	RivalReconciliation precedes Reconciliation	100 %	
Rags2Riches+ShiftingLove			
Role-sharing	Rags2Riches hero is also ShiftingLove former	100 %	50
Sequencing	Aspiration precedes GirlMeetsOtherBoy	71 %	50
	LoveShift precedes Transformation	100 %	
	AspirationAchieved precedes Reconciliation	100 %	
Rags2Riches+Validator			
Role-sharing	Rags2Riches hero is also Validator hero	100 %	50
Sequencing	ValidationRecognised precedes AspirationAchieved	100 %	
ShiftingLove+Validator			
Role-sharing	ShiftingLove hero is also Validator hero	0 %	
Sequencing	LoveShift precedes Tested	100 %	50
	ValidationRecognised precedes Reconciliation	100 %	
Rags2Riches+Rivalry			
Role-sharing	Rags2Riches hero is also Rivalry hero	100 %	50
Sequencing	Transformation precedes Cooperation	100 %	
Rivalry+Validator			
Role-sharing	Rivalry hero is also Validator hero	100 %	50
Sequencing	Cooperation precedes Character'sReaction	100 %	

Textual Rendering
Benson who loved Alan meets different person Hans
Alan develops rivalry with Benson
Benson no longer cares for Alan and now loves Hans
Alan suffers poverty
Alan has aspiration
Alan is transformed
Alan is tested by Benson
Alan cooperates with Benson
Alan reacts to the test by Benson
Alan ends rivalry with Benson
Alan is validated by Benson
Alan sees validation recognised
Alan fulfills their aspiration
Benson makes up with Alan

Table 5 Example of story draft for a basic plot combining four axes of interest, using ShiftingLove as input seed and adding Rivalry, Validator and Rags2Riches as random – connected – extensions. The top part of the table shows the structure of the discourse following the conventions used in Table 2. The middle part of the table shows the scores assigned by the metrics to each pair of AoIs in the combination. The bottom part of the table presents the textual rendering for the story draft.

4.2 Quantitative Comparative Evaluation Informed by Human Judgements

To obtain a quantitative measure of the relative quality of the story drafts generated by the proposed evolutionary solution, we carried out a comparative evaluation between the results obtained by the application of the proposed metrics in the evolutionary process and results of a baseline procedure that did not take the proposed metrics into account. The baseline procedure employed relies on the process of random instantiation of the genetic representation used to build the initial populations for the evolutionary procedures. Because that process does not consider the proposed metrics at any point, any observed

Discourse Structure

TASK	DifficultTask	(setter= **Lilly**, solver = **Sarah**)
RAGS2RICHES	Poverty	(sufferer = **Lilly**)
SHIFTINGLOVE	GirlMeetsOtherBoy	(previous-love= **Lilly**, girl = **Ernest**, boy = **Sarah**)
RAGS2RICHES	Aspiration	(aspirer = **Lilly**)
SHIFTINGLOVE	LoveShift	(lover = **Ernest**, beloved = **Sarah**, rival = **Lilly**)
RAGS2RICHES	Transformation	(transformed = **Lilly**)
ABDUCTION	Abduction	(abducted = **Hugo**, abductor = **Sarah**)
CROSSDRESSING	CrossDressing	(cross-dresser = **Lilly**)
ABDUCTION	Rescue	(abducted = **Hugo**, rescuer = **Lilly**, abductor = **Sarah**)
CROSSDRESSING	Recognition	(recognised = **Lilly**)
RAGS2RICHES	AspirationAchieved	(achiever = **Lilly**)
TASK	Solution	(solver = **Sarah**)
SHIFTINGLOVE	Reconciliation	(lover = **Ernest**, beloved = **Lilly**)

Fitness Scoring

ABDUCTION+RAGS2RICHES			
Role-sharing	Abduction hero is also Rags2Riches hero	100 %	50
Sequencing	Transformation precedes Rescue	100 %	
CROSSDRESSING+SHIFTINGLOVE			
Role-sharing	CrossDressing someone is also ShiftingLove hero	100 %	50
Sequencing	LoveShift precedes CrossDressing	100 %	
	Recognition precedes Reconciliation	100 %	
RAGS2RICHES+SHIFTINGLOVE			
Role-sharing	Rags2Riches hero is also ShiftingLove former	100 %	50
Sequencing	Aspiration precedes GirlMeetsOtherBoy	92 %	50
	LoveShift precedes Transformation	100 %	
	AspirationAchieved precedes Reconciliation	100 %	
ABDUCTION+SHIFTINGLOVE			
Role-sharing	Abduction victim is also ShiftingLove hero	0 %	50
	Abduction villain is also ShiftingLove shadow	0 %	
Sequencing	Abduction precedes LoveShift	84 %	50
	Rescue precedes Reconciliation	100 %	
CROSSDRESSING+RAGS2RICHES			
Role-sharing	CrossDressing someone is also Rags2Riches hero	100 %	50
Sequencing	Aspiration precedes CrossDressing	100 %	
	CrossDressing precedes Transformation	84 %	
	Transformation precedes Recognition	100 %	
	Recognition precedes AspirationAchieved	100 %	
ABDUCTION+CROSSDRESSING			
Role-sharing	Abduction hero is also CrossDressing someone	100 %	
Sequencing	Abduction precedes CrossDressing	84 %	50
	Rescue precedes Recognition	100 %	
ABDUCTION+TASK			
Role-sharing	Abduction villain is also Task hero	100 %	
Sequencing	DifficultTask precedes Abduction	84 %	50
SHIFTINGLOVE+TASK			
Role-sharing	ShiftingLove hero is also Task hero	0 %	
Sequencing	LoveShift precedes DifficultTask	69 %	50
	Solution precedes Reconciliation	100 %	
RAGS2RICHES+TASK			
Role-sharing	Rags2Riches hero is also Task hero	0 %	50
Sequencing	Transformation precedes Solution	100 %	

Textual Rendering

Sarah is set a difficult task by Lilly
Lilly suffers poverty
Ernest who loved Lilly meets different person Sarah
Lilly has aspiration
Ernest no longer cares for Lilly and now loves Sarah
Lilly is transformed
Sarah kidnaps Hugo
Lilly dresses up as a member of the opposite sex
Lilly rescues Hugo from Sarah
Lilly is recognised
Lilly fulfills their aspiration
Sarah solves the task
Ernest makes up with Lilly

Table 6 Example of story draft for a basic plot combining five axes of interest, using CROSSDRESSING as input seed and adding RAGS2RICHES, TASK, ABDUCTION and SHIFTINGLOVE as random – connected – extensions. The top part of the table shows the structure of the discourse following the conventions used in Table 2. The middle part of the table shows the scores assigned by the metrics to each pair of AoIs in the combination. The bottom part of the table presents the textual rendering for the story draft.

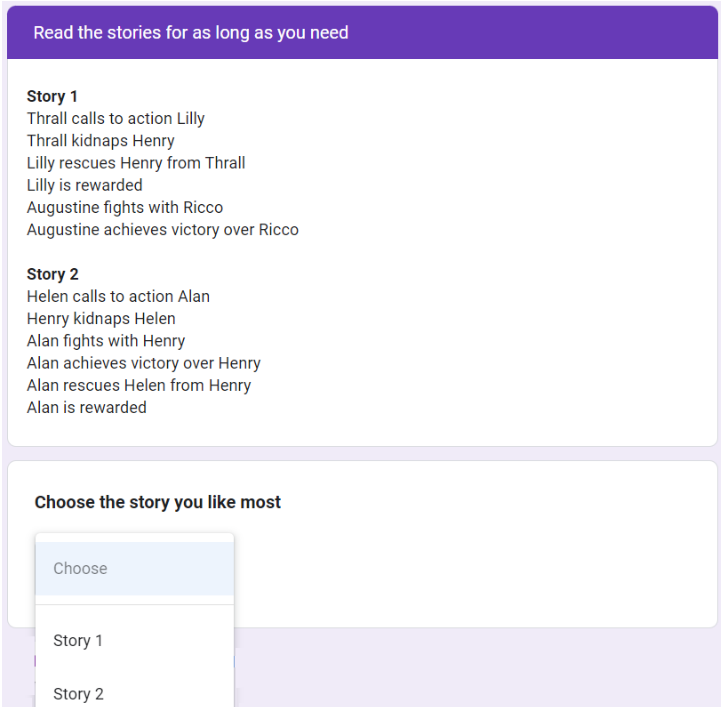

Fig. 3 Sample evaluation screen where the user was presented two plots to select the one with the highest perceived quality

improvements in quality between the baseline and the outputs of the system should be considered an indication of the added value that the metrics used as fitness functions provide.

To discern between the two competing approaches in terms of perceived quality of the stories we rely on a set of human volunteers that were asked to consider pairs of stories and select one of them as more acceptable than the other. Each pair contained – for the same combination of axes of interest – a story draft produced by the evolutionary procedure and a story draft produced by the baseline random instantiation procedure. The pairs were presented in random order to avoid biases arising from the presentation order (see Figure 3).

A set of 10 human volunteers participated in the evaluation, including 7 men and 3 women, with ages ranging from 20 to 60 years old. The level of expertise ranged from Novice to Expert, with 2 considered experts in the field, 3 considered competent, 3 with limited experience and 2 considered novices without any previous experience in narrative generation.

A set of 36 pairs of plots were generated for the evaluation, each pair consisting of the combination of 3, 4 or 5 AoIs as shown in Table 7. The resulting set was divided into 3 subsets with 12 pairs each, and each of them was evaluated by four evaluators, giving rise to 144 evaluations. For each combination,

AoI	Set 1	Set 2	Set 3
3-Abduction	AB-CA-CO-bl	AB-CA-IV-bl	AB-VP-HL-bl
	AB-CA-CO-ev-99-0	AB-CA-IV-ev-83-0	AB-VP-HL-ev-83-0
3-Donor	DO-IV-VP-bl	DO-SL-IV-bl	DO-TA-RE-bl
	DO-IV-VP-ev-75-0	DO-SL-IV-ev-75-0	DO-TA-RE-ev-100-0
3-Rivalry	RI-RR-IV-ev-100-0	Ri-RR-UG-ev-100-0	RI-TA-PU-ev-100-0
	RI-RR-IV-bl	Ri-RR-UG-bl	RI-TA-PU-bl
3-ShiftingLove	SL-RR-RI-bl	SL-UG-JO-bl	SL-UG-RI-bl
	SL-RR-RI-ev-91-0	SL-UG-JO-ev-66-0	SL-UG-RI-ev-75-1
4-Abduction	AB-JO-PU-RG-ev-97-1	AB-RG-RR-TA-ev-83-0	AB-TA-SL-IV-ev-69-1
	AB-JO-PU-RG-bl	AB-RG-RR-TA-bl	AB-TA-SL-IV-bl
4-Donor	DO-CA-CO-RG-ev-99-1	DO-HL-RI-CD-ev-75-0	DO-IV-CA-CO-ev-99-0
	DO-CA-CO-RG-bl	DO-HL-RI-CD-bl	DO-IV-CA-CO-bl
4-Rivalry	RI-AB-VP-DO-ev-87-0	RI-PU-RG-CO-ev-98-0	RI-RG-JO-CA-ev-100-1
	RI-AB-VP-DO-bl	RI-PU-RG-CO-bl	RI-RG-JO-CA-bl
4-ShiftingLove	SL-AB-DO-CD-bl	SL-HL-CD-RR-bl	SL-VA-CA-RR-bl
	SL-AB-DO-CD-ev-66-0	SL-HL-CD-RR-ev-76-1	SL-VA-CA-RR-ev-82-0
5-Abduction	AB-CA-CO-SL-VA-bl	AB-CD-JO-RI-VI-bl	AB-RG-RR-PU-CD-bl
	AB-CA-CO-SL-VA-ev-75	AB-CD-JO-RI-VI-ev-78	AB-RG-RR-PU-CD-ev-81
5-Donor	DO-CO-VA-SL-HL-ev-78	DO-RG-RR-IV-JO-ev-93	DO-TA-VA-AB-CA-ev-82
	DO-CO-VA-SL-HL-bl	DO-RG-RR-IV-JO-bl	DO-TA-VA-AB-CA-bl
5-Rivalry	RI-CO-RR-SL-CA-bl	RI-HL-SL-UG-VP-bl	RI-VA-VP-PU-IV-bl
	RI-CO-RR-SL-CA-ev-77-0	RI-HL-SL-UG-VP-ev-73-2	RI-VA-VP-PU-IV-ev-82-0
5-ShiftingLove	SL-CA-DO-AB-JO-ev-69-0	SL-CA-DO-RI-VP-ev-64-2	SL-CO-JO-VA-AB-ev-70-0
	SL-CA-DO-AB-JO-bl	SL-CA-DO-RI-VP-bl	SL-CO-JO-VA-AB-bl

Table 7 Configuration of the evaluation sets used for the evaluation by human judges

Label	AoI name	Label	AoI name	Label	AoI name
IV	InterdictionViolated	CO	Conflict	HL	HappyLove
AB	Abduction	TA	Task	SL	ShiftingLove
VP	VillainyPunishment	PU	Pursuit	UG	UnrelentingGuardian
CA	Call2ActionReward	DO	Donor	RE	Repentance
VA	Validation	RI	Rivalry	RR	Rags2Riches
JO	Journey	CD	CrossDressing		

Table 8 Key for the two-letter labels for the AoIs used in the examples mentioned in other tables.

the name that appears in the tables includes short labels for each of the AoIs involved, according to the key given in Table 8.

The results of this quantitative evaluation are presented grouped by the number of AoIs combined in each case, to allow consideration of the differences in score results arising from the increase in the number of constraints as the number of AoIs rises (see Tables 9 and 11) as well as by evaluator and evaluation set to show possible differences in the distribution of the plot combinations in each evaluation set (see Table 10).

Table 9 shows the decisions made by the evaluators for each pair of plots in each of the 3 evaluation subsets. The colors in the evaluator columns (EV1 - EV4) show the level of expertise of each evaluator: green for expert, yellow for competent, orange for limited experience and red for novice.

The numeric results in Table 10 point out that there is no relationship between the level of expertise and the number of times the evaluators preferred the evolutionary version of the plot over the randomly generated one, as it might have been expected. For example, Evaluator 3 in evaluation set 2 has preferred the evolutionary version of the plots only 50% of the times (6 out of 12), while the other expert evaluators have chosen the evolutionary version almost in all cases. In contrast, the novice evaluators (Ev2 in evaluation set 2 and Ev4 in evaluation set 3) have consistently chosen the evolutionary versions

Evaluation Set 1	Expected	Ev1	Ev2	Ev3	Ev4
AB-CA-CO-bl AB-CA-CO-ev-99-0	Story 2	Story 2	Story 2	Story 2	Story 1
DO-IV-VP-bl DO-IV-VP-ev-75-0	Story 2	Story 2	Story 1	Story 2	Story 2
RI-RR-IV-ev-100-0 RI-RR-IV-bl	Story 1	Story 1	Story 1	Story 1	Story 2
SL-RR-RI-bl SL-RR-RI-ev-91-0	Story 2	Story 2	Story 2	Story 2	Story 2
AB-JO-PU-RG-ev-97-1 AB-JO-PU-RG-bl	Story 1	Story 1	Story 2	Story 1	Story 2
DO-CA-CO-RG-ev-99-1 DO-CA-CO-RG-bl	Story 1	Story 1	Story 1	Story 1	Story 1
RI-AB-VP-DO-ev-87-0 RI-AB-VP-DO-bl	Story 1	Story 1	Story 1	Story 1	Story 2
SL-AB-DO-CD-bl SL-AB-DO-CD-ev-66-0	Story 2	Story 1	Story 1	Story 2	Story 2
AB-CA-CO-SL-VA-bl AB-CA-CO-SL-VA-ev-75	Story 2	Story 1	Story 1	Story 2	Story 1
DO-CO-VA-SL-HL-ev-78 DO-CO-VA-SL-HL-bl	Story 1	Story 1	Story 2	Story 1	Story 2
RI-CO-RR-SL-CA-bl RI-CO-RR-SL-CA-ev-77-0	Story 2	Story 2	Story 2	Story 1	Story 1
SL-CA-DO-AB-JO-ev-69-0 SL-CA-DO-AB-JO-bl	Story 1	Story 1	Story 1	Story 1	Story 1

Evaluation Set 2	Expected	Ev1	Ev2	Ev3	Ev4
AB-CA-IV-bl AB-CA-IV-ev-83-0	Story 2	Story 2	Story 1	Story 1	Story 2
DO-SL-IV-bl DO-SL-IV-ev-75-0	Story 2	Story 2	Story 2	Story 1	Story 2
RI-RR-UG-ev-100-0 Ri-RR-UG-bl	Story 1	Story 2	Story 1	Story 1	Story 1
SL-UG-JO-bl SL-UG-JO-ev-66-0	Story 2	Story 1	Story 2	Story 1	Story 2
AB-RG-RR-TA-ev-83-0 AB-RG-RR-TA-bl	Story 1	Story 1	Story 1	Story 2	Story 1
DO-HL-RI-CD-ev-75-0 DO-HL-RI-CD-bl	Story 1	Story 2	Story 1	Story 2	Story 1
RI-PU-RG-CO-ev-98-0 RI-PU-RG-CO-bl	Story 1	Story 1	Story 1	Story 1	Story 1
SL-HL-CD-RR-bl SL-HL-CD-RR-ev-76-1	Story 2	Story 2	Story 1	Story 2	Story 2
AB-CD-JO-RI-VP-bl AB-CD-JO-RI-VP-ev-78	Story 2	Story 2	Story 2	Story 2	Story 2
DO-RG-RR-IV-JO-ev-93 DO-RG-RR-IV-JO-bl	Story 1	Story 1	Story 1	Story 1	Story 1
RI-HL-SL-UG-VP-bl RI-HL-SL-UG-VP-ev-73-2	Story 2	Story 1	Story 2	Story 2	Story 1
SL-CA-DO-RI-VP-ev-64-2 SL-CA-DO-RI-VP-bl	Story 1	Story 1	Story 1	Story 2	Story 1

Evaluation Set 3	Expected	Ev1	Ev2	Ev3	Ev4
AB-VP-HL-bl AB-VP-HL-ev-83-0	Story 2	Story 1	Story 2	Story 2	Story 2
DO-TA-RE-bl DO-TA-RE-ev-100-0	Story 2	Story 2	Story 2	Story 1	Story 1
RI-TA-PU-ev-100-0 RI-TA-PU-bl	Story 1	Story 1	Story 1	Story 1	Story 1
SL-UG-RI-bl SL-UG-RI-ev-75-1	Story 2	Story 1	Story 1	Story 2	Story 1
AB-TA-SL-IV-ev-69-1 AB-TA-SL-IV-bl	Story 1	Story 2	Story 1	Story 1	Story 1
DO-IV-CA-CO-ev-99-0 DO-IV-CA-CO-bl	Story 1	Story 2	Story 1	Story 1	Story 1
RI-RG-JO-CA-ev-100-1 RI-RG-JO-CA-bl	Story 1	Story 2	Story 1	Story 1	Story 2
SL-VA-CA-RR-bl SL-VA-CA-RR-ev-82-0	Story 2	Story 1	Story 2	Story 2	Story 2
AB-RG-RR-PU-CD-bl AB-RG-RR-PU-CD-ev-81	Story 2	Story 2	Story 2	Story 2	Story 2
DO-TA-VA-AB-CA-ev-82 DO-TA-VA-AB-CA-bl	Story 1	Story 1	Story 1	Story 1	Story 1
RI-VA-VP-PU-IV-bl RI-VA-VP-PU-IV-ev-82-0	Story 2	Story 1	Story 2	Story 2	Story 2
SL-CO-JO-VA-AB-ev-70-0 SL-CO-JO-VA-AB-bl	Story 1	Story 1	Story 1	Story 2	Story 1

Table 9 Results of the human judgments on the comparison between story drafts produced by the evolutionary procedure and story drafts produced by random instantiation

most of the times (83.33% – 10 out of 12 – in the first case, 75% – 9 out of 12 – in the second).

Results Evaluation Set 1				Results Evaluation Set 2				Results Evaluation Set 3			
Ev1	Ev2	Ev3	Ev4	Ev1	Ev2	Ev3	Ev4	Ev1	Ev2	Ev3	Ev4
10	7	11	5	8	10	6	11	5	11	10	9
83,33	58,33	91,67	41,67	66,67	83,33	50,00	91,67	41,67	91,67	83,33	75,00
68,75				72,92				72,92			

Table 10 Quantitative results of the human judgments on the comparison between story drafts produced by the evolutionary procedure and story drafts produced by random instantiation, per evaluator and evaluation set

Table 10 also shows that there was no significant difference in the composition of the three evaluation sets, as the evaluators of the first set chose the evolutionary versions of the stories 68.75% of the times, whereas the percentage in the other two subsets was slightly higher: 72.92%. While there are some stories in each subset that had a unanimous response by all evaluators (e.g. rows 4, 6 and 12 in evaluation set 1, where all the evaluators chose the evolutionary version as more acceptable than the baseline), or almost unanimous (e.g. first three rows in evaluation set 1), evaluation set 1 also shows that other combinations were not so clearly preferable in the evolutionary version (e.g. rows 5, 8, 10 and 11) or were definitely worse (e.g. row 9, with 3 evaluators chosing the baseline version over the evolutionary one). This explains the slight difference in the results of the three subsets, as in the first subset there are more combinations of the last two cases than in the other two subsets.

AoI	#	%	%%	%%%
3-Abduction	8	66,67	68,75	71,53
3-Donor	8	66,67		
3-Rivalry	10	83,33		
3-ShiftingLove	7	58,33		
4-Abduction	8	66,67	70,83	
4-Donor	9	75,00		
4-Rivalry	9	75,00		
4-ShiftingLove	8	66,67		
5-Abduction	9	75,00	75,00	
5-Donor	10	83,33		
5-Rivalry	7	58,33		
5-ShiftingLove	10	83,33		

Table 11 Quantitative results of the human judgments on the comparison between story drafts produced by the evolutionary procedure and story drafts produced by random instantiation, per number of AoI combinations

As for the results shown in Table 11, two interesting outcomes can be highlighted. The first one is that, counting the number of times the evolutionary version was preferred over the baseline (column 2) for each combination of 3, 4 and 5 AoIs, we can see that the percentage of positive responses increases with the number of AoIs that must be combined (68.75% for 3 AoIs, 70.83% for 4 AoIs and 75% for 5 AoIs). This means that, as the stories gain complexity, it is more difficult to generate meaningful stories randomly, so the evolutionary versions are favoured over the baselines. The second outcome is that, out of the 144 evaluated pairs of plots, 71.53% of the times (i.e. 103 out of 144), the evolutionary versions were considered to have better quality than the baselines. This means that the proposed method to combine subplots generates high quality plots that improve the results provided by the baseline method. Although there is still a wide margin for improvement, the results prove that the proposed method can be successfully used to generate rich, complex stories as the result of combining simpler plots that can be subsequently be generated using other, well established methods (Gervás, 2017).

From the point of view of how these quantitative results can be interpreted in terms of the specific details of the proposed solution, there are two aspects worth discussing. The differences between the evaluation sets may be explained in terms of the interaction between two different factors: the likelihood that the random baseline sometimes produce acceptable results, and the possibility that specific combinations of AoIs are ill-suited for being combined together unless specific metrics are added to consider romantic affinities between characters.

Because the method used as baseline is based on random assignment of genetic information, there is a non-zero chance that it lead to acceptable story drafts. The likelihood of this happening is higher for combinations of a small number of AoIs, where the search space in question is smaller. As the number of AoIs involved increases, the size of the corresponding search space increases exponentially so the likelihood of acceptable results being produced by the random procedure is significantly reduced. This explains the results shown in Table 11, where evaluator preference for the evolutionary versions rises with size of the combinations. The same phenomenon also increases the likelihood that the baseline procedure sometimes produce results that compete in quality with those of the evolutionary approach. This may explain some of the irregularities observed in the results in Table 10. This second consequence may be compounded where it interacts with an observed shortcoming of the solution as it stands, involving conflicts between certain types of AoIs that are not captured by the current set of metrics.

A close examination of the specific results produced shows that there are cases where the chosen combination of AoIs – which are selected at random except for the described filter on combinations that imply temporal inconsistencies – suffer from conflicts at a different level. This happens for instance when two AoIs that involve romantic relationships are combined together in the same story. Each of the AoIs in such a case will postulate specific relations of affinity – or lack thereof – between the characters. The current set of metrics does not include constraints on consistency in affinities between the characters through the story. This is because affinities between the characters are not explicitly modelled in the chosen representation. For this reason, story drafts produced by the evolutionary solution in such cases are likely to include inconsistencies in affinities between characters, such as for instance, having character A make up with character B as resolution of a ShiftingLove AoI but then marry character C as resolution to a RelentingGuardian AoI that has been combined with it. When the outcomes of the evolutionary procedure suffer from this problem, there is a much higher chance that evaluators prefer the outcome of the random baseline. This problem affects only combinations with more than one AoI that involves romantic relationships, so it does not affect combinations without a love interest subplot, or any that have a single AoI involved in the love interest subplot.

To address this particular problem, a future extension of the proposed solution should include explicit consideration of affinities between characters as an additional feature to consider in the metrics that drive the fitness function.

4.3 Relation with Previous Work

The representation of plot in terms of axes of interest had been used before to generate story plot drafts (Gervás, 2019). The procedure employed for combining axes of interest in that instance exhaustively generated all combinations deemed to be valid in terms of whether they matched the probabilities of character continuity across scenes as obtained from a prior corpus. This basically means that two scenes – or plot atoms – are placed contiguously in a candidate story if some character can be found that appears in each of these scenes playing a pair of roles that has been observed before in the corpus. This criterion ensures local consistency, but it has some potential shortcomings. First, it does not take into consideration long ranging connections across non-contiguous scenes. Second, where more than one character from scene A carries on to scene B, a probability-based criterion may validate both links based on different prior stories, but it will not be able to consider the importance of both links occurring together. The new evolutionary solution, by relying on a fitness function based on metrics built heuristically to capture common sense connections across AoIs, improves upon the original on both of these aspects.

The character fusion operation considered here is comparable to binding between characters as used by Fay (2014). In Fay's work, the units being combined are character threads – which tend to gather together all the events in a story in which a given character participates. The procedure proposed by Fay therefore uses fusion –usually involving secondary characters – to combine together threads for different characters into a more elaborate story. The procedure proposed here differs from that approach in two different senses. First, in that the units being combined here are intended to be patterns that focus on plot-relevant connections across elements. This makes it less likely that elements that are not entirely relevant to a particular plot pattern end up included in a story draft simply by virtue of appearing in an existing character thread for a previous story. Second, because the use of the metrics proposed here increases the probability that the bindings established between characters play a relevant role in the narrative structure of the resulting story draft.

With respect to prior approaches that consider evolutionary solutions for story generation, the proposed solution shares characteristics with some of them, but it can also be improved by enrichment with additional features considered in some of them. The use of metrics designed to ensure story consistency is a characteristic shared with the work of McIntyre and Lapata (2010), and it may be comparable to the use of the knowledge-based heuristics of the MEXICA knowledge-based story generator as fitness function as used by Gómez de Silva Garza and Pérez y Pérez (2014). Further features that may be considered to improve the quality of system outputs are: some measure of story interest as used by McIntyre and Lapata (2010), measures of story novelty as used by Fredericks and DeVries (2021), measures of whether the stories satisfy user established goals – as used by Kartal et al (2014) – or a specific curve to describe tensions in the story – as used by de Lima et al (2019). Regardless of these potential extensions, the proposed solution captures typical narrative structures by virtue of the choice of representation units and fitness function metrics.

5 Conclusions

The evolutionary approach to constructing plot outlines for stories by combining axes of interest based on metrics for common sense connections between them provides efficient means for building a population of drafts that satisfy constraints on semantic validity over the final linear discourse for the story. Due to the progressive nature of the metrics used as fitness function the population converges reasonably quickly for a low number of constituent axes of interest. It remains to be seen whether the solution will scale well towards higher numbers of constituents.

Following the approach taken in (Gervás et al, 2022) we assume that the relative chronological order of the scenes in each AoI is respected in the final discourse. Consideration of cases of altered chronology (flashbacks, flashforwards) are left to be addressed in further work. The potential extensions to improve the set of metrics to cover additional features listed in Section 4.3 will also be considered. Explicit consideration of affinities between characters as an additional aspect relevant to the perceived quality of stories should also be attempted. Finally, the procedure described here to generated single plot stories could be combined with a procedure for generating single plotlines into multi-plotline stories such as the one proposed by Concepción et al (2020).

Supplementary information. Not applicable

Acknowledgments. This paper has been partially funded by the project CANTOR: Automated Composition of Personal Narratives as an aid for Occupational Therapy based on Reminiscence, Grant. No. PID2019-108927RB-I00 (Spanish Ministry of Science and Innovation).

Declarations

Ethical Approval. Not applicable.

Competing interests. The authors are not aware of any competing interests.

Authors' contributions. All authors contributed to the study conception and design. Pablo Gervás wrote the code for the proposed solution. Gonzalo Méndez designed and managed the evaluation and processed the results. Eugenio Concepción and Pablo Gervás carried out the knowledge engineering of the required resources. The first draft of the manuscript was written by Pablo Gervás and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding. This paper has been partially funded by the project CANTOR: Automated Composition of Personal Narratives as an aid for Occupational Therapy based on Reminiscence, Grant. No. PID2019-108927RB-I00 (Spanish Ministry of Science and Innovation).

Availability of data and materials. The data are at the present time not publicly available.

References

Booker C (2004) The Seven Basic Plots: Why We Tell Stories. The Seven Basic Plots: Why We Tell Stories, Continuum, URL http://books.google.pt/books?id=qHJj9gOl0j8C

Brehob E (2013) Online academics: The wiki tv tropes as a community of pseudo-academic producers. PhD thesis, University of Michigan

Campbell J, Cousineau P, Brown S (1990) The Hero's Journey: Joseph Campbell on His Life and Work. Collected works of Joseph Campbell, New World Library, URL https://books.google.co.uk/books?id=0LIxpikJraoC

Concepción E, Gervás P, Méndez G (2020) Exploring baselines for combining full plots into multiple-plot stories. New Generation Computing pp 1–41. https://doi.org/10.1007/s00354-020-00115-x, URL https://link.springer.com/article/10.1007%2Fs00354-020-00115-x

Cook W (2011) Plotto: The Master Book of All Plots. Tin House Books, URL https://books.google.es/books?id=HDOPLT0Wer8C

Fay MP (2014) Driving story generation with learnable character models. PhD thesis, Massachusetts Institute of Technology

Fredericks EM, DeVries B (2021) (genetically) improving novelty in procedural story generation. arXiv preprint arXiv:210306935

Gervás P (2017) Comparative evaluation of elementary plot generation procedures. In: 6th International Workshop on Computational Creativity, Concept Invention, and General Intelligence, Madrid, Spain

Gervás P (2019) Generating a search space of acceptable narrative plots. In: 10th International Conference on Computational Creativity (ICCC 2019), UNC Charlotte, North Carolina, USA

Gervás P, Concepción E, Méndez G (2021) Assessing multiplot stories: from formative analysis to computational metrics. In: International Conference on Computational Creativity, Ciudad de México, México

Gervás P, Concepción E, Méndez G (2022) Evolutionary construction of stories that combine several plot lines. In: Computational Intelligence in Music, Sound, Art and Design – 11th International Conference, EvoMUSART 2022, Springer. Springer, Madrid, Spain, https://doi.org/https://doi.org/10.1007/978-3-031-03789-4_5

Kartal B, Koenig J, Guy SJ (2014) User-driven narrative variation in large story domains using monte carlo tree search. In: Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, AAMAS '14, p 69–76

Lehman J, Stanley KO (2004) Exploiting open-endedness to solve problems through the search for novelty. In: Proceedings of the Eleventh International Conference on Artificial Life. MIT Press

León C, Gervás P, Delatorre P, et al (2020) Quantitative characteristics of human-written short stories as a metric for automated storytelling. New Generation Computing 38

de Lima ES, Feijó B, Furtado AL (2019) Procedural generation of quests for games using genetic algorithms and automated planning. In: 18th Brazilian Symposium on Computer Games and Digital Entertainment, SBGames 2019, Rio de Janeiro, Brazil, October 28-31, 2019. IEEE, pp 144–153, https://doi.org/10.1109/SBGames.2019.00028, URL https://doi.org/10.1109/SBGames.2019.00028

McIntyre N, Lapata M (2010) Plot induction and evolutionary search for story generation. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Uppsala, Sweden, pp 1562–1572, URL https://aclanthology.org/P10-1158

Pérez y Pérez R, Sharples M (2001) Mexica: A computer model of a cognitive account of creative writing. Journal of Experimental & Theoretical Artificial Intelligence 13(2):119–139

Polti G, Ray L (1916) The Thirty-six Dramatic Situations. Editor Company, URL https://books.google.es/books?id=AeB_yAEACAAJ

Porteous J, Charles F, Cavazza M (2016) Plan-based narrative generation with coordinated subplots. In: European Conference on Artificial Intelligence (ECAI 2016), IOS Press, pp 846–854

Propp V (1968) Morphology of the Folktale. University of Texas Press, Austin

Gómez de Silva Garza A (2000) An evolutionary approach to design case adaptation. PhD thesis, The University of Sydney, Australia

Gómez de Silva Garza A, Pérez y Pérez R (2014) Towards evolutionary story generation. In: Colton S, Ventura D, Lavrac N, et al (eds) Proceedings of the Fifth International Conference on Computational Creativity, ICCC 2014, Ljubljana, Slovenia, June 10-13, 2014. computationalcreativity.net, pp 332–335, URL http://computationalcreativity.net/iccc2014/wp-content/uploads/2014/06/15.3_Garza.pdf

Tobias R (2012) 20 Master Plots: And How to Build Them. F+W Media, URL http://books.google.pt/books?id=EZgEqCYlBwsC

Young RM, Ware S, Cassell B, et al (2013) Plans and planning in narrative generation: A review of plan-based approaches to the generation of story, discourse and interactivity in narratives. Sprache und Datenverarbeitung 37(1-2):41—–64