

Evolutionary Construction of Stories that Combine Several Plot Lines^{*}

Pablo Gervás^[0000-0003-4906-9837], Eugenio Concepción^[0000-0002-1489-1719],
and Gonzalo Méndez^[0000-0001-7659-1482]

Facultad de Informática, Universidad Complutense de Madrid, Madrid, 28040 Spain
{pgervas,econcepc,gmendez}@ucm.es
<http://nil.fdi.ucm.es>

Abstract. Although the narrative structure of common entertainment products like Hollywood movies or TV series is generally composed of a number of different plot lines combined into a single narrative discourse, efforts on computational modeling of story generation have to this point focused mostly on the construction of stories with a single plot line. The present paper explores an evolutionary solution to the task of building a story that combines more than one plot line into a single linear discourse. This requires: a set of knowledge resources that capture the main features that influence the decisions involved, a representation suitable for evolutionary treatment for discourses with several plot lines, and a set of fitness functions based on metrics related to the quality of the resulting discourses. The proposed solution produces populations of stories with elaborate discourses that combine several subplots.

Keywords: story generation · multiplot stories · evolutionary approach.

1 Introduction

The narrative structure of common entertainment products like Hollywood movies or TV series is generally composed of a number of different plot lines combined into a single narrative discourse. There are very clear mechanisms at work in putting multiple plot lines together into a successful narrative discourse in this fashion, yet efforts on computational modeling of story generation have to this point focused mostly on the construction of stories with a single plot line. This is in part due to the application of a traditional rule of engineering – do not consider complex versions of the problem until the simple versions have been solved – and in part due to oversight of another such rule – if the artifact you are trying to build is made up of several parts, the construction process should

^{*} This paper has been partially funded by the projects 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) and InVITAR-IA: Infraestructuras para la Visibilización, Integración y Transferencia de Aplicaciones y Resultados de Inteligencia Artificial, UCM Grant. No. FEI-EU-17-23.

understand what such elementary parts are and how they come together. The present paper explores an evolutionary solution to the task of building a story that combines more than one plot line into a single linear discourse. This requires a set of knowledge resources that capture the main features that influence the decisions involved – namely a set of basic plot lines as sequences of scenes, each annotated with the characters that take part, the narrative roles they play, and basic semantics such as when characters are born, die, or fall in love. A representation suitable for evolutionary treatment is proposed for discourses with several plot lines, and a set of fitness functions are defined based on metrics related to the quality of the resulting discourses are proposed. The solution defined over these elements produces populations of stories with elaborate discourses that combine several subplots.

The combination of several plot lines into a discourse is more complex than a simple weaving. Plot lines are not actually independent streams of narrative discourse that get combined together into a complex fabric: the character sets from different subplots combined into a story do not remain distinct, they may be merged so that the same character in the overall story often plays different roles in more than one of the subplots. This point is addressed explicitly in the solution proposed in this paper. The process we want to model involves an additional operation of merging (some of the characters in) the narrative threads for the different plotlines by instantiating their narrative roles (either main or secondary) with the set of characters for the overall plot.

2 Related Work

Three topics are considered relevant for this paper: prior solutions for plot line combination, quantitative metrics for stories, and evolutionary approaches to creation of narratives of some kind.

2.1 Plot Line Combination

In recent years there has been an increase in the number of research efforts that consider the construction of stories with more than one plot line. In reviewing research efforts on this topic, we have opted for unifying the terminology under an abstract concept of *plot line* considered as a sequence of plot-relevant elements 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. We will refer to the plot-relevant elements in a plot line as *scenes*. When more than one plot line are involved in a larger story, each of these plot lines is considered a *subplot* of the story.

Two different approaches have been followed to create multi-plotline stories. In the first one, the story is created by adding scenes incrementally, switching between different plot lines so that each of the plot lines involved is also dynamically constructed. In the second one, a set of existing plot lines is considered

at the start, and the construction process involves deciding how these plot lines will be combined in the discourse for the final story.

Under the first approach, Porteous et al [16] describe an interactive storytelling system that builds stories in a way that the resulting story will show multiple interleaved plot lines. Their system relies on a plan-based approach that takes input parameters that govern the number of plot lines that should be involved in the final version and how much of the final story should be taken up by each plot line.

Under the second approach, systems first compile a set of narrative threads from a given source and then search for possible combinations of them that make a story. Fay [3] constructs multi-plotline stories by combining together narrative threads for specific types of characters found in a given story request, so that main characters in the story request are bound to secondary characters in threads for other characters and the elements from each thread are combined into a consistent overall timeline. The Raconteur [6] and StoryFire [5] systems obtain narrative threads from a conceptual description of a chess-game, and builds stories by selecting a *plot schema* to use as a template to be filled in with matching scenes from the available threads. The PlotAssembler system [7] builds stories by combining a set of small spans of narrative discourse called *axes of interest* in ways that interweave their scenes, ensuring that character continuity across scenes is compatible with probabilities mined from a corpus of prior stories. The work of Concepción et al [2] explores baseline solutions for weaving together a set of plot lines into stories. The plot lines considered include additional information on roles played by the characters. The set of weaving procedures includes some based on existing literary techniques and some presented as baselines for computational approaches to the task.

2.2 Computational Metrics for Stories

Over the past few years a body of computational metrics for stories has been developed. As the set of features that might be considered in a story is vast, these metrics generally focus on particular aspects that are relevant for specific purposes. When the goal is to invent a new story, metrics relevant to the purpose focus on aspects that may be related to originality: story novelty [14], similarity between stories [9] and correlations between easy-to-measure features in the stories and the creativity attributed to them by human judges [20].

More recent work addresses the evaluation of multi-plotline stories specifically. An attempt at formative qualitative evaluation of multi-plotline stories was carried out in [2]. A group of human judges were asked to consider the quality of 10 examples of multi-plotline stories, and to identify specific features that they considered positive or negative contributions to the perceived value of the story. A number of interesting conclusions were drawn. First, the most detrimental feature to perceived story quality was the existence of semantic inconsistencies in the story, such as characters that keep taking part in the story after being dead. Second, positive judgments often involved identification of characteristics of the story that were optional rather than necessary, such as the existence of an

overarching plot, or the fact that one subplot was inserted fully within another. In [8], the qualitative analyses presented as part of the formative evaluation were distilled into a set of metrics designed to capture them in a numerical form. These metrics were designed to respond to the set of features that human judges were seen to focus on when asked to assess multi-plotline stories, and to translate the judgments made onto numerical scores.

2.3 Evolutionary Construction of Narratives

Without pretending to be exhaustive, this section reviews some of the existing efforts to generate narratives by means of evolutionary solutions in terms of two different aspects: which elements they combine and what type of fitness function they employ.

In terms of the elements that are combined to make stories, McIntyre and Lapata [13] present a plot generator that builds a plot from the set of entities appearing in a sentence provided by the user. For each entity in the sentence the system retrieves a plot line from a knowledge based automatically extracted previously, and creates a space of possible stories involving the given entities by merging together these plot lines. This search space is then explored by means of genetic algorithms, using as fitness function a combination of story coherence and story interest. Gómez de Silva and Pérez y Pérez [19] propose a model of story construction that combines the MEXICA existing knowledge-based story generator [15] with the GENCAD evolutionary approach for the adaptation stage in case-based solutions to architectural problems [18]. The story construction model relies on the knowledge-based heuristics of MEXICA to build initial populations and applies the evolutionary approach to refine them. Among the features that the fitness functions are designed to detect are individuals with ‘incestuous ancestry’, built by applying more than once rules from the same exemplar, which may lead to repetition of events in the output story.

In terms of the fitness functions employed to drive the evolutionary approach, Fredericks and DeVries [4] describe an application of an evolutionary solution driven by novelty search [11] to improve the novelty of procedurally-generated small narratives fragments for text-based games. Kartal et al [10] use Monte Carlo tree search to drive planning-based narrative generation in support of a interactive framework for user-driven narrative variation. They rely for selection on a combination of the believability of the resulting story and the percentage of the user-defined goals the current story accomplishes. Soares de Lima et al [12] combine planning with an evolutionary search strategy guided by story arcs to generate quests for games. Candidate quests are constructed by a planner as linear sequences of events or tasks to be accomplished by the player. The evolutionary algorithm chooses among them using a fitness function that builds for each candidate quest the sequence of tensions that arise from it, and scores it based on its match with a target curve of evolving tensions provided by the user.

3 An Evolutionary Multiplot Story Composer

A story built by combining plot lines requires a complex set of decisions: in which order to combine the scenes from the different plot lines and how to merge the characters sets from the different plot lines into a single set of characters for the overall story. The search space of possible solutions is immense. Prior attempts to explore this search space in terms of heuristics for informing the required decisions [2] have shown that stories produced in this way are significantly more rigid in structure than human produced stories. An evolutionary solution based on fitness functions that capture some of the desired features for the output stories may provide efficient means of traversing this search space.

3.1 The Knowledge Resources

The experiments reported in this paper are carried out over a set of plot lines extracted from a compilation of classic plots used in cinema [1]. A subset of these plots have been distilled into a set of *plot templates*, which are formalised knowledge resources that describe the structure of the plot as a sequence of scenes. The set of plot templates currently available is: The Benevolent Outsider, The Creation of Artificial Life, Descent into the Underworld, The Destructive Outsider, Faust, Split Person Comic, Split Person Tragic.

Each *scene* describes the characters that take part, the narrative roles they play, the set of semantic annotations that are used by the system to check consistency, and a template for rendering the scene as text. The narrative roles played by the characters are represented by labels that identify the main characters involved in a given plot – say, Creator and Creature in a Creation of Artificial Life plot, or Tempter and Tempted in a Faust plot. The semantic annotations cover aspects about the meaning of the scene that may affect the perceived consistency of the stories in which they appear. The current version considers the following semantic annotations: whether characters are *created/born or die* in the scene and whether characters *fall in love or out of love*. These semantic annotations allow the definition of metrics to identify situations where narrative roles from different subplots are assigned to the same character in the story, and events affecting the different roles are incompatible – characters dying more than once, or serially falling in love. These metrics are used to inform the fitness function for the evolutionary solution.

Each plot template has a particular character identified as the *protagonist* – this is usually one of the characters playing an important role in the plot.

This set of features emerges from the analysis of a number of multi-plot stories described in [2] and the metrics for automated assessment of quality of multi-plotline stories presented in [8]. The insights from these analyses have identified these features as relevant to the perception of quality of a multi-plotline story.

3.2 Character Fusion and Discourse Planning

The analyses carried out to this point on how multi-plotline stories are put together show that, given a set of subplots to be considered as inputs, there are two different processes that contribute to the final result.

The first process is related to the set of characters that take part in each subplot. For a story to be successful, it appears to be important that the sets of characters involved in each subplot have a non-empty intersection. In operational terms, this is achieved by unifying some character a from subplot A with character b from another subplot B , so that a single character – say *John* – in the story undertakes both the part of character a from subplot A and the part of character b from subplot B . We refer to this operation as *character fusion*. In terms of traditional views on natural language generation [17] this operation is a part of the task of *content determination*, since the nature of subplots is somewhat changed in the process of attributing particular characters to the variables in the corresponding template.

The second process is related to the order in which the scenes from the different subplots are presented in the discourse for the final story. This is known to be crucial to the impact that the story has on the audience, and exploited intentionally by authors to achieve effects such as *suspense* – where presentation of certain information is withheld on purpose – or *cliff-hanger breaks* – where transitions to cover other subplots are made at points where tension is high in the current subplot. In terms of traditional views on natural language generation [17] this operation is a fundamental part of the task of *discourse planning*.

Character fusion and discourse planning are tightly interconnected by the information in the semantic annotations for scenes. A fusion between a character a from one template A and a character b from another template B is only possible if it will not result in a final discourse in which the fused character is seen to behave in an impossible manner – such as being born or dying more than once, or being involved in more than one passionate romance without the fact being addressed by the story.¹ The metrics that inform the fitness function should penalise the scores of stories that incur in this type of inconsistency. Even if none of these extreme inconsistencies occur, a particular discourse plan may be incoherent if a fused character takes part in events that happen in the final discourse either before its birth or after its death. Specific metrics for identifying these situations are also needed.

3.3 Representing Multiplot Stories for Evolutionary Construction

In order to apply an evolutionary approach, each story draft needs to be represented in terms of some kind of genetic information that describes how it addresses the main tasks involved: discourse planning and character fusion. To achieve this we propose a solution in several parts.

¹ For clarifications on how romantic conflicts are handled in the current version of the system see the discussion in Section 3.6.

The problem of constructing a multiplot line story considers the following inputs: a set of plot lines to combine, where each plot line is defined by a narrative thread expressed as a sequence of scenes together with a set of characters that appear in it.

This implies that, for a particular problem of combining N plotlines, the length of the final discourse is determined by the total number of scenes in the narrative threads being considered, and the maximum number of possible characters featuring in the story is determined by the union of the sets of characters in the narrative threads being considered.

For simplicity, we are assuming that the relative chronological order of the scenes in each narrative thread is respected in the final discourse. This is not a necessary condition. Indeed, many stories present instances of altered chronology (flashbacks, flashforwards). However, we leave this point to be addressed in further work.

The characterisation of the discourse plan for a given story candidate requires: establishing which narrative thread to start on, defining the specific points in the final sequence in which the discourse switches to a different narrative thread, and defining to which of the other available threads the discourse is switching when it does.

The representation is based on the fact that, for a given story construction problem, the length of the final discourse is known and fixed, and the fact that the set of plotlines being considered is known and fixed.

The information on discourse planning is represented in terms of vectors that define how the narrative threads for the different plot lines are combined into the final linear discourse:

- a single digit (0 or 1) defines which narrative thread 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 narrative thread or it switches to a different thread
- 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 plots are skipped whenever the discourse switches to a different narrative thread

The set of possible characters for the complete story is defined by the union of the sets of variable names for the characters appearing in each narrative thread. These variable names need to be distinct across the different narrative threads to avoid confusion. This is ensured by assigning a prefix with the plotline name to all the variable names that feature in any given narrative thread.

The characterisation of the choices for character fusion for a given story candidate requires an assignment of character names to each of the variables in the joint set of variables for the story.

For simplicity, the set of potential character names for the story is defined to be the set of integers from 0 to C , with C being the cardinality of the joint set of variables for the story. This is sufficient to represent any choices made in

terms of character fusion (with variables in two different positions in the name-assignment vector being assigned to the same integer). The form of the resulting stories would be significantly improved by a later stage of transforming these integer names for the characters into strings representing realistic names.

An example of representation is shown in Figure 1.

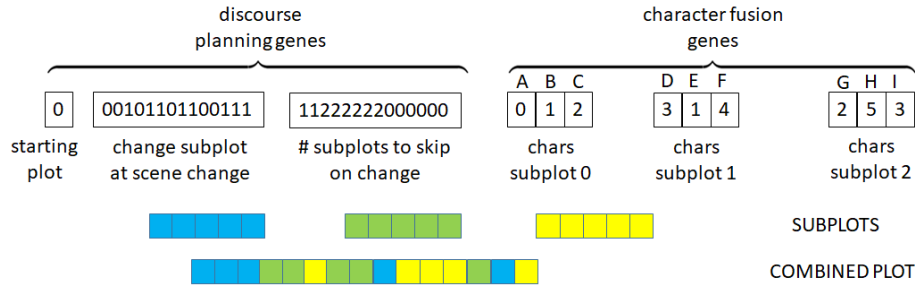


Fig. 1. Genetic representation for a combination of three plots of length 5, each with 3 characters. Fuses characters B (p0) / E (p1), C (p0) / G (p2) and D (p1) / I (p2).

3.4 Constructing an Initial Population

An initial population of story candidates is built by assigning values to the representation described in Section 3.3. 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 narrative thread 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 character to assign to each variable, the choice is more complex. This is because variables from the same narrative thread should not be assigned to the same character, at the risk of confusing the relations between characters in the corresponding subplot. The process of assignment is carried out separately for the set of variables for each thread. For such a set of variables, the process decides at random whether to assign to each variable either a character name chosen at random from those already used in some of the narrative threads already processed, or an entirely new character name chosen at random from the character names that remain free. This ensures the required constraints are satisfied.

3.5 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 each point are either mutated or not depending on a threshold parameter
- for the skip size vector, values at each point are either mutated or not depending on a threshold parameter, and, if required, mutated to a value chosen at random within the required range
- for the character assignment vector, character names at each point are either mutated or not depending on a threshold parameter, and, if required, mutated to a character 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 character assignment vector, the assignments of characters for the two different individuals are swapped over

3.6 Fitness Functions

The construction of potential story drafts as described in the initialisation of the population and the evolutionary operators is mostly random. However not all possible combinations are actually valid. Certain plot lines, when combined in certain orders, give rise to incoherence in the semantics of the story. Incoherence may affect different aspects: characters that act before being born or after being dead, or characters that fall in love serially with no regard for previous romances. It falls to the fitness function to differentiate between valid and invalid story candidates. This is achieved by taking into account the knowledge resources described in Section 3.1.

As there are many different aspects to be considered for a story, a modular approach is applied to the definition of a fitness function for the system. Rather than build a complex fitness function that addresses all the aspects, a number of targeted fitness functions are built, and the fitness for a given individual is computed as a function of the results it achieves under the specific fitness functions.

This approach allows a differentiated consideration of the aspects related to validity of the story draft, the aspects related to satisfaction of optional traits, and the aspects that may actually relate to perceived quality of any given solution.

The different aspects that need to be considered as contributing to the quality of stories differ in the way they impact the overall perception of a reader. These different ways require different solutions to model numerically the expected behaviour. The solution presented here is a tentative proposal that captures the basic intuitions arising from prior empirical studies. Further work may need more detailed empirical studies and matching adaptations of the computational solutions.

Validity Fitness Functions A first set of fitness functions addresses the consistency of the final discourse in terms of the semantics for the scenes as annotated in the knowledge resources described in Section 3.1. All these functions assign a score of 100 if the story is consistent with respect to the corresponding feature, and 0 otherwise.

The current set of fitness functions includes the following aspects that affect the validity of a story draft:

- whether characters are born more than once in the story (rules out combinations of plot lines that merge into a character two subplots that both mention the birth)
- whether characters die more than once in the story (rules out combinations of plot lines that merge into a character two subplots that both mention the death)
- whether characters are active in the story at points that do not lie between their birth and their death (or the boundaries of the story if birth or death are not mentioned)
- whether the romantic behaviour of characters is consistent (rules out combinations of subplots that imply one character is passionately in love with different people)

The fitness function for consistent romantic behaviour is currently an initial approximation to the task. Cases where characters fall in love with more than one person do exist as valid stories, but they are only interesting when the plot addresses the romantic conflict explicitly and resolves it in some way. The limited procedure of story construction applied here – restricted to interweaving scenes from two different plots – cannot address this task in its current form. Further work will consider extensions to the construction procedure and a matching revised fitness function to address this issue.

Optional Traits Fitness Functions The second set of fitness functions addresses aspects noted to be positive traits by the human judges but which constitute optional rather than necessary configurations for a story. These traits add value to a story when present but do not detract from its quality if absent.

Fitness functions are defined to capture the corresponding traits, also scoring 100 if the trait is present and 0 if it is absent.

In the present version of the system, such fitness functions are added to the mix for a given run only if the corresponding trait is desired in the outcome population.

The following traits are considered for discourse plans:

- *inserted subplot*: a complete subplot is inserted as an aside into the story (all scenes from the subplot appear together in the final discourse, and surrounded by scenes from other subplots)
- *overarching plot*: the story starts and finishes with scenes from the same subplot, which therefore appears as a frame for any other subplots in the story – this does occur in cases where there is an inserted subplot but may also occur in other situations, and therefore it is considered as a separate feature

Additional traits may be considered in the same way for character fusions, which establish a link between the casts of the subplots involved in any given story. The importance of the link depends on the relative importance of the character involved in the link with respect to each of the subplots. For the present paper, the following type of links have been considered:

- *shared protagonist*: the same character acts as protagonist of two subplots
- *stitching protagonist*: the protagonist from one subplot plays an important role – different from the corresponding protagonist – in another

Fitness functions of the type explained above are defined for each of these types of links, allowing the outcomes to be driven towards stories satisfying the corresponding criteria.

Combining Fitness Functions to Score Individual Stories To ensure that the different types of fitness functions described above are combined into a single score for each candidate story in a population, stories are assigned as a final score the average of the scores for the set of fitness functions for specific features selected.

This ensures that individual stories that are invalid or do not exhibit the desired traits disappear from the population as early as possible, and that stories with higher quality have a higher chance of survival.

4 Discussion

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

4.1 Results

The proposed system is run in each case with an initial population of 1,000 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 20 generations. At each generation populations are culled by selecting the next generation using a best scoring criterion.

Text	#	Plotline	Scene name	Variable	Role	Name
A is a doctor researching life improving serum. A has dedicated his whole life to science and has no friends or family.	0	SPT	Introduction	CLM-0	Frankenstein	C
A accidentally creates a serum that changes him into B, an evil twin of himself.	1	SPT	Splitting	CLM-1	Creature	D
C is a German nobleman. C lives in a big castle. C is sad. C 's wife died recently. C loves his dead wife. C dreams of bringing to life his wife.	2	CLM	Conception of Idea	CLM-2	-	E
C studies book of magic. C discovers a spell to bring to life a portrait of his wife. C brings to life D , the portrait of his wife.	3	CLM	Creation of new life	SPT-0	DrJekyll	A
C talks every evening with D. C tells D that C loves D. D is kind to C.	4	CLM	The being complies with its master's will	SPT-1	MrHyde	B
C goes to a ball. C meets E in a ball. C falls in love with E. E falls in love with C. C tells D C loves E.	5	CLM	Rebellion and escape	SPT-2	-	C
E visits C. D is jealous of E. D takes E and disappears.	6	CLM	Infringement by creature			
C searches for D across the castle.	7	CLM	Quest for the creature			
D tells C that C must defeat D to save E. C burns the portrait. D is destroyed and E appears magically. C and E are happy.	8	CLM	Death of the creature			
B brutally kills some friends of A. C realises that B is an evil version of A. A confesses to C what has happened.	9	SPT	Entanglements			
A decides to stop taking the serum.	10	SPT	Discovery			
A transforms involuntary into B. B goes on a rampage. Having lost control, A decides to commit suicide, killing B in the process.	11	SPT	Unravelling			
	12	SPT	Denouement			

Table 1. An example of output story combining the plotlines creation of artificial life (CLM) and split person tragic (SPT). Columns show: the *text* for the story broken down by scenes, description of the *discourse plan* indicating for each scene which plotline it comes from, description of the cast indicating the roles they play in the respective plot lines and what names have been assigned to them – which reflect any *character fusions* present. Columns 1 and 2 are alligned by scenes, column 3 applies to the story as a whole.

Some examples of results are shown below.

The stories generated by the system are rendered as text using the templates stored for the corresponding scenes in the knowledge resources, replacing the variables with the assigned character names. An example of complete story is shown in Table 1. Numbers used internally as character names have been replaced with capital letters for ease of reading. This story is obtained with the configuration set to construct stories with an inserted plot. The discourse plan indeed shows all scenes from CLM plot appearing together, bracketted by scenes from SPT. In this case, the two subplots are only very lightly connected by character fusions: variable CLM-0, the protagonist of CLM, playing the role of Frankenstein has been fused with variable SPT-2 that corresponds to a secondary character in SPT.

To illustrate the operation of the fitness functions for optional traits, examples of the structure of different stories are shown in Table 2.

# Plotline	Scene name	# Plotline	Scene name	# Plotline	Scene name
0 SPT	Introduction	0 SPT	Introduction	0 DiU	Lovers Initial Happiness
1 SPT	Splitting	1 FA	Frustrated character regrets his life	1 DiU	Lost
2 CLM	Conception of Idea	2 SPT	Splitting	2 DiU	Mourn and quest for the beloved
3 CLM	Creation of new life	3 FA	Temptation	3 FA	Frustrated character regrets his life
4 CLM	The being complies with its master's will	4 FA	Pact with evil	4 FA	Temptation
5 CLM	Rebellion and escape	5 SPT	Entanglements	5 FA	Pact with evil
6 CLM	Infringement by creature	6 FA	Evil actions	6 DiU	Deal and brief reunion
7 CLM	Quest for the creature	7 FA	Enlightenment	7 FA	Evil actions
8 CLM	Death of the creature	8 SPT	Discovery	8 DiU	Infringement of compromise
9 SPT	Entanglements	9 FA	Redemption	9 DiU	Metamorphosis
10 SPT	Discovery	10 SPT	Unravelling	10 FA	Enlightenment
11 SPT	Unravelling	11 SPT	Denouement	11 FA	Redemption
12 SPT	Denouement				

Table 2. Structures for stories obtained with different configurations for the optional traits concerning discourse structure, using different combinations of plotlines creation of artificial life (CLM), split person tragic (SPT), Faust (FA) and descent into the underworld (DiU). Column 1 shows a story with inserted plots, column 2 shows a story with overarching plot but no inserted plot, column 3 shows a story with neither overarching plot nor inserted plot. For ease of understanding of the structure of the discourse plans, the scenes of one of the subplots are highlighted in bold.

To illustrate the operation of the fitness functions for character fusion, examples of different stories are shown in Table 3. To provide informative views while respecting page limit constraints, only cast descriptions are shown. The first column is an example of shared protagonist (same character A plays Orpheus and Faust). The second column is an example of stitching protagonist (character A plays Orpheus and Mephistopheles and character B plays Euridice and Faust).

DUW-0 Orpheus	A A A
DUW-1 Eurydice	B B B
DUW-2 Hades	D C C
FAU-3 Faust	A B C
FAU-4 Mephistopheles	B A D

Table 3. Examples to illustrate options on character fusion. Names of fused characters are highlighted in bold.

The third column is a weaker example of stitching protagonist (character C plays Hades and Faust).

4.2 Relation with Previous Work

The character fusion operation considered here is comparable to binding between characters as used by Fay [3].

The procedure applied in McIntyre and Lapata [13] shows significant parallels with our own approach: a set of possible stories is created by combining plot lines from different stories and then an evolutionary approach is applied to search for a convincing set of output stories. However, our system differs in that it takes as input a set of already established plot lines that cannot be altered, whereas McIntyre and Lapata explore different choices of elements – tree structures corresponding to sentences – to build the stories.

The metric for detecting individuals with ‘incestuous ancestry’ in [19], intended as it is to avoid repetition of events in the stories, is related to our fitness function to avoid characters in the stories falling in love more than once. Attempts to combine more than two plotlines may require a similar adaptation to avoid, at least initially, stories that include the same plot line several times.

The semantic annotations for the plot templates in our knowledge resources play a similar role to the domain database as considered in [12] to compute tension. The judgement on validity of stories based on these semantic annotations plays a role similar to the believability metric as considered in [10].

5 Conclusions

The evolutionary approach to constructing multi-plotline stories provides efficient means of building a population of drafts that satisfy constraints on semantic validity over the final linear discourse for the story. The drafts in the population can be steered towards stories that satisfy specific features in terms of particular structures in the discourse plan (such as overarching plot or inserted plots) and/or particular choices for character fusion (such as shared protagonists, linking roles or only loose connections between the casts of the different subplots).

The current set of features identified as relevant to story quality is not well suited to numerical ranking in terms of fitness functions. As a result, final populations tend to converge towards scores of 100/100 for all individuals. Nevertheless, the fact that scores for individuals are constructed as a combination of

more specific scores for particular features implies that during the construction procedure – in earlier generations of the evolution – individuals do have scores in the range between 0 and 100. This allows the evolutionary procedure to explore different combinations of the various subparts of the representation vector, to achieve a set of valid solutions in the final population that satisfy the desired constraints.

As future work we will consider extending our system with metrics of the various types used in the other approaches reviewed: believability, story arcs based on tension.

Although user-defined goals are a feature specific to user-driven narratives of the type addressed in [4] and [10], stories built outside these contexts often do have a purpose beyond entertainment – convincing, educating, . . . Moreover, there may not be a single goal but rather a set of goals to achieve. Under this light, extending the fitness functions for our system with metrics for percentage of goals achieved will be considered as future work.

It seems that evolutionary approaches have been used often for story generation as a secondary process applied to a set of stories constructed previously by some other method. Examples of this initial material produced by other means are the stories built by MEXICA in [19] and the grammar-driven quest sketches produced by Tracery in [4]. In our case, we apply the evolutionary solution to combine plot templates that constitute already-built stories. The solution we propose could be used in future work as a secondary process on the results of preceding story generation procedures of a different type.

References

1. Balló, J., Pérez, X.: *La semilla inmortal: los argumentos universales en el cine*. Ed. Anagrama (2007)
2. Concepción, E., Gervás, P., Méndez, G.: Exploring baselines for combining full plots into multiple-plot stories. *New Generation Computing* pp. 1–41 (10/2020–2020). <https://doi.org/10.1007/s00354-020-00115-x>, <https://link.springer.com/article/10.1007%2Fs00354-020-00115-x>
3. Fay, M.P.: *Driving story generation with learnable character models*. Ph.D. thesis, Massachusetts Institute of Technology (2014)
4. Fredericks, E.M., DeVries, B.: (genetically) improving novelty in procedural story generation. arXiv preprint arXiv:2103.06935 (2021)
5. Gervás, P.: Storifying Observed Events: Could I Dress This Up as a Story? In: 5th AISB Symposium on Computational Creativity. AISB, AISB, University of Liverpool, UK (04/2018–2018)
6. Gervás, P.: Composing narrative discourse for stories of many characters: a case study over a chess game. *Literary and Linguistic Computing* **29**(4) (08/14–2014)
7. Gervás, P.: Generating a search space of acceptable narrative plots. In: 10th International Conference on Computational Creativity (ICCC 2019). UNC Charlotte, North Carolina, USA (07/2019–2019)
8. Gervás, P., Concepción, E., Méndez, G.: Assessing multiplot stories: from formative analysis to computational metrics. In: 12th International Conference on Computational Creativity (ICCC 2019). Mexico City, Mexico (09/2021–2021)

9. Hervás, R., Sánchez-Ruiz, A.A., Gervás, P., León, C.: Calibrating a metric for similarity of stories against human judgement. In: ICCBR (Workshops). pp. 136–145 (2015)
10. Kartal, B., Koenig, J., Guy, S.J.: 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. p. 69–76. AAMAS '14, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2014)
11. Lehman, J., Stanley, K.O.: Exploiting open-endedness to solve problems through the search for novelty. In: Proceedings of the Eleventh International Conference on Artificial Life. MIT Press (2004)
12. de Lima, E.S., Feijó, B., Furtado, A.L.: 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. pp. 144–153. IEEE (2019). <https://doi.org/10.1109/SBGames.2019.00028>, <https://doi.org/10.1109/SBGames.2019.00028>
13. McIntyre, N., Lapata, M.: Plot induction and evolutionary search for story generation. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. pp. 1562–1572. Association for Computational Linguistics, Uppsala, Sweden (Jul 2010), <https://aclanthology.org/P10-1158>
14. Peinado, F., Francisco, V., Hervás, R., Gervás, P.: Assessing the novelty of computer-generated narratives using empirical metrics. *MINDS AND MACHINES* **20**(4), 588 (10/2010 2010). <https://doi.org/10.1007/s11023-010-9209-8>
15. Pérez y Pérez, R., Sharples, M.: Mexica: A computer model of a cognitive account of creative writing. *Journal of Experimental & Theoretical Artificial Intelligence* **13**(2), 119–139 (2001)
16. Porteous, J., Charles, F., Cavazza, M.: Plan-based narrative generation with coordinated subplots. In: European Conference on Artificial Intelligence (ECAI 2016). vol. 285, pp. 846–854. IOS Press (2016)
17. Reiter, E., Dale, R.: *Building Natural Language Generation Systems*. Cambridge University Press (2000)
18. Gómez de Silva Garza, A.: *An Evolutionary Approach to Design Case Adaptation*. Ph.D. thesis, The University of Sydney, Australia (2000)
19. de Silva Garza, A.G., y Pérez, R.P.: Towards evolutionary story generation. In: ICCG. pp. 332–335 (2014)
20. Tapscott, A., Gomez, J., León, C., Smailovic, J., Znidarsic, M., Gervás, P.: Empirical evidence of the limits of automatic assessment of fictional ideation. In: C3GI ESSLLI (2016)