Structure-based Clustering of Novels

Mariona Coll Ardanuy
Trier University
ardanuy@uni-trier.de

Caroline Sporleder
Trier University
sporledc@uni-trier.de

Abstract

To date, document clustering by genres or authors has been performed mostly by means of stylometric and content features. With the premise that novels are societies in miniature, we build social networks from novels as a strategy to quantify their plot and structure. From each social network, we extract a vector of features which characterizes the novel. We perform clustering over the vectors obtained, and the resulting groups are contrasted in terms of author and genre.

1 Introduction

In recent years, the interest for quantitative methods of literary analysis has grown significantly. Humanities scholars and researchers are increasingly aware of the potential of data-driven approaches in a field that has traditionally been studied from a 'close reading' perspective. Large repositories of literary text together with the development of promising techniques from fields such as text mining or information extraction offer advantages that open new possibilities to the field of literature studies.

So far, most quantitative studies of literature have focused mainly on form and content. Structure and plot, considered key dimensions in a novel, have often been ignored due to the complexity in quantifying them. In this study, we explore the contribution of features that are directly related to them. With this goal, we represent a novel as a social network of characters (a technique that is not novel in the field of quantitative literary analysis), from which to extract features that can be used to perform document clustering. The outcome of the clustering will be a grouping of novels according to their structural similarity.

This is an exploratory study to determine to what degree the structure and plot of a novel are representative of the genre to which it belongs and characteristic of the style of its author. Two hypotheses are made on the basis of this premise. The first is that the structure and plot of the novel represented as a static and dynamic social network is key to predict the literary genre to which a novel belongs. The second is that the inner structure of the society depicted by the author in a novel is representative of this author. This approach introduces the use of automatically extracted static and dynamic networks to perform large-scale analyses of novels, by representing them as vectors of features that can then be used to compare the novels in terms of genre and authorship.

The rest of this paper is organized as follows. In Section 2 we present the related work. Section 3 describes the method employed in turning a novel into the vector of features chosen to characterize it. The experiments conducted are discussed in Section 4 and the results and analysis of them in Section 5. We discuss the results in Section 6 and conclude in Section 7.

2 Related Work

2.1 Unsupervised Document Classification

Unsupervised document classification (or 'document clustering') consists in automatically grouping a set of documents based on the similarities among them. Unlike its supervised counterpart, it does not require neither labeled training data nor prior knowledge of the classes into which the texts are to be categorized. Instead, documents—represented as vectors of features—that are similar are grouped together, yielding a clustering that is dependent on the features chosen to characterize the document. Due to the lack of supervision, it is not guaranteed that the resulting clustering corresponds to the classes in which we are interested (Zhang, 2013).

Unsupervised authorship analysis from docu-

Proceedings of the 3rd Workshop on Computational Linguistics for Literature (CLfL) @ EACL 2014, pages 31–39,
Gothenburg, Sweden, April 27, 2014.

ments is the task of automatically grouping texts that share the same author, by determining the set of features that distinguish one author from any other. The first approaches focused mainly on stylometrics (Ledger and Merriam (1994), Holmes and Forsyth (1995), Baayen et al. (1996), and Aaronson (2001)). More recent approaches use content-based features, such as Akiva and Koppel (2012) and Layton et al. (2011). Pavlyshenko (2012) brings document clustering by author to the literature domain. The lexicon of the author is in this work represented as semantic fields (the author's idiolect) on which Singular Value Decomposition is applied.

Much less effort has been devoted to the task of clustering documents by the genre in which they fall. Examples of this are Gupta et al. (2005), Poudat and Cleuziou (2003), and Bekkerman et al. (2007). The work of Allison et al. (2011) uses stylometric features to cluster 36 novels according to genre. The resulting clustering is only partially successful, but made its authors realize that the classification was not only obeying to genre criteria, but also to authorship. The stylistic signature of every document corresponded to a strong 'author' signal, rather than to the 'genre' signal.

2.2 Literary Quantitative Analysis

The reviewed approaches have in common that they use stylometric or content-based features. However, a novel should not be reduced to the dimensions of punctuation, morphology, syntax and semantics. This literary form has a depth, a complex structure of plot, characters and narration. The plot of a novel is defined in the Russian structuralism school by the collection of its characters and the actions they carry out (Bakhtin (1941), Propp (1968)). It could be said that every novel is a society in miniature.[1] Moretti (2011), concerned about how plot can be quantified, explores extensively the impact characters have on it. To this end, Moretti represents the characters of William Shakespeare's *Hamlet* as a social network. Several experiments (removing the protagonist, isolates, or a connecting character from the network) show how the plot changes accordingly to the alteration in the structure of characters. Sack (2012) proposes using social networks of characters as a mechanism for generating plots artificially.

One of the first attempts of combining social networks and literature was in Alberich et al. (2002). They built a social network from the Marvel comics in which characters are the nodes, linked by their co-occurrence in the same book. The authors note that the resulting network was very similar to a real social network. In Newman and Girvan (2003), the authors used a hand-built social network with the main characters of Victor Hugo's *Les Misérables* to detect communities of characters that were densely connected. These communities, in the words of the authors, "clearly reflect[ed] the subplot structure of the book".

Elson et al. (2010) introduced an interesting idea: so far, two characters had always been linked by an edge if they occurred in the same text-window. In their approach, characters are linked if they converse. The networks are built in an automatic fashion, and heuristics are used to cluster co-referents. The authors's analysis of the networks debunks long standing literary hypotheses. Celikyilmaz et al. (2010) extracts dialogue interactions in order to analyze semantic orientations of social networks from literature. In order to perform large-scale analyses of the works, both Rydberg-Cox (2011) and Suen et al. (2013) extract networks from structured text: Greek tragedies the first, plays and movie scripts the latter.

All the approaches mentioned above produce static networks which are flat representations of the novel as a whole. In them, past, present, and future are represented at once. By means of static networks, time turns into space. The recent work by Agarwal et al. (2012) questions the validity of static network analysis. Their authors introduce the concept of dynamic network analysis for literature, motivated by the idea that static networks can distort the importance of the characters (exemplified through an analysis of Lewis Carroll's *Alice in Wonderland*). A dynamic social network is but the collection of independent networks for each of the parts in which the novel is divided.

3 Turning Novels into Social Networks

3.1 Human Name Recognition

A social network is a structure that captures the relations between a set of actors. The actors in a novel are its characters, and thus extracting person names from the raw text is necessarily the first step

[1] This is particularly evident in William M. Thackeray's novel *Vanity Fair* through the ironic and mocking voice of the narrator, making the reader aware of his describing much more than just the adventures and missfortunes of a collection of invented characters.

to construct a social network from a novel. To that end, we used the `Stanford Named Entity Recognizer` (Stanford NER)[2], to which we applied post-processing recognition patterns in order to enhance its performance in the literary domain.[3] Stanford NER tags the entities on a per-token basis. The name 'Leicester' might be tagged as `person` in one paragraph and as `location` in the next one. With the assumption that a novel is a small universe in which one proper name is likely to refer to the same entity throughout the novel, we eliminate these inconsistencies by re-tagging the file, so that each entity recognized during the filtering is tagged as `person` throughout the file. Each proper name that has been tagged as a `person` more times than as a `location` is also re-tagged consistently as `person`.

Table 1 shows the evaluation of the person name recognizer in novels both originally in English and translated, both before (`StanfordNER`) and after (`FilteredNER`) the filtering. The filtering improves the performance of the entity recognizer significantly in the case of English literature, and only slightly in foreign literature. We evaluated eight chapters randomly picked from eight different novels.[4]

	$Precision$	$Recall$	$F_1 Score$
StanfordNER-Eng	0.9684	0.8101	0.8822
FilteredNER-Trn	0.9816	0.9970	0.9892
StanfordNER-Eng	0.9287	0.7587	0.8351
FilteredNER-Trn	0.8589	0.8277	0.8430

Table 1: Evaluation of person recognition.

3.2 Character Resolution

A list of person names is not a list of characters. Among the extracted names are 'Miss Lizzy', 'Miss Elizabeth', 'Miss Elizabeth Bennet', 'Lizzy', 'Miss Eliza Bennet', 'Elizabeth Bennet', and 'Elizabeth', all of them names corresponding to one only character, the protagonist of Jane Austen's *Pride and Prejudice*. A social network relates entities, and thus it is a crucial step to group all the co-referents together. The task of character resolution has been done in three steps:

- *Human name parsing.* We used an extended version of the Python module `python-nameparser`[5] to parse the recognized names into its different components, so that a name like 'Mr. Sherlock Homes', would have 'Mr.' tagged as *title*, 'Sherlock' as *first name* and 'Holmes' as *last name*.

- *Gender assignation.* Each human name is assigned a gender (*male*, *female*, or *unknown*). We have four lists: with typical male titles ('Sir', 'Lord', etc.), with female titles ('Miss', 'Lady', etc.), with 2579 male first names[6] and with 4636 female first names[7]. To assign a gender to a human name, first the title is considered. If the title is empty or non-informative, the first name is considered. If none are informative of the gender of the character, immediate context is considered: a counter keeps track of counts of 'his' and 'himself' (on the one hand), and of 'her' and 'herself' (on the other) appearing in a window of at most 3 words to the right of the name. Depending on which of the two counters is higher, the human name is assigned one gender or the other. If the conditions are not met, the gender remains unknown.

- *Matching algorithm.* A matching algorithm is responsible for grouping the different co-referents of the same entity from less to more ambiguous:

  1. Names with `title`, `first name` and `last name` (e.g. 'Miss Elizabeth Bennet').
  2. Names with `first name` and `last name` (e.g. 'Elizabeth Bennet').
  3. Names with `title` and `first name` (e.g. 'Miss Elizabeth').
  4. Names with `title` and `last name` (e.g. 'Miss Bennet').
  5. Names with only `first name` or `last name` (e.g. 'Elizabeth' or 'Bennet').

  For each matching step, three points are considered: a first name can appear as a nick-

[2] `http://nlp.stanford.edu/software/CRF-NER.shtml`

[3] A list of 178 honorifics such as 'Sir', 'Lady', or 'Professor' indicating that the adherent proper name is a person, and a list of 83 verbs of utterance such as 'say', 'complain' or 'discuss' in both present and past forms indicating the immediate presence of a person.

[4] *Little Dorrit* and *The Pickwick Papers* by Charles Dickens, *Pride and Prejudice* from Jane Austen, *Dr. Jekyll and Mr. Hyde* by R. L. Stevenson, *The Hunchback of Notre-Dame* by Victor Hugo, *The Phantom of the Opera* by Gaston Leroux, *War and Peace* by Leo Tolstoy, and *Don Quixote of La Mancha* by Miguel de Cervantes.

[5] `http://code.google.com/p/python-nameparser/`

[6] Source: `http://www.cs.cmu.edu/Groups/AI/areas/nlp/corpora/names/male.txt`

[7] Source: `http://www.cs.cmu.edu/Groups/AI/areas/nlp/corpora/names/female.txt`

name ('Lizzy' is 'Elizabeth')[8], a first name can appear as an initial ('J. Jarndyce' is 'John Jarndyce'), and the genders of the names to match must agree ('Miss Sedley' matches 'Amelia Sedley', but not 'Jos Sedley'). If after these steps a referent is still ambiguous, it goes to its most common match (e.g. 'Mr. Holmes' might refer both to 'Sherlock Holmes' and to his brother 'Mycroft Holmes'. According to our algorithm, 'Mr. Holmes' matches both names, so we assume that it refers to the most relevant character of the novel, in this case the protagonist, 'Sherlock Holmes'.

Evaluating character resolution is not a simple task, since the impact of a misidentification will depend on the relevance of the wrongly identified character. The evaluation that we propose (see Table 2) for this task takes into consideration only the 10 most relevant characters in 10 novels.[9]

	Precision	*Recall*	$F_1 Score$
EnglishLit	0.9866	0.9371	0.9612
ForeignLit	0.9852	0.9086	0.9454

Table 2: Evaluation of character resolution.

The evaluation of the gender assignment task (see Table 3) is done on the total number of characters from six different novels.[10]

	Precision	*Recall*	$F_1 Score$
EnglishLit	0.9725	0.8676	0.9171
ForeignLit	0.9603	0.5734	0.7175

Table 3: Evaluation of gender assignment.

3.3 Network Construction

As mentioned in Section 2, two main approaches to create character networks from literary fiction have been proposed. In the first one (hereafter **conversational network**), nodes (i.e. characters) are related by means of an edge if there is a spoken interaction between them. In the second approach (hereafter **co-occurrence network**), nodes are linked whenever they co-occur in the same window of text. A conversational network is well-suited to represent plays, where social interaction is almost only represented by means of dialogue. However, much of the interaction in novels is done off-dialogue through the description of the narrator or indirect interactions. Thus, using a conversational network might not suffice to capture all interactions, and it would definitely have severe limitations in novels with unmarked dialogue, little dialogue or none.[11]

The networks built in this approach are static and dynamic co-occurrence networks.[12] A **static network** allows better visualization of the novel as a whole, and the features extracted from it correspond to a time agnostic analysis of the novel's plot. A **dynamic network** is a sequence of sub-networks, each of which constructed for each of the chapters into which the novel is divided. In it, one can visualize the development of the characters throughout the novel. In both networks, nodes are linked if they co-occur in the same window of text, which in our case is set to be a paragraph, a natural division of text according to discourse. The graph is **undirected** (the direction of the interaction is ignored) and **weighted** (the weight is the number of interactions between the two linked nodes). In 1^{st} person novels, the off-dialogue occurrences of pronoun "I" are added to the node of the character who narrates the story, in order to avoid the narrator (probably the protagonist of the novel) to be pushed to the background.

We used the python library `Networkx`[13] to construct the networks and the network analysis software `Gephi`[14] to visualize them.

[8] A list of names and their hypocoristics is used to deal with this. Source: `https://metacpan.org/source/BRIANL/Lingua-EN-Nickname-1.14/nicknames.txt`

[9] *The Mystery of Edwin Drood* and *Oliver Twist* by Charles Dickens, *Sense and Sensibility* by Jane Austen, *Vanity Fair* by William M. Thackeray, *The Hound of the Baskervilles* by Arthur Conan Doyle, *Around the World in Eighty Days* by Jules Verne, *The Phantom of the Opera* by Gaston Leroux, *Les Misérables* by Victor Hugo, *The Three Musketeers* by Alexandre Dumas, and *Madame Bovary* by Gustave Flaubert.

[10] *Oliver Twist* by Charles Dickens, *Sense and Sensibility* by Jane Austen, *The Hound of the Baskervilles* by Arthur Conan Doyle, *Around the World in Eighty Days* by Jules Verne, *The Phantom of the Opera* by Gaston Leroux, *On the Eve* by Ivan Turgenev.

[11] Examples are Cormac McCarthy's *On the road*, George Orwell's *Nineteen Eighty-Four*, and Margaret Yourcenar's *Memoirs of Hadrian*.

[12] In section 3.4, we offer a qualitative analysis of some networks. We have already motivated our choice for using co-occurrence networks instead of conversational. Both methods would yield very different networks. The reason why we do not provide compared results between both approaches is that we do not consider them quantitatively comparable, since they represent and capture different definitions of what a social relation is.

[13] `http://networkx.github.io/`

[14] `http://gephi.org/`

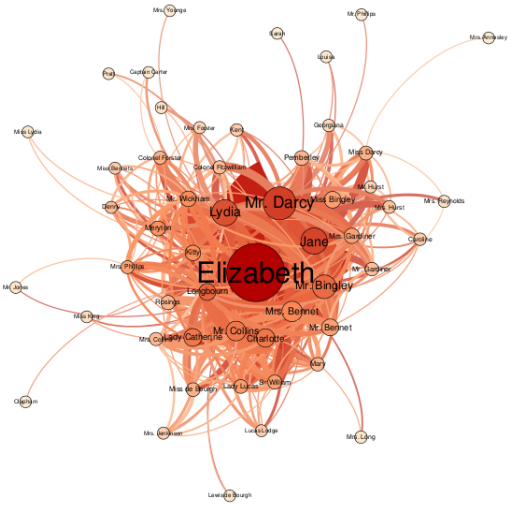

Figure 1: Static network of *Pride and Prejudice*.

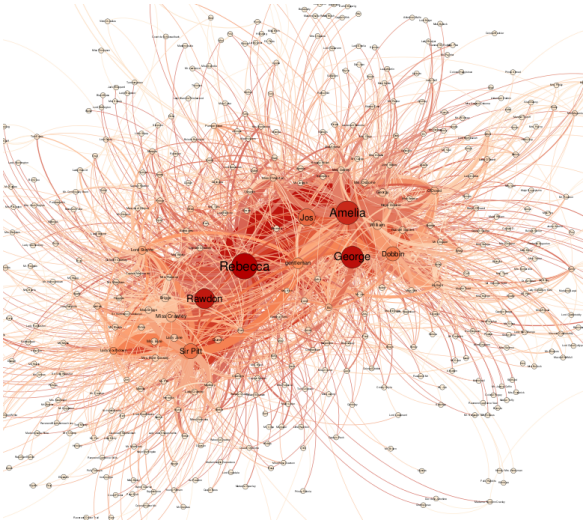

Figure 2: Static network of *Vanity Fair*.

3.4 Network Analysis

The aim of extracting social networks from novels is to turn a complex object (the novel) into a schematic representation of the core structure of the novel. Figures 1 and 2 are two examples of static networks, corresponding to Jane Austen's *Pride and Prejudice* and William M. Thackeray's *Vanity Fair* respectively. Just a glimpse to the network is enough to make us realize that they are very different in terms of structure.

Pride and Prejudice has an indisputable main character (Elizabeth) and the whole network is organized around her. The society depicted in the novel is that of the heroine. *Pride and Prejudice* is the archetypal romantic comedy and is also often considered a Bildungsroman.

The community represented in ***Vanity Fair*** could hardly be more different. Here the novel does not turn around one only character. Instead, the protagonism is now shared by at least two nodes, even though other very centric foci can be seen. The network is spread all around these characters. The number of minor characters and isolate nodes is in comparison huge. *Vanity Fair* is a satirical novel with many elements of social criticism.

Static networks show the skeleton of novels, dynamic networks its development, by incorporating a key dimension of the novel: time, represented as a succession of chapters. In the time axis, characters appear, disappear, evolve. In a dynamic network of Jules Verne's ***Around the World in Eighty Days***, we would see that the character Aouda appears for the first time in chapter 13. From that moment on, she is Mr. Fogg's companion for the rest of the journey and the reader's companion for the rest of the book. This information is lost in a static network, in which the group of very static gentlemen of a London club are sitting very close from a consul in Suez, a judge in Calcutta, and a captain in his transatlantic boat. All these characters would never co-occur (other than by mentions) in a dynamic network.

4 Experiments

At the beginning of this paper we ask ourselves whether the plot of a novel (here represented as its structure of characters) can be used to identify literary genres or to determine its author. We propose two main experiments to investigate the role of the novel structure in the identification of an author and of a genre. Both experiments are considered as an unsupervised classification task.

4.1 Document Clustering by Genre

Data collection.[15] This study does not have a quantified, analogous experiment with which to compare the outcome. Thus, our approach has required constructing a corpus of novels from scratch and building an appropriate baseline. We have collected a representative sample of the most influential novels of the Western world. The resulting dataset contains 238 novels[16]. Each novel

[15]The complete list of works and features used for both experiments can be found in `http://www.coli.uni-saarland.de/~csporled/SocialNetworksInNovels.html`.

[16]Source: `http://www.gutenberg.org/`

was annotated with the genre to which it belongs. The task of assigning a genre to a novel is not trivial. The Russian literary critic Mikhail Bakhtin relates the inherent difficulties in the study of the novelistic genre, being the novel the "sole genre that continues to develop, that is as yet uncompleted" (Bakhtin, 1981). Different sources differ in categorizing the same novels, some novels are labeled with more than one genre, and even some novels are not categorized at all. The process of building and labeling the corpus has therefore been long and laborious.

The decision on how many genres there should be was taken based on observation, resulting in **11 most seen genres**: adventure, historical, romance, satirical, gothic, Bildungsroman, picaresque, mystery, social criticism, science fiction, and children fiction. In order to annotate the data, different sources were contrasted, among which the study guides from Spark Notes[17] and Shmoop[18], popular reading web portals such as Goodread[19], the Wikipedia[20], and different literary research studies for each particular novel. Each novel has been annotated with a maximum of three genres in those cases in which sources did not agree.

Experimental Setup. We propose four different set-ups, representing different fractions of the data set. The **enCorpus** is the set of 184 novels originally written in English. The **trCorpus** is the set of 54 novels originally not written in English, in their translated version. The **alCorpus** is the whole dataset, 238 novels. The **19Corpus** is a subset of 118 British novels from the 19th Century.

4.2 Document Clustering by Author

Data collection. The evaluation of document clustering by author does not pose nearly as many challenges. For this experiment, we have disregarded 1^{st} person narratives.[21] We collected 45 novels from 7 different authors: five British authors from the 19th Century (Jane Austen (*6 novels*), Charles Dickens (*11*), Elizabeth Gaskell (*5*), George Eliot (*7*), and William Thackeray (*6*)), and two Russian realism authors (Ivan Turgenev (*6*) and Fyodor Dostoyevsky (*4*)). For investigative reasons, we have also included the seven novels from the *Harry Potter* fantasy series, by the contemporary British author J. K. Rowling.

Experimental Setup. We propose four different set-ups, focusing on the author. Table 4 shows the authors included in each experiment.

#Corpus	Authors
`Corpus1`	Austen, Dickens, Thackeray, Eliot, Gaskell
`Corpus2`	Austen, Dickens, Thackeray, Eliot, Gaskell, Dostoyevsky, Turgenev
`Corpus3`	Austen, Dickens, Thackeray, Eliot, Gaskell, Rowling
`Corpus4`	Austen, Dickens, Thackeray, Eliot, Gaskell, Dostoyevsky, Turgenev, Rowling

Table 4: Authors in each corpus fraction.

4.3 Feature Selection

The static features that we have used for clustering are mostly well-known metrics drawn from social network analysis. These include measures such as graph density, average clustering coefficient, diameter, radius, proportion of eccentric, central and isolate nodes, and relevance of the main node. Variations of social network analysis metrics are: proportion of male characters, relative weight of the main node, relative weight of the second biggest node, of the ten most important nodes, and of the isolate nodes, and proportion of edges of the main character. Dynamic features control the continued presence of the protagonist throughout the novel, the varying proportion of characters in each stage of the novel, and proportion of characters appearing in only one stage.

In the clustering experiment by genre, we differenciate between features that apply to 1^{st} and 3^{rd} person point-of-view to avoid the disproportionate weight of the narrator to incline the results. Some features not used in the author experiment are added, such as the absolute size of the network both in terms of nodes and of length of the novel, the presence of the main character in the title of the book, the point-of-view, the number of chapters and whether the narrator is known. The author experiment has a total of 27 features, while the genre experiment has 55[22]. The **baseline** we propose is based on content: for each novel a vector with a raw Bag-of-words representation is generated.

For the clustering, we use the `Weka EM` implementation, in which the number of clusters was al-

[17] `http://www.sparknotes.com/`

[18] `http://www.shmoop.com/literature/`

[19] `http://www.goodreads.com/`

[20] `http://www.wikipedia.org`

[21] Whereas the point of view in which the story is written might be indicative of a genre (e.g. some genres might be more prone to use 1^{st} person), in most cases it is not of an author, since they are many the authors that equally use different points of view in their novels.

[22] See footnote 15.

ready pre-defined to the desired number of classes (11 in the case of clustering by genre, 5-8 in the case of clustering by author).

5 Results and Analysis

The results of the clustering are evaluated with respect to the annotated data. The task of evaluating the results of a clustering is not trivial, since one cannot know with certainty which labels correspond to which clusters. In this approach, the labelling of the classes relies on `Weka`'s[23](Hall et al., 2009) **Classes to clusters** evaluation functionality, which assigns a label to the cluster which contains most of the elements of the labeled class, as long as the class is not defining another cluster. The evaluation is based on three popular metrics: purity, entropy and F_1 measure. In the clustering experiments by genre, if one novel is classified as at least one of the correct classes, we consider it to be correct.

#Corpus	Baseline			Our approach		
Metric	Pur	Ent	F_1S	Pur	Ent	F_1S
`enCorpus`	0.45	0.34	0.31	0.46	0.34	**0.33**
`trCorpus`	0.56	0.28	**0.34**	0.44	0.31	0.27
`alCorpus`	0.42	0.35	**0.27**	0.40	0.36	0.26
`19Corpus`	0.53	0.29	0.34	0.58	0.29	**0.40**

Table 5: Genre clustering evaluation.

Table 5 shows the results of both the baseline and our approach in the clustering task by genre.[24] The clustering results are negative, even though not random. The performance is slightly better in works originally written in English (`enCorpus` and `19Corpus`). The reason why the `19Corpus` performs significantly better than the rest of the collections is probably to be found in the fact that all other collections contain documents from very different ages (up to five centuries of difference) and countries of origin. Since novels usually depict the society of the moment, it is not surprising that the more local the collection of texts, the higher the performance of the approach is.

As can be seen in Table 6, the performance of both the baseline and our approach in clustering by author is much higher than by genre.[25] The performance of the baseline approach decreases as it goes away from the same period and same origin, but also as the number of authors in which to cluster the novels increases. Our approach does not suffer too much from the increasing number of classes in which to cluster. Interesting enough, we see how the baseline and our approach yield similar results in both clustering tasks even if the features could not be more different from one vector to the other. As future work, we plan to combine both methods in order to enhance the results.

#Corpus	Baseline			Our approach		
Metric	Pur	Ent	F_1S	Pur	Ent	F_1S
`Corpus1`	0.74	0.20	**0.74**	0.63	0.26	0.63
`Corpus2`	0.64	0.23	0.55	0.60	0.28	**0.60**
`Corpus3`	0.74	0.19	**0.71**	0.71	0.22	**0.71**
`Corpus4`	0.58	0.25	0.52	0.62	0.24	**0.60**

Table 6: Author clustering evaluation.

6 Discussion

6.1 Clustering by Genre

Genres are not clear and distinct classes. By observing the 'incorrectly labeled' cases from our network-based method, we find some interesting patterns: some genres tend to be misclassified always into the same "incorrect" genre. It is the case, for example, of the categories *Bildungsroman* and *picaresque*. Some novels that should have been labeled Bildungsroman are instead considered picaresque, or vice versa. Indeed, one can easily find certain characteristics that are shared in both genres, such as a strong protagonist and a whole constellation of minor characters around him or her. What distinguishes them from being the same genre is that the focus and goal in a Bildungsroman is on the development of the main character. Picaresque novels, on the contrary, usually have no designed goal for the protagonist, and consist of a sequence of adventures, most of them unrelated and inconsequential to each other. The same kind of strong relationship exists, in a lesser extent, between *historical*, *social* and *satirical* genres. These three genres are somewhat intertwined. Social criticism might be carried out through a satirical novel, which might be set to take place in the past, making it a historical novel. Our method classifies these three genres indistinctly together, and this might well be because of their very similar structural characteristics.

We consider this experiment a first step in the task of novel clustering by genre. The method that

[23] `http://www.cs.waikato.ac.nz/ml/index.html`

[24] The yielded clusters and their quality can be found in `http://www.coli.uni-saarland.de/~csporled/SocialNetworksInNovels.html`

[25] See footnote 24.

we have presented is far from being perfected. We have used all the features that we have designed in an unweighted way and without optimizing them. However, it is assumed that some features will have a bigger impact than others at determining genres. A blunt analysis of the role of the features informs that the relevance of the protagonist node is key, for example, to identify genres such as Bildungsroman and picaresque. A high proportion of minor or isolate nodes is, for example, a very good indicator of satirical, social, and historical genres. An unknown narrator is a good indicator that we are in front of a science fiction novel, while a mixed point of view is usually kept for either science fiction, gothic, or mystery novels.

6.2 Clustering by Author

The clustering by author is much clearer than the clustering by genre, and very interesting patterns can be found when looked in detail. One can learn, for instance, that the structure of Jane Austen novels are in the antipodes of the structure of William M. Thackeray's works (as could be inferred from Figures 1 and 2). These two authors are, alongside Rowling, the easiest authors to identify. In fact, a clustering of only the novels by these three authors result in a perfectly clear-cut grouping with no misclassifications. Dickens and Eliot are on the other hand the most difficult authors to identify, partly because their structures are more varied.

An in-depth study of the role of each feature in the clustering provides a very interesting view of the literary work of each author. We can see in our sample that female writers (in particular Austen and Gaskell) have a much higher proportion of female characters than male writers (in particular Dickens, Turgenev, and Dostoyevsky), with Thackeray and Rowling depicting a more equal society. Examples of behaviors that can be read from the clustering are many. The very low graph density of Thackeray's novels contrasts with the high density of the novels by Austen and Turgenev, whereas all of Gaskell's novels have a strikingly similar graph density. In the case of the *Harry Potter* books, the first ones are significantly denser than the last ones. The role of the protagonist also varies depending on the author. It is very important in the works by Austen, Gaskell, and Rowling, in which the presence of the protagonist is constant throughout the novel. Turgenev's protagonists are also very strong, even though their presence varies along the chapters. Thackeray, on the other hand, is by far the author that gives more weight to minor characters and isolates. Turgenev has a high proportion of isolate nodes, while they are almost null in works by Rowling and Austen. The dynamic features show the different distributions of characters over the time of the novel. They allow us see very clearly in which stages coincide the maximum number of characters (the falling action in the case of Austen, the dénouement in the case of Eliot, the rising action in the case of Rowling). They allow us to see also how a very high proportion of characters in Thackeray's novels appear in only one stage in the novel, to then disappear. In the other side of the spectrum are Austen and Dostoyevsky, whose characters arrive in the novel to stay. These are only some of the most evident conclusions that can be drawn from the author-clustering experiment. A more in-depth analysis could be useful, for example, to identify changes in the work of one same author.

7 Conclusion

This work is a contribution to the field of quantitative literary analysis. We have presented a method to build static and dynamic social networks from novels as a way of representing structure and plot. Our goal was two-fold: to learn which role the structure of a novel plays in identifying a novelistic genre, and to understand to what extent the structure of the novel is a fingerprint of the style of the author. We have designed two experiments shaped as unsupervised document classification tasks. The first experiment, clustering by genre resulted in a negative clustering but, if analyzed qualitatively, shows that the approach is promising, even if it must be polished. The second experiment, clustering by author, outperformed the baseline and obtained good enough positive results. Authorship attribution is mostly used for either forensic purposes or plagiarism identification. However, we have shown that an analysis of the features and yielded clustering can also be used to explore structural inter- and intra-similarities among different authors.

8 Acknowledgements

The authors thank the anonymous reviewers for their helpful comments and suggestions.

References

Scott Aaronson. 2001. Stylometric clustering: A comparison of data-driven and syntactic features. Technical report, Computer Science Department, University of California, Berkeley.

Apoorv Agarwal, Augusto Corvalan, Jacob Jensen, and Owen Rambow. 2012. Social network analysis of alice in wonderland. In *Workshop on Computational Linguistics for Literature, Association for Computational Linguistics*, pages 88–96.

Navot Akiva and Moshe Koppel. 2012. Identifying distinct components of a multi-author document. *EISIC*, pages 205–209.

Ricardo Alberich, Josep Miró-Julià, and Francesc Rosselló. 2002. Marvel universe looks almost like a real social network. *cond-mat/*.

Sarah Allison, Ryan Heuser, Matthew Jockers, Franco Moretti, and Michael Witmore. 2011. Quantitative formalism: an experiment. *Literary Lab*, Pamphlet 1.

Harald Baayen, Hans van Halteren, and Fiona Tweedie. 1996. Outside the cave of shadows: using syntactic annotation to enhance authorship attribution. *Literary and Linguistic Computing*, 11:121–131.

Mikhail Bakhtin. 1981. Epic and novel: Towards a methodology for the study of the novel. In J. Michael Holquist, editor, *The dialogic imagination: four essays*. Unversity of Texas Press.

Ron Bekkerman, Hema Raghavan, and James Allan Koji Eguchi. 2007. Interactive clustering of text collections according to a user-specified criterion. In *In Proceedings of IJCAI*, pages 684–689.

Asli Celikyilmaz, Dilek Hakkani-tur, Hua He, Greg Kondrak, and Denilson Barbosa. 2010. The actor-topic model for extracting social networks in literary narrative. In *NIPS Workshop: Machine Learning for Social Computing*.

David K. Elson and Kathleen R. McKeown. 2010. Automatic attribution of quoted speech in literary narrative. In *Association for the Advancement of Artificial Intelligence*.

David K. Elson, Nicholas Dames, and Kathleen R. McKeown. 2010. Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*.

Suhit Gupta, Hila Becker, Gail Kaiser, and Salvatore Stolfo. 2005. A genre-based clustering approach to content extraction. Technical report, Department of Computer Science, Columbia University.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The weka data mining software: An update. *SIGKDD Explorations*, Volume 11.

David I. Holmes and Richard S. Forsyth. 1995. The federalist revisited: New directions in authorship attribution. *Literary and Linguistic Computing*, 10:111–127.

Robert Layton, Paul Watters, and Richard Dazeley. 2011. Automated unsupervised authorship analysis using evidence accumulation clustering. *Natural Language Engineering*, 19:95–120.

Gerard Ledger and Thomas Merriam. 1994. Shakespeare, fletcher, and the two noble kinsmen. *Literary and Linguistic Computing*, 9:235–248.

Franco Moretti. 2011. Network theory, plot analysis. *Literary Lab*, Pamphlet 2.

M. E. J. Newman and M. Girvan. 2003. Finding and evaluating community structure in networks. *Physical Review E*, 69:1–16.

Bohdan Pavlyshenko. 2012. The clustering of author's texts of english fiction in the vector space of semantic fields. *The Computing Research Repository*, abs/1212.1478.

Céline Poudat and Guillaume Cleuziou. 2003. Genre and domain processing in an information retrieval perspective. In *ICWE*, pages 399–402.

Vladimir I. A. Propp. 1968. *Morphology of the folktale*. University of Texas Press.

Jeff Rydberg-Cox. 2011. Social networks and the language of greek tragedy. *Journal of the Chicago Colloquium on Digital Humanities and Computer Science*, 1:1–11.

Graham Alexander Sack. 2011. Simulating plot: Towards a generative model of narrative structure. In *Complex Adaptive Systems: Energy, Information and Intelligence: Papers from the 2011 AAAI Fall Symposium (FS-11-03)*, pages 127–136.

Graham Sack. 2012. Character networks for narrative generation. In *Intelligent Narrative Technologies: Papers from the 2012 AIIDE Workshop, AAAI Technical Report WS-12-14*, pages 38–43.

Caroline Suen, Laney Kuenzel, and Sebastian Gil. 2013. Extraction and analysis of character interaction networks from plays and movies. Retrieved from : http://dh2013.unl.edu/abstracts/ab-251.html, July.

Bin Zhang. 2013. *Learning Features for Text Classification*. Ph.D. thesis, University of Washington.