

Semantic annotation of narrative media objects

Vincenzo Lombardo · Rossana Damiano

Published online: 26 May 2011
© Springer Science+Business Media, LLC 2011

Abstract This paper addresses the annotation of the narrative features of media objects. Based on a relevant narratological and computational background, we introduce an ontology-based model called Drammar, an annotation schema for the narrative features of media objects based on Drammar and a software tool, Cinematic, for annotating these objects and validating the annotation. Annotated media objects can also be automatically edited into sequences, with the twofold goal of testing the validity of the annotation—through the reconstruction of the baseline sequence—and exploring the possibility of alternative sequences. The software tool encodes both the narrative model and the annotation itself in ontological format, and relies on external ontologies for representing world knowledge and limit the arbitrariness of the annotation. The paper opens the way to the design of a general annotation schema for narrative multimedia with the long-term goal of building large corpora of annotated video material and of bridging the gap between the low-level signal analysis and the high-level semantic representation of the narrative content of the media objects. Finally, the paper illustrates a few projects elaborated with the Drammar annotation and the Cinematic tool, with purposes of artistic research and cross-media analysis, that provide an empirical validation of the annotation process.

Keywords Narrative media object · Multimedia corpus annotation · Automatic video editing

V. Lombardo · R. Damiano (✉)
CIRMA and Dipartimento di Informatica, Università di Torino, C.so Svizzera 185, Torino, Italy
e-mail: rossana@di.unito.it

V. Lombardo
e-mail: vincenzo@di.unito.it

1 Introduction

The use of narratives in communication is ubiquitous across cultures and is widespread in multimedia, especially in linear audiovisuals, such as television news, soap operas, advertisements, and, more recently, in non linear, interactive storytelling applications. A recent article by a writer in a general newspaper [Paul Schrader, *The Guardian*, Friday 19 June 2009, <http://www.guardian.co.uk/film/2009/jun/19/paul-schrader-reality-tv-big-brother>] crunches some hypothetical numbers in estimating that an average 30-year-old person today has seen approximately 35,000 h of audiovisual narrative.

Narrative media introduce real and fictional worlds by structuring events and characters into stories. The codes and conventions that regulate them convey further elements that enrich their pure meaning. The annotation of narrative features in multimedia data is useful for several reasons.

- Media production. Traditional media production undergoes several complex tasks, handled by different professionals (writer, director, cinematographer, editor, actors, ...). The annotation of the narrative features can be started during the authoring process and propagated through the production phases to make all the team aware of what is going on (common tools used for this goal are storyboards and screenplay breakdowns, that are not machine readable).
- Search/Retrieval. The annotation of narrative features is important in the search/retrieval of individual media assets, that are component parts of some larger object. Again, this is relevant for media production itself (for example, for the editor to retrieve some footage shot), but also for content aggregators (for example, in the compilation of programme schedules or trailers).
- Access to cultural heritage archives. Narrative-based access can support institutions in providing services that are of interest to general public and scholars, making the role of the archives more active in the society.
- Analysis of narrative media objects. The analysis of narrative media objects is part of the training in film and media studies. Annotated narratives can support this training, especially through dedicated IT tools for the methodology of analysis.
- Real time editing of video. Annotation of narrative features is important for the implementation of automatic real-time editing of video segments.

Notwithstanding the advantages of having narrative metadata, there exists no neutral approach to the annotation of narrative media objects. In fact, though a number of annotations were devised for different projects (see the related work section below), there has never been an effort towards a common annotation of the narrative features. In this paper, relying on a background of narratological and drama studies, we introduce an ontology-based model called *Drammar*, an annotation schema for the narrative features of media objects based on *Drammar* and a software tool, *Cinematic*, for annotating these objects and validating the annotation. Though retaining the core assumptions of most narratological theories, *Drammar* provides a theory *a-specific* description of the narrative content of media objects. Such a model is aimed at building large corpora of narrative media (with no specific constraint on format), that can be employed to build and test specific models and applications in the areas mentioned above.

The basic elements of Drammar are the segmentation of a media object into several objects that can work as narrative units (a notion borrowed from media production), the character-centered representation of the story (mainly derived from drama studies), and the representation of causal relations among units, according to a line of research that, in artificial intelligence, proceeds from situation calculus to agent theories. So, for each segment, the annotation schema describes what actions are displayed in it, the characters' motivations for executing those actions, and how they affect the state of the world, and the characters' beliefs and emotions. The annotation relies on formal ontologies to categorize actions and emotions. The graphical interface of the annotation tool enforces the annotation schema through the display of templates and the navigation of the ontologies for filling the fields in the templates. The tool includes a function that, given the annotated units, automatically edits the objects to reconstruct the original sequence and to discover alternative ways of editing. Such editing function provides a first, baseline validation of the annotation, provided that the annotation contains enough information to allow the reconstruction of the baseline sequence.

Both the Drammar model and the Cinematic tool are not intended as instruments for the creation of stories, but merely for the annotation of story elements in media objects. This opens the way to the construction of large corpora of annotated media objects, as the preliminary step to filling the semantic gap from the low-level intrinsic features to the high-level story-based description of narrative multimedia, or, stated in other words, to connect the story elements devised by the author and the representation of data in media objects.

The paper is organized as follows. The next section introduces some literature that is relevant for the annotation of the narrative features (Section 2). Then, in Section 3, we provide the background and describe the Drammar model and the annotation schema, applied to an example. In the fourth section (Section 4) we illustrate the software tool Cinematic for annotating the media objects and validating the annotation through an editing function. Finally, we present some applicative projects based on Drammar and Cinematic (Section 5) and discuss the results (Section 6).

2 Related work

This section reviews a number of works and approaches to narrative description of media objects, with particular attention to the area of video production. We cite both modular methods and complete systems for editing videos. The structure of the survey accounts for the influence each research initiative had on the design of the Drammar model and the annotation schema of Cinematic (described in Section 3).

2.1 Narrative units

From the point of view of media production, a narrative object consists of a number of segments (or sub-objects), collected in a repository, from where they are retrieved for editing purposes. Each unit has a role in the development of the object it belongs to, and such a role is acknowledged at each step of production, from authoring and directing to editing. For example, the segments produced by film shooting (the

so-called shots) are indexed by the production logs, and the video editor selects such segments from the repository to order them in a sequence [16]. It is worth noticing that the span of a film shot usually does not match a narrative unit, because production plans are geared to optimize location usage and cast availability, while units having a narrative relevance usually consist of several shots [38]. The work of the editor is largely supported by non-linear editing software programs, such as Avid Media Composer [<http://www.avid.com/US/products/family/Media-Composer>, 2011] or Final Cut Studio [<http://www.apple.com/it/finalcutstudio>, 2011], that allow for free annotation of the segments.

The evolution of annotation is towards a notion of video segments as “standard” computational objects [17], oriented to parameterization and re-use. The increased IT contribution to media production through the use of metadata should produce benefits in good quality editing and the reduction of the ratio between the shot footage and the actual use in a video project (about 10:1 according to [17]). A concrete idea for implementing parameterization and re-use is to work with templates that encompass multiple versions of some asset (e.g., personalized commercials with audience faces) and are annotated by leaving their content metadata somewhat underspecified for later production; then, metadata are updated automatically when video segments are edited, by mapping typical editing operations to data base operations such as merge (for segment combining) and projection (for segment extraction), as suggested by [33]; these operations guarantee the consistency of the new metadata with respect to the original ones and to the annotation format. The idea of elementary units as the base for the annotation of narrative media objects is reflected in the Drammar model, described in Section 3; the annotation includes both the content description of the unit and its role in the whole narrative object.

2.2 Formal annotation of units

The formal annotation of units in computational systems ranges from keyword systems to predicate logic, with pioneering works that have laid the bases for the automatic manipulation of video objects. For example, the Strata system [56] employs keywords that are relevant over a frame span of a video (implicit segments). Keyword association can be partially overlapping and this produces a layered annotation of the video, with each layer representing some association among the segments. The LIVE (Lancaster Intelligent Video Editor) system [5] is an automatic editor that operates over a data base of video segments with a set of transformation rules that encode film editing rules. Segments are manually annotated with Prolog clauses. The transformation (or “fragment construction”) rules fire in response to user queries that specify the meaning of the desired video, mapping it onto the content of the footage available in the database.

More recently, the emphasis has shifted towards a standardization of the annotations (also abstracting from video to general multimedia objects) as well as the application to practical, real world, cases, especially news and sport reports. MPEG-7 has posed an emphasis on providing a set of descriptors of audio-visual content [40]. The description defines the syntax and the semantics of some features of the media unit (ranging from low-level to high-level, e.g., image color, camera motion, characters in the video, ...). Given the difficulty of integrating MPEG-7 with semantic technologies, some efforts have been made to bridge the gap between

signal-oriented content description and multimedia semantics (starting from COMM ontology [1], as surveyed by [15]).

An annotation method with practical applications is the A4SM (Authoring System for Syntactic, Semantic and Semiotic Modelling) framework [41], for the semi-automated annotation of audiovisual objects in a news production environment. A4SM is based on a semantic network for data storage and management: as production goes on, semantic structures evolve. This facilitates the dynamic use of audiovisual material, by establishing a multi-layered spatio-temporal perspective, and enables the semantic connection between segments. In the domain of news, the authors show a practical example where a basic set of 18 syntactic and semantic descriptors are used to annotate video material in real time, then used for retrieval and editing.

Both the theoretical premises and the annotation principles proposed here are encompassed by the Drammar model: for narrative features, we provide a basic set of descriptors (based on characters' mental states and observable behaviors) and we interpret the video editing process as an evolution of semantic structures inspired by a well known paradigm in computational systems, namely the situation calculus [29]. With respect to MPEG-7 related approaches, the model still lacks the connections to the low-level features of the audiovisual material: though a number of issues are immediately accountable even in the narrative domain (take, e.g., the case of luminance and color properties that typically change at unit boundaries), we have left to a future work the integration of low-level descriptors in the current annotation.

2.3 Narrative computational systems

Together with annotation methods and editing programs, there are a number of complete systems developed with artistic goals or specific genres in mind.

The application Soft Cinema [<http://softcinema.net>, 2005–2011] hybridizes the paradigms of cinema and human-computer interaction, and enhances the new media practices in video production. In Soft Cinema, media elements are stored in a large database, and the narrative is generated through the selection and editing of clips (a 'database narrative' according to [34]). Each clip is assigned keywords that describe both its "content" (geographical location, presence of people in the scene, ...), and its "formal" properties (i.e., dominant color, contrast, camera motion). Soft Cinema assembles a video track by selecting clips, using different systems of rules (e.g., color neighborhood, type of motion, content, etc.). The combination of the two paradigms is realized by associating the temporal editing of clips to the spatial composition of images in one screen, an idea borrowed from GUI's, employing windows of differing proportions and sizes, re-used here for aesthetic reasons. During the playback, when the program assembles movies in real time, individual clips are assigned various sections on a partitioned screen, at various resolutions.

A genre that received much attention in annotation and real-time editing is the documentary, likely because of its nature of loose screenwriting, the authors' goal to show more material than what can be reasonably accommodated in a standard video duration, and the need to take into account audience's reaction along the presentation. Korsakow [<http://korsakow.org/>, 2000–2011], Vox Populi [3], and Terminal Time [37] are interactive systems for editing documentaries from video segments in a data base. Terminal Time is a project for the realization of

ideologically-biased documentaries in response to audience feedback during the projection in a movie theater. In response to the intensity of the applause, Terminal Time creates historical narratives that attempt to mirror and often exaggerate the audience's expressed biases.

Thematic sequences are the output of the Auteur system [39]. Auteur operates from arbitrarily annotated video material, implemented on the exemplar theme of humor. The annotation approach, based upon multiple overlapping intervals (such as Strata above), allows for the implementation of complex editing rules. Auteur embodies strategies for the generation of film sequences and the presentation of humorous concepts.

In the field of interactive TV, the ShapeShifting TV [61] is a system for generating interactive TV narrative that encompasses both the annotation of the media items, and the formal description of the narrative structure through the Narrative Structure Language. This system, employed for the production of interactive tv movies (“Accidental Lovers”, 2006) and documentaries (“A Golden Age”, 2007), is posited between the use of templates and the declaration of editing rules, to offer larger expressive possibilities to the author.

The Narrative Abstract Model [27], differently from the works above, implements the analysis of narrative videos for summarization purposes. This model splits the video into segments, with algorithms based on stylistic elements (such as shot and scene detectors), informed by a manually built story structure. Then, based on this representation, it computes video abstracts for TV review services of soap operas. The soap-opera episode is represented as a graph of interconnected narrative nodes, with an evaluation of the degree to which each interconnection realizes a story progress; the intelligent component of the program detects the sequence of narrative units that mostly contribute to the progress of the story (Degree of [story] Progression—DoP—measure), generates a graph that encodes all the possible sequences of connected narrative elements, and selects the sequence (path) with the maximum DoP to propose an abstract.

These complete systems go beyond the goals of the Drammar model, by accounting for the authorial goals and motivations that lie behind an audiovisual work. Providing a control over the visual aspects of a video, the rhetorical structure of a documentary, the message that emerges from the unit sequentialization, and the compliance with a (humorous) theme require the representation of an operational knowledge that exceeds the pure description of the narrative form. Where the Drammar model can provide a basis for the interoperability among several descriptions, such specific applications put an emphasis over the control system for creating the audiovisual works.

2.4 Interactive storytelling systems

The field of interactive storytelling, that generates narrative audiovisuals on-the-fly taking into account the input from the users, provides some useful inspiration for the annotation of narrative media objects. There are two broad categories of interactive storytelling systems: story-based [51, 55, 60] and character-based [35, 46].

Story-based systems are characterized by centralized architectures, in which the system is driven by the unifying principle of a story. Story-based architectures tend to incorporate sophisticated story models to account for the structural aspects of

narration, ranging from semiotic structuralism [25, 45, 59] to cognitive models of story understanding [58]. This knowledge can be encoded in the form of logical rules, as in the DEFACTO [55] and the IDtension [59] systems, where they select characters' actions to form a coherent plot, or in the form of planning operators [51], which combine into a sequence of incidents from the initial state to the final state.

Character-based systems rely on the autonomous behaviour of characters, and their interaction, to create situations, which are then interpreted as emergent narrative structures [57]. This approach has been encouraged by the availability of conceptual and practical tools that implement the characters' deliberation through the notion of intelligent agent. For example, the "Friends" system [8], an interactive version of a well-known TV series, characters are committed to specific goals (such as, e.g., seducing another character) and devise and execute plans to achieve them. The Façade system [36], an interactive drama performance, adopts a mixed strategy by encoding multi-agent plans for the characters (included the user, seen as a character) and controlling the story advancement through numerical parameters that represent the so-called "story value". A related work of this category, although with a mild narrative flavor, is the Dramatour system [11], that realizes a guide to a historical site on a mobile device. An animated character, an anthropomorphized spider, tells a story through the composition of a sequence of narrative units on-the-fly; the selection of the next unit takes into account the story progression, the location of the visitor in the site, and the character's communicative intentions; units are annotated according to these issues, as well as with visual properties for continuous editing. Lombardo et al. [32] describe the Dramatour system, and the use of metadata, with respect to the model of the canonical processes of semantically annotated media production [24].

The Drammar model draws inspiration from both structural descriptions of story-based systems and behavioral accounts of character-based systems. The logs of interactive systems can be described in Drammar terms both as a sequence of transitions between world states, determined by the user's choices, and as a sequence of characters-enacted actions.

2.5 Conclusions

In general, all the approaches reviewed here generally do not exhibit an explicit annotation for the narrative aspects of video segments. The Narrative Abstract Model is limited to the interconnection between segments, and does not describe the content of segments in semantic terms. Moreover, the annotations used in the applications described above do not exhibit an effort for standardizing the narrative aspects, and are geared to the purposes of the specific application.

3 The Drammar model

In this section, we illustrate the Drammar model and introduce the narrative annotation schema. We also provide a short description of the underlying theoretical background, more details are in [31]. The model and the annotation schema have been designed for any media object that has a temporal evolution and whose content can be defined as narrative at large, e.g., media objects featuring characters who

intentionally perform actions of some type. The model does not cover the relation among the narrative content, such as characters and actions, and their physical expression in the media objects, such as sentences in text or regions in moving images, leaving the mapping to narrative feature to low-level features to future work. So, the annotation may apply indifferently to a short movie, a novel, or a play session of a videogame. The model we propose is not geared to any specific application, unlike most of the works reviewed above; our effort was to identify those issues that properly characterize a narrative media object independently on the application that can produce or embed it.

By considering both the related work and the literature on narratology, three main elements emerge as relevant to describe the content and function of a narrative media object, namely story structure, character, and world state.

- First, from the notion of narrative unit (Sections 2.1 and 2.2), we acknowledge the fact that narrative media objects are modular, i.e. that they can be decomposed into sequences of smaller objects, each characterized by an inner homogeneity (and discontinuity with adjacent ones) related to its narrative function within the story. In the annotation, the *story structure* component describes the object as its position in the succession of objects and as part of larger objects. By doing so, it also takes a structuralist stance on story analysis, following a well known paradigm in narratology [23].
- Second, as exemplified by the architecture of storytelling systems (Section 2.3), the annotation must account for the paramount role of *characters* in story. Characters are the medium through which the story is conveyed to the audience; by displaying rational intentions and emotions, they provide meaning to the story and mediate the audience participation through the mechanism of identification. These two components, story structure and characters, account for the ubiquitous claim of literary studies (e.g., [48]), that storytelling develops along two orthogonal axes, characters and plot. This claim is also empirically confirmed by the two major approaches devised by storytelling systems, namely story-based and character-based.
- Finally, the third component of the description of narrative media objects accounts for the role of causation to determine the feasibility of story [52]. The *world state* component describes the sequencing relations that an object can satisfy. In particular, given that a story forms a causal chain of incidents that bring about changes in the story world, the world state specifies the conditions at which an object can participate in a sequence, namely the preconditions that must hold in the world for the object to be displayed (or for the characters' actions to be executed), and the effects that hold after the object display (i.e. the effects of the characters' actions).

Clearly, the three components are not unrelated, but the description schema is aimed at keeping them distinct, to manage the complexity of narrative works. In particular, while the first component establishes a formal template for story structure, allowing the annotation to describe the place/role of an object in the story, the latter component states what are conditions the object must fulfill to be actually inserted in the structure. These conditions, in turn, partially refer to the mental states (e.g. beliefs) that are acknowledged by the description of characters.

In the following, we illustrate the three components in turn. For clarifications, we refer to the story example in Fig. 1.

3.1 Story structure

Narrative units provide a way to discretize the advancements in the story. Abstracting from the conventions of specific media, we define a *narrative unit* as any media object that contributes to the story advancement by bringing about significant changes in the story world, i.e., changes that affect the characters' goals and intentions. In Drammar, a narrative unit can be recursively expanded into a number of children units, forming the *plot tree*, such that the story advancement provided by a unit is part of the story advancement provided by its parent unit. The expansion of narrative units into smaller narrative units stops when, at the basic recursion, narrative units are expanded into the actual sequence of elementary narrative units (the leaves of the plot tree).

The annotation schema represents the plot tree through the parent notation: for each narrative unit, it indicates the parent unit. There are no restrictions on the number of layers and the number of units at each level. The leaves of the plot tree constitute the linear sequence of story advancements and are enriched with the annotation of the third component, namely the world state. The story structure of the example in Fig. 1 is represented in Fig. 2.

The hierarchical structure of the plot is acknowledged by literary studies: in theatre, for example, narrative units are called acts, sequences and scenes, in descending order of size [28]. From the point of view of the annotation, the advantage of representing the plot as a tree instead of a linear sequence of elementary units is that we can abstract the the role of some units into a larger unit. This representation can

World War II. 1944, Italy.

[1] *Two partisans, Tenebra, the commander, and Echo, his subordinate and guide, are on a mission in the North of Italy.*

[1.1] The mission begins and they leave. [1.1.1] Echo says goodbye to his girlfriend, Agnese.

[1.1.2] Days after, while they walk in the woods, Tenebra reveals that they have to meet an Allied officer in another county. [1.1.3] During the walk, they talk about their life as civilians.

[1.1.4] They make a halt and decide to stop at Agnese's farm to get food and water. [1.1.5] As they resume the walk, Tenebra and Echo have a quarrel about Echo's reliability as a guide.

[1.1.6] They start to talk again about their previous life.

[1.2] They get to the farm. [1.2.1] At the farm, they find that Agnese is being tortured by a brigade of fascists.

[2] *The fight with the fascists.*

[2.1] Echo and Tenebra split up. [2.1.1] While Tenebra decides to stick with the mission, Echo decides to rescue Agnese.

[2.2] Echo attacks. [2.2.1] With a strategem, he kills most of the brigade, except the officer, who asks him to surrender in exchange for Agnese's life.

[2.3] The rescue. [2.3.1] Tenebra decides to go back. [2.3.2] Unnoticed, Tenebra reaches the officer from behind and shoots him.

[3] *Echo and Tenebra resume the mission.*

[3.1] Echo and Tenebra walk away from the farm. [3.1.1] Walking through a field of corn, Echo and Tenebra abandon the fight scene.

Fig. 1 The synopsis of the example story, the short film "1944", directed by A. Scippa, Italy, 2007. The structure of the plot is marked by the numbers in *square brackets*

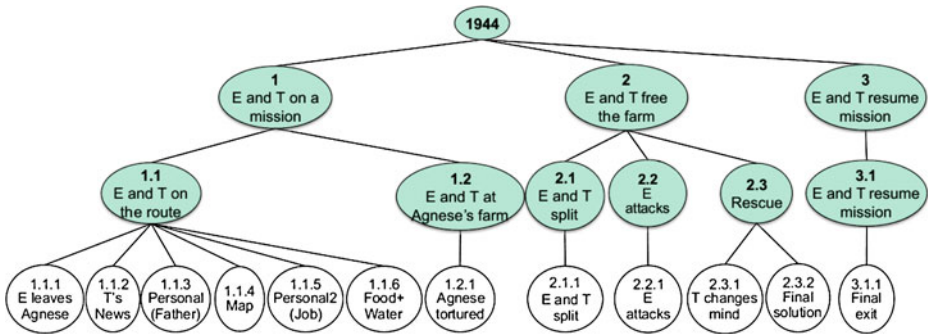


Fig. 2 The plot tree of the example story (Fig. 1)

describe works at different levels of granularity, accommodating complex works such as novels or feature films in annotations of increasing complexity. The idea of a hierarchical subdivision allows for the explicit representation of how the authorial goal (a larger unit) is realized through a number of simpler units. According to drama writing literature (e.g. [19, 38, 47]), authors typically relate a general premise, a situation, or a protagonist's achievement to a number of constituent units. In the example story, the unit 2.3 consists of the rescuing action to free Agnese. This is realized through two sub-units: in the first, Tenebra changes his mind with respect to continuing the mission alone and decides to help Echo in rescuing Agnese; in the second, the final solution of the rescuing action, Tenebra kills the fascist officer who was threatening Echo and holding Agnese hostage.

The annotation of the plot tree in the narratives generated by computational systems deserves further comments. In character-based storytelling systems, the notion of unit is only emergent, since the story is generated from the execution of the characters' actions. So, though narrative units are not explicitly represented within the knowledge manipulated by those systems, it is possible to identify boundaries in story advancement within the logs of the interactive sessions. As for the layered structure, although the computational systems surveyed in Section 2 do not explicitly represent it, the production (or the automatic generation) of media objects induces a type of discretization that is orthogonal to the narrative-based segmentation. For example, the units identified by the automatic editing projects (e.g., the LIVE system) refer in most cases to shooting units (and not narrative units, see above).

3.2 Character

Characters are the medium through which the story is conveyed to the audience. As such, they provide a powerful instrument of identification for the audience [7], leading to what Coleridge termed the "suspension of disbelief", i.e. the immersive experience of the story world [9]. Characters have received much attention in the literature, from literary structuralism [22, 49] to aesthetics [7, 20]. Starting from [6], characters also play a crucial role in the scriptwriting, where authors usually rely on a descriptive schema usually referred to as "the characters' bible" [38, 54]. The

identification of character relies, to a large degree, on their capability to display a believable rational and emotional behavior. Since the pioneering work of Schank and Abelson [53], in fact, cognitive experiments have shown that the audience ascribes intentions to story characters as part of the process of story understanding. This in line with Dennett's 'intentional stance' [18] as an intrinsic part of human cognition.

In order to account for such findings, the second component of the Drammar model refers to the characters. A number of formal frameworks were devised for defining characters, starting from their non-dramatic counterparts, called agents. A well-established formal framework for intelligent rational agents is the BDI model [4]: according to this model, the behavior of an agent is entirely determined by its mental state, which consists of Beliefs about the world, Desires (to be translated into goals), and Intentions (i.e. commitment to action plans aimed at achieving goals). In computational storytelling, this model has proven effective both as an operational solution for creating believable characters [42] and as a theoretical framework for analyzing narratives [20].

However, the BDI model, by itself, is not sufficient to capture the essentials of characters, so Drammar integrates it with emotions and moral values. As recently pointed out in [12, 44], moral and emotional aspects must be accommodated into a rationality-inspired model in order to account for dramatic characters. Recognizably subjective [62], values have emerged as a motivating component in the characters' cognitive structures, tightly linked with goals. Typically, narrative plots are organized so as to put at stake values of increasing importance, until a climax of characters' struggling; after the climax, value-based conflicts tend towards a resolution. For example, in 007 movies, the hero Bond must defeat an arch-villain who threatens the human kind; as he devises a clever plan to neutralize his antagonist, the value at stake, initially set to a generic "security of the country", becomes increasingly higher up to include his own life, as an effect of the counter attacks of the antagonist. A model of how characters modify their goals (and intentions) in response to values at stake is described in [13].

On the account of the characters' emotions, especially in conjunction with the BDI model, the cognitive framework of emotions of Ortony et al. [43] has received much attention since the pioneering work of [2]. According to this framework, emotions such as hope, fear, or shame stem from the appraisal of one's and others' actions, based on a combination of self-interest (achievement of one's own goals) and moral evaluation. The integration of emotions in the annotation of characters follows the methodology described by [14].

The annotators' arbitrariness for the terms that describe characters' goals and actions, as well as their emotions, must be limited, relying on external knowledge sources, to improve effectiveness of annotation and re-use of the annotated objects. As for actions and events, we currently adopt the ontology of processes included in the IEEE Standard Upper Merged Ontology [<http://www.ontologyportal.org/>, 2003–2011], SUMO. SUMO is a formal ontology, written in KIF (Knowledge Interchange Format) language; it contains the Upper Ontology Itself, a mid-level ontology (MId Level Ontology, or MILO), and a set of domain ontologies, ranging from Transportation to Engineering and Communications. The set of all ontologies, combined, contains about 20,000 terms and 70,000 axioms when all domain ontologies are combined. The process ontology, in particular, contains the axiomatization describing intentional and non-intentional processes, necessary to describe narrative

actions and events, as also shown by [10]. Emotions are described with reference to the Ortony et al. [43] model mentioned above (OCC model), which was originally expressed as a taxonomy of emotion types.

The annotation of characters' actions is particularly relevant, since actions constitute a descriptive feature of primary importance when referring to plot incidents. So, their encoding is relevant for searching stories and retrieving story units. In the annotation of a story, actions (and goals as their originating, motivational source) have different levels of granularity across the segmentation levels of the plot tree. When a unit that represents the entire story is described in terms of the characters' actions, annotations tend to refer to abstract actions (such as "saving the mankind" for Bond's movies or, in our example film, "going on a mission"). Clearly, these labels represent actions that cannot be directly executed as such, but only as sequences of more specific actions (in our example, "getting food", "walking to the meeting", etc.). So, the media objects that are contained in a larger object are likely to display the execution of these more specific actions, and the level of detail increases with the granularity of the object—that may eventually display only one non decomposable action (e.g., the mere act of "firing a gun"). Computational ontologies model the relationship between more general and more specific actions as subsumption relations (formally sustained by the use of description logics), thus providing a sound basis for the annotation of units.

In a bottom-up perspective, the annotation of actions in lower units is necessary for the retrieval of media objects based on the actions performed by the characters (such as "firing a gun"), and on the actual achievement of the character's goal (e.g., "killing someone").

The descriptors of the annotation schema feature characters' beliefs, goals, values, actions, and emotions, in a structured list. Though these entities should be automatically connected through the ontological knowledge, the current development of the annotation tool leaves such consistency check to the annotator.

3.3 World state

The third component of Drammar, the world state, models the dynamics of the world along the story development. This component accounts for the narratologists' claim that plot incidents must be causally connected to each other as a very precondition for story construction [52]. It is orthogonal with respect to the story structure, since it develops along the horizontal axis of the plot tree, and provides a consistency check over the behaviors of characters, so extending the second component.

Drammar defines two world state components for a unit: the preconditions that must hold for the unit to be displayed and the effects that hold after the unit display. So, e.g., in unit 2.2.1 of the example story, one precondition for Echo's attack to the farm is that Echo realized that Agnese was being tortured by the fascists, while one effect of his attack is that Agnese becomes hostage of the fascist officer.

The necessity of including the description of the world state goes beyond the account of the story dynamics. In fact, the character component only ensures the consistency of the characters' behavior (as grasped by the BDI model), but does not enforce the overall consistency of the plot, which emerges from the interplay of the actions pursued by all the characters in the plot's elementary units and the events that occur in them. So, the world state component of the model accounts for the

plot consistency. No matter, in fact, how the characters' goals change along the story, still the sequence of actions they perform in the elementary units—interwoven with the events that occur in these units—must give rise to a causally sound sequence, avoiding meaning gaps.

In order to link the preconditions and the effects to the characters' intentions, so that the character and the world state components are not unrelated in the model, Drammar assumes that the preconditions include the characters' mental state (beliefs and goals) and the actual facts about the world. So, for example, both Echo's goal to free Agnese and the fact that he really is at the farm are preconditions to his action of attacking the fascists. Following the BDI model, actions are motivated by the goals and beliefs characters hold before the unit occurs in the story, and their execution affects, as a result, both the state of the world (through practical effects) and the characters' mental state. For example, the action of attacking the fascists is motivated by Echo's goal of rescuing Agnese and provokes the death of part of the brigade. After this action, the goal of rescuing Agnese is not removed from Echo's mind, since she is even more in danger than before, but Echo's beliefs are updated with the new information about the death of the other components of the brigade.

The annotation of the state of the world before and after a certain unit accounts for the audience's point of view. If the characters' beliefs about the state of the world preceding and following the units differ from the audience's, they must be encoded in the characters' subjective beliefs, in order to model the role of "dramatic irony", i.e. the audience knowing more than the characters. This annotation is particularly important for the editing process: an automatic system may elide some parts of the action sequences because they are not necessary for the audience's understanding and can be inferred from the context.

Finally, the annotation of the state of the world in narrative units is the bridge between the annotation of media objects and the validation of the annotation through the automatic editing function (see below). As we will see, a baseline method for validating the annotation is the reconstruction of the original video sequence through the automatic editing of the narrative units, by taking into account the evolution of the world state, as prescribed by the preconditions and the effects of the units. Starting from an initial world state, the automatic editor randomly selects one unit such that the preconditions of the unit match the world state and updates the world state with the effects of the unit. The editing process continues until no more units can be selected or the maximum number of units has been reached. In the Section 4, after having introduced the annotation process, we present the editing algorithm and a working example.

3.4 The annotation schema

The Drammar model is translated into an annotation schema, with a structured list of descriptors that represent the narrative features discussed in the previous sections. For a number of descriptors (action, goal, character, unit, emotions), we refer to the concepts of an ontology that describes the complex relations over such concepts. However, the Cinematic tool, described below, does not currently exploit such representation and does not provide any reasoning service. Another descriptor (link to media object) refers to some media file. The rest of the descriptors are

arbitrary alphanumeric strings. In all cases, name and function consistency checks are left to the annotator. The annotation schema is summarized in Table 1.

STORY STRUCTURE

- **Unit Id:** a unique identifier (a string);
- **Level:** an integer, the level of the unit in the plot hierarchy (0 for the root, 1 for the first (“act”) level, 2 for the second (“sequence”) level, 3 for the third (“scene”) level, 4, 5, etc.
- **Children Narrative Units:** a list of lower level narrative unit instances (an array of Unit Id).
- **Description:** a string containing the natural language description of the content of the drama unit.
- **Link:** the link to the actual media object.

CHARACTERS

- **A set of characters:**
 - **Character:** a character instance. The instance of a character is characterized by its static properties (name, profession, age, appearance, scale of values, etc. as defined in the character model.)
 - **Character’s belief:** a string containing a set of ground formulae. They encode only the character’s subjective beliefs, as long as they are not included in, or are inconsistent with, the *world state* component. If not specified, character’s beliefs are assumed to be consistent with the world state.
 - **Character’s goal:** the instance of an action in the ontology of action types. It represents the character’s active goal (a goal of actional type).

Table 1 The descriptors in the annotation schema of Cinematic

Descriptor name	Purpose	Drammar concept	Type
Unit ID	The unit identifier	Unit	String
Level	The level of unit in the plot hierarchy	–	Integer
Children	A subunit list	ChildrenUnits	Ordered list of URI’s
Description	Textual description of the unit content	Description	String
Link	The link to the media object	Link	URI
Character	A character instance	Character	URI
Character.Belief	Character’s beliefs	Believes	String
Character.Goal	Character’s goals	Intends	String
Character.Action	The action performed by the character	Does	URI
Character.Values	Character’s values at stake	HasValue	String
Character.Emotions	Character’s emotional state	Feels	URI
World.Preconditions	The preconditions of the unit	Preconditions	String
World.Effects	The effects of the unit	Effects	String

- **Goal achievement:** a boolean. It expresses whether the goal is achieved or not in the drama unit. The notion of goal achievement is needed to model the dynamics of goals and the activation of emotions.
- **Character’s action:** an action instance. May correspond to the character’s goal (an action itself) or be an instance of a more specific action.
- **Values at stake:** a set of value instances, with an integer indicating their priority. They represent the character’s values at stake, associated with the character’s instance.
- **Emotions:** a set of emotion instances, i.e., the character’s current emotions in the drama unit.

WORLD STATE

- **Preconditions:** a set of ground formulae describing the world state when the unit begins.
- **Effects:** a set of ground formulae describing the world state when the unit ends.

In order to illustrate how the model applies to actual media objects, we resort to the example story (Fig. 1). As already mentioned, action types in characters’ actions and goals refer to the IEEE Standard Upper Merged Ontology; the descriptors concerning the dramatic elements, namely characters and units, refer to a purposely built ontology, called the Drammar ontology. The plot structure of the example story is represented in Fig. 2.

The opening scene of “1944” shows Agnese and Echo saying goodbye, as annotated in the *Description* field of the schema. The *Unit Id* and *Level* fields encode the position of the unit in the plot hierarchy. As for the annotation of the characters, Agnese is sad (*Emotions*) for the imminent departure and worried that Echo’s may be hurt or die; so, she cries. Echo is also sad (*Emotions*). Both of them are willing to sacrifice themselves in the name of the freedom of mankind (*freedom*). Echo’s goal is to leave for the mission (*Character’s goal*) and this goal is accomplished (*Goal achievement*) by saying goodbye to Agnese (*Character’s action*). Agnese’s goal is more limited in scope, as she simply wants to bid him farewell.

Finally, the annotation schema records the state of the world holding before and after each narrative unit. For example, in the annotation of the unit 1.1.1, the preconditions and effects explicitly state the connection between the action of leaving (the accomplished goal of Echo in the unit) with the fact that, as a consequence of the leaving, Echo begins his mission (*not(on_mission(Echo))* and (*on_mission(Echo)*) before and after the unit). The annotation of the unit 1.1.1 is in Fig. 3.

Notice that the annotation above only describes the literal content of the narrative unit. The full account of Echo’s intentional behavior emerges only by considering the units that constitute the context of the example unit in the plot hierarchy (the parent unit, 1.1 (“The mission begins ...”), and the subsequent unit, 1.1.2 (“They walk in the woods...”), as shown in Fig. 2). In the parent unit, 1.1, Echo’s goal is annotated as “meet(Echo+Tenebra,officer)”, a high-level action that consist of two more specific actions: “going” and “meeting”. From the context of the parent unit, it is clear that Echo’s goal in unit 1.1.1, “leaving”, and the action of “walking”, displayed in unit 1.1.2, constitute, respectively, the initial part of the action of “going” and the proper execution.

- **Unit Id:** 1.1.1
 - **Level:** 3.
 - **Children Drama Units:** <>.
 - **Description:** Echo says goodbye to his girlfriend, Agnese, to go on a mission.
 - **Link:** “media_objects/1944_1_1_1.avi”
 - **Characters:**
 - **Character:** Echo.
 - **Character’s belief:** occupied(Italy).
 - **Character’s goal:** Leave.
 - **Goal achievement:** true.
 - **Character’s action:** ExpressingFarewell.
 - **Values at stake:** freedom.
 - **Emotions:** sadness about departure; love for Agnese.
 - **Character:** Agnese.
 - **Character’s belief:** none.
 - **Character’s goal:** ExpressFarewell.
 - **Goal achievement:** true.
 - **Character’s action:** ExpressingFarewell.
 - **Values at stake:** freedom.
 - **Emotions:** fear for Echo; love for Echo.
- **Preconditions:** *not(on_mission(Echo)); together(Echo, Agnese).*
 - **Effects:** *on_mission(Echo); not(together(Echo, Agnese)).*

Fig. 3 Annotation of the unit 1.1.1 of the example story (Fig. 1)

The point of using ontologies resides not only in the fact that they provide agreed-upon labels to describe actions, but also in the axiomatization that accompanies the action labels. For example, the fact that the action of “leaving” constitutes the initial part of the action of “going” is explicitly expressed by the axiomatization of the “leaving” process in SUMO ontology (*sub Process ?LEAVE ?GO*). Or, the axioms describing the “meeting” process in SUMO explicitly states that the agents who meet must be near each other for the process to take place, thus expressing the rational motivation for going to the meeting point: (*orientation ?AGENT1 AGENT2 Near*). Although the Cinematic tool does not currently exploit this type of ontological knowledge, it is available for some external application to reason on the actional consistency of stories starting from the annotation. This is particularly useful for developing further tools that can assist story authoring or support the automatic analysis of narrative media.

3.5 The modeling process with Drammar

The modeling process implied by Drammar consists in the segmentation of a media object into narratively significant units, the annotation of such units in the three components, and the validation of the annotation through a run of the editor, trying to yield the original sequence in the video.

Taking a synoptic view over the Drammar model, we realize that the story structure and the world state represent the two orthogonal axes of the inclusion of a unit within a larger unit (vertical axis) and the linear dynamics of a story (horizontal axis), respectively, while the characters represent the basic engine for the

story incidents. The modeling process, applied to a specific story, reflects the work of an author in developing the story, as results from our empirical observation of authors as users of Cinematic and from the analysis of the scriptwriting manuals. So, the annotator operates through a reverse engineering process. S/he segments some media object to yield the major narrative units on an intuitive basis, and starts the annotation process by identifying the elements that characterize the story. Then, s/he proceeds either top–down or bottom–up: objects are further segmented into sub-objects (narrative sub-units), until s/he reaches the elementary actions that are significant from the narrative point of view; vice versa, elementary narrative units, identified from an actional point of view, are used to segment basic media objects, that are then concatenated to form larger objects (and larger narrative units). At the same time, the characters that drive the actions populate the unit annotation, with their emotions and goals to motivate the actions, and each unit is provided with the preconditions and effects that connect it to the adjacent units to form the evolving world state. The process applies to both complete stories and story units respectively, in a recursive manner.

During the development of the model, we studied both classical Hollywoodian and more experimental examples, in order to widen the empirical basis and identify the necessary elements of the model. However, only a thorough testing and the construction of a large corpus can ensure the robustness and the applicability of the model.

4 Annotation and editing of video segments in the Cinematic tool

Cinematic is a software tool for the annotation and automatic editing of narrative audiovisual segments. Cinematic allows the user to annotate video segments according to the tripartite Drammar model described above: segments are described as narrative units, and annotated with their position in the plot tree, the mental and emotional state of the characters, and the description of the world state before and after they occur, to make their contribution to the story progression explicit. The complete annotation schema is applied only to elementary narrative units (the leaves of the plot tree); higher nodes are only annotated with the first component of the schema (the position in the plot tree).

Cinematic was initially developed with educational purposes, namely to introduce the basic notions of formal logic to media students. Then, the partial incorporation of Drammar made Cinematic evolve towards a tool for the validation of the model and a basis for the construction of a corpus of video material annotated with narrative features.

The perspective adopted by Cinematic follows the video editor metaphor (see Section 2.1) in providing an immediate interface that lists all the sub-units of a narrative media object. We notice again that such narrative units are not equivalent to the shooting units that are dealt with in the editing task. Shooting units do not necessarily accomplish an advancement in the story progression, while narrative units do. However, the unit annotation employed by video editors typically consists in free textual notes inserted by human editors with purposes of easing the tasks of identifying, retrieving and positioning the units on the timeline, while Cinematic relies on a formal notation based on ground predicates of first–order logic for

annotating preconditions and effects. Video editors do not use annotations for automatic editing purposes; on the contrary, Cinematic, being an annotation tool, requires the user to fill the several fields that form the tripartite schema illustrated above. Then, Cinematic uses the annotation to provide a rudimentary editing facility, that validates the annotation of units in terms of continuity of the story progression. This makes Cinematic similar to the automatic editors of the Sections 2.2 and 2.3, where units are annotated with some formal language expressions, that the system exploits to produce a linear audiovisual. The difference with these systems relies on the editing purpose: these systems, being oriented to some rhetorical or thematic structure, employ sophisticated editing rules that map the users' goals to the unit annotation.

With respect to the systems described in the Section 2.4, Cinematic draws inspiration from both story-based systems, since the annotation schema represents the story structure, and character-based systems, because most of the descriptors of the annotation schema concern the characters. The goal of Cinematic is not the creation of a story, but to provide an interface for the annotation of narrative units and validate the adequacy of such an annotation through the reconstruction of the original linear sequence.

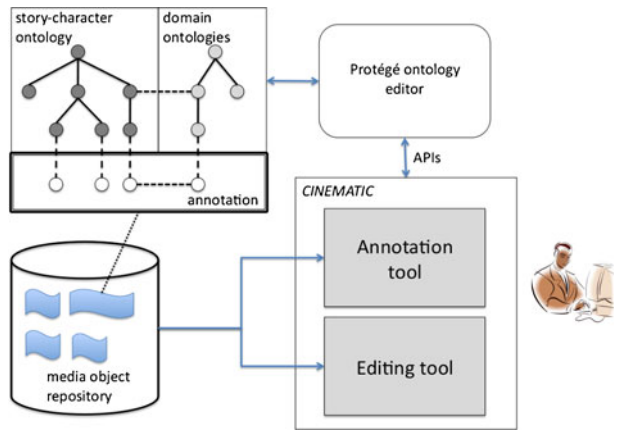
The content of a narrative unit is described as the state that holds in the story world before and after the unit is played, rather than as the contribution it brings to the rhetorical or thematic structure of the complete object. Cinematic adopts a situation calculus perspective over units, by considering them as operators that transform a state of affairs into another state of affairs. The situation calculus is a well established paradigm for reasoning about how actions modify the state of the world, although has proven inefficient in practical contexts [29]. In short, situation calculus, axiomatizing the preconditions and effects of actions, views actions as operators whose execution leads from one state to another as a logical consequence. As a result, the meaning of actions (here, narrative units) is self-contained and can be accessed by external applications as such, being at the same time application-independent. The computational tractability of the situation calculus is not a relevant issue in the case of Drammar, because the requirement of causal connection among units limits the combinatorial explosion of the possible concatenations (this is achieved through a careful handcrafting of preconditions and effects on behalf of the annotator); also, in the Cinematic tool, we pose an upper limit to the number of units being concatenated (see below).

4.1 The Cinematic tool

The Cinematic tool consists of two main components, the annotation module and the editing engine. The system architecture is sketched in Fig. 4.

The whole architecture builds on the top of Protégé ontology editor [<http://protege.stanford.edu>, 1995–2011]. So, the Drammar model is represented with an ontology, and the annotation format is in Protégé ontology language, i.e., every narrative unit is described by a set of (instances of) concepts represented in the Drammar ontology. For example, if a certain unit features a certain character, the value of the “character” descriptor of that unit refers to a specific instance of the concept of character in the ontology, an instance that represents that specific character. Then, given the intrinsic properties of the ontology languages, every

Fig. 4 The architecture of the Cinematic annotation system. The annotation and editing system (*bottom right*) access the media object repository (*bottom left*) through the annotations of the objects (*top left*), encoded as Protégé ontologies (*top right*)



instance of the character class automatically inherits all the properties of the concept: for example, the properties of having a name, a social role, etc.

As for the characters' role in a unit, also the concept of “unit” in the Drammar ontology describes the relation between the characters who appear in it and the actions they perform; also, this relation is inherited by any unit instance: so, for each unit, the character instances are listed as agents of certain “actions” (i.e., instances of process types in a separate process ontology). Figure 5 shows a screenshot of Protégé containing a portion of the Drammar ontology. In particular, the properties of the Narrative Unit can be seen (as slot in Protégé terms) in the right part of the figure, where we can recognize some of the fields listed in the Table 1. Media objects are located in a repository, and are linked to the instances that represent them by means

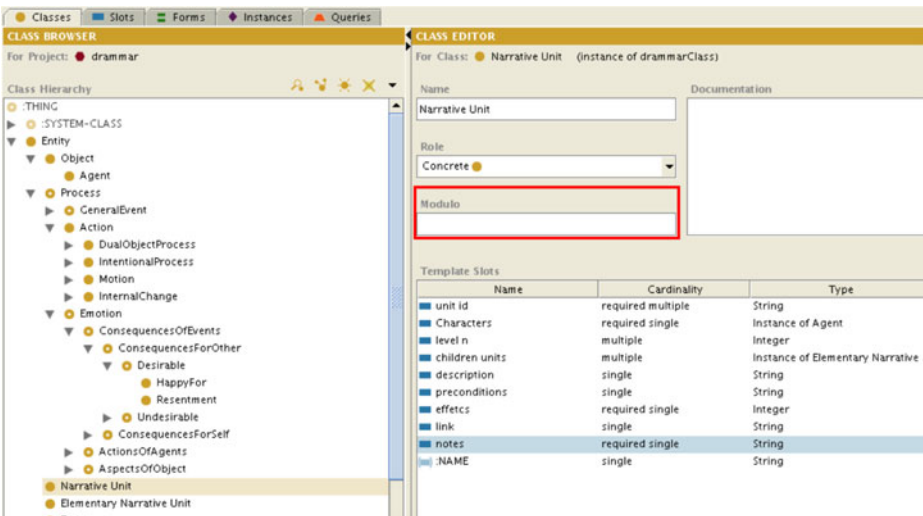


Fig. 5 A screenshot of Protégé with a portion of the story–character ontology. The *right pane* lists the slots (properties) of the drama unit class (entity)

of unique identifiers. The annotator selects an object from the repository, possibly previewing it, and annotates its narrative content and its position in the plot structure through a form-based user interface.

The interface of Cinematic is divided into three parts (snapshots can be seen in the Figs. 6 and 7). In the left pane, the user annotates the structural aspects of the unit (characters, actions and goals, etc.). This template is enforced by form filling; in some parts of the annotation, such as the description of the action and emotion types, the user can rely on a decision procedure, opened in a pop-up window, that helps her/him to select the right type of values by answering questions. Action and emotion types can also be selected by the annotator by navigating the ontology as shown in Fig. 6. Further, new action types can be created by the user during the annotation process and become available for future annotations. To keep the modeling of new actions simple, they can only be added to the fringe of the class hierarchy, but cannot be further described through the annotation interface. To add axioms, users need to access the process ontology separately.

The lower pane of the interface for annotation deals with the annotation of the world state (preconditions and effects) of the unit. Preconditions and effects of a unit are encoded through a set of statements, such as *not(on_mission(Echo))* and *proximity(Echo, Agnese)* in Fig. 7, that describe the state of the world at the beginning and at the end of some unit (technically, ground formulae of predicate logic). So, this pane contains two text areas for preconditions and effects, where the user inserts, removes or modifies the statements one by one. As stated in Section 3, the annotation schema assumes that the unit preconditions and effects include the

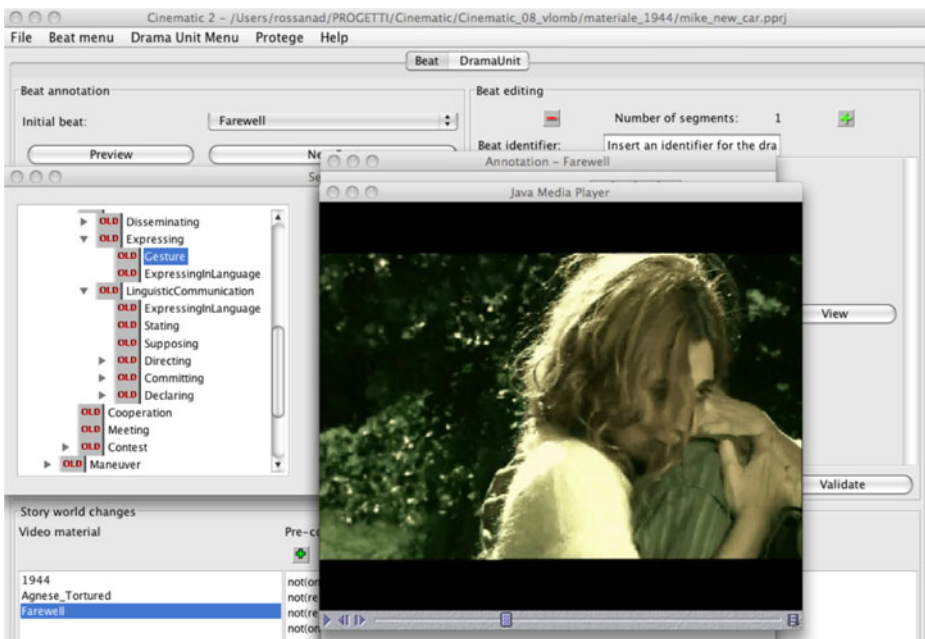


Fig. 6 A screenshot of Cinematic interface for annotating narrative units. On the left, the pop-up window for navigating the ontology of processes in the annotation of characters' action types

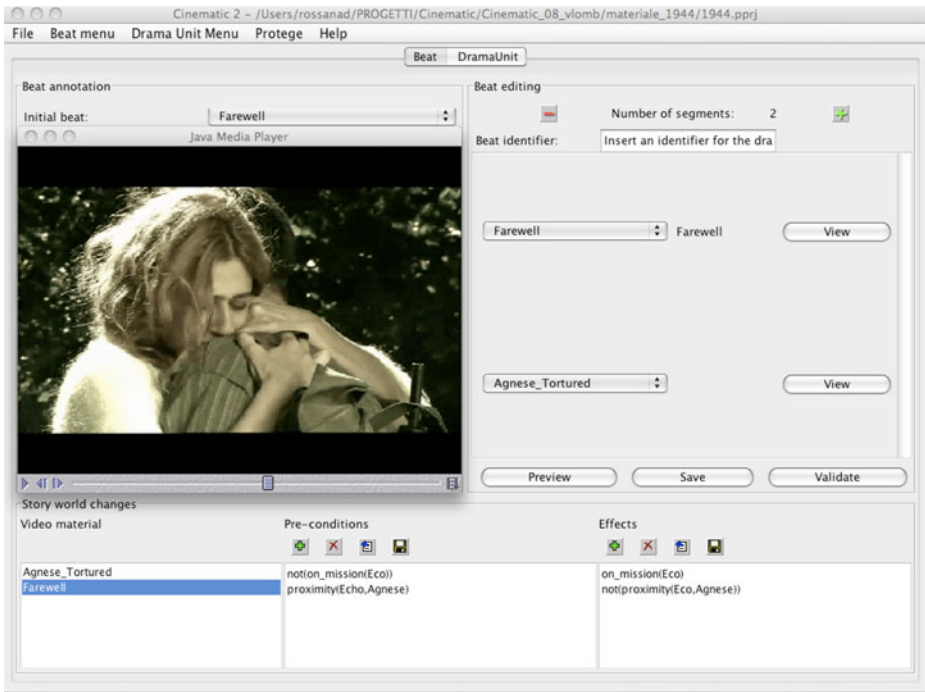


Fig. 7 The interface for editing. On the *left*, over the annotation pane, the preview window. On the *right*, the editing pane. In the *lower part* is the annotation pane for preconditions and effects, that determine how the story world changes

belief and goals of the characters who appear in that unit. However, the annotation interface does not enforce this constrain, leaving the annotator the decision of explicitly including the characters' mental state within the preconditions and effects.

The right pane deals with the editing function (Fig. 7). Given an arbitrary number of units to be edited, a search engine explores the space of the possible sequences through an all-path algorithm. By clicking the “validate” button, the software proceeds either by proposing possible sequences to the human editor or validating sequences that are edited by hand. Generated sequences are saved as instances in the annotation project, but can also be saved as video files in AVI format.

Cinematic is developed in Java (SDK 1.4.2) and relies on Java Media Framework (JMF) for the display of videos from within the application. It embeds the Protégé ontology editor and is interfaced with Protégé through its APIs. In order to support interoperability, it adopts Protégé ontology language as its native format for encoding both the annotation schema and the annotation of media objects, Protégé 3.x (Protégé-frames). So, the annotation of a video is saved as a Protégé project and can be accessed standalone as such. The transition to more recent ontology technologies, embedded in Protégé's latest versions, requires a refactoring of the software architecture that will be addressed in the CADMOS project [<http://www.cadmos-project.org>, 2011].

4.2 Annotation process

It is possible to use Cinematic either in a top-down fashion, by providing a narrative structure and then generating the edited sequences after identifying the elementary narrative units, or in a bottom-up fashion, by editing elementary units into larger and larger narrative units, until the complete video.

The top-down approach is similar to the implementation of a *videoboard*, a technique that is typical of animation films, and consists in sketching the various scenes and putting them in sequence with a soundtrack; as the animated sequences are ready, they replace the sketches in the videoboard until the final animated sequence. The bottom-up perspective adopts the viewpoint of the script supervisor in a production team. The script supervisor is responsible for checking the continuity constraints in a sequence of segments during film production, in order to prevent the introduction of inconsistencies or unmotivated meaning gaps that could frustrate the audience's attention and understanding.

As stated above, the annotation process relies upon two external ontological sources to label the actions performed by characters, to prevent much arbitrariness. For the description of the characters' actions and the events that occur in the narrative units, Cinematic relies on domain ontologies. Both the SUMO ontology for action labelling and the OCC ontology for emotion labelling are already embedded in Cinematic.

4.3 Editing in cinematic

The editing tool allows the annotator to validate a given sequence of narrative units, or to generate new sequences automatically by considering all the possible sequences. Editing relies on the annotated preconditions and effects, considering a narrative unit as an action operator, in the perspective of situation calculus. The paradigm of situation calculus, however, is not applied to the search of a valid sequence of operators to obtain a goal state, a perspective in which it has proven inefficient, but to validate the concatenation of random sequences of fixed length. The editing algorithm is the following:

```

Given a number  $N$  of units
Select the initial unit  $U_0$ ;
Set the Situation to the effects of  $U_0$ ;
REPEAT
  Randomly select the next unit  $U_i$  such that preconditions of  $U_i$  match the Situation;
  Add effects of  $U_i$  to Situation (while Removing facts that are inconsistent with effects);
UNTIL
 $N$  is reached OR no more selection is possible.

```

The editing tool produces in output a new unit instance, that represents the newly generated sequence. The tool may output other sequences than the original one. This can result from annotation flaws or from some motivated underspecification in the annotation. In the example above, it is indifferent in what sequence the editing tool orders the two units in which Echo and Tenebra talk, respectively, about their families left at home and their jobs (see the leaves of the plot tree in Fig. 2).

The baseline result is to yield the same sequence of the original video. To this aim, the annotator's methodology is to identify the minimal set of preconditions and

effects that yield the original playback. Once minimal preconditions and effects have been identified, they can be manipulated (by replacing, modifying, etc.) to generate new sequences. For example, the annotations can be modified by obliterating all but one character, so that the story is edited from that character’s point of view. In this sense, the editing tool lends itself to the exploitation in artistic productions (see Section 5), both to stimulate the author’s creativity in the content production process and to test the possible re-uses of existing media objects.

In order to describe how the editing works, we refer to the annotation and editing of two units of the example story (refer to Fig. 8, where the dynamics of the world state is encoded in the *Situation_i* columns and the Unit columns represent the annotation of a unit. In the unit 1.2.1, Echo and Tenebra get to Agnese’s farm and they find (“Learning” action type in SUMO) that she is being tortured by a brigade of fascists. So, in this unit, both Echo and Tenebra come to know that Agnese’s life is threatened and the value “human_life”, acknowledged by the two characters, is put at stake. The annotation represents this state of affairs by putting “human_life” as a value at stake into the characters’ properties in the unit, and by listing the formula *hostage(Agnese)* in the unit effects.

For the unit 2.1.1, the preconditions require that *hostage(Agnese)* holds in the situation (Fig. 8). The diverging characters’ goals, in this unit, account for their divergent moral judgements. For Echo, “human_life” has a higher priority than “freedom”, so he changes his goal from “Get(food)” (Echo’s goal in the previous unit) to “Rescue(Agnese)” from the fascists (an instance of the “UnilateralGetting” action type in SUMO). On the contrary, Tenebra’s commitment remains unaffected, since the value of freedom, for him, still has a higher priority. In order to mark



SITUATION i-1	UNIT	SITUATION i	UNIT	SITUATION i+1
				
	STORY STRUCTURE 1.2.1		STORY STRUCTURE 2.1.1	
	CHARACTER		CHARACTER	
on_mission(E) on_mission(T) at(E+T, A's_farm) T.VaS={freedom(1)} E.VaS={freedom(2)} T.I=meet_officer(E+T) E.I=meet_officer(E+T)	Tenebra VaS: human_life (2) Goal: Get(food) Action: Learning Emotions: Hope for freedom Echo VaS: human_life (1) Goal: Get(food) Action: Learning Emotions: Anger for A tortured	hostage(A) on_mission(E) on_mission(T) T.B={hostage(A)} E.B={hostage(A)} T.VaS={freedom(1), human_life(2)} E.VaS={freedom(2), human_life(1)} T.I=meet_officer(E+T) E.I=meet_officer(E+T)	hostage(A) not(on_mission(E)) on_mission(T) T.B={hostage(A)} E.B={hostage(A)} T.VaS={freedom(1), human_life(2)} E.VaS={freedom(2), human_life(1)} T.I=meet_officer(E+T) E.I=rescue(E,A)	
	WORLD STATE		WORLD STATE	
	Preconditions		Preconditions	
	at(E+T, A's_farm)		hostage(A)	
	Effects		Effects	
	hostage(A)		not(on_mission(E))	

Fig. 8 The log of the editing of two “1944” units in Cinematic (described in Section 4.3). Below the screenshots, the three components of the annotation. The description of the state of the world precedes and follows each unit. Predicates in *bold* are the most recently added to world state through the assertion of the effects of some unit; *arrows* connect matching predicates

the fact that Echo is not on a mission anymore, the annotator has also decided to explicitly state *not(on_mission(Echo))* among the effects of the unit 2.1.1. The latter choice, arbitrarily taken by the annotator, was motivated by a simulation of the characters' goal activations, where actions are part of larger plans. In this example, Echo's plan of accomplishing a mission is suspended because of the adoption of the new goal of rescuing Agnese.

When a sequence of N units is validated through the automatic editing function, Cinematic generates a new unit that results from the concatenation of the N units, with preconditions set to the ones of the first unit and effects set to the ones of the last unit, respectively. Further runs of the editing tool produce other sequences, until the search space has been completely explored.

In the current implementation of Cinematic, the editing does not account for the interplay of characters' goals and values described by the Drammar model (Section 3.2), since the tool does not constrain them to be consistent with the mechanism assumed by Drammar; so, they are currently annotated only for the sake of completeness. They can be taken into account only if they are explicitly listed among the unit preconditions and effects. In a future release of Cinematic, we will add the connections between the character and the world state components, as stated by the Drammar model, and will implement reasoning services that make some explicit annotations redundant.

5 Testing the Drammar annotation and the Cinematic tool in applicative projects

In order to test the adequacy of the Drammar annotation schema and the viability of using Cinematic to build a corpus of annotated narrative media objects, we have carried out a few experiments of annotation and (re-)use. In the following, we describe the different types of experiments.

5.1 Media analysis

Annotation experiments were set up with the cooperation of the students of film and media studies of the School of Media and Arts of the University of Torino, through two academic years. The goal of these experiments was to check the possibility of using Cinematic for media analysis with short movies (including cartoons) and scenes from feature films and other narrative media, such as opera and theatrical performances.

Students only received a basic training in the logic language to express the formal annotations (ground formulae of predicate logic and computational ontologies). Then, students were required to segment and annotate narrative audiovisuals with Cinematic and to check the validity of the annotation by producing the original sequence through the editing tool; also, we asked students to vary preconditions and effects so as to relax the constraints of the original video, in order to understand what aspects of the narrative were mostly responsible for the original sequence.

Most of the students decided to apply the Drammar annotation to short movies, especially cartoons, where actional gags are easier to annotate and characters' features usually are exaggerated versions of human characters, so that they would easily fit into SUMO categories. However, we also received some original projects.

For example, some students carried out authoring experiments by describing the actions played by the characters in a hypothetical script, in order to find out what possible sequences would be generated before continuing the writing process. In other cases, they took the logs of Cinematic executions to build interactive stories, by proposing the audience to choose among possible continuations.

The general comments were that they found the tool interesting to use in both analysis and production. They complained about the length of the segmentation process (done outside the Cinematic tool, with standard slicers). While the annotation of precondition and effects (free of references to some external source) was considered challenging but easy to perform, one of the most difficult tasks in the annotation was to select the appropriate action types in the SUMO ontology. In general, users found the Drammar annotation easy to understand and use, though they would prefer more support from the software tool in copying annotations from some unit to another or to re-use some predicate. The automatic editing for validation turned out to be very useful for testing the annotation hypotheses, and the strategy they typically adopted was of try-and-error and stepwise refinement over first guesses.

5.2 Media production

The concept project conceived by the artist Pino Cappellano consists of automatic content aggregation from four different sources (video objects), all different realizations of Shakespeare's Hamlet for cinema, namely 1948 Laurence Olivier's "Hamlet" (A), 1980 Rodney Bennet's "Hamlet, Prince of Denmark" (B), 1990 Franco Zeffirelli's "Hamlet" (C), 2000 Campbell Scott's "Hamlet" (D). The project focuses on an individual scene, the duel between Hamlet and Laertes (5th act, 2nd scene), which leads to the tragic final where all the main characters die and the well-known sentence, "The rest is silence", is uttered by the protagonist.

The annotated scene consists of 13 numbered narrative units (Fig. 9). The annotation produced a one-level plot tree with a single narrative unit that results from the sequencing of all the 13 units.

1. The Court led by the King meets Hamlet.
2. Hamlet asks Laertes for forgiveness.
3. The distribution of swords (with Laertes selecting the one with the poisoned tip).
4. The King betting on Hamlet's victory.
5. The first assault (Hamlet hurts Laertes).
6. The King offers Hamlet a poisoned cup.
7. The Queen drinks the poisoned cup while the second assault occurs.
8. The third assault (that causes Hamlet's wound with the poisoned sword).
9. The Queen dies.
10. Laertes accuses the King of the plan to kill Hamlet.
11. Hamlet kills the King.
12. Laertes asks Hamlet for forgiveness and dies.
13. Hamlet dies while telling Horace his last wills.

Fig. 9 The thirteen narrative units of the duel between Hamlet and Laertes (Hamlet, 5th act, 2nd scene)

In the four films, with only a few variations, the scene precisely realizes the Shakespearian text and features a well marked and rhythmic dynamics, a key feature for the full realization of the project. The only exception is that the first unit (Court enters) is absent in film B and the 12th beat (Laertes asks for forgiveness) is absent in film C. The annotation of elementary units relies on two specific choices, that revealed tricky issues in the annotation.

The first is the goal to preserve the narrative connections of the original tragedy. For example, the unit no. 7 includes two distinct dramatic situations, the second assault of the duel and the Queen drinking from the poisoned cup. According to a pure actional analysis, the two situations could be torn apart and form two separate units. However, as is well marked in the scene writing, the Queen observes the cup several times during the assault before drinking, and splitting the unit would lose this “crescendo”. This choice leads us to understand that a pure actional analysis, though desirable for re-use purposes can be poor for the specificity of some project (especially on the emotional account).

The second is the fine tuning of the cuts in the film to yield the units. Instead of pursuing a simplistic cutting strategy that satisfies the actional analysis of the characters’ behaviors, the artist took into account the unit transition over the four films, since, e.g., some utterance could be present at the end of the unit_{*i*} in one film and at the beginning of the unit_{*i+1*} in another film. So, in automatic editing the utterance would be possibly repeated, with a loss of believability and a consequent loss of dramatic tension. So, in general, the segmentation into units, had to cope with several issues of visual or sound juxtapositions. We had to introduce specific technical tools to extend one of the tracks beyond the duration of the other in segmentation and model these effects in automatic editing.

Preconditions and effects have been carefully designed to preserve the story coherence. So, the situations that are tested in the preconditions and updated in the effects are in Fig. 10.

Across the four films, units numbered equally received the same annotation in terms of preconditions and effects. By operating with random choices, Cinematic provided a large number (4^{12}) of possible sequences. The aesthetic effect intended by the artist relied more on interleaving of units from the four different sources than

Fig. 10 Situations for the preconditions/effects of the annotation of the duel between Hamlet and Laertes

1. Court in the room.
2. Hamlet asked for forgiveness.
3. Sword in Hamlet’s hand.
4. Sword in Laertes’ hand.
5. King’s bet occurred.
6. Laertes wounded by Hamlet.
7. Hamlet wounded by Laertes.
8. Cup drunk by the Queen.
9. Accusation by Laertes.
10. Queen died.
11. King died.
12. Laertes died.
13. Hamlet died.

the modification of the original sequence. The spectator was also invited to generate montages and save the best results in terms of invisible cuts between different films.

5.3 Cross-media analysis

This experiment concerns the support to cross-media analysis, in particular the issue of delivering multiple versions of a story through various media, such as cinema, television and videogames, as recently pointed out by contemporary narratology [50]. The analysis spans over a set of representative case studies across different media, encompassing traditional and new media. The example we report here concerns the notion of adaptation of the same baseline story [30], namely the novel “Dune” by Frank Herbert (1965), to the homonymous film by David Lynch (1984) and the videogame developed by Cryo Interactive in 1992. The story is about the war between the protagonist, Paul Atrides, and the cruel Baron Harkonnen for the control of the planet Dune, the only place in the universe where the precious “spice” (that makes intergalactic journeys possible) grows. Paul wins the war, rescues the inhabitants of Dune, the Fremmen nation, from the bloody domination by the Baron Harkonnen, and gains the control of the universe through the spice.

The analysis especially addressed the structure of the narrative and its emotional qualities, comparing the segmentation and the type and number of emotional states of the characters in the three different expressions of the baseline story. Here, we report the results of the analysis for the main character, Paul Atrides. The novel features the most complex narrative structure, as expected. The emotional state of the protagonist is described by 12 emotional states; there are 52 changes in the characters’ emotional states along the story. Changes are less frequent at the beginning of the novel, become more frequent in the central part and diminish towards the end, thus showing the pattern of a reverse U-shaped dramatic arc first described by Freytag [21]. Prospect-based emotions like fear and hope characterize the first half of the story, and are most related with Paul’s doubts about himself, his own nature and mission in life. As long as he engages in the fight with Harkonnen, gains new powers and gets stronger, he acquires self-confidence and starts to be worried mostly about the outcomes of the battles with Harkonnen.

The film has a canonical three-act structure and features 28 scenes. The narrative structure is a simplified version of the novel, with 9 emotional states and 45 overall changes of emotional states. Paul’s inner conflict to emerge as leader remains a central subject of the story. Similarly to the film, a dramatic arc can be observed, with most emotions gathering in the second third of the story structure, where Paul is faced with his inner conflict, the conflict with Harkonnen and the hostile environment of the planet Dune itself (notice that this schema reflects the types of conflicts theorized by McKee [38]) It is worth noting that the character of Harkonnen is quantitatively and qualitatively much less developed in the film than in the novel.

The analysis of the videogame shows a deeply different structure. First of all, the characters are largely simplified. Although the videogame feature 26 scenes and 12 emotional states, with a overall number of 61 changes, the structure of the story is very repetitive on the actional part (the actions of the characters are always the same), due to the strategic nature of the game that spans along a set of fighting episodes characterized by the two emotions of hope and fear. The player (as Paul)

feels hope and fear all along the conflict, but although the emotional intensity does not leave room for the inner conflict that constitutes the character of Paul in the other two media.

6 Conclusions

In recent years, a number of relevant research efforts have addressed the annotation of video for the purpose of retrieval, re-use and (semi)automatic editing. However, most projects focus on the description of the visual and content properties of video objects, but do not specifically address the question of how each segment contributes to the narrative structure. In particular, the structural aspects of story have not been explored to depth in video annotation, despite the research advances made by computational storytelling to define story structures in formal terms. In this paper, we have proposed an annotation schema based on the Drammar model and the Cinematic tool for the annotation and editing of video segments. The Drammar model relies upon a widely accepted background of theoretical models of story and consists in the tripartite schema of the annotation: the hierarchical organization of narrative units, the character-centered analysis of the actions displayed in the narrative units and their motivational counterpart in characters' goals, the control of story progress through the state changes that units bring in the story world. In particular, the annotation of units with their position in the plot tree locates the function of each narrative unit within the overall story structure; the character-centered annotation part supports the narrative analysis of media objects, since it formally traces the relations between the characters' goals and the actions they perform along the story to reach them. The annotation of preconditions and effects of narrative units describes them as independent from the story context, as standalone operators, open to re-use. Differently from other models, such as the Ontomedia ontology [26], Drammar does not represent how the story is expressed in the specific narrative, leaving the relation between the narrative content and the expression to future work.

The long term goal of this approach is the production of a golden reference corpus for the annotation of large corpora of narrative audiovisuals. Annotated corpora can then be used in a number of tasks related to media analysis, production, search/retrieval of segments, access to cultural heritage archives, automatization of several functions, such as the methods that bridge the gap between the low-level intrinsic features of a video and high-level semantic descriptors of the content.

Designed for non expert users (our test users only received an informal introduction to the annotation task), the Drammar annotation schema and the tool Cinematic are suitable to support a range of narrative audiovisuals and applicative scenarios, spanning from educational projects about the mechanisms of screenwriting and editing, to film analysis and media production. In order to limit the arbitrariness of the choices related to the classification of characters' goals and actions, the Drammar model relies on computational ontologies and Cinematic is interfaced with the Protégé editor to the SUMO ontology, although different top and mid-level ontologies could be adopted. Also, by relying on the use of ontologies, semi-

automatic tools to support the annotation process may be developed in the future. In fact, we aim at the development of a web-based annotation tool, that integrates semantic technologies, reasoning services, and multiple ontologies, with references to lexical resources. Especially the last innovation, i.e. the integration of a natural language processing engine, could be extremely important to reduce the amount of manual work in searching the correct annotation term through a navigation of the ontology. This was often commented as particularly onerous by our annotators, who would like to propose a free annotation in natural language and receive suggestions from the system about some standardized term to specify and replace the free annotation.

The Drammar annotation schema and the Cinematic tool have shed a light upon the content description of narrative media objects and the automatic editing process. In particular, the character component provides a link between the agent theories, coming from artificial intelligence and digital storytelling systems, and the classical narratological theories. Also, the world state component establishes a parallel between the situation calculus and the story coherence in terms of causal connections. The automatic editing provides a basic validation method for the annotation, but revealed to be also useful for artistic and analysis purposes. However, the annotation fails in describing how the story is conveyed to the audience at the expression level by the use of specific stylistic means (such as mood, editing, etc.). A better knowledge of these limits is fundamental for assessing how far the automatization of some procedures and tasks can go. As future work, we envisage the development of more sophisticated validation and editing functions, but also the evaluation of the tool and of its interface through systematic experiments with professional users, in order to assess its adequacy in the workflow of video production and unit editing in the re-use perspective.

A final word concerns the applicability of the Drammar annotation schema to interactive narrative. As we mentioned throughout the paper, we can apply (and actually we did, in the case of the game “Dune”) to the annotation of the session logs of interactive products, which share with linear audiovisuals the fact of having a temporal extent and consisting of actions carried out by the characters (that here include the user). We do not know if this metaphor of interpreting the user as a further character, with established goals and actions, is applicable to all of the interactive narratives, especially those where the user operates as a decision maker for orienting the story towards some direction. In these cases, we probably need an extension to the classical narratological and drama studies, that can inspire an enriched annotation schema, and may incorporate an explicit notion of authorial goals.

Acknowledgements This work is partially supported by the CADMOS Project (Character-centered Annotation of Dramatic Media ObjectS), funded by Regione Piemonte, Pole of Innovation for the Digital Creativity and Multimedia, 2010–2012.

We thank the anonymous reviewers who put a lot of time and effort in helping us to improve the paper’s quality and clarity.

We thank Giulio Valente and Stefano Vajna de Pava for their contribution in the implementation of Cinematic and Fabrizio Nunnari for software engineering; Antonio Pizzo for his contribution to the theoretical aspects of this work; the artist Pino Cappellano and the graduate student Irene Fornerone for their projects in Cinematic.

References

1. Arndt R, Troncy R, Staab S, Hardman L, Vacura M (2007) COMM: designing a well-founded multimedia ontology for the web. *The semantic web*, pp 30–43
2. Bates J, Loyall A, Reilly W (1994) An architecture for action, emotion, and social behaviour. *Artificial Social Systems*. LNAI, vol 830
3. Bocconi S, Nack F, Hardman L (2008) Automatic generation of matter-of-opinion video documentaries. *Web Semantics* 6(2):139–150. *Semantic Multimedia*
4. Bratman ME, Israel DJ, Pollack ME (1988) Plans and resource-bounded practical reasoning. *Comput Intell* 4:349–355
5. Butler S, Parkes A (1997) Film sequence generation strategies for automatic intelligent video editing. *Appl Artif Intell* 11:367–388
6. Campbell J (1949) *The hero with a thousand faces*. Princeton University Press, Princeton
7. Carroll N (2001) *Beyond esthetics: philosophical essays*. Cambridge University Press, New York
8. Cavazza M, Charles F, Mead S (2002) Interacting with virtual characters in interactive storytelling. In: *Proc. of the first int. joint conf. on autonomous agents and multiagent systems*
9. Coleridge ST (1817, 1985) *Biographia literaria*. Princeton University Press
10. Cua J, Ong E, Manurung R, Pease A (2010) Representing story plans in SUMO. In: *Proceedings of the NAACL HLT 2010 second workshop on computational approaches to linguistic creativity*. Association for Computational Linguistics, pp 40–48
11. Damiano R, Gena C, Lombardo V, Nunnari F, Pizzo A (2008) A stroll with carletto: adaptation in drama-based tours with virtual characters. *User Model User-Adapt Interact* 18(5):417–453
12. Damiano R, Lombardo V (2009) Value-driven characters for storytelling and drama. In: *Proc. of AI*IA10, LNCS 5883-0441*
13. Damiano R, Lombardo V (2011) An architecture for directing value-driven artificial characters. *Agents for Games and Simulations II: Trends in Techniques, Concepts and Design*, pp 76–90
14. Damiano R, Pizzo A (2008) Emotions in drama characters and virtual agents. In: *AAAI spring symposium on emotion, personality, and social behavior*
15. Dasiopoulou S, Tzouvaras V, Kompatsiaris I, Strintzis M (2010) Enquiring MPEG-7 based multimedia ontologies. *Multimed Tools Appl* 46(2):331–370
16. Davenport G, Morgenroth L (1994) Video database design: convivial storytelling tools. Technical report, MIT IC
17. Davis M (2003) Editing out video editing. *IEEE Multimed* 10(12):2–12
18. Dennett DC (1987) *The intentional stance*. The MIT Press, Cambridge
19. Egri L (1946) *The art of dramatic writing*. Simon and Schuster, New York
20. Feagin SL (2007) On Noel Carrol on narrative closure. *Philos Stud* (135):17–25
21. Freytag G (1863) *Die Technik des dramas*. Hirzel, Leipzig
22. Genette G (1983) *Narrative discourse: an essay in method*. Cornell University Press
23. Greimas A (1977) Elements of a narrative grammar. *Diacritics* 7(1):23–40
24. Hardman L, Obrenović Ž, Nack F, Kerhervé B, Piersol K (2008) Canonical processes of semantically annotated media production. *Multimedia Syst* 14(6):327–340
25. Hartmann K, Hartmann S, Feustel M (2005) Motif definition and classification to structure non-linear plots and to control the narrative flow in interactive dramas. In: *Third International Conference, Virtual Storytelling, ICVS, using virtual reality technologies for storytelling*, pp 158–167
26. Jewell M, Lawrence K, Tuffield M, Prugel-Bennett A, Millard D, Nixon M, Shadbolt N (2005) *OntoMedia: an ontology for the representation of heterogeneous media*. In: *Proceeding of SIGIR workshop on multimedia information retrieval*. ACM SIGIR.
27. Jung B, Song J, Lee Y (2007) A narrative-based abstraction framework for story-oriented video. *ACM Transactions on Multimedia* 3(2):1–28
28. Lavandier Y (1994) *La dramaturgie. Le clown et l'enfant*, Cergy
29. Lesvesque H, Pirri F, Reiter R (1998) Foundations for the situation calculus. *Electron Trans Artif Intell* 2:159–178
30. Lohse L, Thomas M (2005) Storytelling for the small screen: authoring and producing reconfigurable cinematic narrative for sit-back enjoyment. In: *Proceedings of int. conf. on virtual storytelling 2005, LNCS 3805/2005*.

31. Lombardo V, Damiano R (2010) Narrative annotation and editing of video. In: ICIDS'10, proceedings of the third joint conference on interactive digital storytelling. Springer-Verlag, pp 62–73
32. Lombardo V, Nunnari F, Damiano R, Pizzo A, Gena C (2008) The canonical processes of a dramatized approach to information presentation. *Multimedia Systems* 14(6):385–393
33. Madhwacharyula CL, Davis M, Mulhem P, Kankanhalli MS (2006) Metadata handling: a video perspective. *ACM Transactions on Multimedia* 2(4):358–388
34. Manovich L (2001) *The language of new media*. The MIT Press
35. Mateas M, Stern A (2003) Integrating plot, character and natural language processing in the interactive drama Faade. In: TIDSE 03
36. Mateas M, Stern A (2005) Structuring content in the faade interactive drama architecture. In: *Proceedings of artificial intelligence and interactive digital entertainment*.
37. Mateas M, Vanouse P, Domike S (2000) Generation of ideologically-biased historical documentaries. In: *Seventeenth national conference on artificial intelligence (AAAI)*, pp 36–42
38. McKee R (1997) *Story*. Harper Collins, New York
39. Nack F (1996) AUTEUR: the application of video semantics and theme representation in automated video editing. PhD thesis, Lancaster University
40. Nack F, Lindsay AT (1999) Everything you wanted to know about mpeg-7. *IEEE Multimedia* 6(3):65–77
41. Nack F, Putz W (2001) Designing annotation before it's needed. In: *Proceedings of the 9th ACM international conference on multimedia*, pp 251–260
42. Norling E, Sonenberg L (2004) Creating interactive characters with BDI agents. In: *Proceedings of the Australian workshop on interactive entertainment IE2004*.
43. Ortony A, Clore G, Collins A (1988) *The cognitive structure of emotions*. Cambridge University Press.
44. Peinado F, Cavazza M, Pizzi D (2008) Revisiting character-based affective storytelling under a narrative BDI framework. In: *Proc. of ICIDIS08*. Erfurt, Germany
45. Peinado F, Gervás P (2004) Transferring game mastering laws to interactive digital storytelling. In: *Proceedings of the 2nd international conference on technologies for interactive digital storytelling and entertainment*
46. Pizzi D, Charles F, Lugin J, Cavazza M (2007) Interactive storytelling with literary feelings. In: *The second international conference on affective Computing and Intelligent Interaction (ACII2007)*. Springer, Lisbon, Portugal
47. Polti G (1895). *Les trente-six situations dramatiques*. Mercure de France, Paris
48. Prince G (2003) *A dictionary of narratology*. University of Nebraska Press
49. Propp V (1968) *Morphology of the folktale*. University of Texas Press
50. Ryan ML (2004) *Narrative across media: the language of storytelling*. University of Nebraska Press
51. Riedl M, Young M (2006) From linear story generation to branching story graphs. *IEEE Journal of Computer Graphics and Applications*, pp 23–31
52. Rimmon-Kenan S (1983) *Narrative fiction: contemporary poetics*. Routledge
53. Schank RC, Abelson RP (1977) *Scripts, plans goals and understanding*. Lawrence Erlbaum, Hillsdale
54. Seger L (1990) *Creating unforgettable characters*. Henry Holt and Company, New York
55. Sgouros N (1999) Dynamic generation, management and resolution of interactive plots. *Artif Intell* 107(1):29–62
56. Smith TGA, Davenport G (1992) The stratification system. a design environment for random access video. In: *ACM workshop on networking and operating system support for digital audio and video*
57. Spierling U (2007) Adding aspects of “implicit creation” to the authoring process in interactive storytelling. In: *Proceedings of virtual storytelling 2007*. LNCS, 4871:13
58. Swartjes I, Theune M (2006) A fabula model for emergent narrative. *Technologies for Interactive Digital Storytelling and Entertainment (TIDSE)*. LNCS, vol 4326, pp 95–100
59. Szilas N (2003) Idtension: a narrative engine for interactive drama. In: *Proc. 1st international conference on Technologies for Interactive Digital Storytelling and Entertainment (TIDSE 2003)*. Darmstadt, Germany
60. Theune M, Faas S, Heylen D, Nijholt A (2003) The virtual storyteller: story creation by intelligent agents. In: *Proceedings TIDSE 03*. Fraunhofer IRB Verlag, pp 204–215

61. Ursu MF, Kegel I, Williams D, Thomas M, Mayer H, Zsombori V, Tuomola ML, Larsson H, Wyver J (2008) Shapeshifting tv: interactive screen media narratives. *Multimedia Syst* 14(2):115–132
62. van Fraassen B (1973) Values and the heart's command. *J Philos* 70(1):5–19



Vincenzo Lombardo PhD Associate Professor of Informatics, University of Turin, Italy (www.di.unito.it/~vincenzo) Born in 1964, he graduated in Computer Science at the University of Turin (1987), and doctorated in Computer Science in the Turin-Milan University Consortium (1993). He is co-founder of CIRMA (Centre for Research on Multimedia and Audiovisuals—www.cirma.unito.it) and of the School of Multimedia and Arts of the University of Torino. He also teaches in the Cognitive Science programme of the Doctoral School in Neurosciences (dott-neuroscienze.campusnet.unito.it). At the Virtual Reality & Multi Media Park company (www.vrmmp.it), he runs the Art-Science Alliance Laboratory (ASA Lab—www.vrmmp.it/asalab/), for applied research projects in interactive multimedia, and the School (www.edu.vrmmp.it/), with training programmes for professionals in the audiovisual industry.

He has been the coordinator of the EU-funded project Virtual Electronic Poem (www.edu.vrmmp.it/vep/) and of the national project DramaTour (www.dramatour.unito.it). Currently he is the coordinator of the projects CADMOS (www.cirma.unito.it/cadmos) and MESH—Mise-en-scène Helper (www.edu.vrmmp.it/mesh/). He also directs the virtual agent development of the ATLAS project (www.atlas.polito.it/en/).

His research concerns methodologies, models and applications of informatics to the production processes of multimedia artefacts and the simulation of cognitive processes. His papers are published in international journals, books, conference proceedings. He carries on some production activity in multimedia communication, hosted by events at the international level.



Rossana Damiano got a PhD in Computer Science from the University of Turin, Italy, in 2001, with a dissertation on artificial intelligent planning for multi-agent, normative systems. Since 2001, she has been teaching at the School of Arts and Media of the same university. Her research interests

span from AI applications for cultural heritage and multimedia to artificial characters for drama and computer-human interaction. She belongs to the research group “Interaction Models: Agents, Natural Language and Expression” at the Department of Computer Science of the University of Turin.

Rossana Damiano has published research papers in several international conferences and journals in the area of artificial intelligence and multimedia. In the last decade, she has taken part into applicative projects ranging from intelligent virtual agents (such as the ATLAS project for avatar-based translation from Italian Sign Language, ongoing) and character-based guided tours for historical locations (DramaTour project, 2006) to the CADMOS project (2010–2012) where she is part of the scientific committee.