

Adapting Proppian Morphology for Generating Narrative Structures

Pablo Gervás

Facultad de Informática
Universidad Complutense de Madrid
Madrid, 28040, Spain

Abstract

Vladimir Propp's "Morphology of the Folk Tale" presented a meticulous analysis of a basic type of story into several layers of constituents that could be recombined to make new stories. Early adaptations of a very limited portion of his account to generate stories were criticised on the grounds that the representation was too simple. The present paper considers how the more complex aspects of Propp's analysis might be implemented computationally. The original descriptive representation of stories is augmented with solutions inspired by reverse engineering some of the more complex examples of stories analysed by Propp in his book. The revised computational model includes important aspects of narrative such as the distinction between fabula and discourse, the use of embedded stories and construction of narrative structures with more than one plot line.

Introduction

The Morphology of the Folk Tale proposed by Vladimir Propp has been used as inspiration by many computational attempts at generating stories. However, in most cases the corresponding computational systems departed considerably from Propp's original formulation. The present paper explores a solution that: (1) focuses on the generation of sequential tales matching the nature of those in Propp's corpus, (2) enriches Propp's original formulation with the details necessary to make it operational from a computational point of view, and (3) at all levels prioritises faithful compliance with Propp's account over implementation details considered traditional for specific computational approaches.

Related Work

Three branches of previous work need to be reviewed: Propp's formalism, relevant concepts of narratology and related computational story generators.

Propp's Morphology of the Folk Tale

Vladimir Propp (1928) was a Russian formalist interested in the form of the tale. With the aim of developing a useful tool for adequate classification, factual description and comparison across tales he studied the component parts of a tale – and of their relations to each other and to the whole – over a selection out of a corpus of Russian folk tales (Afanasyev,

1859). The tales studied involved a hero facing a complication (a villainy or a lack) who sets out, undergoes some tribulations and eventually succeeds and is rewarded. Propp's analysis is best known for the formulation of *character functions* – abstractions of character behaviour relevant to the plot – and a *canonical sequence* – a relative order in which all possible character functions appear in a tale with respect to one another. His analysis includes a number of additional elements that are less well known – possibly because their description in the book is much less detailed – but which have a significant potential for computational consideration. First, a set of roles for the characters in the narrative (the *dramatis personae*) which include, among others: the *hero* (who sets out on a journey), the *dispatcher* (who dispatches the hero on his journey), the *villain* (that the hero faces during the story), the *donor* (who provides a magical agent to the hero), the *false hero* (who competes with the protagonist for the role of hero of the story). Second, the concept of *move*: a single pass over the canonical sequence selecting character functions to include in a tale for a given hero. According to Propp, tales beyond his simple initial description could be analysed as a combination of several moves. He includes several analyses of this type for considerably complex tales in the appendices of his book. Third, he describes a procedure to "create a tale artificially", based on selecting character functions out of the canonical sequence and then assigning to them the characters for the tale while respecting the restrictions imposed by which character functions involve which roles of the characters: "In order to create a tale artificially, one may take any *A*, then one of the possible *B*'s then a *C*↑, followed by absolutely any *D*, then an *E*, the one of the possible *F*'s, then any *G*, and so on. In doing this, any elements may be dropped (...). If one then distributes functions according to the *dramatis personae* of the tale's supply or by following one's own taste, these schemes come alive and become tales. Of course, one must also keep motivations, connections, and other auxiliary elements in mind" (p. 111-112). This description served as inspiration for the paper.

Narratological Concepts Relevant for Story Generation

Narrative involves two separate chronologies: the timeline of the events being narrated and the timeline of their telling

(Abbott, 2008). The events of a narrative in the chronological order in which they happened is referred to as the *fabula*. The events of a narrative in the chronological order in which they are told in the narrative is referred to as the *discourse*.

Characters in a story sometimes do themselves tell stories to other characters. These are known as *embedded stories* (Herman, Jahn, and Ryan, 2010). The overarching narrative in which this telling takes place is known as the *frame story*.

Computational Story Generators

Over the years, Propp’s Morphology of the Folk Tale has been revisited often as inspiration for computational approaches to narrative. These approaches have ranged over a wide set of formalisms and technologies: generative grammars (Lakoff, 1972), case based reasoning (Fairclough, 2004) and case based reasoning combined with planning (Turner, 1993). Of those, Fairclough (2004) considers moves – in Propp’s sense – as cases, and story construction as a process of selecting moves from a casebase and combining them into a new story. Gervás (2019) explicitly links the concepts of character function and narrative role into a structured concept of *plot element* that explicitly includes representation of which specific characters contribute in which roles with respect to the story. It also introduced the concept of *axis of interest* (or AoI) as a short sequence of plot elements 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.

Propp’s idea of multi-move tales resonates with recent efforts to construct stories with more than one plotline, such as Fay (2014) – that combines plot lines extracted from existing stories and uses bindings between characters in them to connect them together – and Gervás, Concepción, and Méndez (2022) – that combines two different operations: *discourse planning* – decide on the relative order of appearance in the discourse of contributions from different subplots – and *character fusion* – bind together for the final story roles played by different characters in different subplots.

The ProppOSER Framework

The ProppOSER (**Propp** Order **S**anction **E**ntwine **R**ewriter) system has been developed to explore how the complete set of concepts proposed by Propp in his book might be exploited to generate complex stories. To follow this goal, instead of exploiting existing AI techniques that might be useful, an effort has been made to come up with computational representations of Propp’s elements, and to design specific procedures that combine them in a manner that is either faithful to the procedures described explicitly by Propp or implements adequate procedures designed to produce outcomes that are equivalent in narrative structure to the complex tales described by Propp in the appendices of his book.

Proppian Data Structures

The work presented here relies on the concepts *plot element* and *axis of interest* as proposed by Gervás (2019). A plot element is a data structure that includes a character function

AXISofINTEREST =	DONOR
Tested	characters(tested=X,tester=Y)
Character’sReaction	characters(tested=X,tester=Y)
ProvisionOfAMagicalAgent	characters(tested=X,tester=Y) objects(gift=Z)
UseOfAMagicalAgent	characters(user=X) objects(gift=Z)

AXISofINTEREST =	CONFLICT
Struggle	characters(attacker=X,defender=Y)
Victory	characters(winner=X,loser=Y)

Table 1: Examples of axes of interest.

label and a set of roles that characters need to play in it. These roles can be instantiated by identifiers for particular characters. An axis of interest is a data structure that captures dependencies between plot element that normally occur together in stories, including the relative order in which they appear and which of their roles have to be instantiated by the same characters. Two examples of axes of interest are shown in Table 1.

In Propp’s description a given move is characterised by a particular set of instantiations of the roles for the dramatis personae with specific characters. To capture this concept we introduce a *role combo* data structure, which can hold the set of instantiations of roles for the the dramatis personae with specific characters for a particular move. The set of actual roles that Propp mentions in his book differs slightly across the different mentions. We have chosen to include the following roles: hero, villain, victim, authority figure, dispatcher, donor and false hero.

A *Proppian move* data structure then holds together: a set of AoIs chosen to be included in the move, a table of which specific characters instantiate the various roles in the plot elements of the AoIs, and a role combo data structure that specifies which characters are assigned in this move to the roles for the dramatis personae.

A version of the *canonical sequence* in which character functions should occur in a move has been created based on the order in which character functions are tabulated in the analyses presented by in Appendix III.

Generating Single Moves Propp’s Way

We construct instances of Proppian move data structures by compiling a set of AoIs that match a given complication, and splicing them all together under the same combo for narrative roles. This involves a stage of interweaving the plot elements from the different AoIs, and a stage of establishing which characters are shared across different AoIs. The interweaving is done by following explicitly the procedure proposed by Propp: follow the canonical sequence and at each point select the plot element from the compiled AoIs that appears next in the sequence. The assignment of characters to roles in AoIs has to be done respecting the particular narrative role associated with each character slot in the corresponding plot elements and the assignment of roles to characters given in the role combo for the move.

Tested	Character'sReaction	Deception	Deceived	DifficultTask	Solution
Character'sReaction	PromiseOfHelp	Struggle	Victory	Departure	Return
Character'sReaction	ProvisionOfAMagicalAgent	Trickery	Complicity	ProvisionOfAMagicalAgent	UseOfAMagicalAgent
Character'sReaction	Validation	Command	CommandObeyed	PromiseOfHelp	HelpReceived
HelpReceived	Transference	NoticeFact	InferCause	Validation	ValidationRecognised
UseOfAMagicalAgent	Transference	Discovery	Communicate	FalseClaims	FalsehoodExposed
HelpReceived	RecoverEaten	Pursuit	RescueFromPursuit	DecisionToReact	Reward
UseOfAMagicalAgent	RecoverEaten	Victory	TokenAcquired	Lack	LackResolved
HelpReceived	Resuscitation	TokenDisplayed	Recognition	Maiming	RegenerateMaiming
UseOfAMagicalAgent	Resuscitation	Deceived	Maiming	Blinding	ReturnSight
HelpReceived	Rescue	Deceived	Blinding	Devour	RecoverEaten
UseOfAMagicalAgent	Rescue	Deceived	Devour	UnrecognisedArrival	Recognition
HelpReceived	Departure	Deceived	Abduction	Murder	Resuscitation
UseOfAMagicalAgent	Departure	Deceived	Capture	Abduction	Rescue
HelpReceived	Transference	Transference	UnrecognisedArrival	Capture	Rescue
FishForInfo	InfoRelinquished	CallToAction	DecisionToReact	TokenAcquired	TokenDisplayed

(a) attracted character functions

(b) repulsed character functions

Table 2: Pairs of character functions considered: (a) attracted to one another and (b) repulsed by one another.

Going Multi-Move: When One is Not Enough

Propp’s description of how his framework might be used to generate stories focuses on picking out selected character functions from this canonical sequence and considering them as the sequence describing a new story. However, the tales analysed in the appendices have more than one such sequence, connected together by having characters that play one role in one sequence and play a different role in another. Propp describes these as tales with more than one move. A move in this context is some extract from the canonical sequence, featuring a particular set of characters instantiating the relevant narrative roles.

Multi-move tales may be constructed by building a number of moves following the procedure suggested by Propp to build one move, and then combining the resulting set of moves together. Combining moves together involves two different levels of decision: instantiating some characters in common between the move (to provide a link between them) and interweaving the events of the various moves.

Character fusion across moves is done in terms of the set of roles in the role combos for the different moves. So we can have fusions between the heroes of two moves (which packs the moves into a single hero storyline) or a number of combinations of the supporting casts of the different moves (hero becomes victim, victim becomes hero, shared villain or multiple villains, single or multiple victims, shared donor, dispatcher of one move is also helper in another...). This approach allows for richer outcome stories.

Decisions on the relative order in which the plot elements from the different moves should appear in the final story require additional considerations. A baseline that combines the plot elements from the moves at random results in stories that make little sense, but Propp gives no indication of what criteria might be used to inform this task. We have therefore looked at the examples of tales that he analyses for inspiration. We have opted for a solution based on identifying: (1) character functions should appear together in the tale because they are causally connected, (2) character functions intended to span larger segments of the story, and (3) character functions that inactivate a particular character.

Even though Propp suggest that any choice of character functions out of the canonical sequence is valid, if someone is tested but their reaction is not mentioned the result-

Devour	RecoverEaten	victim
Murder	Resuscitation	victim
Abduction	Rescue	victim
Capture	Rescue	victim

Table 3: Character functions pairs that inactivate a given character, identified by that the character that is inactivated plays in the character function.

ing story will not make sense. Or if someone uses a magical agent, it makes sense that the story follows with some achievement that can be understood as a result of that use, like resurrecting someone or curing someone. In a similar fashion, there are character functions that really should not appear contiguously in a tale, such as departing on a journey and returning, or someone being abducted and immediately rescued. Ideally, breaks in the telling of one move to tell about events in another should not happen in between pairs of attracted character functions, but are welcome between pairs of repulsed character functions. To capture this concept we define two sets of pairs of character functions, one of those that attract each other and one of those that repulse each other. The two sets of pairs of character functions defined as related in this way are given in Table 2. The introduction of this kind of connection across character functions may partially address – at the simplest possible computational level – Propp’s injunction that “one must also keep motivations, connections, and other auxiliary elements in mind” (p. 111-112).

Finally, some character functions (like abduction or capture) imply that a certain character becomes inactive until released by a different character function (such as rescue). Events from another move where the corresponding character is active should not be inserted within the span in which the character has been inactivated for the current move. Table 3 lists the set of character functions related in this fashion. As a baseline the events for the moves in a story are interwoven following a greedy algorithm informed by the set of resources described.

Low Level Choices: Form, Cast and Text

The computational steps described to this point represent our proposal for the first decision that Propp considers relevant

in the process of building a tale (“choice of those functions which he omits, or, conversely, which he uses” p. 112). The next step he describes (“choice of the means (form) through which a function is realized.” p. 112) corresponds to selecting, for character functions that are defined at a very abstract level – such as villainy – instantiations of a specific type. In our case, this is covered by having specific villainy-related AoIs that feature plot element classified as villainies – abduction, murder, ...) –combined with the corresponding resolutions – rescue, resurrection, ... The choice between them is made at random. Other types of function, such as Difficult Task or Lack need to be instantiated with a particular form of the corresponding concept. Instances of these specific forms for the concepts are provided by dedicated knowledge resources.

The next step required to emulate Propp’s proposed procedure is to assign story characters to particular slots in functions. Propp explains that the freedom postulated in his description of this point in his procedure is often not exploited in full, with characters often instantiated with a typical villain, a typical donor. . . . Our computational solution for this step has been to create resources that serve on demand options – chosen at random – for typical instantiations of the various roles. The system then queries these resources and uses the returned instances to replace throughout the story the variables representing the roles that the characters play. As observed in the stories mentioned in Propp’s book, heroes are assigned proper names, but other characters are described in terms of some common name (usually an occupation).

The final step described by Propp (“The story teller is free in his choice of linguistic means.” p. 113) involves transcribing the conceptual representation of the story that results from the steps above into a linguistic rendering. This task is carried out by a simple template-based text realizer that relies on templates for the various character functions: A template holds a sentence in English with gaps to be filled with text descriptions of characters or plot-related concepts. Two additional tasks had to be resolved. First, a basic module for referring expression generation (Reiter, 2010) has been added to adapt each occurrence of characters described by common names (first mention relies on an indefinite article, subsequent mentions on a definite one). Second, a number of the character functions involve anaphoric reference to prior events, and this needs to be addressed during realization: for instance, the Connective Incident – in which some character informs the hero about the villainy – or the False Claims – in which the false hero claims for himself the merit of the hero’s actions. In each case the referred character function needs to be retrieved, transcribed into text, and used as a secondary clause to the main sentence. An additional specific module of the realizer addresses this task. When a sentence of text has been produced by these modules, the text for the story is put together by packaging the sentences from different moves into different paragraphs.

Results and Discussion

An example of story generated by the system is presented in Table 4. Move labels have been added to each paragraph

for ease of discussion. We can see here a two-move story, a main move involving the apothecary as the villain and Katoma as the hero, and a secondary move with Katoma as the villain, the apothecary as victim, and Peter as a hero who comes to help the apothecary. Each move sees the conflict between Katoma and the apothecary from a different point of view. This is known as *focalization* (Genette, 1980).

It is important to keep in mind that the system is not intended to generate text for the stories, but rather focuses on their narrative structure. The text given is merely provided as indication of the structure that has been constructed. Such structures may be provided to a neural text generator to be fleshed out into interesting texts, though attention must be paid so that the structure is respected.

The system as described is only a tentative baseline to demonstrate how some of the more elaborate concepts mentioned by Propp might be represented computationally. It constitutes a valuable contribution because it demonstrates that simple computational models can capture complex narratological concepts. Narratologists have in the past observed that computational approaches to story telling fail to address concepts that are considered extremely relevant in the field of narratology (Lönneker et al., 2005). The very simple system described here contemplates the distinction between fabula (the set of moves considered separately) and discourse (the linear sequence that results from interweaving the events for the different moves), simple instances of embedded stories (where someone tells the hero about the villainy or where the false hero claims to have been the protagonist of the hero’s deeds), the possibility of having more than one plot line in the story and focalization on different characters.

Early attempts that adopted Propp’s views as inspiration, often intended to model types of narrative that differ significantly from the tales in Propp’s corpus, and in so doing departed freely from the original framework. Lakoff (1972) proposed a context free grammar (Chomsky, 1957) for the discourse of Russian Fairy tales. His proposal uses phrase structure rules to assign typical characters to variables of character functions, and transformational rules to connect related character functions kidnapping and rescue and to combine more than one move into complex stories. There is no computational implementation associated with this model. Turner (1993) mentions Propp as inspiration of his MINSTREL storytelling system, but he himself confesses that in the final system “‘Vladimir Propp’s intriguing little grammar was nowhere to be seen’ (Chapter 1, page 1). Lang (1997) developed a definite clause grammar that modeled – and could generate – Russian folk tales in the spirit of Afanasyev (1859), but his approach relies more on concepts of story grammar from Rumelhart (1975) and logic programming to model causality and goal-directedness.

Fairclough (2004) in his OPIATE system does include representations of character functions, character roles, and moves, but he adapts these representations heavily to match the requirements of a case-based approach. In doing so, he elaborates character functions into larger units, grouping them into small related sets, redefines the roles into a more generic set of labels, considers a move as the minimum sense

(move-0) An apothecary tries to discover from a seamstress where valuables are hidden. The seamstress unwittingly reveals about where valuables are hidden.

(move-1) Katoma attempts to deceive the apothecary. The apothecary is taken in by the deception. Katoma devours the apothecary. Peter always wanted diamond. Peter recovers the apothecary who had been eaten by Katoma. Peter acquires diamond. Peter fights with Katoma. Peter achieves victory over Katoma.

(move-0) The seamstress is captured by the apothecary. A duchess discovers the seamstress is captured by the apothecary. The duchess tells Katoma that the seamstress is captured by the apothecary. Peter calls to action Katoma. Katoma decides to react to the seamstress is captured by the apothecary. Katoma rescues the seamstress from the apothecary. Katoma fights with the apothecary. Katoma achieves victory over the apothecary. Katoma is rewarded.

Table 4: Example of story generated by the system.

carrying unit to use as a case, and relies on case adaptation to cast the roles in the move. OPIATE being a system for interactive visual narrative, the story is always focalized on the player, who plays the role of the hero. Gervás (2016) operates much closer to Propp’s account in that it creates plots for Russian folk tales as a sequence of character functions. This approach relies on long-range dependencies identified between character functions in the sequence to capture some of connections left underspecified by Propp, but it departs from Propp’s framework in that it creates the fabula for the tales by instantiating the character functions with story actions. To do so, it relies on pre and post conditions for story actions, driven by unification procedure. Specific additional steps are included to assign a cast of characters to the tales and render them as text using a template-based solution. Gervás (2019) proposes a number of extensions to Propp’s formalism, such as grouping character functions into axes of interest (covers long-range dependencies between character functions) and explicitly adding roles to capture restrictions on character instantiation. This approach relies on probability of co-instantiation over existing corpus of examples to assign characters to roles.

Another issue is that Propp’s account is significantly underspecified. Developers extending it to make it operational often did so by relying on specific technologies that brought their own mechanisms both for representation and operation – such as transformational grammar, case based reasoning or planning. Lakoff (1972) transfers a significant amount of the task of modeling the task to the mechanisms of transformational grammar. In his account these mechanisms carry the burden of filling in the details, but their complexity makes it difficult for the reader to grasp intuitively what is happening in each case. Fairclough (2004) brings in techniques for case-based planning and case adaptation. Again, these may indeed fill in the gaps left in Propp’s description, but it is unclear whether the result matches the expectations in terms of what Propp intended. Gervás (2016) proposed explicit representation of long-range dependencies identified between character functions, which are further formalised in Gervás (2019). The proposal in the present paper attempts to bring together all the various extensions of Propp’s account identified as useful at some point.

The decisions not to resort to specific techniques in the modeling process is related to the underlying goal of the research, which is not to generate stories but to understand how it is that stories come about, and how they are structured. Whereas specific techniques do provide valuable

working solutions, they come at the cost of clouding the principles underlying the process. Both transformational grammar and case-based reasoning work to generate stories, but it is a daunting challenge to understand what the two may have in common that serves as description of how narrative works.

A possible criticism is that the system as described here introduces a number of concepts – such as attraction/repulsion between characters functions or characters rendered inactive – that were not considered by Propp. This has been necessary to cover specific points where Propp was not informative about the details underlying some of his descriptions. On this point, we take him at his word when he says: “In the interest of a shorter and more vivid presentation it became necessary to renounce many things which would be relevant for a specialist” (...) “The experienced reader will be able to complete the outline himself.” (Vladimir Propp (1928), Author’s foreword, 15 July 1927). Any errors made in that process remain, of course, our own.

Conclusions

Propp’s Morphology of the Folk Tale has been revisited as inspiration for a computational generator of narrative structures for stories. Extensions of the framework considered in prior attempts have been compiled into a working system that explicitly represents all the steps involved. The initial outcomes show potential for capturing together in a single system of rather simple architecture some complex narratological concepts that in the past have required different specific story generators, each focused on a particular AI technique that favoured the phenomenon chosen as focus.

Acknowledgments

This paper has been partially supported by 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), project DARK NITE: Dialogue Agents Relying on Knowledge-Neural hybrids for Interactive Training Environments, Grant No. PID2023-146308OB-I00 (Spanish Ministry of Science and Innovation) and project ADARVE (Análisis de Datos de Realidad Virtual para Emergencias Radiológicas) funded by the Spanish Consejo de Seguridad Nuclear (CSN).

References

- Abbott, H. 2008. *The Cambridge Introduction to Narrative*. Cambridge Introductions to Literature. Cambridge University Press.
- Afanasyev, A. N. 1859. *National Russian Tales (in Russian)*.
- Chomsky, N. 1957. *Syntactic structures*. Mouton de Gruyter.
- Fairclough, C. 2004. *Story Games And The Opiate System: Using Case Based Planning For Structuring Plots With An Expert Story Director Agent And Enacting Them In A Socially Simulated Game World*. Ph.D. Dissertation, Trinity College Dublin.
- Fay, M. P. 2014. *Driving story generation with learnable character models*. Ph.D. Dissertation, Massachusetts Institute of Technology.
- Genette, G. 1980. *Narrative discourse : an essay in method*. Cornell University Press.
- Gervás, P.; Concepción, E.; and 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*. Madrid, Spain: Springer.
- Gervás, P. 2016. Computational drafting of plot structures for russian folk tales. *Cognitive computation* 8(2):187–203.
- Gervás, P. 2019. Generating a search space of acceptable narrative plots. In *10th International Conference on Computational Creativity (ICCC 2019)*.
- Herman, D.; Jahn, M.; and Ryan, M. 2010. *Routledge Encyclopedia of Narrative Theory*. Taylor & Francis.
- Lakoff, G. 1972. Structural complexity in fairy tales. *The Study of Man* 1:128–150.
- Lang, R. R. 1997. *A formal model for simple narratives*. Ph.D. Dissertation, Tulane University.
- Lönneker, B.; Meister, J. C.; Gervás, P.; Peinado, F.; and Mateas, M. 2005. Story generators: Models and approaches for the generation of literary artefacts. In *ACH/ALLC 2005 Conference Abstracts. Proceedings of the 17th Joint International Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing*, 15–18.
- Propp, V. 1928. *Morphology of the Folk Tale*. Akademija, Leningrad.
- Reiter, E. 2010. Natural language generation. *The handbook of computational linguistics and natural language processing* 574–598.
- Rumelhart, D. E. 1975. Notes on a schema for stories. In *Representation and understanding*. Elsevier. 211–236.
- Turner, S. R. 1993. *Minstrel: a computer model of creativity and storytelling*. Ph.D. Dissertation, University of California at Los Angeles, Los Angeles, CA, USA.