Extraction and Analysis of Fictional Character Networks: A Survey

VINCENT LABATUT, Laboratoire Informatique d'Avignon – LIA EA 4128
XAVIER BOST, Orkis and Laboratoire Informatique d'Avignon – LIA EA 4128

A *character network* is a graph extracted from a narrative in which vertices represent characters and edges correspond to interactions between them. A number of narrative-related problems can be addressed automatically through the analysis of character networks, such as summarization, classification, or role detection. Character networks are particularly relevant when considering *works of fiction* (e.g., novels, plays, movies, TV series), as their exploitation allows developing information retrieval and recommendation systems. However, works of fiction possess specific properties that make these tasks harder.

This survey aims at presenting and organizing the scientific literature related to the extraction of character networks from works of fiction, as well as their analysis. We first describe the extraction process in a generic way and explain how its constituting steps are implemented in practice, depending on the medium of the narrative, the goal of the network analysis, and other factors. We then review the descriptive tools used to characterize character networks, with a focus on the way they are interpreted in this context. We illustrate the relevance of character networks by also providing a review of applications derived from their analysis. Finally, we identify the limitations of the existing approaches and the most promising perspectives.

CCS Concepts: • **Computing methodologies** → **Network science**; *Information extraction*; • **Information systems** → *Multimedia and multimodal retrieval*; • **General and reference** → Surveys and overviews;

Additional Key Words and Phrases: Information retrieval, character network, work of fiction, narrative, graph extraction, graph analysis

ACM Reference format:
Vincent Labatut and Xavier Bost. 2019. Extraction and Analysis of Fictional Character Networks: A Survey. *ACM Comput. Surv.* 52, 5, Article 89 (September 2019), 40 pages.
https://doi.org/10.1145/3344548

1 INTRODUCTION

The first works of fiction possibly date back to as far as the Paleolithic period, and have constituted a major part of human culture since then [88]. Nowadays, it is estimated that, on average, adults are in contact with fictional stories for 6% of their time awake [88]. Besides their artistic and entertainment aspects, fictions are assumed to fulfill various social and psychological purposes, e.g., improvement of communication [59], development of empathy and collaboration skills [94], elaboration of social norms [87], proxy to understand the real world [105], assessment of

Authors' addresses: V. Labatut, Laboratoire Informatique d'Avignon – LIA EA 4128, 339 chemin des Meinajariès, BP 1228, 84911 Avignon Cedex 9, France; email: vincent.labatut@univ-avignon.fr; X. Bost, Orkis, Laboratoire Informatique d'Avignon – LIA 4128, 339 chemin des Meinajariès, BP 1228, 84911 Avignon Cedex 9, France, Orkis and 610 rue Georges Claude, Pôle d'activité, 13290 Aix-En-Provence, France; email: xbost@orkis.com.

0360-0300/2019/09-ART89 $15.00
https://doi.org/10.1145/3344548

social strategies [88], constitution of a collective memory [88]. It is therefore natural that they are abundantly studied by academia and that fiction-related business is a significant part of the economy [9, 71].

A work of fiction takes the form of a *narrative*, i.e., a report of events telling a story. This report can be conducted through a variety of communication means: text, speech, image, music, gesture, and others, under a variety of forms: fables, tales, novels, plays, but also movies, TV series, video games, cartoons, and comics. The collection of events explicitly reported by the narrative constitutes its *plot*. These events are often ordered to form a chronological and/or causal chain [12]. By comparison, the *story* contains all the plot events, plus those imagined or inferred by the audience, based on both the plot and a number of contextual factors [12]. As an illustration, an *ellipsis* consists in removing events from the plot without affecting the story, as the audience will interpolate the missing parts. Put differently, the plot is *what* is told, whereas the narrative is *how* it is told, and the story is what the audience *perceives* of the plot through the narrative.

Historically, narratives have been studied from the Aristotelian perspective, which argues that the most important part of a narrative is its *plot*. However, more modern approaches focus on *characters* instead and consider that they are the agents that advance the plot through their actions [73]. This is exemplified by Woloch in the field of *literary analysis* [119]. He defines the notion of *character-space* as the narrative environment of characters in a novel, i.e., their position relative to the other elements of the plot (place, time, other characters). In other words, this is how characters are described in the narrative. The concept of *character-system* extends this notion to the narrative as a whole and corresponds to the union of all character-spaces. This approach has been noticeably used to study and understand how writers and directors build a narrative.

In addition to the characters themselves, researchers have started to take into account the way characters *interact*, which is considered as the backbone of the narrative [24, 84]. In such a context, graphs are a natural modeling paradigm, as they allow representing and studying a system through the interactions of its constituting elements. A *character network* is a graph describing a narrative by representing the characters through its vertices, and the interactions between them through its edges. As we will see later, there are many methods to extract this type of network from some raw data representing the considered work, depending not only on the nature of these data, but also on the information that one wants to encode in the produced network and on what one wants to do with it eventually. Moretti has shown that such an approach allows to handle more formally Woloch's concepts [74]. In a graph, the subgraph induced by a vertex and its neighborhood can be seen as a projection of the social aspects of the notion of character-space, whereas the whole graph, which contains all characters and their relations, represents the character-system [89]. Woloch emphasizes the fact that character-spaces must be considered jointly, and this is precisely what graphs, a naturally relational modeling framework, allow.

1.1 Value of Character Networks

This relevance of graphs for modeling works of fiction is illustrated by the number of articles dealing with character networks in the literature, and the variety of purposes for which they are used. We distinguish three categories of such articles.

First, in the context of *Narrative Analysis*, character networks are generally extracted manually for a very small number of narratives (typically, a single one). Authors use them in a "distant reading" fashion to obtain a simplification of the plot [74], characterize the plot structure at various levels [90], detect relevant patterns and narrative events, identify character roles (e.g., protagonist vs. antagonist) or particularly important characters [89], assess the validity of literary theories [32, 35], and produce graphical representations [79, 113, 120]. In addition to the description of individual plots, they are also used to compare them, for example, among episodes of a given series [40] or works belonging to the same genre [90], period [50], or author [85]. In other social science domains, character networks are also used for educational purposes [11] and to study certain psychological mechanisms [14].

Second, another category of works also adopts a descriptive and comparative approach, but relying on a *Complex Systems* paradigm. These authors consider that character networks are a type of *Complex Network*, and as such they apply the standard tools developed to analyze them [84] and/or propose new ones [101]. The network itself is the object of the study. Like for Narrative Analysis, these works generally consider a few narratives, as the networks are often extracted manually. Many articles of this type compare the topological properties of character networks with those of other kinds of complex networks, e.g., real-world social networks [5, 67], random models [5], or other fictions [104].

Third, a large number of works originate from the *Artificial Intelligence* domain. They focus more on automating the network extraction process, which requires solving various text, speech, image, and/or video processing problems, depending on the media used. Compared to both other categories, this allows using much larger corpora. These works also consider character networks as models of the plot and take advantage of this to solve higher-level problems: role detection [52], genre classification [7, 102], storyline detection [117], story segmentation [118], movie scene segmentation [65], video abstraction [109], recommendation systems [63], and others. The results obtained by solving some of these problems can be used to treat higher-level tasks, e.g., the detected roles can help summarize a plot. Certain authors directly relate character networks to novel fields such as *movie information retrieval* [82], which consists in obtaining and exploiting valuable information from collections of movies.

Character networks even reach the mainstream audience, mainly for their relevance as a visualization tool. Numerous non-academic or educational Web pages display character graphs extracted from pop culture works such as *Star Wars,*[1] as well as from classics like Shakespeare's plays.[2]

1.2 Specific Features of Fiction Works

The extraction and use of character networks concern all types of works, including non-fictional ones, such as biographies, professional meetings, journal articles, and broadcast news. So, in theory, it is possible to apply these methods developed for non-fiction to deal with fiction. However, in practice this does not necessarily lead to good results, because works of fiction possess some specific features absent from non-fiction. These can result in specific issues whose resolution requires suitable processing, but they can also correspond to additional information one can leverage through appropriate methods to improve performance. We give examples of both aspects in the following:

First, there can be differences in the *structure* of the narrative. For instance, plays and TV or movie scripts are *semi-structured* in the sense that scenes are explicitly bounded and speakers are explicitly named. This feature can be harnessed during network extraction [76] and does not appear in non-fictional narratives (or most other types of fictional ones, for that matter). In video-based narratives, the set of camera and editing rules, conventions, and guidelines, sometimes metaphorically called *film grammar* [15], is quite different in fiction and non-fiction works. For instance, the so-called *180 degree* rule states that during a scene, the relative positions of the characters on the screen must not change. The *shot alternation* (or *shot/countershot*) rule is particularly used during conversations: it specifies that consecutive shots alternatively show the involved characters. Yeh et al. [121] leverage both of them to improve character detection in movies. Comics and

[1] http://evelinag.com/blog/2015/12-15-star-wars-social-network/index.html.
[2] http://www.martingrandjean.ch/network-visualization-shakespeare/.

animated films are apart, as in their cases, the medium itself is unlike anything related to real-life. Characters can be highly deformed humans, non-anthropomorphic beings, or even inanimate objects, which makes ineffective the methods designed to detect faces or persons in photographs [9] or live action movies [111]. Moreover, the structure of comic narratives is unique in the sense that they include information under a variety of forms encoded in both text (captions, speech balloons, onomatopoeia) and drawings (pose, graphical conventions) not found in other media and whose extraction requires specific methods [9].

Second, there are generally significant *stylistic* differences. In texts, literary prose is considered as more complex than journalistic prose [31], and even more so when the work is older [41]. One of the effects of style is actually to give a unique identity to the work and to distinguish it from both non-fictional works and other fictions [13]. Stylistic differences are so marked that it is possible to assign works automatically to their creators [7]. They significantly affect the performance of generic methods on a variety of NLP tasks: plot modeling, character detection and story generation [30], text summarization [54], named entity recognition (NER) [8, 110], and co-reference resolution [57, 110]. There are a number of reasons for this drop in performance. For instance, for character detection in novels [30]: many characters are relatives and share the same last name; they bear nicknames; some fictional characters are inanimate objects in real life; writers use specific honorifics corresponding to complex, possibly outdated, and even imaginary social conventions; and they craft names to convey certain meaning or function. For co-reference resolution, the problem comes from longer sentences, more frequent use of pronouns and direct speech, more numerous and shorter co-reference chains [57]. Similarly to text, certain characteristics of fiction works make generic audiovisual processing tools inefficient [121]. For instance, movie directors use a variety of complex, possibly genre-related, editing techniques. At a lower level, the same face can appear under a variety of lights, colors, angles, expression, and other deformations that do not correspond at all to the very controlled conditions under which non-fictional works are recorded (e.g., news or talk shows). Speech-wise, conversations are subject to background noise or music, involve more participants, and a way of speaking that is unlike that found in other forms of audiovisual productions [19].

Third, fictions often are *closed-worlds* in the sense that they are self-contained and involve recurring entities, possibly with made-up names. Generic tools ignore this characteristic, which sometimes can help handling certain tasks [51] such as alias resolution (finding the different variants of a character's name). On the contrary, most generic tools rely either on a training corpus or on external databases: in both cases, the described entities are likely to be completely different from those occurring in a fiction. For example, a standard approach when performing face-matching in news is to leverage pictures from press articles and their captions: this cannot be done for movies containing fictional characters [127]. If anything, this method is more likely to return the actor's rather than the character's name. Similarly, many NER systems rely on gazetteers or services such as DBPedia, Wikidata, or YAGO [13], which are likely to include only the main characters (if any) of the considered fiction. For instance, none of the proper nouns used in Tolkien's *The Lord of the Rings* would be present in a standard list of first names or places.

Fourth, there is also a difference in the way characters interact in works of fiction, compared to real life [89]. A real-world social network represents an auto-organized system whose structure emerges from the interactions between some agents acting according to their own agenda. By comparison, the writer or director controls all actions of fictional characters and arranges them according to a plot. Put differently, real-world networks are the result of microscopic processes, whereas fictional ones are caused by a macroscopic process [89]. There is no reason to suppose that the writer tries to mimic actual social relationships when producing the work of fiction. As we will see later, studies show that this is generally not the case, as numerous character networks extracted from fictions do not exhibit realistic topological properties. This is because other constraints come into play, such as the intelligibility and appeal of the plot. Analyzing a different structure is likely to require specific tools, compared to real-world networks (including non-fictional character networks).

1.3 Perimeter and Organization of the Survey

The first publications related to fictional character networks date back to the early 2000s, e.g., References [5, 98]. As explained before, both extracting and leveraging these networks involve solving specific problems. However, there is no synthetic review describing the solutions proposed in the literature. With this survey, we want to fill this gap. Not only do we consider articles directly related to fictional character network extraction and/or usage, but also articles focusing on certain specific steps of this process (without necessarily trying to deal with such networks). Note that certain authors extract other types of graphs (non-character-based) from works of fiction, such as scene transition graphs [124], but we do not include them in this review. Our contributions include the description of the tools currently available and the approaches currently adopted to detect characters and their interactions from all forms of narratives, as well as the methods leveraging them to build character networks. We also contribute by identifying open problems at all levels of the extraction and analysis processes, and proposing perspectives to solve them.

Terminology-wise, we need to distinguish scientific work from works of fiction. For this purpose, we will use the words *author* and *article* to refer to scientific authors and their work, whereas *writer* (or *director*, *playwright*, or any medium-specific term) and simply *work* will refer to artistic authors and their works of fiction. The rest of the survey is organized into two parts. We first focus on the process of extracting a character network from a work of fiction. We introduce it in a generic way (Section 2) before describing its three main steps: the identification of characters and their occurrences (Section 3), the detection of their interactions over the narrative (Section 4), and the extraction of the graph itself (Section 5). In the second part, we focus on how to leverage character networks by examining a selection of tools developed to solve specific problems (Section 6). Finally, we discuss the main issues of the field and conclude with some perspectives (Section 7).

2 OVERVIEW OF THE EXTRACTION PROCESS

The process of extracting character networks from works of fiction depends a lot on the form of the considered narrative, e.g., novels are not treated like movies. To give the reader a general overview, in this section, we make abstraction of these differences and present this process in a very generic way. In the rest of the survey, on the contrary, we focus on their differences. We consider that this process consists of three main steps, represented in Figure 1: (1) the identification of characters; (2) of their interactions; and (3) the extraction of the proper graph. Each of them can be conducted in a number of ways, depending not only on the nature of the considered narrative, but also on the planned usage of the character network and on certain methodological choices.

The first step is the most dependent on the form of the narrative, as it starts with the raw material, i.e., the work of fiction itself. We distinguish two substeps. The first is to detect occurrences of characters in the narrative—for instance, looking for people's names in a novel or looking for faces in a movie. The second is to unify these occurrences, i.e., to determine which ones correspond to the same character. In a text, the same character can appear under different names, whereas in a movie, the same face can be shown under a variety of scales, colors, light, and angles. The output of this step takes the form of a chronological sequence of unified character occurrences.

The second step consists in detecting interactions between characters. Note that it is sometimes more efficient or convenient to conduct parts of this process during the first step, but this is generally not the case. We identify five different definitions for the notion of interaction. Many authors

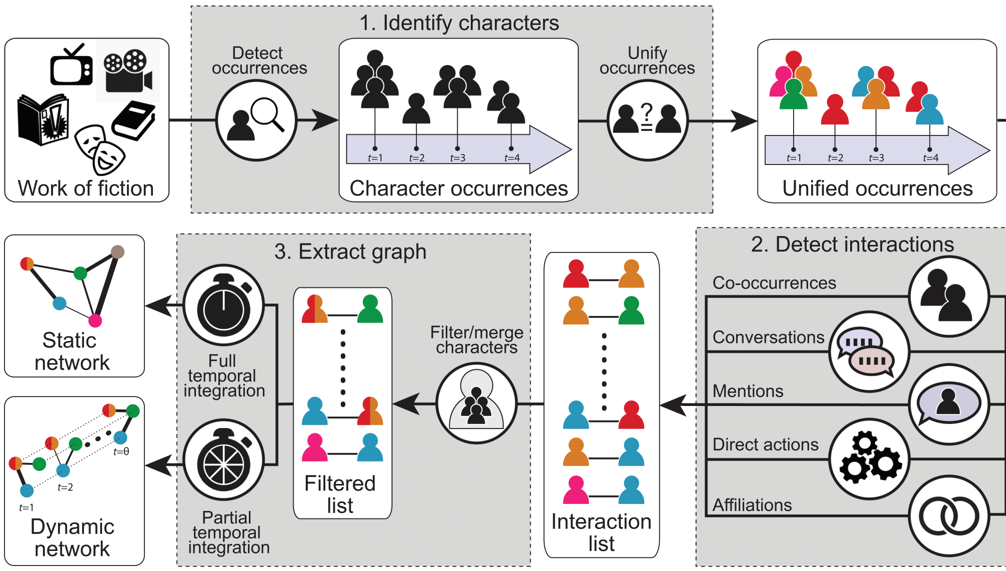

Fig. 1. Overview of the generic character network extraction process. Figure available at 10.6084/m9.figshare.7993040 under CC-BY license.

consider that a simple *co-occurrence* between two characters is enough to infer an interaction between them. Others prefer to identify *explicit* interactions, which is generally a more difficult process. One way of doing this is to take into account *conversations* and to consider that two characters interact when one talks to the other. With certain forms of narrative such as plays, in which speakers are given, this task is relatively straightforward. An alternative is to focus on the content of the conversations and to leverage *mentions*, i.e., situations where one character talks about the other. Some authors consider all sorts of actions one character can perform on the other (besides conversing). This is particularly the case with novels, a form of narrative in which such actions are explicitly described. Finally, certain authors do not focus on actions and prefer to use *affiliations*, i.e., explicit or inferred social relationships such as being married, being relatives, or working together. Note that it is possible to combine these definitions of the notion of interaction; for instance, by looking for both co-occurrences and conversations.

The output of the second step is a chronological sequence of interactions between characters. The third step is therefore relatively generic, as it relies only on this list and is thus independent from the nature of the original narrative. We distinguish two substeps: The first, which is optional, consists in simplifying this sequence by *filtering and/or merging* some of the characters under certain conditions. For example, when considering co-occurrences, some authors merge characters that always appear together; this allows simplifying the network. The second substep defines how the graph is extracted through *temporal integration*, i.e., the aggregation of the previously identified interactions. There are a number of approaches for this purpose, which we separate into two groups: those performing a *full* integration and therefore leading to a *static* network, and those performing only a *partial* integration and producing a *dynamic* network.

3 CHARACTER IDENTIFICATION

Character identification consists in detecting *which* characters appear in the considered narrative, and *when* exactly they appear in this narrative. As mentioned before, the form under which characters appear in the narrative varies much depending on the medium. In the case of text, they can be represented in three ways [31, 108]: *proper nouns* (e.g., "Sherlock Holmes"), *pronouns* (e.g., "He"), and *nominals*, i.e., anaphoric noun phrases referring to characters (e.g., "The consulting detective"). For videos, they either can appear onscreen or be mentioned in the audio stream (again as a proper noun, pronoun, or nominal). In comics, characters can either appear as drawings or be mentioned in the text (again, under the same three forms).

Automating character identification is quite challenging, which explains why many non-specialists prefer to perform this task manually. We describe this manual approach separately, as there are various ways of proceeding (Section 3.1). But our focus is rather on automatic approaches, for which we distinguish two subproblems: first, finding character occurrences in the narrative (Section 3.2); and second, determining which of these occurrences represent the same character (Section 3.3). For both subproblems, there are two very different categories of approaches, which depend on whether characters are represented in a textual vs. audiovisual way. Note that this dichotomy does not necessarily match the type of narrative; for instance, a movie can be treated as a video or as a text (through its transcript or script).

3.1 Manual Approaches

Some authors adopt a fully manual approach to detect character occurrences, in which case there is no need to distinguish occurrence detection from occurrence unification, as both tasks are conducted at once.

Direct Annotation. The most widespread method is direct annotation, which consists for the authors in annotating by themselves the narrative they want to study, e.g., [14] for novels, [67] for myths, [74] for plays, [24] for movies, and [117] for TV series. Researchers opt for such a manual approach due to technical limitations, e.g., they do not have access to efficient automatic methods [74]. However, it can also be a methodological choice, e.g., to better focus on the assessment of the other steps of the network extraction and/or analysis process [2, 3]. It is not clear exactly how annotators deal with occurrence unification. However, context suggests they generally perform this task, and do so manually (e.g., Reference [74]), as the extra cognitive cost is marginal.

Character Index. Instead of doing the annotation work themselves, certain authors take advantage of predefined resources, which are also manually constituted. For certain classic fiction works, experts have constituted so-called *character indexes*, indicating at which point of the plot each character appears. This is the case, for instance, of Rousseau's *Les confessions* in Reference [89], Park's *Toji* in Reference [81], and Marvel comics in References [5, 38]. There are mainly two limitations when using such resources: First, they are not or poorly documented, and it is not clear how they were elaborated, what their reliability is, and how character unification was handled. Second, they impose a predefined level of precision on the rest of the extraction process. For instance, character occurrences are expressed in terms of pages for *Les confessions*, and comic issues for the Marvel Universe.

Crowdsourcing. In practice, it is hard to handle more than a few works of fiction when using direct annotation or predefined character indexes. A workaround is to turn to *crowdsourcing*, as Rochat and Kaplan do in Reference [90] to constitute their own indexes. Interestingly, this study is also characterized by its multimedia nature, as the authors consider a corpus of science-fiction works including novels, comics, movies, TV series, and video games. For the latter, they experiment with three different base materials: walk-throughs (i.e., texts explaining how to finish the game), which are treated like novels; cinematic scenes, which are treated like other videos (movies and TV series), and transcriptions of these scenes, which are treated like scripts. The manual approach has the advantage of allowing a more uniform network extraction process over the variety of considered media, and therefore makes it possible to compare them.

3.2 Detection of Character Occurrences

We now switch to approaches that are at least partially automated. As mentioned before, the process of character identification largely differs depending on whether the narrative is visual (Section 3.2.3) or textual, which is why we separate them in our description. Moreover, certain texts such as plays and scripts possess a structure that can be leveraged for character occurrence detection, so we distinguish such semi-structured text (Section 3.2.2) from free text (Section 3.2.1).

3.2.1 Free Text. As mentioned before, a character can appear under three forms in text: proper noun, nominal, and pronoun. The methods used in the literature all handle the first form, but not necessarily the two others, as detecting them is generally a much harder problem, and they are often not considered as informative. A simple way to detect character names is to use a predefined list of these names and proceed through exact matching [42, 47, 61]. Such a list is generally constituted manually, either by the authors themselves or through an external source such as the Wikipedia page of the considered novel. Constituting it is not a trivial task, as characters can be referred to through a variety of *aliases*, i.e., variations of their name. For instance, Sherlock Holmes can also be called simply "Sherlock," "Holmes," or "Mr. Holmes."

Detecting character names can be viewed as a specific version of the Named Entity Recognition (NER) problem. NER consists in finding expressions in the text corresponding to proper nouns and identifying their category (e.g., *Location*, *Person*, *Organization*). A number of authors apply off-the-shelf NER tools to novels, e.g., References [32, 45, 96], and then only retain the *Person* entities. Incidentally, those are generally much more frequent than other proper nouns in literary texts [112]. It is possible for a NER tool to assign different categories to distinct instances of the *same* string, because of contextual differences. For instance, "France" is a country, but also a first name. However, in the context of novels, such a situation is likely an error: It is generally agreed upon that a novel is a small, self-contained world [7] and that the writer would not confuse the reader by using the same name to denote entities of different types (such as a person and a place) [8]. A straightforward solution to the multiple category issue is then to keep the majority category, as in References [7, 8]. Valls-Vargas et al. specifically train a classifier to distinguish characters from other types of mentions [111].

Fiction texts have certain characteristics that are leveraged by some authors either to perform some post-processing after having applied an off-the-shelf NER tool to find missed mentions and/or discard incorrectly detected ones or to design new fiction-specific NER tools. A simple method is to remove infrequent names, as they are likely to be errors. For instance, Elsner [30] and Sack [91] remove names appearing fewer than five times. Some authors also perform a manual verification to fix the errors of the automatic tools [91]. Some automatic approaches use honorifics (titles such as "Sir" or "Madam"), generally by relying on manually predefined resources. One can take advantage of a list of honorifics to detect them in the text and check the surrounding text for character names [7, 8] or look for a set of patterns describing the various possible combinations of honorifics, first names, and last names [108]. Some approaches proceed similarly with action verbs, as only characters are likely to be their subjects. For instance, Ardanuy and Sporleder [7] use a manually constituted list of speech verbs (e.g., to say, to discuss). Zhang et al. also use the grammatical structure of the sentence through part-of-speech (PoS) tagging [126]. Finally, some approaches consist in looking for relations of possession (through genitive marks, such as "'s" in English) [108], as only characters are supposed to own things. These approaches are more robust than generic NER tools, in the sense that they allow detecting non-human characters behaving as humans [53].

These last methods are likely to return not only proper nouns but also nominals (anaphoric noun phrases referring to characters). Some authors propose methods specifically designed to detect these nominals, generally through regular expression matching. Elson et al. [33] look for structures of the form: a *determiner* (article, possessive, number . . .), an optional *modifier* (e.g., an adjective), and a *head noun* (not necessarily a proper noun). They manually compile lists of determiners and head nouns based on their corpus and external linguistic resources such as WordNet. They use them to detect the determiner and head noun first and consider the text located in between as the modifier. The task of detecting pronouns is more or less difficult, depending on the considered language. For English, exact matching based on a manually defined list is a simple and efficient approach [31].

3.2.2 Semi-structured Text. A number of narratives can take the form of a script: theatrical plays, movies, TV series. A script is essentially a conversation-based text, with specific structure and formatting described as *semi-regular* by certain authors [1]: scene boundaries are clearly indicated, the characters involved in a scene are explicitly listed at its beginning in upper-case, and the name of the character speaking a line is indicated right before it, also in upper-case. When the script is properly formatted and structured, it is relatively straightforward to extract this information automatically. Authors have proposed methods based on exact string matching [27, 52, 75], regular expression [102, 120], or a custom parser [85].

However, this structure and formatting is not a proper standard and can vary from one script to the other [68]. It is even possible to find inconsistencies in the same script. Machine learning can help solve this issue. Agarwal et al. [1] propose a method to identify which parts of the script correspond to character lists, dialogues, speaker names, scene boundaries, and scene instructions. Tan et al. [104] take advantage of this type of decomposition to only focus on character mentions associated to utterances to ignore passive characters that are present in a scene but do not intervene.

Discrepancies can also appear in the speaker names. In this case, a simple approach consists in using an *a priori* list of the characters involved in the script [56, 104, 120] with their associated aliases. This list is generally constituted manually or by taking advantage of publicly available resources (generally also constituted manually), such as the Wikipedia page of the considered work of fiction. Again, machine learning–based method can be more robust than such simple matching-based approaches. Makris and Vikatos [68] take advantage of the Wikipedia pages of the movies they study to train a classifier into identifying which character speaks which line. Certain authors directly apply off-the-shelf NER tools [39, 127] to detect speaker names.

Identifying the characters involved or speaking during a scene is enough when extracting co-occurrence (cf. Section 4.1) or conversational networks (Section 4.2), respectively. Dealing with other types of interactions between characters requires identifying character mentions in the rest of the text [60, 75]; not only explicitly identified speakers, but also scene metadata, spoken lines, and/or stage directions. In this case, one can apply similar approaches to those already described for free text. For instance, Krishnan and Eisenstein [56] train a classifier to detect addressees mentioned in utterances to determine who speaks to whom exactly. They detect not only proper nouns, but also nominals, including titles and placeholder names (e.g., “bro,” “dude,” “sir”).

3.2.3 Visual Narratives. In audiovisual narratives, detecting character occurrences amounts to solving several distinct but related problems, depending on whether one focuses on the video or the audio stream. In videos, these are face detection and face tracking [121].

Face detection consists in identifying which parts of a still image correspond to faces. Jung et al. note that current face-detection methods are efficient mainly on front views of the faces [52]; this is a strong limitation in our context, as a character can be filmed under a variety of angles. Weng et al. also observe that current automatic methods do not reach satisfying enough performances, which is why they first proceed manually [117]. However, they later train their own model to obtain acceptable performances on their dataset [118]. They experimentally find the community structure of the extracted networks to be relatively robust to face-detection errors. A number of authors proceed automatically using off-the-shelf tools [65, 77, 109, 121].

The face-detection problem is relatively similar when dealing with comics, except that the images are drawings. This implies a number of additional difficulties: the characters can be very deformed, non-human, or even non-anthropomorphic. Moreover, the structural lines defining the characters and objects composing the panels are mixed with textures, screentones, and stylistic elements. For these reasons, methods designed to handle photographs generally perform poorly on comics, which require specific approaches [23, 100]. One such approach consists in adapting features or models originally developed for photographs, e.g., facial landmarks (points corresponding to specific parts of the face such as eyes or mouth) in Reference [100]. Takayama et al. [103] handcraft features to fit the specific case of mangas (skin and hair colors, jawline shape, symmetry). More recent articles focus on training Deep Neural Networks [23], but there is not enough publicly annotated data yet to reach the full potential of such approaches [9]. Finally, it is worth noticing that before being applied, many face-detection approaches require some preprocessing, in particular detecting panel bounds and speech bubbles [23], which in turn constitute specific problems [86, 99].

Face tracking builds upon face detection and aims at identifying chronological sequences of faces corresponding to the same person in a video. These sequences are called *face tracks* and can be considered as character occurrences in videos. Performing face tracking requires accounting for changes in pose, scale, rotation, expression, color, light, angle, and blur. Most authors use off-the-shelf tools [65, 127], usually based on some form of similarity-based classification of the detected faces.

Somandepalli et al. [95] detect characters in *animated* movies. This task proves to be much more difficult than with live-action videos, as the design of the characters can vary widely, including non-human and even non-anthropomorphic shapes. The authors first list character candidates by detecting salient objects in a generic way before taking advantage of graphical and saliency features to discard irrelevant ones. They then use an off-the-shelf tool to track deformable objects.

When using the audio stream, detecting character occurrences amounts to solving the *speaker segmentation* (or *speaker change detection*) problem. It consists in partitioning the audio stream into segments associated to unique speakers. Put differently, one wants to find the moments corresponding to switches between speakers. This task is sometimes performed simultaneously with that of *segment clustering*, which consists in grouping the segments spoken by the same person. Performing both these tasks sequentially or simultaneously is called *speaker diarization*. However, we treat this later in Section 3.3.2 and focus here only on speaker segmentation. Certain existing systems work well in controlled environments, but this performance strongly drops when applied to fiction works, e.g., movie trailers and cartoons [26], and TV series [34]. This is mainly due to the presence of background music and sound effects, the higher number of speakers [16], the spontaneous (though acted) nature of the exchanges, and the shorter speech turns [19]. Results improve when using methods specifically designed or trained on fictional audiovisual narratives, e.g., Reference [17]. It is worth noting that compared to video-based methods, audio-only tools do not allow identifying characters that appear in a scene *without speaking* [52].

Certain approaches try to combine several types of information, be it video-, audio-, or language-based. A few multimodal methods able to perform speaker segmentation using both audio and video have been proposed, but we describe them later, as they all additionally solve speaker clustering (and therefore speaker diarization). Scripts can be used to distinguish speakers or on-screen characters as in References [52, 82]. A script is not time-stamped, so this approach requires first solving an additional problem called *script alignment*, which consists in determining the exact time at which each line contained in the script occurs in the video. In Reference [22], Chen et al. use transcript extracted from TV series. They apply off-the-shelf tools to detect all three forms of textual character mentions (proper nouns, nominals, pronouns).

3.3 Unification of Character Occurrences

The second step of character identification is occurrence unification, which consists in determining, for each detected occurrence, to which character it corresponds. Like for occurrence detection, the methods proposed for this purpose vary much depending on the medium of the narrative: textual (Section 3.3.1) vs. visual (Section 3.3.2). However, this time there is no distinction to make between free and semi-structured text, as all the additional information of the latter has already been used during the detection step.

As the literature shows, character unification is often not performed at all. There are mainly two reasons for this: this task is generally harder than occurrence detection (especially in text); and in certain situations, it is simply unnecessary. For instance, when extracting a purely conversational network (Section 4.2) from a clean script, the speakers explicitly named in the script are enough, e.g., References [27, 74]; or when extracting a network from a novel by considering chapter co-occurrences (Section 4.1) [55]—it is likely that all characters participating in the chapter will be explicitly named. Some authors show this empirically, e.g., Seo et al. [93] argue that restricting their analysis to explicit proper nouns (and thus, ignoring pronouns and noun phrases) is enough to perform their targeted tasks (character ranking and edge prediction) without significant performance loss.

3.3.1 Textual Narratives. As mentioned before, characters' occurrences appear under three forms in text: proper nouns, nominals, and pronouns. Unifying these occurrences can be considered as a specific version of the *coreference resolution* problem, which consists in identifying sequences of expressions, called *coreference chains*, that represent the same concept. Generic tools exist to solve this problem, but their performance does not necessarily translate to fiction works. In particular, they tend to overlook minor characters such as those mentioned only through nominals (e.g., "the detective") [110]. Moreover, in our case, the referents are necessarily persons (characters), a category of entities possessing certain characteristics (e.g., gender) that can be leveraged to improve performance. Two variants of the problem appear in the literature: certain authors focus only on *alias resolution* (e.g., Reference [32]), which consists in grouping proper nouns referring to the same character, while others additionally solve pronominal and/or nominal anaphoras.

The task of alias resolution arises because of the variability of proper nouns appearing in fiction works. On the one hand, in addition to their full name, characters are generally called by a variety of aliases depending on context, style, and other factors. For instance, Sherlock Holmes can also be called "Mr. Holmes" or "Sherlock." On the other hand, some aliases cannot be unequivocally associated to a character, e.g., "Mr. Holmes" can refer to both Sherlock Holmes and his brother Mycroft Holmes. Most authors use some form of *name clustering* to perform alias resolution, each cluster corresponding to all the names encountered for a specific character. Roughly, they use two factors to determine that two aliases point at the same character: *string similarity* and *gender compatibility* [7, 30, 33, 79, 110]. The gender of a character mention can be detected using gendered honorifics (e.g., "Mr." vs. "Mrs.") and gendered first names (e.g., "Stephen" vs. "Stephanie"), matched to a manually constituted list or some external resource such as WordNet [33, 79].

A straightforward approach to compare strings is to use an appropriate distance function [49]. However, by doing so, one ignores the structure of the names: potential presence of honorifics, initials, multiple first or last names, distinction between first and last names. Moreover, a number of conventions are culture-specific (e.g., the use of patronyms in Russian names). Also, the relative proportions of first- and last-name occurrences is likely to vary considerably from one work to the other, as it is tied to stylistic aspects: it is assumed to reflect the level of intimacy in the narrative [112]. Certain authors propose to perform direct comparisons through predefined patterns [79] or rules [7, 8]. Elsner [30] first compares only multiword names to deal with the ambiguity of isolated first or last names. He constitutes clusters of similar and compatible multiword names and then only assigns single word names to these clusters whenever possible. The remaining names are assigned based on spatial proximity in the text and lexical frequency. Certain authors use a generative approach [32, 110]; based on multiword names found in the text, they produce potential variants thanks to predefined recombination rules (e.g., addition of honorifics, omission of first names) and resources such as gazetteers. These artificial names are then matched to those found in the text. Vala et al. [110] use additional constraints to prevent certain names from being grouped together: co-occurring names, names with the same last name but different first names, names containing different honorifics.

The other types of anaphoras are more difficult to handle, as they convey additional issues. Some pronouns or nominals may not be connected to any proper noun (and therefore character), if their referent is missing. They can also have split referents, e.g., "They" and "the Holmes brothers" can both refer to "Sherlock and Mycroft Holmes." Certain anaphoric expressions can also refer to non-character entities. Many authors use off-the-shelf tools to solve automatically pronominal anaphoras, e.g., References [96, 101, 108]. Lee and Yeung additionally define a distance limit between the reference and the referent to discard relations deemed too remote [62]. Vala et al. [110] extend to pronouns and nominals the cluster-based approach used for alias resolution. As for proper nouns, gender compatibility can be leveraged for certain pronouns (e.g., "she" vs. "he") and nominals (e.g., "uncle" vs. "aunt"). To identify anaphoras referring to characters (as opposed to other types of entities), they constitute a list of verb-noun co-occurrences considered as frequent in novels and perform a grammatical dependency parsing: only the expressions involved in such situations are considered as character mentions. In Reference [49], Jannidis et al. use a co-reference resolution tool that they previously developed specifically for novels in German [57]. In particular, they use linguistic resources to associate close synonym nominals to the same character.

3.3.2 Visual Narratives. Like before, handling audiovisual narratives amounts to solving very different problems that depend on whether one uses video or audio data. When dealing with videos, the problems at hand are face track clustering and possibly face-name matching.

Face track clustering consists in identifying groups of face tracks (output from the occurrence detection step) corresponding to the same face, hence character. Certain authors use off-the-shelf tools [65, 109, 121], but others consider that these generic methods are not sufficiently efficient when applied to fiction works [117]. Their results can be improved by training them on a corpus of such works [118], but this requires additional work and resources.

As mentioned before, when using the audio stream, the problem is to perform *speaker clustering* based on the output of the speaker segmentation step, the whole process being called *speaker diarization*. Speaker clustering consists in grouping audio segments spoken by the same character (akin to face track clustering). Generic methods are subject to the same limitations as observed for speaker segmentation [19, 26, 34]. Methods specifically developed for fictions obtain better results. In Reference [17], Bost and Linarès treat TV series through a two-step method first solving the problem locally at the scene level, then combining these partial results at the global level to deal with the whole character set. In Reference [18], they turn to a multimodal approach to enhance their method: in addition to their audio-based tool, they independently perform speaker diarization based on low-level video and audio features, before performing optimal matching to combine both resulting outputs. However, after further experimentation, Bost et al. consider the obtained performance is not sufficient and eventually turn to manual annotation [16].

Like for character occurrence detection, certain authors adopt multimodal approaches. Some prefer to extract the transcript of the work and apply text-based methods instead of directly using the video or audio stream. This is the case for Chen et al. [22], who, after having detected character occurrences in this text, use existing off-the-shelf co-reference tools to detect chains of mentions to the same character. They identify the concerned character by using predefined rules, exploiting the presence of a proper noun in the chain, or connections to utterances whose speaker could be identified. They propose an automatic method based on agglomerative convolutional networks to take advantage of the latter type of information when solving co-references and identifying characters associated to co-reference chains. In Reference [19], Bredin and Gelly combine face track clustering and speaker diarization: they detect speakers through standard speech activity detection tools before using face embeddings to cluster the face tracks corresponding to the resulting speech segments.

An additional issue specific to audiovisual narratives is to determine the name of the characters detected by grouping faces or speech segments. In the former case, this is called *Face-name matching*, and in the latter, *speaker identification*. In both cases, solving this problem requires using linguistic information: speech content (via transcripts, scripts, or subtitles), text overlaid in the video, predefined list of characters, other external resources. For instance, based on the assumption that the title of the fiction work is known, Tran et al. [107] retrieve its list of characters from IMDb, look for their picture using Google Image, and leverage this information to infer character names in the movie through matching. In the context of fiction works, though, the favored approach is to leverage scripts [52, 65, 127]. This again raises the issue of *script alignment* (with the corresponding video, transcript, or subtitle), as scripts are not time-stamped. After alignment, the script directly allows recovering speaker names and inferring addressee names (for instance, by crosschecking names mentioned in the conversation and on-screen faces).

For comics, a variety of methods has been proposed for *character detection* (or *face recognition* [103]), i.e., to match the multiple occurrences of a character's face. The general approach consists in defining some form of similarity measure, which is then leveraged to group occurrences corresponding to the same character. This is what Takayama et al. [103] do, based on the features they use for face detection (cf. Section 3.2.3). Stricker et al. [100] adopt a similar approach, but to compare their sets of facial landmarks. The problem is difficult and still open; its resolution will likely require large annotated corpora [9].

4 INTERACTION DETECTION

Based on the character occurrences, the next step of the extraction process consists in detecting all interactions happening in the narrative between each pair of characters. Such an interaction can be explicitly described, but also inferred from the narrative, depending on what one considers to be an interaction. We identify five distinct approaches in the literature:

The first (Section 4.1) is co-occurrence-based and relies on a decomposition of the narrative into smaller narrative units. Two characters are considered to interact when they *jointly appear* in the same such unit. The second approach (Section 4.2) considers only *direct verbal interactions* between the characters. This is particularly appropriate for dialogue-oriented narratives such as plays. The third (Section 4.3) requires one character to *explicitly mention* another one to infer an interaction between them. The fourth (Section 4.4) takes into account other types of *direct interaction* than conversation (e.g., fighting, kissing). The fifth (Section 4.5) focuses on explicitly expressed *affiliations*, such as family relationships or being coworkers.

4.1 Co-occurrences

The co-occurrence-based approach is the most widespread in the literature, probably because it is the easiest to apply: detecting interactions in a more precise way can be a difficult problem, even for humans [90]. This approach consists in breaking down the considered work into smaller *narrative units* and in assuming that two characters interact when they occur together within the same unit. A few authors use additional constraints to ensure that co-occurrences actually capture interactions. Some want the narrative unit to contain *only* the two characters of interest, and no one else [30, 48]. Others take into account only *consecutive* occurrences, i.e., not separated by another character [41].

Using co-occurrences presents several limitations mainly caused by their imprecise nature. Indeed, co-occurrence is only a proxy for actual interaction, as it is possible for two characters to appear together without interacting at all (e.g., they both are spectators of some event [73] or one mentions the other in his absence [84]). The first limitation is that this imprecision propagates to the network itself: the set of co-occurrence-based interactions theoretically contains the conversation-, mention-, action-, and affiliation-based ones, plus some false positives. In practice, though, Ardanuy and Sporleder [8] argue that false positives are rare in the sense that two co-occurring characters are almost always related in one way or the other. But this holds only for their experimental results, obtained by integrating co-occurrences over whole narratives.

The second limitation also directly comes from the imprecise nature of co-occurrences. As they encompass a number of different types of interactions, it is not possible to assign them a direction, and they are therefore regarded as some form of bilateral interaction. Furthermore, for Kwon and Shim [60], due to their imprecise nature, co-occurrences ignore intimate aspects of interactions, such as opinions and emotions. The third limitation, according to Prado et al. [84], is that using co-occurrences results in more importance being given to otherwise minor characters, when later analyzing the obtained character network. As discussed in our introduction, all these arguments must be balanced accordingly to the possibly very specific nature of the considered narrative.

We discuss the choice of the narrative unit in Section 4.1.1, as it depends on the type of narrative and can affect the end result. Besides the detection of interactions under the form of co-occurrences, certain authors additionally assign a numerical score to each interaction to include more information in the character network eventually extracted; we review such approaches in Section 4.1.2.

4.1.1 Narrative Unit.

Novels. In novels, Rochat and Kaplan use the page as a narrative unit [89], as imposed by the predefined character index they leverage during character identification (see Section 3.1). Such a partitioning of the text, based on purely *physical* (and therefore arbitrary) aspects, results in the possible split of chapters, paragraphs, or even sentences. It is therefore very likely to miss co-occurrences. Rochat and Kaplan try to overcome this problem through a two-page unit with a one-page overlap (instead of consecutive pairs of pages).

Other authors use smaller narrative units that can be considered as more natural, in the sense that they at least avoid such arbitrary splitting: 1 sentence [62], 10 sentences [81], 1 paragraph [30], 10 paragraphs [32]. However, the length of sentences and paragraphs can vary considerably from one writer to another, to the point where one writer's paragraph can be shorter than another's sentence [48]. Using word spans instead solves this problem: the literature contains examples ranging from 5 [47] to 1K words [42]. However, word spans are as arbitrary a narrative unit as the page and suffer from the same limitation.

Certain authors prefer to use a larger unit, especially the chapter [8, 55, 73] or the pericope [40] (its counterpart in the context of biblical writings). Like the sentence and the paragraph, however, it does not lead to split sentences, but its size can vary significantly from one writer to another. Moreover, it can be considered as too long a narrative unit for this usage, as many events can take place in the same chapter. Bolioli et al. [11] use what they call a *narrative sequence*: a small portion of text characterized by its unity of location and involved characters (basically, a scene).

Scripts. For scripts, certain authors use the line [120], but the most widespread narrative unit is simply the scene, e.g., References [27, 39, 85, 102]. However, as noted by Suen et al. [102], if a script contains long scenes, this can lead to the connection of characters involved in completely different parts of the story. To solve this issue, Stiller et al. [97, 98] prefer to use *subscenes*, a concept similar to Bolioli et al.'s narrative sequence.

Comics. In comics, Rochat uses a two-page narrative unit [90], like he does for novels (see above). In References [5, 38], the authors use the comic book issue, but this narrative unit is dictated by the predefined database they use as raw material. There seems to be a lot of room for exploration here, as comic book formats are very diverse, and there are very few articles dealing with this question. For instance, a seemingly natural choice would be to transpose the concept of scene and consider a sequence of panels involving the same characters as a narrative unit. However, this requires efficiently solving certain lower-level problems, in particular *panel identification* and *panel ordering*. Existing methods to detect the boundaries of panels take advantage of the black lines generally outlining them, or of the white space called *gutter* separating them [9, 99]. But a number of artists use complex page layouts, which makes both panel detection and ordering much harder [99].

Videos. In videos, certain authors use a fixed time interval as a narrative unit, e.g., 10s [63], 1min [90]. Similarly to using the page as a narrative unit for novels, this approach seems quite arbitrary. Indeed, it is likely to split the plot at any moment, including, for instance, the middle of an action. Several more *natural* units are traditionally used when dealing with videos [82]: The *frame* is the smallest, it is a single image. The *shot* is a sequence of frames continuously recorded by a single camera. The *scene* is a collection of consecutive and semantically related shots. The *group* is a logical subdivision of a scene (e.g., a conversation in a scene containing several of them [82]), and the *sequence* is a series of consecutive scenes constituting a short story.

A number of authors prefer to work with scenes, but this requires solving the difficult problem of scene boundary detection. Certain authors perform this task manually, e.g., Weng et al. in Reference [117]. They later adopt a semi-automatic approach, consisting in applying first an existing automatic tool and then manually detecting and correcting its errors [118]. Others proceed automatically, using an off-the-shelf tool [109, 127]. In Reference [106], Tran et al. segment movies by detecting stable periods in terms of character on-screen presence and merging them according to criteria related to character similarity and temporal proximity.

4.1.2 Interaction Score.

Positive Scores. While most authors adopt a Boolean approach when detecting interactions (i.e., presence vs. absence), certain authors try to assess their *intensity* under the form of numerical scores. Some use the distance between the two concerned character occurrences, expressed as a number of some smaller narrative unit separating them (or, more exactly, some decreasing function of this quantity). For instance, Park et al. [81] use a narrative unit of 10 sentences and experiment with two distance functions: the numbers of words and sentences separating the two occurrences. Grayson et al. [41] count the number of tokens between occurrences.

Others use the duration of the co-occurrence, e.g., the number of lines spoken in the scene [102], or the number of seconds during which both characters jointly appear on screen in a video [106]. Rieck and Leitte [85] use the proportion of words spoken by the considered pair of characters, relative to the total number of words spoken during the scene. This amounts to giving more importance to the most talkative characters. Yeh et al. [121] take advantage of film-editing guidelines, especially the *180 degrees* and *shot alternation* rules (cf. Section 1.2): their score is the number of consecutive shots showing alternatively both concerned characters.

Signed Scores. A few authors try to detect the *polarity* of an interaction, i.e., whether or not it is friendly or hostile, by leveraging the context of the co-occurrence. This results in a *signed* score (i.e., it can be negative). Some adopt an *indirect top-down* approach consisting in assessing the general polarity of the concerned narrative unit and extending it to all interactions occurring in it. Min and Park do so on novels through sentiment analysis [72], and Ding and Yilmaz on movies by training a classifier using a set of visual and auditory features [27]. They later experiment with affinity learning to take advantage of *visual concepts*, a higher-level information describing the scene *context* through the objects and environments it involves [28]. Alternatively, the *indirect bottom-up approach* consists in inferring the polarity of the relation based on the emotional state of the concerned individual characters. Lee et al. do so for movies [63] through off-the-shelf tools leveraging both conversational (movie script) and visual (facial expressions) cues. In comics, a similar approach could be applied by exploiting graphical elements such as effect lines, characters' facial expressions and poses, and conventional symbols used to reflect emotions. However, this is not possible yet, as only a handful of articles propose methods to detect such elements [9]. Finally, certain authors propose a *direct* way to monitor the polarity of an interaction. For novels, Chaturvedi et al. [21] use a Markov model to represent the chronology of the interactions between a given pair of characters and detect friendly vs. hostile phases based on the surrounding text.

4.2 Conversations

A number of authors focus only on verbal interactions between two characters, i.e., when one explicitly *talks* to another, to later extract so-called *conversational networks*. Technically, the process of detecting them is generally harder than for co-occurrences, as one has to detect that a conversation is taking place, as well as to distinguish the involved speakers and addressees. A fundamental difference between co-occurrence and verbal interaction is that the latter is naturally *unilateral*: one character talks, the other listens. This generally leads to the extraction of *directed* networks [82].

As explained before, verbal interactions can be assumed to be subsumed by co-occurrences, as two characters need to co-occur to converse, but can co-occur without necessarily talking to each other [16]. One could therefore suppose that only focusing on conversations leads to some information loss. However, certain authors argue that this is not the case, as many aspects of interpersonal relationships [43], if not most [116], are conveyed through conversation. The validity of this argument actually depends on the type of considered narrative, or even on the specific narrative itself. Several authors note that conversational networks are suitable only for narratives rich in verbal interactions [62, 72]. Lee and Yeung [62] give the counterexample of the *Book of Genesis*, in which the tense relationship between Abraham's wife and servant is mentioned frequently in the text, while they never speak to each other. Ardanuy and Sporleder [8] identify plays as the most appropriate form of narrative for conversational networks, as they are essentially scripted dialogue, by comparison with novels, in which most of the action takes place off-dialogue (e.g., McCarthy's *The Road*, Yourcenar's *Mémoires d'Hadrien*). Visual media such as comics tend to rely only lightly on dialogues.

4.2.1 Textual Narratives. In text, one can distinguish utterances (quotations of a character) from proper narration parts (description of the action occurring). Verbal interactions can take one of two forms: *direct* speech, which consists in explicitly quoting the utterance, and *indirect* speech, which consists in reporting what the speaker said (e.g., "Arthur told his knights to go home").

Conversational networks are built upon the former, whereas the latter belongs to the class of direct interaction considered in Section 4.4.

Identifying verbal interactions in fictional free text is a difficult task. First, the text generally contains much more proper narration parts than utterances, and both are tightly intertwined (e.g., speech verbs connecting utterances during a conversation). Second, the typographical conventions used to identify utterances in the text are not always respected (not all spoken text is quoted), and are sometimes ambiguous (e.g., sneer quotes). Third, the speakers are often not identified explicitly [37], and when they are, this information can be embedded in the narration. Therefore, automating the detection of verbal interactions requires the resolution of several NLP-related problems: detecting utterances, assigning a speaker to each of them, and possibly also identifying its listeners. Maybe for this reason, certain authors decide to proceed manually [35].

By comparison, semi-structured text is easier to handle, as typography generally allows distinguishing utterances from the rest of the text, and speakers are explicitly indicated. Thus, the only remaining problem is to identify addressees. As for novels, this task is difficult to automate, which is why certain authors perform it manually [74].

Quote Detection. It is relatively straightforward to detect quotes when the text is clean by simply relying on the presence of quotation marks [20, 32, 116] and/or using simple rules [123]. It is worth noting that some complications can appear, depending on the file encoding: for example, Unicode includes tens of distinct glyphs likely to be used as quotation marks.

But quotes are not always used correctly, or the considered language can be without typographic quoting convention, e.g., old Chinese punctuation style [61]. In this case, the authors rely on the presence of quotative verbs and on the sentence structure. Zhang et al. [126] train a decision tree based on morphological and typographical features to recognize quoted speech and detect speaker changes between two consecutive utterances. Mamede and Chaleira [69] propose 12 heuristics to identify both direct and indirect utterances in Portuguese. These are based on typographical clues, verb tense, presence of certain pronouns, temporal adverbs, and interjections.

Quote Attribution. The literature contains a number of articles dealing with quoted speech attribution for general texts [31], but we focus on the methods proposed specifically to handle fictions. For a given utterance, one can distinguish two cases: either the speakers are mentioned explicitly (be it by their name or some anaphora) or implicitly and can be inferred from the context.

The most widespread method to deal with *explicit* speakers relies on detecting the speech verb associated to the concerned utterance and considering its subject as the speaker. Certain authors propose to manually define [43, 116] or automatically learn appropriate rules [37]; others look for patterns [57, 61]. In the absence of a speech verb, certain authors proceed similarly with nearby action verbs [69], whereas others look for the nearest character around the utterance [126].

The first step in identifying *implicit* speakers is generally to select a subset of candidates among all known characters. One first has to estimate the boundaries of the conversation; this can be done by leveraging long proper narration parts separating sequences of utterances [116]. The candidates correspond to the characters present in the obtained portion of text. A number of authors assume that the characters involved in the conversation respect certain rules relative to conversation turns (a.k.a. the *conversation turn assumption*) such as: consecutive utterances are not spoken by the same character or one speaker answers to the previous one. It is then possible to leverage the explicit speakers identified before to infer the missing ones. Certain authors identify speaker alternation patterns and use them to define rules [57, 61, 116], or to learn them in a supervised way [37]. Some authors additionally use gender compatibility to improve utterance attribution [43].

Some authors try to detect both implicit and explicit speakers simultaneously. Elson and McKeown train a classifier [32, 33] based on standard low-level features as well as the syntactic category of the utterance and the identity of the last speaker. O'Keefe et al. [80] consider the conversation turn assumption as unrealistic and consider instead the problem as a sequence labeling one. Using the same features as Elson and McKeown, they experiment with various sequence decoding methods. Finally, some authors assume speakers can be distinguished depending on the topics they talk about. For instance, Celikyilmaz et al. [20] build an Actor-Topic model (ACTM) by leveraging textual content and use it to predict who speaks a given utterance.

Addressee Identification. The addressee is sometimes mentioned *explicitly* in the utterance when the speaker directly calls him. Certain authors define specific rules to take advantage of this situation [75]. Krishnan and Eisenstein [56], in particular, focus on the level of formality of these verbal interactions. He et al. [43] train a classifier to detect these cases based on features related to punctuation and typical forms of interjection.

When the addressee is *implicit*, one uses the *conversation turn assumption* already leveraged for speaker identification. Certain authors define rules assuming that a character talks to the preceding speaker in the conversation [60, 76], or the few preceding speakers [102], and/or to the following one [70, 75, 83]. In the case of conversations involving more than two characters, certain authors assume that a speaker talks to everyone present [75], which can also be considered as co-occurrence with an additional constraint (not just being present, but also speaking). Yeung and Lee [123] propose a conversation turn–based approach to assign simultaneously both speakers and addressees to utterances. It relies on two CRF classifiers trained using morphological, grammatical, and positional features.

4.2.2 *Visual Narratives.*

Comics. We could not find any article dealing with the extraction of conversational networks from comics, mainly because this requires solving a number of lower-level problems for which there is not much literature [86] and no efficient solutions yet [9]. The first ones are panel detection and ordering, already mentioned in Section 4.1.1. Then, it is necessary to detect speech bubbles and captions. This problem is difficult, because the shape and position of bubbles widely vary depending on the artist, culture, context, meaning, and a number of other factors. Existing approaches range from the exploitation of low-level color-based features (e.g., detecting white blobs) to adaptive outline detection methods [86].

One also needs to retrieve the shape of the bubbles, as it generally conveys a specific meaning: smooth for speech, cloudy for thoughts, spiky for screams. But there is no definitive convention; for instance, certain artists use captions to show the characters' thoughts. The location of the bubble is also important. First, relative to the other bubbles, it directly affects the reading order and is needed to properly extract a conversation. Second, relative to the characters, it indicates who speaks. Bubbles often possess a tail directed at their speaker, which helps solve the speaker attribution problem. Rigaud et al. [86] adopt a graph-based approach leveraging the distance between bubbles and characters and the angle of the tail. But again, using a tail is not a universal convention, which makes it harder to perform this task in general.

Finally, the last step is optical character recognition (OCR). Comics can be very challenging for standard OCR models [9] for a number of reasons: (1) many authors write by hand; (2) use of a variety of fonts, sizes, and colors, sometimes in the same bubble, sometimes mixed with pictographs; (3) the environment is very noisy due to the drawings surrounding the text, which is sometimes integrated to the background (e.g., sound effects).

Videos. There are not many articles describing the extraction of conversational networks based on audiovisual narratives either. The problems are roughly the same as for other media: detecting utterances, identifying their speaker, and possibly their addressees.

When using the video stream, one has to detect which parts of a face track correspond to a speaking character. Certain authors use off-the-shelf tools to identify lip motion [127], which allows simultaneously detecting utterances and assigning them to speakers. When using the audio stream, the speaker diarization tools described in Section 3.3.2 also solve the two same problems simultaneously. In theory, it is also possible to apply Automatic Speech Recognition tools to the audio stream to obtain a transcription and get access to the utterance content, or to use the script directly if it is available; however, in practice, we did not find any use of these approaches in the fictional network extraction literature.

Regarding the identification of addressee, one approach is to use the *conversation turn assumption* that we already mentioned for text. For TV series, for instance, Bost et al. [16] consider that a speaker talks to another character if the latter speaks before and after the considered utterance. For utterance starting or ending the conversation, they just consider the following or preceding speaker. When there are more than two speakers involved in the conversation, they use temporal proximity to determine the addressees. Alternatively, it is also possible to take advantage of lip motion detection: one can consider that characters appearing without moving their lips are addressees [82]. However, this approach has the major drawback of missing addressees that are not shown on-screen.

4.2.3 Score and Direction.

Score. The ways scores are handled for verbal interactions are quite similar to those already described for co-occurrences (Section 4.1.2). Some authors adopt a Boolean approach, only identifying whether a verbal interaction is occurring or not. This can be for the sake of simplicity [75] or by lack of technical resources [74]. A number of authors use a numerical score to represent the intensity of each interaction. The simplest approach is to rely on the length of the interaction, be it expressed in words in the case of text [49, 102] or some temporal unit in the case of videos [16]. Some authors prefer to compute the distance separating the utterances of the two concerned users, expressed in numbers of utterances, and apply a decreasing function, e.g., linear [102] or stepwise [83].

Like for co-occurrences, certain authors take advantage of the content of the conversation to compute signed scores reflecting the polarity of the exchange. In plays, Nalisnick and Baird [76] use an existing lexicon designed for sentiment analysis to associate a polarity (positive or negative) to certain words. The score of an utterance is the sum of these values over all of its constituting words.

Direction. Certain authors consider verbal interactions as symmetric [32, 74, 75, 116] in the sense that both the speaker and the addressee are interchangeable. It is an important simplification, though, as the information flows from the speaker towards the addressee, making the interaction asymmetric by nature; many authors prefer to consider that the speaker acts on the addressee [60, 61]. As we have seen before, the addressee is generally much more difficult to identify than the speaker, which could explain this choice. Certain authors go further in the simplification and consider that all characters participating in a conversation speak to each other [116], which is conceptually very close to simply detecting co-occurrences.

Interestingly, Park et al. [82] consider that a verbal interaction can involve a *single* character as both the speaker and the addressee, in the case of *soliloquy* (i.e., when one talks to oneself). In terms of graphs, this results in loops, or self-edges (an edge connecting a vertex to itself). Note that such graphs are relatively unusual in the domain of complex network analysis, and most descriptive tools are not designed to take advantage of this information.

4.3 Mentions

Certain authors focus only on explicit mentions of characters during conversations. Compared to the conversational approach described in Section 4.2, the difference is that they consider that there is an interaction not when one character speaks *to* another, but when he speaks *about* another. Thus, like for conversational interactions, this requires detecting utterances and their speakers, but not their addressees. Also like them, this approach is appropriate for speech-oriented media. Once the utterances are identified, one only needs to list which character occurs within them: this is trivial, since character identification is the object of the first step of the extraction process (Section 3). This is the approach adopted to study novels, e.g., Reference [32] (as a baseline, the main focus being conversational interactions) and TV or movie scripts [39].

As with other types of interactions, it is possible to compute a score to characterize the intensity of a mention-based interaction, although most authors do not in this case. Elson et al. [32] count the number of times the character is mentioned in the concerned utterance and divide by its length. Implicitly, they consider that the interaction is more intense if the utterance is short. Kwon and Shim [60] take advantage of the utterance content to compute signed scores in scripts. They look for utterances expressing the opinion of the speaker regarding the mentioned character. They use an off-the shelf sentiment analysis tool to detect the polarity of the utterance and perform a manual pass to improve the result.

As a mention is by nature an asymmetric interaction, in the sense that some character performs an action involving another one [39], most authors consider them as such. However, a few authors choose to ignore this information [32, 75], either because they do not consider it as useful for the task at hand or to simplify the processing.

4.4 Direct Actions

Instead of focusing only on verbal interactions, certain authors take into account all forms of direct actions that one character can perform on another (e.g., thinking about someone) or that two characters can perform jointly (e.g., fighting). By comparison with conversational interactions, detecting *general* actions allows handling narratives in which most of the interactions takes place off-dialogue (see Section 4.2). However, the task is even more difficult, as one needs to identify the action taking place as well as the characters performing and undergoing it.

Some authors dealing with textual narrative proceed manually, possibly because they focus on certain classes of actions, semantically speaking: one character supporting another [14], physical encounters [46], actions that influence the development of the story [113]. In Reference [24], Cipresso and Riva are interested in identifying the interactions between movie characters according to four basic emotions: anger, fear, sadness, and joy. For this purpose, they rely on a survey conducted over a sample of 11 persons watching the movie and detecting these interactions manually.

The automatic approaches proposed in the literature are all designed for textual narratives. Indeed, processing visual narratives implies recognizing actions in videos or images, which is a difficult problem called *human-human interaction detection* [114]. Generic methods exist for videos [114], some of which have been applied to works of fiction (mainly movies). However, the literature does not seem to contain any article leveraging them to extract character networks. For comics, the problem can be considered as even harder due to the static nature of the medium: state-of-the-art works only focus on the lower-level task of *pose detection* [9]. Most automatic methods for text first employ parsing techniques to detect so-called SVO triplets (subject-verb-object) before filtering them to retain only those involving characters as subjects and objects [96, 108]. Certain authors take advantage of external resources such as *FrameNet* to additionally filter the remaining triplets by focusing on certain classes of verbs, semantically speaking. To summarize, FrameNet is a database containing *frame semantics*, i.e., structured representations of situations involving various agents and objects and semantic relationships between them (e.g., generalization) allowing to perform inference. Certain authors use predefined classes of interest, e.g., social interactions [62], while others estimate them ad hoc, e.g., the 50 most frequent [101].

Certain types of actions can be considered as naturally *unilateral*, as one character acts on another one, whereas others are *bilateral*, as several characters act together. In terms of graph extraction, this means extracting directed vs. undirected edges. Certain authors consider only unilateral [101] or bilateral actions [62] and ignore the rest. Others use both, but consider all of them as bilateral to avoid the identification of the subject and object characters, and therefore significantly ease the process [108]. Agarwal et al. [2] distinguish between two types of actions: unilateral ones in which the object character is not aware of the action occurring (e.g., thinking about someone), which they call *observations*, vs. all other actions. They handle them separately to compare them relative to the task at hand.

Very few authors compute a score to represent the intensity of the interaction. Trovati and Brady manually identify three categories of verbs [108]: *friendly*, *hostile*, or *unknown*. They ignore interactions described by verbs belonging to the latter and assign a signed score to the others, depending on the category. Srivastava et al. [96] use sentiment analysis to take advantage of the semantics of the textual context and associate a polarity to the interaction (*cooperative* vs. *adversarial*).

4.5 Affiliations

Unlike the other types of interactions listed in this document, affiliations do not correspond to actions but rather to states. Among others, affiliations include: being blood-related, being married, and belonging to the same social group. They are explicitly mentioned in the narrative, which means that they can appear either in the conversation or in the narration. It is clearly the less-popular approach, as only a few authors leverage this type of interaction. Moreover, the methods proposed in the literature only concern textual narratives.

Srivastava et al. [96] leverage an *a priori* selection of keywords (e.g., father, wife) to identify family relationships explicitly mentioned in the text. Krug et al. [58] propose a semi-supervised method to train a maximum entropy classifier in detecting family, romantic, professional, and social relationships in literary texts. Starting from a set of annotated instances, it uses uncertainty-based active learning to select appropriate examples during training and ask the user to label them.

In Reference [62], Lee and Yeung detect affiliations using a heuristic approach, based on a dependency parsing and the linguistic resource *FrameNet*. They distinguish two cases: the relations stated *directly* ("Sherlock is the brother of Mycroft") and the ones mentioned *in passing* ("Watson was referring to Sherlock's brother Mycroft"). The authors treat both of them by looking for predefined patterns, designed using a development dataset. More specifically, they look for sentences involving several characters and fitting one of the manually selected semantic frames.

5 GRAPH EXTRACTION

At this stage of the process, the characters and their interactions have been identified. The last step consists in building the character network based on this information. This requires making two methodological choices: how to define vertices and edges.

Vertices. In almost all the approaches presented in the literature, characters are represented as individual vertices. However, a few authors alternatively consider certain characters *collectively* and represent them as a single vertex. This can be due to several reasons. The first is that some groups of people are mentioned in an indistinguishable way in the raw material; for instance, townfolk in Twain's *Huckleberry Finn* [55] or Olympian gods and Greek soldiers in Homer's *Iliad* [113]. Second, it can also be that some characters always appear simultaneously in the plot: considering them collectively can be viewed as some form of simplification. However, this can have an effect on many of the topological measures computed to describe the network, unless the multiplicity of the vertex is encoded in some way (e.g., vertex weight) and used when processing said measures. Third, it is possible that one wants to model the narrative at the level of groups rather than individuals (e.g., Reference [66]). Handling such collective vertices is difficult when their composition evolves over time. Moreover, it is possible for the same character to appear in the work of fiction both as an individual and as a part of a group. For these reasons, most authors choose to simply ignore such groups altogether, as they are generally secondary [2].

Edges. The methods proposed to define edges differ much more from one author to another, which is why we focus on them in this section. One has to consider several aspects of the interactions, which must be translated in graph-related concepts: their laterality, score, polarity, and temporality. The general approach is to represent unilateral interactions by directed edges, and bilateral ones either by undirected edges or by pairs of reciprocal directed edges. The scores computed by certain authors to measure interaction intensity are modeled by edge weights, which can be signed to represent the polarity of the interaction (friendly vs. hostile). This results in graphs that can be (un)directed, (un)weighted, and (un)signed. A signed network has both *positive* and *negative* edges and is used to model antagonistic relationships [44].

Interestingly, certain authors decide to ignore information that could be modeled as edge directions or weights, even if they have it at their disposal. This can be due to some methodological priorities; for instance, Yose et al. state that using directed edges to model unilateral interactions would result in a loss of statistical power when testing certain assumptions on the resulting network. The choice can also be caused by the will to adopt a simple approach. For example, Yose et al. justify their use of unweighted networks by the fact that they are interested only in the presence of a relation between two characters and not by its intensity [125].

The most important methodological issue one has to solve when extracting edges is to decide how to handle time. It is possible to consider the complete set of interactions occurring between two characters of interest over the narrative and create a single edge summarizing all of them; this results in *static* networks (Section 5.1). However, this complete temporal integration leads to some information loss. Certain authors propose methods aiming at solving this issue and produce *dynamic* networks (Section 5.2) instead.

5.1 Static Networks

In most of the approaches presented in the literature, the extracted character network is *static*, i.e., it represents the interactions between the characters for the period of time corresponding to the whole narrative. As interactions can occur anywhere in the storyline, obtaining such a network requires some form of temporal integration. On the one hand, the most widespread approach is to consider the set of successive interactions between two characters and derive an edge by applying a simple mathematical function, e.g., by counting them (Section 5.1.1). On the other hand, certain authors propose more advanced methods based on the explicit modeling of inter-character relationships (Section 5.1.2).

5.1.1 Count-based Approaches. The most widespread methods to derive an edge from a series of interactions are relatively simple, relying on the existence or number of such interactions. Certain authors then discard some of the resulting edges, which they deem unreliable. Finally, a different extraction approach consists in extracting networks whose vertices represent groups of characters and edges correspond to various forms of overlap between them.

Interaction Aggregation. The simplest form of temporal integration, and also the most widespread, results in an *unweighted* network. It consists in creating an edge between two vertices if at least one interaction is detected between the corresponding characters over the whole narrative, e.g., [46, 73, 74, 98]. A few authors extract bipartite co-occurrence networks in the first place [5, 40, 117], but then convert them in unipartite graphs for the same result. When interactions are described by their polarity, it is possible to produce a *signed* network: The sign of an edge is obtained by considering the majority sign among the set of interactions between the two concerned characters [108].

It is straightforward to generalize this approach to produce a *weighed* network, in which the weight of an edge reflects the overall intensity of all the interactions occurring between the two considered characters. Certain authors use a frequency-based weight, the most straightforward being the number of times they interacted [32, 61, 75, 89] or the proportion of interactions [101]. Amancio [6] applies a probabilistic normalization to the number of interactions, which favors the stronger relationships.

If the interactions are already associated to some numerical score describing their intensity, it is possible to combine them (instead of just counting the interactions). Certain authors sum the scores of all the interactions detected for the considered pair of characters [41, 52, 83] or take their average value [81, 93]. Alternatively, others normalize the sum, for instance, by dividing by the total network weight [82] or through some non-linear function [92].

Edge Filtering. Certain authors consider that the weakest relationships amount to noise and choose to remove edges with low weights. This can be done through a fixed threshold, e.g., at least three [89, 91] or five [48] co-occurrences to keep an edge, whereas Stiller and Hudson just remove isolates [97]. Park et al. use a normalized weight, for which they empirically define a minimal threshold [81]. Sudhahar and Cristianini [101] do not deal with a single narrative, but with a *collection* of related stories, at once. They filter relationships depending on the number of documents in which they appear. The same method could be applied to subdivisions of a single fiction work. In Reference [92], the authors transform their weights so they represent distance in place of intensity. They then extract the minimum weighted spanning tree of their co-occurrence network and use it in place of the original network, therefore discarding the rest of the edges.

Networks of Character Groups. Certain authors aggregate interactions, and more precisely co-occurrences, not only temporally, but also over *groups of characters*. Tsai et al. [109] extract a direct network in which a vertex corresponds to a group of characters co-occurring during the same scene, and possibly several ones. An edge going from one vertex to the other represents an inclusion between the character sets they represent. As described in Section 6.6, this type of network is designed specifically to summarize the plot. Grener et al. [42] proceed similarly, but an edge represents any non-empty intersection between two groups (instead of strict inclusion). Hettinger et al. [45] model temporal proximity rather than group similarity by connecting two groups appearing successively in the plot.

5.1.2 Model-based Approaches. Certain authors propose specific models to represent the relationships between two characters, taking advantage of the sequence of interactions between them, but also possibly of additional information such as the context of these interactions in the narrative. All these models rely either on co-occurrence- or conversation-related interactions.

Co-occurrences. Elson et al. [32] describe characters by vectors constituted of their numbers of occurrences during each unit of the considered narrative. They compute the correlation matrix over this representation of the character set and use it as a weighted adjacency matrix to obtain what could be called an occurrence similarity network. Hutchinson et al. [47] propose to compare the context in which the characters occur in terms of topics. They first perform a Latent Semantic Analysis (LSA) to identify the lexical context of the occurrences, followed by a Singular Value Decomposition (SVD) for dimension reduction. In the resulting matrix, each character is represented by a real vector. They then use the cosine similarity to compute the weight associated to each pair of characters. The result is a topic similarity network. Nan et al. [77] proceed similarly with TV series by combining visual (objects surrounding the characters at the time of the occurrences) and textual information (subtitles).

Ding and Yilmaz [27] assume the existence of two conflicting groups of characters in the plot. As explained in Section 4.1.2, at the interaction detection step they leverage the narrative to compute a score representing the polarity of each scene. They then propose a Bayesian model taking into account the sequence of scene polarity scores and each character's group membership. It allows estimating the overall signed weight representing the polarity of the relationship between any two characters for the whole narrative, resulting in a signed weighted network. In Reference [96], Srivastava et al. define a supervised approach to estimate the polarity of the relationships. They first extract an unsigned graph based on co-occurrences, affiliations, and other interactions. They then train a classifier to predict the polarity of the relations, based on a number of textual features describing the context of the interactions, as well as structural features related to the notion of structural balance (categories of signed triads).

Conversations. Celikyilmaz et al. [20] focus on the content of the utterances. They propose a Bayesian model describing each character in terms of the topic about which he talks. From this, they build a topical similarity network: a fully connected graph in which the weight of the edge connecting two characters reflects how much they talk about the same things. According to the authors, this allows identifying hidden relationships between characters.

In Reference [56], Krishnan and Eisenstein take advantage of *how* two characters address each other over all their conversations to estimate the nature of their relationship: *informal* vs. *formal*. The resulting network is therefore signed, positive and negative edges representing informal and formal relationships, respectively. In addition to linguistic content, their probabilistic model is able to take into account the structure of the graph, for example, by assuming it is consistent with structural balance.

5.2 Dynamic Networks

Static networks present an obvious limitation, as identified by numerous authors in the context of character networks [7, 16, 78, 89]: they result in a significant information loss, as they completely hide the chronology of the interactions. Yet, the order in which events occur is crucial to characterize a story, and for the writer it is generally at the core of the writing process. Moreover, the relationship between two characters is likely to evolve with the plot. This can lead to poor performance when solving certain problems based on a static network, or even completely prevent any resolution. Even from the descriptive point of view, Prado et al. [84] show empirically that the most central characters detected in static vs. dynamic networks can differ dramatically.

Certain authors try to cope for this by allowing multiple edges between vertices. For instance, in their signed network, Yose et al. [125] can have both a positive edge and a negative one connecting the same pair of characters to model a relation evolving over time. Although simple, this is a very imperfect and incomplete solution. Another possibility is to extract a dynamic network. It takes the form of a sequence of character graphs called *time slices*, each one corresponding to a certain sub-period of the story. Like a static graph, a time slice is obtained through temporal integration, as it represents all the interactions occurring over a period of time. But this integration is performed at a much smaller time scale, which allows limiting the information loss. We distinguish two types of approaches: those using a fixed-size window to obtain these time slices (Section 5.2.1) vs. the others (Section 5.2.2).

5.2.1 Temporal Window. The fixed-size temporal window approach is clearly the most widespread. The notion of window depends on both the nature of the considered narrative and the utilization of the extracted network. The literature exhibits a variety of window sizes. For novels, most authors use one chapter [2, 7, 41, 84]. Grener et al. [42] use a 1K-word window, whereas Seo et al. [93] arbitrarily split the novel in 10 equal-sized pieces. In theater scripts, Kwon and Shim [60] use a one-act window. For movies, authors use one scene [28] or automatically detected segments roughly corresponding to scenes [63]. For TV series, authors use one scene [16], one episode [66], or one season. Furthermore, certain authors use an overlap between consecutive windows to preserve continuity as much as possible, e.g., for novels: 15 pages with a 14-page overlap [89], 100 sentences with a 10-sentence overlap [78]. Weights can be processed for each slice, similarly to what is done for static networks (cf. Section 5.1.1).

Certain authors extract a dynamic but *cumulative* network [73, 76, 116]. This means that, to obtain the graph representing a given moment in the narrative, they do not use only the corresponding time slice, but rather the portion of the narrative going from the very start to the considered moment. Other than that, the principle is the same as before: they perform a temporal integration over this period and do so for every moment in the narrative to get a series of graphs constituting their dynamic network. As before, some authors consider unweighted graphs [116], and others use the number of interactions [73, 120] or the total interaction scores [107] as edge weights. Nalisnick and Braid [76] proceed similarly, but as their interaction scores are signed, the weights of the edges do not necessarily increase with time. On the contrary, one of the benefits of their approach is that a moment in the plot corresponding to a sign inversion for a given relationship is supposed to be narratively important for the concerned characters.

Generally speaking, using a fixed-size window to discretize a time series requires knowing which window size to use. This is no trivial task, and it was shown that this parameter can have a very strong effect on the extracted network and consequently on the process conducted on this network [25]. With character graphs in particular, on the one hand, too-large windows will miss many details, as relationships in works of fiction can evolve quickly; on the other hand, too-small windows will result in very unstable networks that may hide relevant information [78]. Bost et al. [16] have shown this experimentally, and also that a window too small can lead to mistaking irrelevant events as important ones. Finally, a window too large may also cause problems when later studying the network: When the time slices differ much in terms of involved characters, it makes little sense to compare them through topological measures, as these are generally defined in a relative way [84]. Grayson et al. [41] propose to increase the window size until the graph density reaches a plateau, but this can produce very dense networks, likely to be uninformative.

5.2.2 Other Methods. In addition to the difficulties of estimating the best window size, there is no reason to suppose that this size should even be fixed: the tempo of the narrative is likely to change in any way, e.g., accelerating during action scenes and slowing down during emotional ones. Very few authors try to take this point into account.

Instead of considering fixed-size time slices, Mutton [75] uses an event-based approach to extract the sequence of graphs constituting his dynamic network. Each modification (creation or deletion of an edge, revision of a weight) results in the addition of a new graph in the sequence, representing the state of the character relationships after the modification. In some sense, this is the smaller possible time slice. Like some methods from Section 5.2.1, he adopts a cumulative approach. However, he also includes a temporal decay mechanism that decreases the weights of an edge depending on the last time it was updated.

Another important point is that the narrative is linear, but not necessarily the story or even the plot.[3] The narrative does not reflect the full state of all inter-character interactions at all times:

[3]The notions of plot (*what* is told), narrative (*how* it is told), and story (*perceived*) are defined in the introduction.

It focuses on certain characters, presents under a sequential form some events actually occurring simultaneously, shows past events as flashbacks. A character can be absent from the narrative for a while but still be active under the scene, performing actions that will be revealed later to the audience. Some authors consider it necessary to model this hidden evolution to represent fully the character graph. Elsner [30] does so by applying a linear interpolation to determine the weight of a relation at a given point in the narrative for which this relation does not appear explicitly.

In Reference [16], Bost et al. propose *narrative smoothing*, an interpolation method allowing both to solve this problem and to mitigate the fixed-size window issue. Starting from the list of verbal interactions occurring at each scene, and weighted depending on their duration, they apply the following transformation to get their smoothed dynamic network: For a given pair of characters, if a verbal interaction occurs at the considered scene, the smoothed weight is the raw weight (duration). Otherwise, they take into account the last and next interactions of the considered characters: The raw weight is decreased depending on how much the characters interact with *others* in the meantime. Note that this decrease does not depend on the amount of time between two interactions of the considered characters to avoid the fixed-size problem. This method allows estimating the weight of the relationship between any two characters at any moment of the narrative, even if they are not interacting at the time.

6 APPLICATIONS

In this section, we present a selection of methods and results aimed at solving a specific problem and relying on character networks for their purpose. We overlook the many articles from the field of literary studies that focus on a single fiction work (or a few) and aim at describing it in great detail [89]; these are case studies, so their goal is not to propose a generic tool that one could subsequently apply to other works. We also ignore articles only using character networks for visualization [42, 113, 120], as they are somewhat limited, at least from this perspective. Finally, we also ignore purely descriptive tools that rely on various topological measures to characterize networks. The main such measures are described in the *Supplementary Material*, with a focus on their interpretation in the context of character networks.

6.1 Assessment of Literary Theories

Some articles aim at assessing the validity of literary theories. Those are often evaluated qualitatively and/or on a small number of fiction works [32]. Automated approaches can help at testing them in a more quantifiable way and/or on larger corpora.

In Reference [32], Elson et al. focus on two literary assumptions related to the size of the community surrounding the protagonist in 19th century British novels. The first one is that it depends on the amount of dialogue occurring; it would not be possible to show many conversations when there are many characters to consider. This is formally translated as the presence of a strong correlation between the amount of dialogue and the number of characters in the novel. The second assumption is that it depends on the social setting of the novel; there would be more verbal interactions in rural than urban communities. This is handled by manually determining the setting of the considered novels. Elson et al. adopt conversational networks to model the 60 classic novels constituting their corpus. They compute using a number of features, some describing the networks (numbers of 3- and 4-cliques, average degree, density) and others extracted from the detected verbal interactions (e.g., numbers of speaking and non-speaking characters, variance of the number of utterances by character). They find a slightly positive correlation between the numbers of characters and utterances and an even stronger one when counting only speaking characters; this invalidates the first theory. They do not observe any significant difference in the numbers of characters (speaking or not) or any tested feature relative to the type of setting (rural vs. urban), which invalidates the second theory. Instead, the most influential factor seems to be the narrator's perspective: characters are much more loosely connected when the story is told in the first-person than in the third.

Jayannavar et al. [50] later revisit these results using a different analysis method. Some other authors consider different theories: Grayson et al. [41] focus on social exclusivity in the social systems described in Austen's novels through the co-occurrence network of *Sense and Sensibility*; and Falk [35] studies certain aspects of the *Bildungsroman*, a category of novels dealing with the formative years of young characters.

6.2 Level of Realism and Historicity

A popular problem is to determine whether the network of social interactions extracted from a fictional narrative is realistic in the sense that it displays topological properties similar to real-world social networks. Stiller et al. [98] study a corpus of 10 plays by Shakespeare in an effort to determine whether the success of the playwright relies on his ability to mimic some basic properties of real social networks in his writing. Alberich et al. [5] and Gleiser [38] assess how realistic cooperation between Marvel characters is through the comparison of their co-occurrence network with real-world collaboration networks (e.g., IMDb, DBLP). In a series of articles studying myths, tales, and other mythological stories, Mac Carron et al. [67, 125] assume that such narratives convey both actual and fictional information. They assess their historicity by determining where they stand between completely real and purely fictional social networks.

The approach adopted in these articles is globally the same: they describe their character networks using standard topological descriptors (e.g., degree distribution, average distance, transitivity) and compare their counterparts obtained for real-world and/or random networks. They generally find both similarities and differences. For instance, the Marvel network possesses certain properties observed in real-world networks: presence of a giant component, small-world (small average distance and high transitivity), scale-free (power law-distributed degree). However, it also presents important differences: Its transitivity is much lower, and its degree is much smaller, reflecting the fact that Marvel characters tend to collaborate more often with the same people than real-world agents do [5]. This can be explained by legal matters related to the *Comics Authority Code* [38]. In the case of myths, *parts* of the networks are realistic, whereas others are not. For instance, the Old English epic poem *Beowulf* is modeled by a relatively realistic network, provided one ignores the eponymous character; this is likely caused by the concentration of fantastic plot elements on him, whereas the other characters are based on real people [67].

6.3 Role Detection

A number of authors leverage character networks for role detection in plays, novels, and movies. i.e., assigning one among several predefined narrative roles to each character. A consensual definition would be that a character plays a role in the story in the sense that it follows "characterization archetypes" [89], such as being a protagonist or an antagonist. Once detected, roles can be used to solve higher-level problems such as plot classification (Section 6.4), storyline detection (Section 6.5), and plot summarization (Section 6.6).

The different detected roles differ significantly from one author to another, and some assign several roles to certain characters. In the simplest case, one wants to distinguish between *major* and *minor characters* [63, 70, 97]. Certain authors try to split the latter class into *supporting characters* and *extras* [82, 89]. Dealing with adversarial stories, in which two sides are opposed, adds another dimension: character alignment, i.e., whether a character assists or opposes the hero. This leads to the detection of protagonists vs. antagonists [60] and can be combined with the major/minor distinction to get a finer typology [52]. It is also possible to focus only on certain roles; for instance, Agarwal et al. [2] want to identify the main protagonist and the narrator (which is not necessarily a character). Finally, some roles are specific to the considered narrative; for instance, Pope et al. [83] distinguish cops, gangsters, and informants in the TV series *The Wire*.

All the methods appearing in the literature rely on nodal topological measures: Degree or strength are generally considered as the most discriminative, then transitivity, betweenness, closeness, and Eigenvector centrality. Most of them are generally distributed according to a power law (or at least a strongly heterogeneous distribution), which fits the assumption that there are many more minor characters than primary ones [74]. The simplest approach is to focus on a single nodal topological measure and use predefined or automatically estimated thresholds to distinguish between roles [52, 82]. However, most authors prefer to simultaneously use several such measures, as these are deemed complementary. They adopt various approaches: multiple thresholding [2, 60, 97], cluster analysis [70], supervised or semi-supervised classification [39, 83], definition of a composite measure [63, 106], and PCA (Principal Components Analysis) completed by visual inspection [89]. In addition to purely structural features, certain authors dealing with alignment-related or work-specific roles use the narrative content: part-of-speech associated to each role [83], sentiment analysis applied to the dialogues [52], or co-occurrence context [39].

6.4 Classification of Fiction Works

Based on the assumption that the character network extracted from a work of fiction is affected by a number of factors (time, space, genre, writer/director ...), several authors take advantage of the network structure to predict some of these traits. Most of them use a supervised approach: Suen et al. train a classifier to distinguish scripts based on various criteria (e.g., cinema vs. theater, publication year) [102]; Hettinger et al. classify German novels in terms of genre [45]; Li et al. do the same for plays [64]; Holanda et al. try to distinguish texts according to three classes (pure fictions, pure biographies, and legends—a mix of both) [46]. A few authors adopt a non-supervised, more exploratory approach, through clustering: Ardanuy and Sporleder group similar novels and study the uniformity of such groups in terms of genre and writer [7, 8]; Waumans et al. want to gather episodes of the same series. Rochat and Triclot [90] manually classify works in four plot classes (heroic-core, unicentered, acentric, polycentric).

The general approach consists in extracting the character network of the considered fiction work before computing a collection of topological measures used as features by the classifier: network size, density, transitivity, degree distribution, diameter, radius, centrality measures, and others. Li et al. compare these to their weighted generalization of graph motifs [64]. Most authors additionally use meta-data (medium type, genre, writer, date, length/duration) and content-related features (point of view of the narrator, word frequency, topics). Ardanuy and Sporleder [7, 8] also leverage the plot dynamics: They split each narrative into a few predefined phases (exposition, rising action, climax ...) and separately compute the features for each of them.

In the end, the nature of the most discriminative features largely depends on the considered data and classification task. For instance, Suen et al. [102] observe that the best features to distinguish between plays and movies are degree-related, because plays generally have a single largely dominating character, whereas movies possess several of them. When discriminating between fictions and biographies, Holanda et al. [46] remark that the most discriminant feature is the number of *hapax legomena* (single-occurrence characters), which increases with the level of realism of the book. They assume that novel writers tend to use characters several times once they have made the effort of introducing them, whereas character occurrences are dictated by historical events in biographies.

6.5 Story Decomposition

Character networks are used in the literature to break down a story into its constitutive parts. Solving this problem is relevant for numerous information retrieval tasks, such as plot summarization or comparison, and can be expressed under various forms.

Storyline detection consists in identifying the multiple subplots intertwined by the writer/director to constitute the plot of his narrative. The methods proposed in the literature are applied to movies [82, 117] and TV series [118]. They postulate that each storyline is organized around a leading character. They first use role detection to detect these main characters, and then community detection to determine their entourage. Each scene is then associated to one of the communities depending on the involved characters, and storylines are obtained by ordering them chronologically. Scenes involving characters from multiple communities are points of contact between the storylines: They generally correspond to key moments in the plot, and are therefore particularly interesting for higher-level tasks such as automatic summarization. Certain authors observe that main characters are likely to appear in multiple subplots and therefore assume that they can belong to several communities [82].

Story segmentation constitutes another form of story decomposition. It consists in splitting the plot into consecutive and meaningful phases. When dealing with movies and TV series, this amounts to gathering shots constituting scenes or substories (so-called *sequences*, cf. Section 4.1.1). Traditional methods focus only on low-level audiovisual features [118], but recent approaches take advantage of the plot-related semantics encoded in character networks. Weng et al. [118] define a character similarity measure based on neighborhood proximity in the character network, then use it to compute scene similarity based on the similarity of the involved characters. The plot is then split by identifying strongly dissimilar consecutive scenes. Min and Park [73] want to detect standard plot phases (exposition, rising action, climax, falling action, resolution) in novels. For this purpose, they use a dynamic cumulative network and detect significant increases of vertices and edges, as well as stable periods, to identify these phases. They subsequently use sentiment analysis and topic detection to improve their tool by taking the content into account.

6.6 Summarization

Several approaches use character networks to summarize audiovisual narratives in place of or in addition to the traditional low-level audiovisual features. This task consists in producing an extractive summary by combining important scenes picked from the original narrative.

Tran et al. [106] propose a method for the automatic summarization of movies. They first extract a co-occurrence network and use a combination of standard topological measures to identify the main characters. They then assign a so-called *social score* to each scene depending on the nature of the involved characters. They combine this score with video-related features such as inter-scene distance and duration to identify the scenes of particular interest, which are finally used to generate the summary. Bost et al. [15] propose an extractive method to summarize TV series seasons from the perspective of a character of interest (as opposed to summarizing the whole plot). They extract a dynamic conversational network using narrative smoothing (cf. Section 5.2.2) to take into account the non-linear nature of the narratives (e.g., parallel subplots). Their assumption is that important events occurring in the storyline of characters are associated to durable changes in their social environment. They use the vertex neighborhood as a proxy to such a social environment and detect these events by clustering the sequence representing the successive states of this neighborhood. Their algorithm combines this graph-based approach to a more traditional analysis of the filmmaking grammar to generate the character-oriented summaries.

Tsai et al. [109] use a network of character groups instead of individual characters (cf. Section 5.1.1): Each vertex represents possibly overlapping subsets of characters, and edges between them model inclusion relations involving these subsets. The authors assume that a movie is typically built by first exposing the characters in scenes involving few of them before developing the proper story by making more of them interact during more crowded scenes. They use an incoming degree-based centrality measure to identify the most important character groups in the plot and partition the network around them. The video summary is finally built by solving an optimization problem that consists in selecting the scenes involving the most central vertices (character groups) under some user-specified length constraints.

7 DISCUSSION, OPPORTUNITIES AND PERSPECTIVES

Our review of the literature reveals that there are still a number of issues to solve and directions to explore at each of the steps constituting the network extraction (Section 7.1) and exploitation (Section 7.2) processes.

7.1 Network Extraction

We identify three main types of open problems regarding the extraction of character networks: the improvement and development (1) of tools to identify characters and interactions in the various types of media used to create narratives (Section 7.1.1); (2) of approaches to construct dynamic character networks based on this information (Section 7.1.2); and (3) of evaluation methods to assess the performance of these identification and construction tools (Section 7.1.3).

7.1.1 Character and Interaction Identification. It appears from this survey that all media are not equal regarding the character and interaction identification processes. This is because they rely on the resolution of a number of medium-specific lower-level problems for which the state-of-the-art performance is not always as advanced in all domains.

With textual narratives, the literature shows that automatic methods work relatively well when dealing with the simplest tasks: Character occurrences can be identified through NER and detecting their co-occurrences is quite straightforward. A number of authors extract character networks based on these tools only. However, there is still work to do to leverage more of the information conveyed by text. Character unification is not efficiently solved, as it requires detecting all forms of character aliases and most of all performing anaphora resolution, which is a difficult NLP problem. There are also research opportunities with the identification of advanced types of interactions between characters besides simple co-occurrences. This task is much harder, as it requires analyzing the semantics of the text, for instance, by detecting action verbs and their respective subjects and objects. For all these open problems, as explained in the survey, the specific characteristics of fiction offer both additional difficulties and leverage compared to non-fiction content.

By comparison with text, the results obtained with automatic tools on audiovisual narratives are much less satisfying, to the point where many authors perform certain tasks such as face recognition manually. This difference might explain why there are many more articles dealing with text than audiovisual narratives. The low-level tasks necessary to detect character occurrences (face detection and tracking, speaker segmentation) and unify them (face track clustering, speaker clustering) are still open problems when performed in such uncontrolled conditions. Regarding interaction detection, in practice authors basically focus on co-occurrences only, as identifying more precise interactions requires solving even lower-level problems. For instance, to get a conversational network, one has to perform lip motion detection or speaker identification to determine who is speaking. Furthermore, interlocutors may be tricky to detect when multiple characters are involved in the same scene. Obtaining an action-based network would require determining which character performs which action on which other character [114]; there is no trace of such an attempt in the literature. It seems difficult to automate the whole character network extraction pipeline as long as all these basic processing steps cannot be efficiently performed. Moreover, unlike for textual narratives, here the specific properties of fiction result in significant difficulties compared to non-fiction and yield little additional leverage. On the bright side, multimodal approaches constitute a very promising perspective, as some authors try to combine information extracted from the video and audio streams, but also from text (scripts, transcriptions, external sources such as wiki pages).

This might be surprising, but comics appear to be even harder to process as a narrative than video content. This is because the information they convey can take a number of different forms (text, drawings, balloons, panels, onomatopoeia, effect lines, textures), each one needing some specific processing. Moreover, the medium itself is not very normalized in the sense that distinct artists may express the same ideas in very different graphical ways. As a result, all the (few) authors extracting character networks from comics operate manually. Detecting and unifying character occurrences in comics requires solving certain problems similar to those met for videos, such as face detection, but in an even more adverse context (non-anthropomorphic and/or highly deformed characters). Moreover, they also require performing specific lower-level tasks such as panel segmentation or bubble detection, which are still open problems. For interaction detection, like for audiovisual narratives, co-occurrences are easy to get once the characters have been identified (provided the panels have been properly segmented), but identifying more precise interactions is hard. Leveraging conversations would essentially depend on the extraction of text and locating of bubbles and captions, two difficult tasks that are not satisfactorily tackled by state-of-the-art methods. Identifying specific actions that characters perform on each other would require associating semantics to specific body postures, which is also an open problem. To finish with comics, it is worth noticing that we could not find any article dealing with *photonovels*; on the one hand, their economical interest is questionable, as they felt out of fashion, but on the other hand, they might be easier than comics, as they rely on photographs instead of drawings while retaining certain conventions of comics.

To summarize, independently from the type of medium, there are many low-level open problems regarding the extraction of characters and interactions in fiction works. Recently, deep learning methods have proven to be efficient on classic NLP and multimedia tasks (e.g., Reference [36]) and are therefore a promising perspective to solve most of these problems. Alternatively, a promising change of perspective would be to break with the pipeline approach, which is currently the norm for character network extraction, as illustrated by Figure 1. All approaches first attempt to detect characters before trying to capture interactions, possibly propagating errors made at the earlier stages of the process. Instead, both tasks could probably benefit from one another and be performed jointly, as demonstrated by Yeh and Wu for face clustering [122]. Another possible option to avoid or reduce the pipeline limitations is to leverage deep learning methods and adopt a supervised *end-to-end* approach over parts, or even the whole extraction process. However, deep learning methods require to be trained on large corpora, and current fiction-related corpora are small or do not even exist for certain tasks (especially for comics). Thus, before being able to leverage deep learning methods, it is likely that the community will have to constitute or extend such corpora, a tedious work often performed collectively. An alternative would be to take advantage of existing corpora to perform *transfer learning* based on existing models built on non-fiction data.

7.1.2 Dynamic Networks. Many authors using character networks to solve higher-level problems identify the dynamics of the story as a very important aspect to be taken into account. This is especially true in long-term narratives such as TV or novel series in which relationships and characters are likely to change over time. Yet, the overwhelming majority of methods proposed in the literature rely on *static* networks. This may be because extracting *dynamic* networks requires making tough methodological choices such as choosing the size of the temporal window. Moreover, time can be modeled in several ways in a network [4]. Finally, there are much fewer off-the-shelf tools to analyze dynamic networks, and these are more difficult to use. However, this issue deserves to be explored, as dynamic networks are likely to improve both our understanding of the structure of fictional character networks and the performance of the methods taking advantage of this structure to solve problems such as those presented in Section 6. It is even likely that such networks are necessary to solve certain intrinsically dynamic problems such as narrative generation (see Section 3 in *Supplementary Material*).

7.1.3 Evaluation and Methodological Choices. As described before, extracting a reliable character network requires efficiently solving a number of media-specific problems (e.g., NER, face detection), most of them difficult. This raises several questions, the first one being that of *performance assessment*. As mentioned before, all existing extraction methods rely on the pipeline approach. Consequently, each processing step constituting the pipeline must be assessed separately. Performing such a task requires an appropriate ground truth designed for the considered problem. Such annotated corpora already exist for most problems, but they are generally based on non-fiction data, in which case they do not account for the specific characteristics of fiction described throughout this survey. To obtain a reliable performance evaluation, the test dataset must be built on fiction, which ties this point to the issue of machine learning training that we mentioned before: for a number of problems, such corpora are still to be constituted. But it is also important to estimate the relevance of the extracted character networks, i.e., the output of the pipeline. Up to now, only a few authors have tried to conduct this type of assessment on character networks [3, 62]. This task necessitates ground-truth networks to which one can compare the networks extracted using the pipeline, so here, too, there is a significant manual annotation work to do. Besides these data-related aspects, some methodological questions remain to be solved. There are many possible ways to compare two graphs, and it is not obvious which ones are the most appropriate to the case of character networks. In addition, this aspect could be dependent on the high-level use one wants to make of the network (e.g., story decomposition, summarization).

Solving the data and methods issues described above is important not only to evaluate the performance of the processing steps constituting the extraction pipeline, but also to drive methodological choices. Indeed, there are a number of open questions regarding which approaches are the most appropriate to handle these steps, or if some steps should be handled at all, and assessing their effect on the overall pipeline performance is necessary to discriminate between them. For instance, in textual narratives, it is not clear whether detecting all forms of character mentions (including nominals) instead of just looking for proper nouns, and/or solving anaphoras instead of ignoring them, would significantly improve the quality of the extracted networks. The same goes when detecting interactions: We do not know if conversation-based networks are more informative than co-occurrence-based ones, yet they are much harder to extract. To generalize, we do not know whether solving such harder low-level problems would result in better character networks; this remains to be tested. Solving these problems efficiently requires significant extra efforts, so this is a very important question, yet it is still open. Very few [29, 50] of the reviewed articles try to tackle the problem by assessing the quality of the extracted networks. Solving the evaluation-related issues would also allow assessing the relevance of using dynamic networks instead of static ones, and/or to identify which parameter values are optimal (e.g., size of the narrative unit when detecting co-occurrences, or of the sliding window when building dynamic networks). Finally, it is important to note that the relevance of the extracted network is likely to depend greatly on the high-level task for which one plans to leverage this network.

7.2 Network Analysis

We distinguish between the development of methods aimed at describing character networks (Section 7.2.1) and the exploitation of character networks to solve high-level problems (Section 7.2.2).

7.2.1 Descriptive Tools. The tools used in the literature to describe character networks are standard, very widespread complex network topological measures (see *Supplementary Material*). Yet, the field has produced many more tools that are not as known (e.g., motif-based measures, backbone detection, core-periphery structures, community structure–related node roles), but still likely to be relevant to study character networks. Therefore, there is a large number of opportunities to explore here. Some of these tools (e.g., motif-based measures) are designed for large networks, which means they would be relevant only for networks representing collections of narratives, like the comic networks mentioned in the survey [5, 38]. Another interesting possibility is to adapt existing measures to the specific case of character networks. For instance, certain authors identify one or two characters as extremely central in the studied narrative (protagonist vs. antagonist). Then when computing distance-based centrality measures such as the closeness, why not focus only on the shortest path joining the character of interest to the protagonist, instead of considering all pairs of characters?

Another promising perspective is to follow the evolution currently taking place in the field of complex network analysis and to integrate more information in the network to apply tools able to leverage this additional information. A few authors mentioned in the survey have started extracting such networks by including time (dynamic networks), relationship polarity (signed networks), or individual information (attributed networks) in their model. However, they focus more on the extraction process than on the way to efficiently use this information. For instance, several approaches exist to detect significant changes in dynamic networks [4], which could be applied to identify plot twists or to segment the narrative. Regarding signed networks, there are alternatives to the concept of structural balance; for instance, relaxations allowing to identify mediator groups in a situation of conflict; this could be of interest when analyzing so-called *adversarial* movies [27]. Node attributes (see Section 1.2 in the *Supplementary Material*) can be the object of the description through measures such as assortativity, which allows quantifying how homophilic the network is. But they can also help interpreting the output of other tools; for instance, in the context of community detection, by associating certain attribute values to specific communities. To conclude, note that it is possible to combine all these aspects (time, attributes, polarity) in one network, which would require very specific descriptive tools.

7.2.2 Applications. It is worth noting that the articles using character networks to solve high-level problems are quite segregated depending on the medium of the considered narratives (textual vs. audiovisual). This makes sense for the low-level part of the network extraction process, as it is medium-specific but not for the exploitation of the extracted networks, as these do not vary much depending on the narrative medium. Consequently, it seems possible to transpose the methods designed to solve certain problems on a given medium to another one. For instance, text-based networks have been used to assess literary theories, but we could not find any article trying to do the same for cinematographic ones, e.g., the widespread use of the so-called *Hollywood Formula* to design movie plots. On the contrary, character networks have been used to generate summaries of TV series; the same general principle could be applied to novels.

A number of high-level applications taking advantage of character networks consist in solving some classification problem, such as grouping narratives depending on their genre, time of publication, or creator. A related task is that of narrative recommendation (e.g., movie recommendation), as it can also be formulated as a prediction problem. In both cases, all authors feed their classifier with a few topological measures, generally the same very standard measures mentioned in Section 7.2.1. But there is no reason why these specific measures would be more appropriate for a classification task than the other measures defined in the literature. Also, the most relevant measures are not necessarily the same, depending on the considered classification task. A first improvement would therefore be to adopt a more exhaustive approach and consider a much larger number of topological measures, picking them so they characterize the network at various scales and scopes. Another improvement would be to use representation learning instead of manually selecting discriminant features. This could be performed by using *graph embeddings*, a recent transposition to graphs of the NLP concept of *word embeddings*. This approach allows directly learning the most appropriate numeric representation of the character network for the considered classification task. However, it requires access to a large amount of data.

Recently, cross- and trans-media narratives have become popular, and their automatic processing constitutes another promising perspective. Crossmedia means that the same narrative is repeated over several different media, e.g., a novel and its movie adaptation. Transmedia means that different narratives belonging to the same fictional universe are expressed using different media, e.g., the sequel of a movie taking the form of a novel. This raises the interesting problem of network alignment, i.e., determining which character in one network corresponds to which character in the other (see also Section 3 of the *Supplementary Material*). In the case of transmedia, one could use character networks to look for plot intersections between the different narratives and possibly detect divergences. They could also be used to help the user navigate among the sometimes numerous and complex narratives (e.g., superhero universes). In the case of crossmedia, adapting a narrative for a different medium generally involves changing the plot or even the story and adding or removing characters. Identifying which changes a character network undergoes during such adaptation could help in understanding and maybe automating this process. Moreover, having access to the same story under different forms could be leveraged to improve network extraction itself by adopting and extending the multimodal approach already mentioned in Section 7.1.1.

Finally, automatic plot generation constitutes another promising perspective. Performing this task is likely to require leveraging some additional information, as mentioned in Section 7.2.1, in particular: dynamic networks, to represent existing plots and study them to model their typical evolution; and signed networks, to deal with antagonistic stories, which are very frequent. A few recent articles deal with plot generation (see Section 3 of the *Supplementary Material*), but there is still much work to do. Designing models to generate networks that would be realistic according to some criteria of interest constitutes an important part of the complex networks field,[4] so this task could benefit from the work already conducted on this issue. However, it is worth noting that if generating a sequence of events corresponding to a plot looks achievable in the short-term; automatically converting it into a proper narrative seems like a very difficult problem. It is possible to adopt an extractive approach, i.e., to build the narrative by selecting narrative bits among a predefined collection. The alternative is to generate the narrative outright, but this seems realistic only for certain media, in particular text, for which efficient language-based models exist.

REFERENCES

[1] A. Agarwal, S. Balasubramanian, J. Zheng, and S. Dash. 2014. Parsing screenplays for extracting social networks from movies. In *Proceedings of the 3rd Workshop on Computational Linguistics for Literature (CLFL'14)*. ACL, 50–58. http://www.aclweb.org/anthology/W14-0907.

[2] A. Agarwal, A. Corvalan, and J. R. O. Jensen. 2012. Social network analysis of *Alice in Wonderland*. In *Proceedings of the Workshop on Computational Linguistics for Literature (NAACL'12)*. http://www.cs.columbia.edu/apoorv/Homepage/Publications_files/naacl2012.pdf.

[4] E.g., the Watts-Strogatz [115] and Barabási-Albert [10] models respectively produce *small-world* and *scale-free* networks.

[3] A. Agarwal, A. Kotalwar, and O. Rambow. 2013. Automatic extraction of social networks from literary text: A case study on *Alice in Wonderland*. In *Proceedings of the International Joint Conference on Natural Language Processing (IFCNLP'13)*. 1202–1208. http://www.aclweb.org/website/old_anthology/I/I13/I13-1171.pdf.

[4] C. Aggarwal and K. Subbian. 2014. Evolutionary network analysis: A survey. *ACM Comput. Surv.* 47, 1 (2014), 10. https://doi.org/10.1145/2601412

[5] R. Alberich, J. Miro-Julia, and F. Rossello. 2002. Marvel Universe looks almost like a real social network. Retrieved from: *arXiv* cond-mat.dis-nn (2002), cond–mat/0202174. http://arxiv.org/abs/cond-mat/0202174

[6] D. R. Amancio. 2015. Network analysis of named entity interactions in written texts. *EPL* 5, 114 (2015), 58005. https://doi.org/10.1209/0295-5075/114/58005

[7] M. C. Ardanuy and C. Sporleder. 2014. Structure-based clustering of novels. In *Proceedings of the 3rd Workshop on Computational Linguistics for Literature*. 31–39. http://www.aclweb.org/anthology/W14-0905.

[8] M. C. Ardanuy and C. Sporleder. 2015. Clustering of novels represented as social networks. *Linguistic Issues in Language Technology* 12 (2015). http://csli-lilt.stanford.edu/ojs/index.php/LiLT/article/view/60.

[9] O. Augereau, M. Iwata, and K. Kise. 2018. A survey of comics research in computer science. *J. Imaging* 4, 7 (2018), 87. https://doi.org/10.3390/jimaging4070087

[10] A.-L. Barabási and R. Albert. 1999. Emergence of scaling in random networks. *Science* 286, 5439 (1999), 509–512. https://doi.org/10.1126/science.286.5439.509

[11] A. Bolioli, M. Casu, M. Lana, and R. Roda. 2013. Exploring the betrothed lovers. In *Proceedings of the CMN*. 30–35.

[12] D. Bordwell and K. Thompson. 1993. *Film Art: An Introduction*. McGraw-Hill. https://www.mhprofessional.com/9780073535104-usa-film-art-an-introduction.

[13] C. Bornet and F. Kaplan. 2017. A simple set of rules for characters and place recognition in French novels. *Front. Digital Hum.* 4 (2017), 6. https://doi.org/10.3389/fdigh.2017.00006

[14] G. Bossaert and N. Meidert. 2013. "We are only as strong as we are united, as weak as we are divided" A dynamic analysis of the peer support networks in the *Harry Potter* books. *Open J. Appl. Sci.* 3 (2013), 174–185. https://doi.org/10.4236/ojapps.2013.32024

[15] X. Bost, S. Gueye, V. Labatut, M. Larson, G. Linarès, D. Malinas, and R. Roth. 2019. Remembering winter was coming: Character-oriented video summaries of TV series. *Multimed. Tools Appl. (in revision)* (2019).

[16] X. Bost, V. Labatut, S. Gueye, and G. Linarès. 2016. Narrative smoothing: Dynamic conversational network for the analysis of TV Series plots. In *Proceedings of the Advances in Social Networks Analysis and Mining - 2nd International Workshop on Dynamics in Networks*. 1111–1118. https://doi.org/10.1109/ASONAM.2016.7752379

[17] X. Bost and G. Linarès. 2014. Constrained speaker diarization of TV series based on visual patterns. In *Proceedings of the IEEE Spoken Language Technology Workshop*. 390–395. https://doi.org/10.1109/SLT.2014.7078606

[18] X. Bost, G. Linarès, and S. Gueye. 2015. Audiovisual speaker diarization of TV series. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*. 4799–4803. https://doi.org/10.1109/ICASSP.2015.7178882

[19] H. Bredin and G. Gelly. 2016. Improving speaker diarization of TV series using talking-face detection and clustering. In *Proceedings of the ACM International Conference on Multimedia*. 157–161. https://doi.org/10.1145/2964284.2967202

[20] A. Celikyilmaz, D. Hakkani-Tur, H. He, G. Kondrak, and D. Barbosa. 2010. The actor-topic model for extracting social networks in literary narrative. In *Proceedings of the NIPS Workshop: Machine Learning for Social Computing*. https://webdocs.cs.ualberta.ca/denilson/files/publications/nips2010.pdf.

[21] S. Chaturvedi, S. Srivastava, H. Daumé, and C. Dyer. 2016. Modeling evolving relationships between characters in literary novels. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. 2704–2710. https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/viewPaper/12408.

[22] H. Y. Chen and J. D. Choi. 2016. Character identification on multiparty conversation. In *Proceedings of the 17th Annual SIGdial Meeting on Discourse and Dialogue*. 90–100. http://www.aclweb.org/anthology/W16-3612.

[23] W.-T. Chu and W.-W. Li. 2017. Manga FaceNet: Face detection in manga based on deep neural network. In *Proceedings of the ACM International Conference on Multimedia Retrieval*. 412–415. https://doi.org/10.1145/3078971.3079031

[24] P. Cipresso and G. Riva. 2016. Computational psychometrics meets Hollywood. *Front. Psychol.* 7 (2016), 1753. https://doi.org/10.3389/fpsyg.2016.01753

[25] A. Clauset and N. Eagle. 2007. Persistence and periodicity in a dynamic proximity network. In *Proceedings of the Workshop on Computational Methods for Dynamic Interaction Networks (DIMACS'07)*. http://arxiv.org/pdf/1211.7343

[26] P. Clément, T. Bazillon, and C. Fredouille. 2011. Speaker diarization of heterogeneous web video files. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*. 4432–4435. https://doi.org/10.1109/ICASSP.2011.5947337

[27] L. Ding and A. Yilmaz. 2010. Learning relations among movie characters: A social networks perspective. In *Proceedings of the European Conference on Computer Vision*. Lecture Notes in Computer Science, Vol. 6314. 410–423. https://doi.org/10.1007/978-3-642-15561-1_30

[28] L. Ding and A. Yilmaz. 2011. Inferring social relations from visual concepts. In *Proceedings of the International Conference on Computer Vision*. 699–706. https://doi.org/10.1109/ICCV.2011.6126306

[29] M. Edwards, L. Mitchell, J. Tuke, and M. Roughan. 2018. The one comparing narrative social network extraction techniques. *arXiv* cs.SI (2018), 1811.01467. https://arxiv.org/abs/1811.01467

[30] M. Elsner. 2012. Character-based kernels for novelistic plot structure. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*. 634–644. http://www.aclweb.org/anthology/E12-1065.

[31] D. K. Elson. 2012. *Modeling Narrative Discourse*. Ph.D. Dissertation. Columbia University. http://www.cs.columbia.edu/delson/pubs/Modeling-Narrative-Discourse_Elson_R4.pdf.

[32] D. K. Elson, N. Dames, and K. R. McKeown. 2010. Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. 138–147. http://dl.acm.org/citation.cfm?id=1858696

[33] D. K. Elson and K. R. McKeown. 2010. Automatic attribution of quoted speech in literary narrative. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. 1013–1019. https://dl.acm.org/citation.cfm?id=2898607.2898769

[34] P. Ercolessi. 2013. *Extraction multimodale de la structure narrative des épisodes de séries télévisées*. PhD Thesis. Université de Toulouse. http://www.theses.fr/2013TOU30131.

[35] M. Falk. 2016. Making connections: Network analysis, the bildungsroman and the world of *The Absentee*. *J. Lang. Lit. Cult.* 63, 2–3 (2016), 107–122.

[36] S. Ghannay, A. Caubriere, Y. Estève, N. Camelin, E. Simonnet, A. Laurent, and E. Morin. 2018. End-to-end named entity and semantic concept extraction from speech. In *Proceedings of the IEEE Spoken Language Technology Workshop*. https://doi.org/10.1109/slt.2018.8639513

[37] K. Glass and S. Bangay. 2006. Hierarchical rule generalisation for speaker identification in fiction books. In *Proceedings of the Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries*. 31–40. https://doi.org/10.1145/1216262.1216266

[38] P. M. Gleiser. 2007. How to become a superhero. *J. Stat. Mech.* 2007, 9 (2007), P09020. https://doi.org/10.1088/1742-5468/2007/09/P09020

[39] P. J. Gorinski and M. Lapata. 2015. Movie script summarization as graph-based scene extraction. In *Proceedings of the Annual Conference of the North American Chapter of the ACL*. 1066–1076. http://www.aclweb.org/anthology/N15-1113.

[40] M. Grandjean. 2017. Comparing the relational structure of the Gospels. *DBL* 2 (2017). In press. http://www.martingrandjean.ch/communications.

[41] S. Grayson, K. Wade, G. Meaney, and D. Greene. 2016. The sense and sensibility of different sliding windows in constructing co-occurrence networks from literature. In *Proceedings of the International Workshop on Computational History and Data-Driven Humanities*. 65–77. https://doi.org/10.1007/978-3-319-46224-0_7

[42] A. Grener, M. Luczak-Roesch, E. Fenton, and T. Goldfinch. 2017. *Towards a Computational Literary Science: A Computational Approach to Dickens' Dynamic Character Networks*. Technical Report. Zenodo. https://doi.org/10.5281/zenodo.259499

[43] H. He, D. Barbosa, and G. Kondrak. 2013. Identification of speakers in novels. In *Proceedings of the ACL*. 1312–1320.

[44] F. Heider. 1946. Attitudes and cognitive organization. *J. Psychol.* 21, 1 (1946), 107–112. https://doi.org/10.1080/00223980.1946.9917275

[45] L. Hettinger, M. Becker, I. Reger, F. Jannidis, and A. Hotho. 2015. Genre classification on German novels. In *Proceedings of the 26th International Workshop on Database and Expert Systems Applications*. https://doi.org/10.1109/DEXA.2015.62

[46] A. J. Holanda, M. Matias, S. M. S. P. Ferreira, G. M. L. Benevides, and O. Kinouchi. 2017. Character networks and book genre classification. *arXiv* cs.SI (2017), 1704.08197. https://arxiv.org/abs/1704.08197

[47] S. Hutchinson, V. Datla, and M. M. Louwerse. 2012. Social networks are encoded in language. In *Proceedings of the 34th Annual Conference of the Cognitive Science Society*. 491–496. https://mindmodeling.org/cogsci2012/papers/0096/paper0096.pdf.

[48] M. Iyyer, A. Guha, S. Chaturvedi, J. Boyd-Graber, and H. Daumé. 2016. Feuding families and former friends: Unsupervised learning for dynamic fictional relationships. In *Proceedings of the Conference of the North American Chapter of the ACL: Human Language Technologies*. 1534–1544. https://doi.org/10.18653/v1/N16-1180

[49] F. Jannidis, I. Reger, M. Krug, L. Weimer, L. Macharowsky, and F. Puppe. 2016. Comparison of methods for the identification of main characters in German novels. In *Proceedings of the DH*. 578–582. http://dh2016.adho.org/abstracts/297.

[50] P. A. Jayannavar, A. Agarwal, M. Ju, and O. Rambo. 2015. Validating literary theories using automatic social network extraction. In *Proceedings of the 4th Workshop on Computational Linguistics for Literature*. 32–41. http://anthology.aclweb.org/W/W15/W15-07.pdf#page=46.

[51] J. Jovanovic, E. Bagheri, J. Cuzzola, D. Gasevic, Z. Jeremic, and R. Bashash. 2014. Automated semantic tagging of textual content. *IT Professional* 16, 6 (2014), 38–46. https://doi.org/10.1109/MITP.2014.85

[52] J. J. Jung, E. You, and S.-B. Park. 2013. Emotion-based character clustering for managing story-based contents: A cinemetric analysis. *Multimed. Tools App.* 65, 1 (2013), 29–45. https://doi.org/10.1007/s11042-012-1133-x

[53] F. Karsdorp, P. van Kranenburg, T. Meder, and A. van den Bosch. 2012. Casting a spell: Identification and ranking of actors in folktales. In *Proceedings of the 2nd Workshop on Annotation of Corpora for Research in the Humanities*. https://pure.knaw.nl/ws/files/481270/karsdorp_et_al2012b.pdf.

[54] A. Kazantseva and S. Szpakowicz. 2010. Summarizing short stories. *Comput. Linguist.* 36, 1 (2010), 71–109. https://doi.org/10.1162/coli.2010.36.1.36102

[55] D. E. Knuth. 1993. *The Stanford GraphBase: A Platform for Combinatorial Computing*. Vol. 37. Addison-Wesley Reading. http://www-cs-faculty.stanford.edu/knuth/sgb.html.

[56] V. Krishnan and J. Eisenstein. 2015. "You're Mr. Lebowski, I'm the Dude": Inducing address term formality in signed social networks. In *Proceedings of the Annual Conference of the North American Chapter of the ACL*. 1616–1626. http://www.aclweb.org/anthology/N15-1185.

[57] M. Krug, F. Puppe, F. Jannidis, L. Macharowsky, I. Reger, and L. Weimer. 2015. Rule-based coreference resolution in German historic novels. In *Proceedings of the 4th Workshop on Computational Linguistics for Literature*. 98–104. http://www.aclweb.org/anthology/W15-0711.

[58] M. Krug, I. Reger, F. Jannidis, L. Weimer, N. Madarász, and F. Puppe. 2017. Overcoming data sparsity for relation detection in German novels. In *Proceedings of the Digital Humanities*. https://dh2017.adho.org/abstracts/340/340.pdf.

[59] D. Kruger, M. Fisher, and I. Jobling. 2003. Proper and dark heroes as DADS and CADS. *Human Nature* 14, 3 (2003), 305–317. https://doi.org/10.1007/s12110-003-1008-y

[60] H.-C. Kwon and K.-H. Shim. 2017. An improved method of character network analysis for literary criticism: A case study of Hamlet. *Int. J. Contents* 13, 3 (2017), 43–48. https://doi.org/10.5392/IJoC.2017.13.3.043

[61] J. Lee and T.-S. Wong. 2016. Conversational network in the Chinese Buddhist Canon. *Open. Ling.* 2, 1 (2016), 427–436. https://doi.org/10.1515/opli-2016-0022

[62] J. Lee and C. Y. Yeung. 2012. Extracting networks of people and places from literary texts. In *Proceedings of the 26th Pacific Asia Conference on Language, Information and Computation*. 209–218. https://aclanthology.info/papers/Y12-1022/y12-1022.

[63] O.-J. Lee and J. J. Jung. 2019. Modeling affective character network for story analytics. *Future Generation Computer Systems* 92 (2019), 458–478. https://doi.org/10.1016/j.future.2018.01.030

[64] J. Li, D. Yang, and P. Lv. 2019. Visualize classic play's composing patterns. *Multimed. Tools App.* 78, 5 (2019), 5989–6012. https://doi.org/10.1007/s11042-018-6405-7

[65] C. Liang, Y Zhang, J. Cheng, and C.-S. Xu. 2009. A novel role-based movie scene segmentation method. In *Proceedings of the Pacific-Rim Conference on Multimedia*. Lecture Notes in Computer Science, Vol. 5879. 917–922. https://doi.org/10.1007/978-3-642-10467-1_82

[66] D. Liu and L. Albergante. 2017. Balance of thrones: A study on Game of Thrones *Game of Thrones*. *arXiv* cs.SI (2017), 1707.05213. https://arxiv.org/abs/1707.05213

[67] P. Mac Carron and R. Kenna. 2012. Universal properties of mythological networks. *EPL* 99, 2 (2012), 28002. https://doi.org/10.1209/0295-5075/99/28002

[68] C. Makris and P. Vikatos. 2016. Community detection of screenplay characters. In *Proceedings of the 12th IFIP International Conference on Artificial Intelligence Applications and Innovations*. IFIP Advances in Information and Communication Technology, Vol. 475. 463–470. https://doi.org/10.1007/978-3-319-44944-9_40

[69] N. Mamede and P. Chaleira. 2004. Character identification in children stories. In *Proceedings of the International Conference on Natural Language Processing in Spain*. Lecture Notes in Computer Science, Vol. 3230. 82–90. https://doi.org/10.1007/978-3-540-30228-5_8

[70] V. H. Masías, P. Baldwin, S. Laengle, A. Vargas, and F. A. Crespo. 2017. Exploring the prominence of Romeo and Juliet's characters using weighted centrality measures. *Digit. Scholarsh. Hum.* 32, 4 (2017), 837–858. https://doi.org/10.1093/llc/fqw029

[71] D. McNary. 2019. 2018. Worldwide Box Office Hits Record as Disney Dominates. Retrieved on February 20, 2019 from https://variety.com/2019/film/news/box-office-record-disney-dominates-1203098075/.

[72] S. Min and J. Park. 2016. Mapping out narrative structures and dynamics using networks and textual information. *arXiv* cs.CL (2016), 1604.03029. http://arxiv.org/abs/1604.03029

[73] S. Min and J. Park. 2016. Network science and narratives: Application to *Les Misérables*. In *Proceedings of the 7th Workshop on Complex Networks CompleNet*. Studies in Computational Intelligence, Vol. 644. 257–265. https://doi.org/10.1007/978-3-319-30569-1_19

[74] F. Moretti. 2011. Network theory, plot analysis. *Stanford Lit. Lab.* 2 (2011). http://litlab.stanford.edu/LiteraryLabPamphlet2.pdf.

[75] P. Mutton. 2004. Inferring and visualizing social networks on internet relay chat. In *Proceedings of the 8th International Conference on Information Visualisation*. 35–43. https://doi.org/10.1109/IV.2004.1320122
[76] E. T. Nalisnick and H. S. Baird. 2013. Character-to-character sentiment analysis in Shakespeare's plays. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. http://www.aclweb.org/anthology/P13-2085.
[77] C.-J. Nan, K.-M. Kim, and B.-T. Zhang. 2015. Social network analysis of TV drama characters via deep concept hierarchies. In *Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 831–836. https://doi.org/10.1145/2808797.2809306
[78] D. Oelke, D. Kokkinakis, and D. A. Keim. 2013. Fingerprint matrices: Uncovering the dynamics of social networks in prose literature. *Comput. Graph. Forum* 32, 3pt4 (Jun 2013), 371–380. https://doi.org/10.1111/cgf.12124
[79] D. Oelke, D. Kokkinakis, and M. Malm. 2012. Advanced visual analytics methods for literature analysis. In *Proceedings of the 6th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. 35–44. https://dl.acm.org/citation.cfm?id=2390364
[80] T. O'Keefe, S. Pareti, J. R. Curran, I. Koprinska, and M. Honnibal. 2012. A sequence labelling approach to quote attribution. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. 790–799. https://dl.acm.org/citation.cfm?id=2391033
[81] G.-M. Park, S.-H. Kim, H.-R. Hwang, and H.-G. Cho. 2013. Complex system analysis of social networks extracted from literary fictions. *Int. J. Mach. Comput.* 3, 1 (2013), 107–111. https://doi.org/10.7763/IJMLC.2013.V3.282
[82] S.-B. Park, K.-J. Oh, and G.-S. Jo. 2012. Social network analysis in a movie using Character-Net. *Multimed. Tools App.* 59, 2 (Jan. 2012), 601–627. https://doi.org/10.1007/s11042-011-0725-1
[83] F. Pope, R. A. Shirvani, Mugizi R. Rwebangira, M. Chouikha, A. Taylor, A. Alarcon Ramirez, and A. Torfi. 2015. Automatic detection of small groups of persons, influential members, relations and hierarchy in written conversations using fuzzy logic. In *Proceedings of the International Conference on Data Mining*. 155–161. https://arxiv.org/abs/1610.01720v1
[84] S. D. Prado, S. R. Dahmen, A. L. C. Bazzan, P. Mac Carron, and R. Kenna. 2016. Temporal network analysis of literary texts. *Adv. Complex Syst.* 19, 3 (2016), 1650005.
[85] B. Rieck and H. Leitte. 2016. "Shall I compare thee to a network?": Visualizing the topological structure of Shakespeare's plays. In *Proceedings of the 1st Workshop on Visualization for the Digital Humanities*. https://doi.org/10.11588/heidok.00023477
[86] C. Rigaud, T. N. Le, J.-C. Burie, J.-M. Ogier, M. Iwata, E. Imazu, and K. Kise. 2015. Speech balloon and speaker association for comics and manga understanding. In *Proceedings of the 13th International Conference on Document Analysis and Recognition*. 351–355. https://doi.org/10.1109/ICDAR.2015.7333782
[87] D. Robson. 2016. Heroes and villains. *The Psychol.* 29 (2016), 610–613. https://thepsychologist.bps.org.uk/heroes-and-villains.
[88] D. Robson. 2018. Our fiction addiction: Why humans need stories. Retrieved on June 28, 2018 from http://www.bbc.com/culture/story/20180503-our-fiction-addiction-why-humans-need-stories.
[89] Y. Rochat and F. Kaplan. 2014. Analyse de réseaux sur les personnages des Confessions de Jean-Jacques Rousseau. *Cahiers Num.* 10, 3 (2014), 109–133. https://doi.org/10.3166/LCN.10.3.109-133
[90] Y. Rochat and M. Triclot. 2017. Les réseaux de personnages de science-fiction. *ReS Futurae* 10 (2017). https://doi.org/10.4000/resf.1183
[91] G. A. Sack. 2012. Character networks for narrative generation. In *Proceedings of the 8th Artificial Intelligence and Interactive Digital Entertainment Conference - Intelligent Narrative Technologies Workshop*. 38–43. http://www.aaai.org/ocs/index.php/AIIDE/AIIDE12/paper/view/5550.
[92] J. Seo, G.-M. Park, S.-H. Kim, and H.-G. Cho. 2013. Characteristic analysis of social network constructed from literary fiction. In *Proceedings of the International Conference on Cyberworlds*. 147–150. https://doi.org/10.1109/CW.2013.72
[93] J.-K. Seo, S.-H. Kim, H.-S. Tak, and H.-G. Cho. 2014. A structural analysis of literary fictions with social network framework. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*. 634–640. https://doi.org/10.1145/2554850.2555049
[94] D. Smith, P. Schlaepfer, K. Major, M. Dyble, A. E. Page, J. Thompson, N. Chaudhary, G. D. Salali, R. Mace, L. Astete, M. Ngales, L. Vinicius, and A. Bamberg Migliano. 2017. Cooperation and the evolution of hunter-gatherer storytelling. *Nature Comm.* 8 (2017), 1853. https://doi.org/10.1038/s41467-017-02036-8
[95] K. Somandepalli, N. Kumar, T. Guha, and S. S. Narayanan. 2018. Unsupervised discovery of character dictionaries in animation movies. *IEEE T. Multimedia.* 20, 3 (2018), 539–551. https://doi.org/10.1109/TMM.2017.2745712
[96] S. Srivastava, S. Chaturvedi, and T. Mitchell. 2016. Inferring interpersonal relations in narrative summaries. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. 2807–2813. https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/viewPaper/12173.

[97] J. Stiller and M. Hudson. 2005. Weak links and scene cliques within the small world of Shakespeare. *J. Cult. Evol. Psychol.* 3, 1 (2005), 57–73. https://doi.org/10.1556/JCEP.3.2005.1.4

[98] J. Stiller, D. Nettle, and R. I. M. Dunbar. 2003. The small world of Shakespeare's plays. *Hum. Nature* 14, 4 (2003), 397–408. https://doi.org/10.1007/s12110-003-1013-1

[99] M. Stommel, L. I. Merhej, and M. G. Müller. 2012. Segmentation-free detection of comic panels. In *Proceedings of the International Conference on Computer Vision and Graphics*. Lecture Notes in Computer Science, Vol. 7594. 633–640. https://doi.org/10.1007/978-3-642-33564-8_76

[100] M. Stricker, O. Augereau, K. Kise, and M. Iwata. 2018. Facial landmark detection for manga images. *arXiv* cs.CV (2018), 1811.03214. https://arxiv.org/abs/1811.03214

[101] S. Sudhahar and N. Cristianini. 2013. Automated analysis of narrative content for digital humanities. *Int. J. Adv. Comp. Sci.* 3, 9 (2013), 440–447. https://doi.org/10.1.1.429.3639

[102] C. Suen, L. Kuenzel, and S. Gil. 2013. Extraction and analysis of character interaction networks from plays and movies. In *Proceedings of the Digital Humanities*. http://snap.stanford.edu/class/cs224w-2011/proj/laneyk_Finalwriteup_v1.pdf.

[103] K. Takayama, H. Johan, and T. Nishita. 2012. Face detection and face recognition of cartoon characters using feature extraction. In *Proceedings of the IIEEJ Image Electronics and Visual Computing Workshop*. http://www.iieej.org/trans/IEVC/IEVC2012/PDF/4B-1.pdf.

[104] M. S. A. Tan, E. A. Ujum, and K. Ratnavelu. 2014. A character network study of two Sci-Fi TV series. In *Proceedings of the Frontiers in Physics (AIP Conference Proceedings)*, Vol. 1588–246. https://doi.org/10.1063/1.4866954

[105] M. Tandiwe Myambo. 2016. How reading fiction can help students understand the real world. Retrieved on February 02, 2019 from https://theconversation.com/how-reading-fiction-can-help-students-understand-the-real-world-52908.

[106] Q. D. Tran, D. Hwang, and J. J. Jung. 2015. Movie summarization using characters network analysis. In *Proceedings of the 7th International Conference on Computational Collective Intelligence*. Lecture Notes in Computer Science, Vol. 9329. Springer, 390–399. https://doi.org/10.1007/978-3-319-24069-5_37

[107] Q. D. Tran, D. Hwang, and J. J. Jung. 2017. Character-based indexing and browsing with movie ontology. *J. Intell. Fuzzy Syst.* 32, 2 (2017), 1229–1240. https://doi.org/10.3233/JIFS-169122

[108] M. Trovati and J. Brady. 2014. Towards an automated approach to extract and compare fictional networks. In *Proceedings of the 25th International Workshop on Database and Expert Systems Applications*. https://doi.org/10.1109/DEXA.2014.58

[109] C.-M. Tsai, L.-W. Kang, C.-W. Lin, and W. Lin. 2013. Scene-based movie summarization via role-community networks. *IEEE T. Circ. Syst. Vid.* 23, 11 (2013), 1927–1940. https://doi.org/10.1109/TCSVT.2013.2269186

[110] H. Vala, D. Jurgens, A. Piper, and D. Ruths. 2015. Mr. Bennet, his coachman, and the Archbishop walk into a bar but only one of them gets recognized: On the difficulty of detecting characters in literary texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. http://piperlab.mcgill.ca/pdfs/Vala_EMNLP_2015.pdf.

[111] J. Valls-Vargas, S. Ontanón, and J. Zhu. 2014. Toward automatic character identification in unannotated narrative text. In *Proceedings of the 7th Intelligent Narrative Technologies Workshop*. 188–194. https://pdfs.semanticscholar.org/b755/1e222321cca2fdb1992991c434ef0aa1a2fd.pdf.

[112] K. van Dalen-Oskam, J. de Does, M. Marx, I. Sijaranamual, K. Depuydt, B. Verheij, and V. Geirnaert. 2014. Named entity recognition and resolution for literary studies. *Comput. Ling. Netherlands J.* 4 (2014), 121–136. https://clinjournal.org/node/62.

[113] T. Venturini, L. Bounegru, M. Jacomy, and J. Gray. 2017. How to tell stories with networks. In *Datafied Society*. Amsterdam University Press. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3043857.

[114] M. Vrigkas, C. Nikou, and I. A. Kakadiaris. 2015. A review of human activity recognition methods. *FRAI* 2, Article 28 (2015), 28. https://doi.org/10.3389/frobt.2015.00028

[115] D. J. Watts and S. H. Strogatz. 1998. Collective dynamics of "small-world" networks. *Nature* 393, 6684 (1998), 440–442. https://doi.org/10.1038/30918

[116] M. C. Waumans, T. Nicodème, and H. Bersini. 2015. Topology analysis of social networks extracted from literature. *PLoS ONE* 10, 6 (2015), e0126470. https://doi.org/10.1371/journal.pone.0126470

[117] C.-Y. Weng, W.-T. Chu, and J.-L. Wu. 2007. Movie analysis based on roles' social network. In *Proceedings of the ICME*. 1403–1406. https://doi.org/10.1109/ICME.2007.4284922

[118] C.-Y. Weng, W.-T. Chu, and J.-L. Wu. 2009. RoleNet: Movie analysis from the perspective of social networks. *IEEE T. Multimed.* 11, 2 (2009), 256–271. https://doi.org/10.1109/TMM.2008.2009684

[119] A. Woloch. 2003. *The One vs. the Many: Minor Characters and the Space of the Protagonist in the Novel*. Princeton University Press. http://press.princeton.edu/titles/7622.html.

[120] A. Xanthos, I. Pante, Y. Rochat, and M. Grandjean. 2016. Visualising the dynamics of character networks. In *Proceedings of the DH*. 417–419. https://github.com/maladesimaginaires/intnetviz/raw/master/DH2016_xanthos_et_al.pdf.

[121] M.-C. Yeh, M.-C. Tseng, and W.-P. Wu. 2012. Automatic social network construction from movies using film-editing cues. In *Proceedings of the IEEE International Conference on Multimedia and Expo Workshops*. 242–247. https://doi.org/10.1109/ICMEW.2012.48

[122] M.-C. Yeh and W.-P. Wu. 2014. Clustering faces in movies using an automatically constructed social network. *IEEE MultiMed.* 21, 2 (2014), 22–31. https://doi.org/10.1109/MMUL.2014.24

[123] C. Y. Yeung and J. Lee. 2017. Identifying speakers and listeners of speech in literary works. In *Proceedings of the IJCNLP*. 325–329. http://www.aclweb.org/anthology/I17-2055.

[124] M. Yeung, B.-L. Yeo, and B. Liu. 1996. Extracting story units from long programs for video browsing and navigation. In *Proceedings of the 3rd lEEE International Conference Multimedia Computing & Systems*. 296–305. https://doi.org/10.1109/MMCS.1996.534991

[125] J. Yose, R. Kenna, M. MacCarron, and P. MacCarron. 2018. Network analysis of the Viking age in Ireland as portrayed in Cogadh Gaedhel re Gallaibh. *Roy Soc Open Sci* 5, 1 (2018), 171024. https://doi.org/10.1098/rsos.171024

[126] J. Y. Zhang, A. W. Black, and R. Sproat. 2003. Identifying speakers in children's stories for speech synthesis. In *Proceedings of the Eurospeech*. 2041–2044. https://www.cs.cmu.edu/ awb/papers/eurospeech2003/esper.pdf.

[127] Y.-F. Zhang, C.-S. Xu, H.Q. Lu, and Y.-M. Huang. 2009. Character identification in feature-length films using global face-name matching. *IEEE T. Multimed.* 11, 7 (2009), 1276–1288. https://doi.org/10.1109/TMM.2009.2030629

Received November 2018; revised April 2019; accepted June 2019