Orbit: Writing Around Pynchon

ISSN: 2044-4095

Author(s): Christos Iraklis Tsatsoulis

Affiliation(s): Institute of Technology Blanchardstown, Dublin, Ireland and Supreme Joint War College, Thessaloniki, Greece

Title: Unsupervised text mining methods for literature analysis: a case study for Thomas Pynchon's *V.*

Date: 2013

Volume: 1

Issue: 2

URL: https://www.pynchon.net/owap/article/view/44

DOI: http://dx.doi.org/10.7766/orbit.v1.2.44

Abstract:

We demonstrate the use of unsupervised text mining methods for the analysis of prose literature works, using Thomas Pynchon's novel *V*. as an example. Our results suggest that such methods may be employed to reveal meaningful information regarding the novel’s structure. We report results using a wide variety of clustering algorithms, several distinct distance functions, and different visualization techniques. The application of a simple topic model is also demonstrated. We discuss the meaningfulness of our results along with the limitations of our approach, and we suggest some possible paths for further study.

Unsupervised text mining methods for literature analysis: a case study for Thomas Pynchon's *V.*

Christos Iraklis Tsatsoulis

1. Introduction

The application of algorithmic and computational techniques and methods to literature and humanities studies has lately resulted in the emergence of a novel research field termed *digital humanities*[1]: there is already an organization called the Alliance of Digital Humanities Organizations (ADHO), the Blackwell's *Companion to Digital Humanities*[2] and *Companion to Digital Literary Studies*,[3] and, to the best of our knowledge so far, at least seven dedicated peer-reviewed academic journals, namely the *Journal of Digital Humanities* (open access), the *Digital Humanities Quarterly* (open access), the *Digital Medievalist* (open access), the *Digital Studies / Le champ numérique* (open access), the *Journal of Digital Culture & Electronic Scholarship* (open access), the *Journal of Data Mining & Digital Humanities* (open access), and the *Literary & Linguistic Computing*. It should come as no surprise that natural language processing and text mining techniques have come to play a central part in this emerging field,[4] and it is exactly in this context that the present work should be placed.

In this article, we demonstrate the capabilities of unsupervised[5] text mining techniques in revealing useful and meaningful information about the structure of prose literature works. Our exposition aims at simplicity and clarity of the general methods used, so as to be of introductory merit to an uninitiated reader. We have chosen Thomas Pynchon's novel *V*. as our example, which should be familiar to *Orbit* readers, as it is well known that the novel exhibits a highly heterogeneous structure, with two minimally intersecting storylines running in parallel. Our purpose is to explicitly demonstrate that the computational techniques employed are capable of revealing this heterogeneous structure at the chapter level, possibly along with other, less expected information and insight. Hence, our point of departure is not a literary question ("Is there a heterogeneous plot structure in *V*.?"), whose answer is arguably known and well established;

DOI: http://dx.doi.org/10.7766/orbit.v1.2.44

rather, it is to provide adequate evidence that the computational analysis results indeed converge to the already known answer. In other words, here we aim to legitimize the use of such techniques in the eyes of the uninitiated, and possibly skeptical (or even suspicious[6]) scholar, by verifying that they confirm the existing critical readings of the novel. Nevertheless, on the way, we have come upon a slight revision of the accepted division of the novel between the two storylines, as we explain in detail in Section 2.

Trying to clarify further, and to avoid possible misunderstandings regarding the scope of the present study: this article is written from the point of view of a data scientist, and our strategic objective is a) to convince Pynchon scholars that there is indeed merit in using such techniques to aid the critical analysis, and b) possibly to help initiate their application in critical problems and questions yet unanswered, or even not yet posed. We will pause here, to come back to this discussion in the final section of the article.

A work like the present one can easily grow to an inconvenient (and possibly threatening) length and complexity, if one attempts to take notions like “rigour” and “completeness” at face value, and thus tries to introduce in detail all the technical concepts involved. As our stated objective is to provide a convincing demonstration for the uninitiated reader, we deliberately choose not to go down this path: hence, we mainly introduce the relevant concepts in an intuitive manner, just enough to facilitate the reader’s smooth engagement with the main findings. In every case, appropriate references are cited, which the interested reader can consult for delving further into the computational techniques employed.

The general structure of the rest of this article is as follows: in Section 2 we provide a brief overview of the novel, and we frame more precisely our research question; the basic framework of our computational approach is introduced in Section 3; in Sections 4-7 we present our computational experiments and findings, introducing also the relevant concepts in the above stated manner; Section 8 concludes with a comprehensive discussion regarding the interpretation of our findings, the limitations of our approach, and some suggestions for possible future work.

2. Overview of the novel

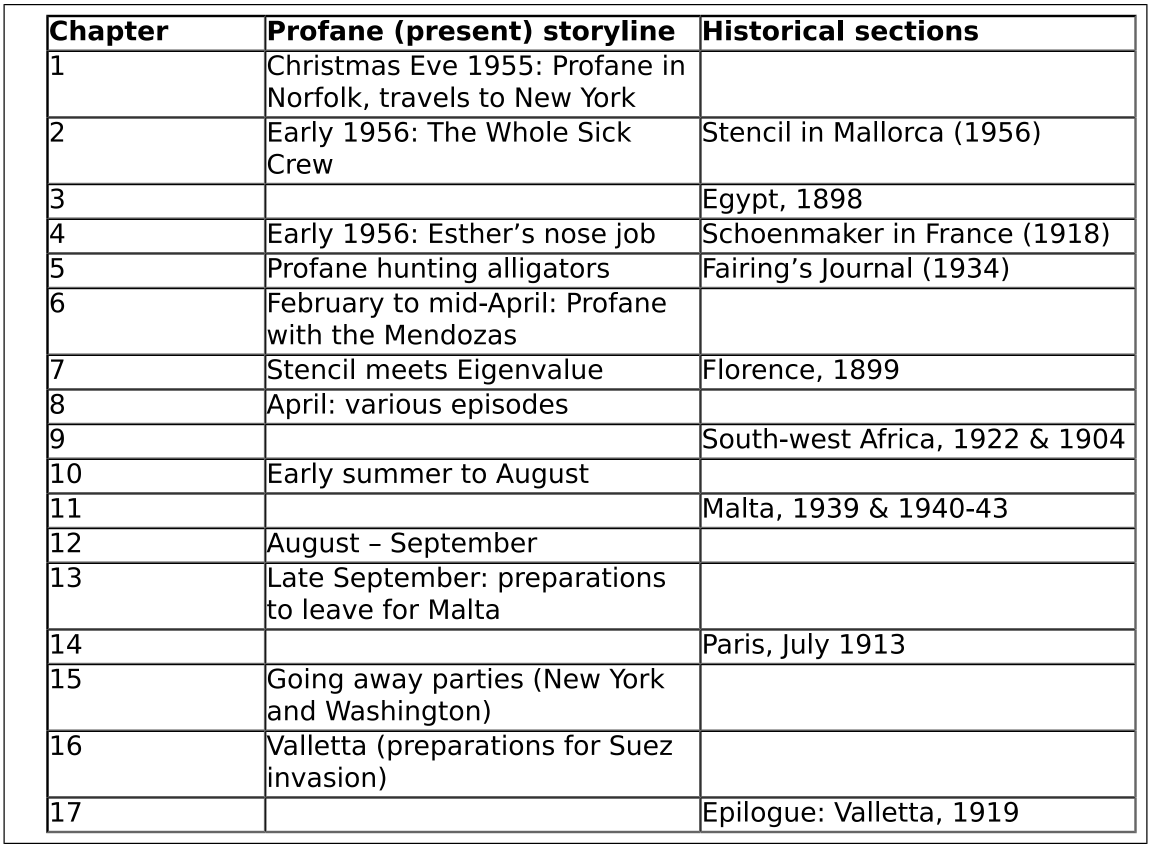
As already mentioned, Thomas Pynchon's novel *V*. consists of two minimally intersecting storylines running in parallel, a fact that is rather universally recognized in the relevant critical literature:

> Dualism structures Pynchon's first novel, *V.*, a multifaceted work stretched between two picaresque plots. The first plot involves Benny Profane […] Profane wanders the "streets" of the present – a period of several months in 1955 and 1956. His motion […] frames the other episodes in the novel. Profane's travels intersect those of Herbert Stencil […] In contrast to Profane's, Stencil's movements have purpose: He is searching for manifestations of a mysterious female called V., who he believes has appeared at various social and political junctures since the turn of the century.[7]

> Like *In Our Time* and *U.S.A.* [*V.*] intercalates sections within a linear narrative set in 1956 in order to broaden the scope of that narrative. The two main sequences which alternate with each other and thus establish one of the novel's rhythms, are the latter which centres on a character called Benny Profane and takes place mainly in New York, and a series of historical chapters which spread from 1898 to 1943. The historical sections are linked by the search of one Herbert Stencil for a mysterious figure called V.[8]

> "[T]here are two main threads or plots to [the novel's] structure, threads that begin far apart from each other but ultimately intersect and interweave, forming a "V" in the plot itself. One storyline of the book details the life and adventures of Benny Profane and is set in the mid 1950s; the other line of the book describes Herbert Stencil's quest for V. "herself," and includes most of the key, calamitous events of the twentieth century.[9]

Somewhat to our surprise, despite this universal agreement regarding the existence of two different storylines in the novel, it seems that there has never been an attempt to *exclusively* map each chapter to *one and only one storyline*. Indeed, and to the best of our knowledge, the closest one has come to such a distinction is a relevant table in David Seed's *The Fictional Labyrinths of Thomas Pynchon*, which we reproduce in Table 1 below.



Chapter	Profane (present) storyline	Historical sections
1	Christmas Eve 1955: Profane in Norfolk, travels to New York	
2	Early 1956: The Whole Sick Crew	Stencil in Mallorca (1956)
3		Egypt, 1898
4	Early 1956: Esther’s nose job	Schoenmaker in France (1918)
5	Profane hunting alligators	Fairing’s Journal (1934)
6	February to mid-April: Profane with the Mendozas	
7	Stencil meets Eigenvalue	Florence, 1899
8	April: various episodes	
9		South-west Africa, 1922 & 1904
10	Early summer to August	
11		Malta, 1939 & 1940-43
12	August – September	
13	Late September: preparations to leave for Malta	
14		Paris, July 1913
15	Going away parties (New York and Washington)	
16	Valletta (preparations for Suez invasion)	
17		Epilogue: Valletta, 1919

Table 1: The exact distribution of present and historical episodes per chapter (adapted from Seed, pp. 71-72)

As it can be seen from Table 1, most of the chapters are indeed “pure”, in the sense that they belong to one and only one storyline; nevertheless, there are four chapters (2, 4, 5, and 7) that seem to contain elements from both storylines. As we intend to use the individual chapters of the novel as our “base units”, we need a way to obtain a one-to-one mapping of chapters to storylines. So, we put forward the following, “operational” definition, for mapping individual chapters to each one of the two storylines:

If a chapter takes place at the novel's present *and* it involves Benny Profane,[10] it belongs to the Profane storyline, irrespectively of what other nested stories it may contain; otherwise, it belongs to the V. storyline.

We will strongly argue that the above definition, irrespectively of its “operational” potential for the purposes of our study, is indeed a natural and intuitive one, and the most probable answer once the relevant question of chapters-to-storylines mapping has been posed. Also, as we shall see, our computational results justify this choice ex post facto.[11]

The above definition leaves us with 11 chapters in the Profane storyline, and 6 chapters (including the epilogue) in the V. storyline.[12] This chapter division, along with some other relevant information, is summarized in Table 2.

#	Title	Storyline	Place	Time
1	In which Benny Profane, a schlemihl and human yo-yo, gets to an apocheir	Profane	New York	Present
2	The Whole Sick Crew	Profane	New York	Present
3	In which Stencil, a quick-change artist, does eight impersonations	V.	Egypt	1898
4	In which Esther gets a nose job	Profane	New York	Present
5	In which Stencil nearly goes West with an alligator	Profane	New York	Present
6	In which Profane returns to street level	Profane	New York	Present
7	She hangs on the western wall	V.	Florence	1899
8	In which Rachel gets her yo-yo back, Roony sings a song, and Stencil calls on Bloody Chiclitz	Profane	New York	Present
9	Mondaugen's story	V.	South-west Africa	1922
10	In which various sets of young people get together	Profane	New York	Present
11	Confessions of Fausto Maijstral	V.	Malta	1939-1943
12	In which things are not so amusing	Profane	New York	Present
13	In which the yo-yo string is revealed as a state of mind	Profane	New York	Present
14	V. in love	V.	Paris	1913
15	Sahha	Profane	New York	Present
16	Valletta	Profane	Malta	Present
17	Epilogue, 1919	V.	Malta	1919

Table 2: Overview of the book chapters, including the mapping of chapters-to-storylines

With the novel structure as depicted in Table 2, our research question can be now stated as follows: given the heterogeneous nature of the narrative as imposed by the two different storylines, can we construct relatively simple unsupervised text mining techniques that can reveal structural heterogeneities at the chapter level? In other words, can we come up with relatively simple algorithms that can distinguish between the two storylines, so as to group the corresponding chapters separately in a meaningful way? And if yes, how stable and consistent can such groupings be with varying methods, algorithms, parameterizations, and preprocessing tasks applied?

In answering the above questions, and in what follows, we must keep in mind two things: first, it is naturally and intuitively expected that the 11 chapters of the Profane storyline bear a greater degree of similarity among them regarding word usage than with the V. storyline chapters; at the same time, the 6 chapters of the V. storyline are not expected to bear an analogous degree of similarity among them, as their narrative intra-diversity is much wider than that of the Profane chapters.[13] These observations will serve as a means of model validation, in order to justify or not the results produced.

3. Basic framework and concepts of the computational approach

With the exception of Section 6, all results reported here are based on the *bag-of-words* assumption, i.e. we simply count individual words and compute word frequencies, without taking into account combinations of words or any other higher-order semantic structure of the text.[14] The limitations of such an approach are apparent, but, as already stated, our purpose here is to keep the techniques used as simple as possible, in order to demonstrate their power and applicability in a most elementary setting; several ways by which this assumption can be relaxed and extended are discussed in Section 8. In accordance with the quantitative text analysis framework and the relevant terminology, we consider the novel as a *document collection*, where the individual documents are indeed the book chapters.

Based on the above mentioned bag-of-words assumption, the simplest approach in order to quantify the content of a document is simply to compute the frequencies of the individual words (terms) contained in it, and then represent the document as the weighted set of these terms, with the weights being the computed term frequencies. This is called (simple) *term-frequency* (TF) weighting, and it is indeed a valid document representation approach. Nevertheless, it happens that we can also do somewhat better, the rationale being as follows: we would like to give more weight to terms that may appear very frequently in only one (or a subset) of our documents, as these terms are possibly the exact ones that mostly signify the *differences* in the content of our documents. This leads to the *term-frequency/inverse-document-frequency* (TF-IDF)[15] weighting, which can be shown to possess the following qualitative properties:[16]

1. It is highest, for a term that occurs *many* times within a *small* number of documents in the collection (thus lending high discriminative power to those documents).
2. It is lower, for a term that occurs fewer times in a document, or occurs in many documents of the collection.
3. It is lowest, for a term that appears in virtually all documents of the collection.

We employ both TF and TF-IDF weighting schemes in our experiments, indicating our choice each time.

The end result of the above “text quantification” process is the representation of a document as a mere list of numbers,[17] where the list length is equal to the number of different terms contained in the document, and the list entries are the (TF or TF-IDF) weights of each individual term.[18] Stacking these lists together for all documents in a given collection, we get the *term-document matrix*, which exhibits how exactly the various terms in a collection are distributed among its constituting documents.

Having effectively transformed the text of the novel into a term-document matrix (with the documents being the individual chapters), i.e. a matrix of *numbers*, as described above, we can now process it, using several quantitative and computational techniques appropriate for our purpose.

Of fundamental importance in what follows – actually in almost every approach in the quantitative analysis of text – are the notions of similarity and distance. Informally speaking, the *similarity* between two data objects (i.e. two documents, in our case) is a numerical measure of the degree to which the two objects are alike. In analogy, the dissimilarity is a measure of the degree to which two data objects are different. Often, the term *distance* is used as a synonym for dissimilarity.[19] Applied to our case, it should be obvious from the above definitions that the *lower* the distance between two documents (expressed as entries in a term-document matrix), the more *similar* these two documents are, while a higher distance denotes a greater dissimilarity between two documents.[20]

There are several different measures which can be used in order to quantify the distance between data objects, and the choice is usually dictated by the specific problem at hand.[21] The *Euclidean distance*[22] is a generalisation of our usual notion of distance between two points in our everyday, 3-dimensional space. The *cosine similarity*[23] is frequently used for text and document analysis; as it has been shown that it exhibits a very high and almost perfect negative correlation with the Euclidean distance (i.e. the higher the cosine similarity between two objects, the lower their Euclidean distance),[24] and since we utilize the Euclidean distance in what follows, we will not employ the cosine similarity here. Going into the details of the different distance functions employed is clearly beyond the scope of the present article; the relevant list is shown in Table 3 (Section 4) below, and more technical details can be found in the provided references.

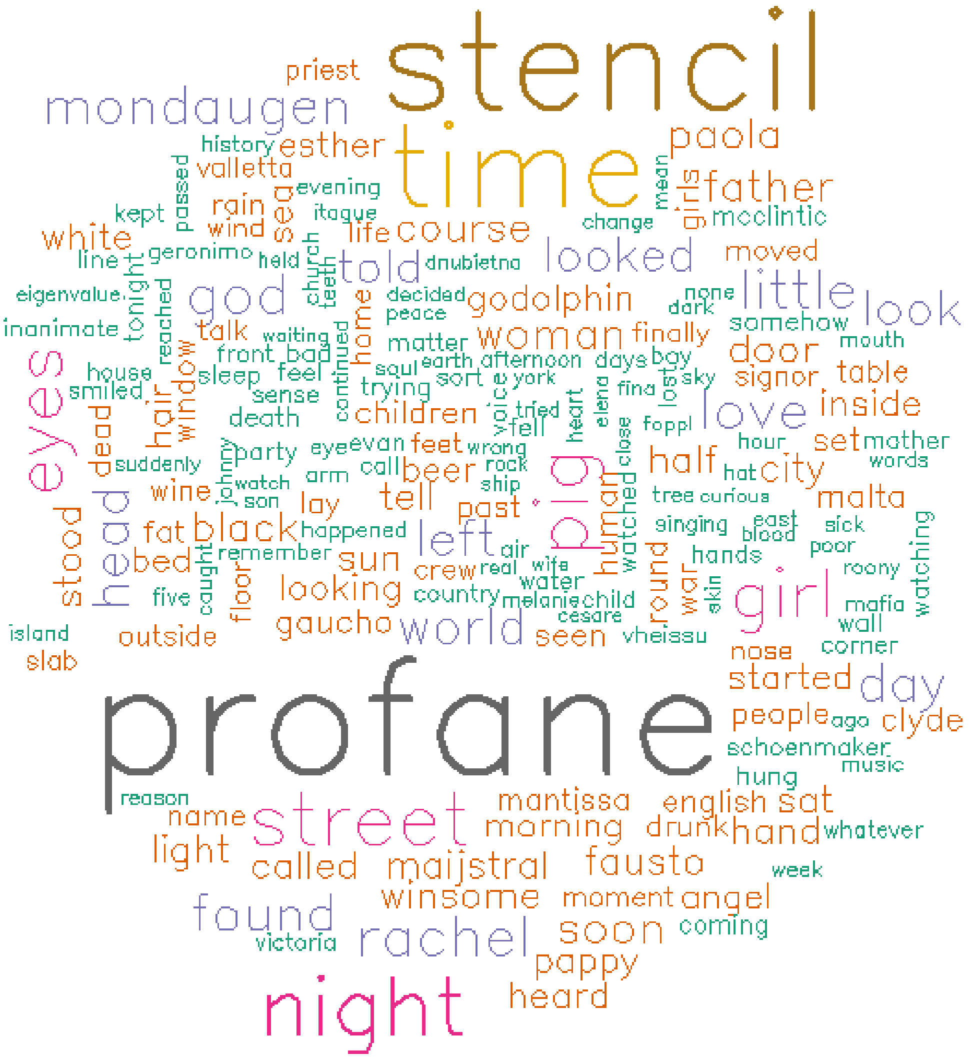


Figure 1: The 300 most frequent terms in the whole novel

In what follows, except otherwise mentioned, the typical text preprocessing tasks of stop words[25] and punctuation removal have been applied. As this is a literary text, we did not perform word stemming.[26] We also found that conversion to lowercase or otherwise has occasional impacts to the results, so we keep it as a parameter for experimentation. As a kind of kick-off, and before proceeding to our main results, in Fig. 1 we present a wordcloud visualization of the 300 most frequent terms in the whole novel. It can be seen that, excluding the character names “Profane”, “Stencil”, and “Pig”, the most frequently occurring terms are “time”, “night”, “street”, “girl”, and “eyes”.

4. Hierarchical clustering

Intuitively speaking, *clustering* refers to the grouping of similar objects together, whereas the groups (clusters) thus produced are thought of as being meaningful, useful, or both.[27] Cluster analysis has a rather long history in various fields of physical and social sciences, including the quantitative analysis of documents and texts. There are several different types and methods for clustering data; here we will restrict the discussion to what is termed as agglomerative hierarchical clustering, which is the type of clustering most often used for this kind of text analysis.[28]

A *hierarchical clustering* is a set of nested clusters that are organized as a tree, and frequently visualized as a tree-like diagram called a *dendrogram*.[29] Usually, the leaves of the tree are singleton clusters of individual data objects,[30] which, the reader is reminded, in our case are the individual book chapters. *Agglomerative* means that the procedure starts with the single data objects as the (trivial) individual clusters and, at each step, merges the closest pair of clusters, according to the particular distance function (see Section 3) used.[31] It should be intuitively obvious, even from this short discussion, that book chapters that are “close” together are expected to be found in the same branch of the corresponding dendrogram visualizations, and “away” from other chapters, with which they are less similar.

Given a particular distance function, there are several different hierarchical clustering methods available, depending on how exactly the distance *between clusters* is defined: in the *single linkage* method, this distance is defined as the distance between the closest two points that are in different clusters, while in the *complete linkage* method it is the distance between the farthest two points in different clusters;[32] UPGMA and WPGMA methods stand for “unweighted/weighted pair group method using arithmetic averages” respectively,[33] while Ward’s method uses a somewhat different cluster distance measure, involving the increase in the variance of the distances between the data objects and the cluster centroids, when the two clusters are merged.[34]

At this point, we would like to urge the reader not to let himself or herself be discouraged by the introduced technical terminology, which serves only for making the present article self-sufficient regarding terms and definitions: we argue that the rest of this section can be safely read and comprehended intuitively, without any direct reference to the technical definitions given above.

That said, in the rest of this section, we present hierarchical clustering results involving six (6) different distance functions between data objects (book chapters), each one of them tried with five (5) different clustering methods.[35] We stress that, to the best of our knowledge, this is far from typical in similar studies of literary texts, where the clustering experiments are usually limited to the Euclidean or cosine distance functions, with Ward’s or complete linkage clustering methods,[36] with usually no justification provided for the choice of a particular distance function or clustering method. The rationale for employing such a rather numerous variety of clustering approaches in our study is discussed and justified in the final section of the article.

To begin with, the results of hierarchical chapter clustering using Ward's method with the Manhattan distance[37] are shown in Fig. 2 below.

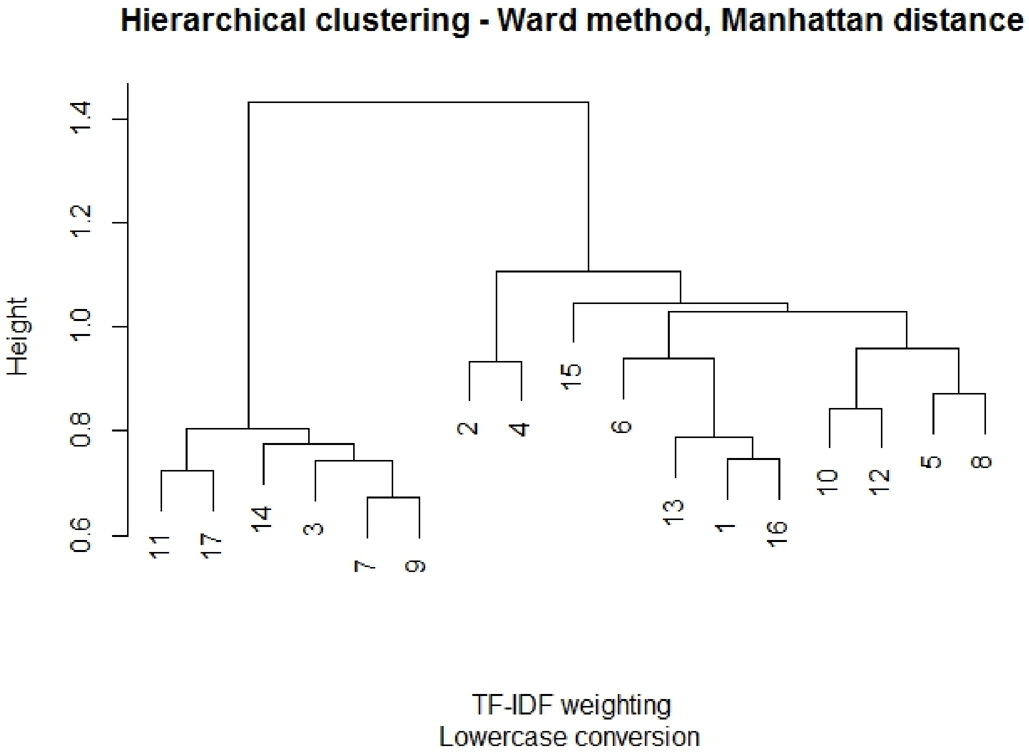


Figure 2: Hierarchical clustering of chapters using Ward's method with the Manhattan distance. The six V. storyline chapters stand out clearly on the left branch of the dendrogram. The picture is very similar using simple term frequency (TF) weighting without conversion to lowercase.

In Fig. 2, the six chapters of the V. storyline (3, 7, 9, 11, 14, and 17) are clearly grouped together and “away” from the chapters of the Profane storyline, which are also themselves grouped together. The clustering of Fig. 2 is produced by weighting the terms according to their TF-IDF count (see Section 3) with lowercase conversion, but the picture is qualitatively very similar with simple TF weighting and preservation of the uppercase characters. Two more clustering examples with very similar results are shown in Figs. 3 and 4, using different clustering methods, distance functions, and term weighting.

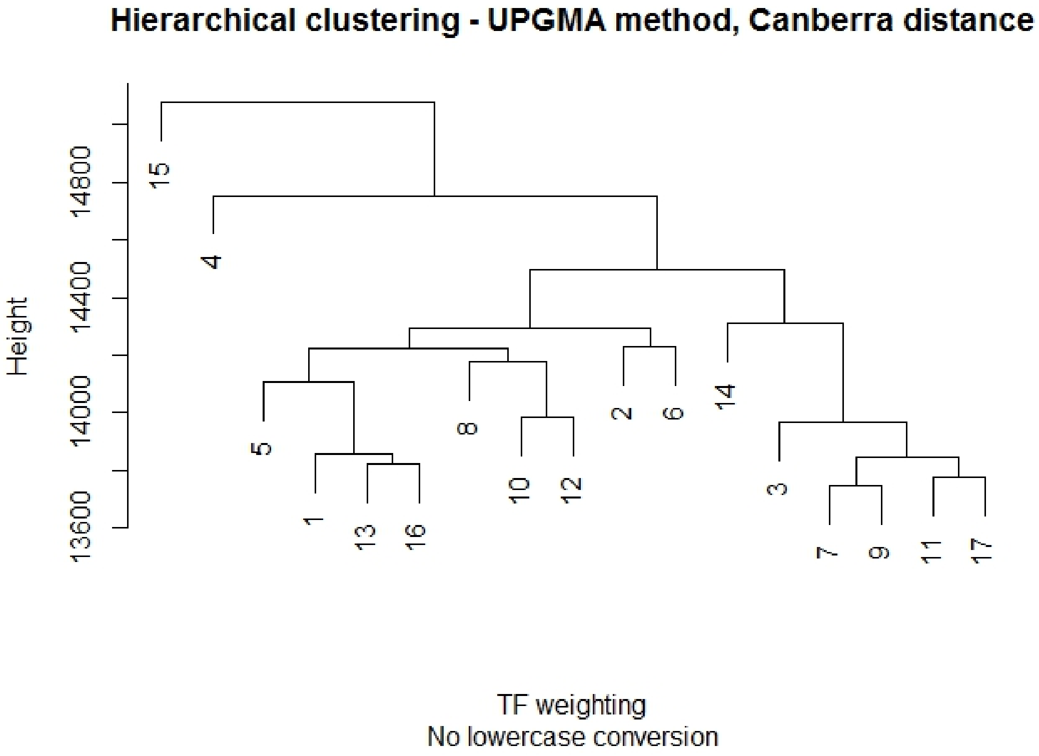


Figure 3: Hierarchical clustering of chapters using the UPGMA method with the Canberra distance. The six V. chapters stand grouped together in the rightmost branch of the dendrogram (14, 3, 7, 9, 11, 17). Changing the weighting to TF-IDF or converting to lowercase gives practically identical results.

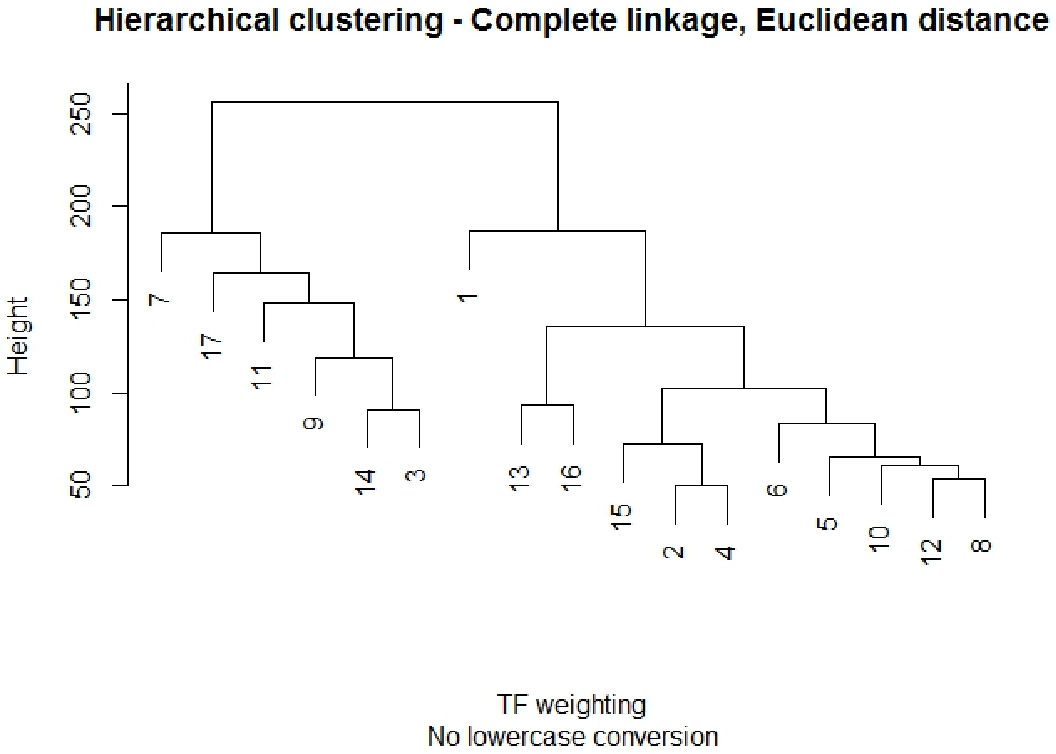


Figure 4: Hierarchical clustering using the complete linkage method with Euclidean distance. Again, the six V. storyline chapters can be seen to occupy their own dedicated (left) branch of the dendrogram.

The results of our thorough experiments with hierarchical clustering are summarized in Table 3 below. The tick symbols mean that, for the particular combination of clustering method (columns) and distance function (rows), we were always able to find a hierarchical clustering similar to that of Figs. 2-4 above, by varying the term weighting (TF or TF-IDF), the percentage of sparse terms removed, and, rarely, the conversion or not to lowercase. When using the Canberra and binary distance functions with the single linkage clustering method, we had again a dendrogram branch consisting exclusively of five chapters of the V. storyline, but Chapter 14 was not included. For the UPGMA method with the Euclidean distance, we had the case that Chapter 16 was grouped together with the six V. storyline chapters, again in a dedicated branch of the corresponding dendrogram with no other chapters of the Profane storyline.

Method Distance	Ward	Complete Linkage	Single linkage	UPGMA	WPGMA
Euclidean	✓	✓	✓	#16 into V. storyline	✓
Manhattan	✓	✓	✓	✓	✓
Canberra	✓	✓	#14 misgrouped	✓	✓
Maximum	✓	✓	✓	✓	✓
Binary	✓	✓	#14 misgrouped	✓	✓
Minkowski	✓	✓	✓	✓	✓

Table 3: Summary of results for five hierarchical clustering methods used in combination with six different distance functions. For the Minkowski distance, several different values for the parameter *p* were tried (3-5, 10, 30), with generally similar (positive) results.

From the results shown in Table 3 and Figs. 2-4 above, it is apparent that the clustering algorithms employed are capable of capturing the heterogeneities among the book chapters in a robust and consistent way, across a rather wide spectrum of settings, approaches, and parameters.

5. Graph visualizations

Utilizing the distance calculations produced as a part of the clustering approaches presented in the previous section, we are able to come up with a different, ad hoc visualization technique that can highlight the book structure from an alternative viewpoint. The idea behind it is simple: we visualize the chapters as nodes in a graph; we apply a certain threshold to the distance measures, so that if the distance between two chapters is lower than this threshold, we connect these two chapters with a link; otherwise, if the distance between two chapters is greater than this threshold (i.e. if two chapters are more *dissimilar* according to the particular distance function used), we do not connect them. That way, we expect to get a graph where the most similar chapters will be connected between them, and disconnected from the other ones. By applying this idea to the Euclidean distance function between our chapters, we get the graph shown in Fig. 5.

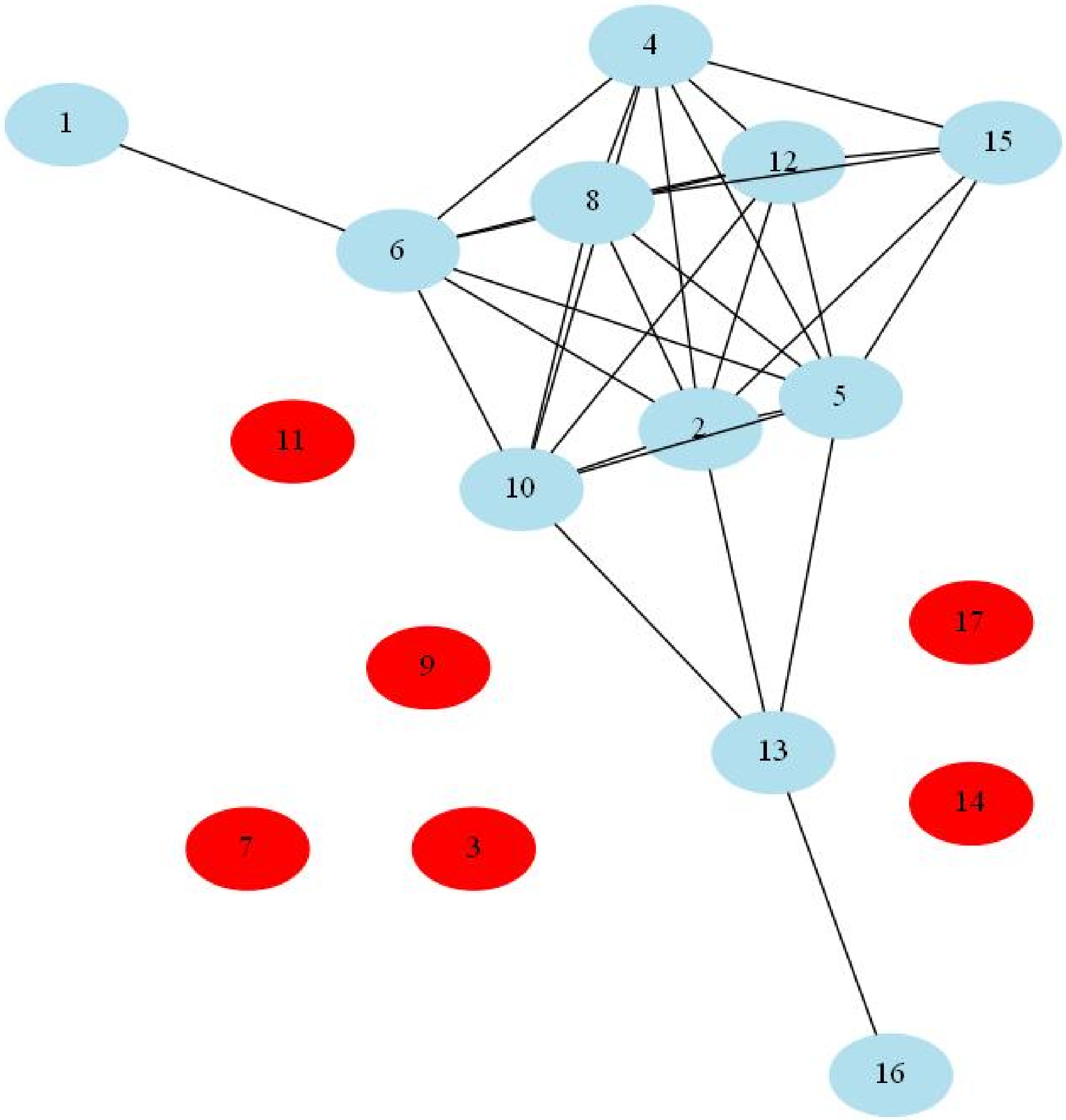


Figure 5: Graph visualization of the book chapters. Links correspond to the Euclidean distance being below a certain threshold. For convenience, the Profane storyline chapters are depicted in blue, and the V. storyline chapters in red.

From Fig. 5, we can observe the following:

1. All the chapters of the Profane storyline are connected in a main cluster, while all six chapters of the V. storyline stand out as single-node islands.[38]
2. Chapter 13 is depicted as a kind of gateway, between the main body of the Profane storyline and the final Chapter 16. In reality, Chapter 13 is the one where the main characters of the Profane storyline get ready for their passage to Malta.

3. Chapter 16 is connected with the main body of the Profane storyline only through Chapter 13 (the main characters are already in Malta), and it stands out naturally as a kind of terminal (or a cape!).

Looking at the terminal- or cape-like depiction of Chapter 16 in Fig. 5, we cannot help but recall the actual ending of the chapter (of the whole Profane storyline, in fact), with Benny Profane running towards the literal edge of Malta:[39]

> Later, out in the street, near the sea steps she inexplicably took his hand and began to run. The buildings in this part of Valletta, eleven years after war's end, had not been rebuilt. The street, however, was level and clear. Hand in hand with Brenda whom he'd met yesterday, Profane ran down the street. Presently, sudden and in silence, all illumination in Valletta, houselight and streetlight, was extinguished. Profane and Brenda continued to run through the abruptly absolute night, momentum alone carrying them toward the edge of Malta, and the Mediterranean beyond.[40]

As with hierarchical clustering, our results here seem also to be robust and persistent under different settings and parameterizations: Fig. 6 shows a similar graph visualization, this time with the Manhattan distance. Despite some differences, most notably the non-connection of Chapter 1 with the main body of the Profane storyline, the similarity between the two figures is striking.

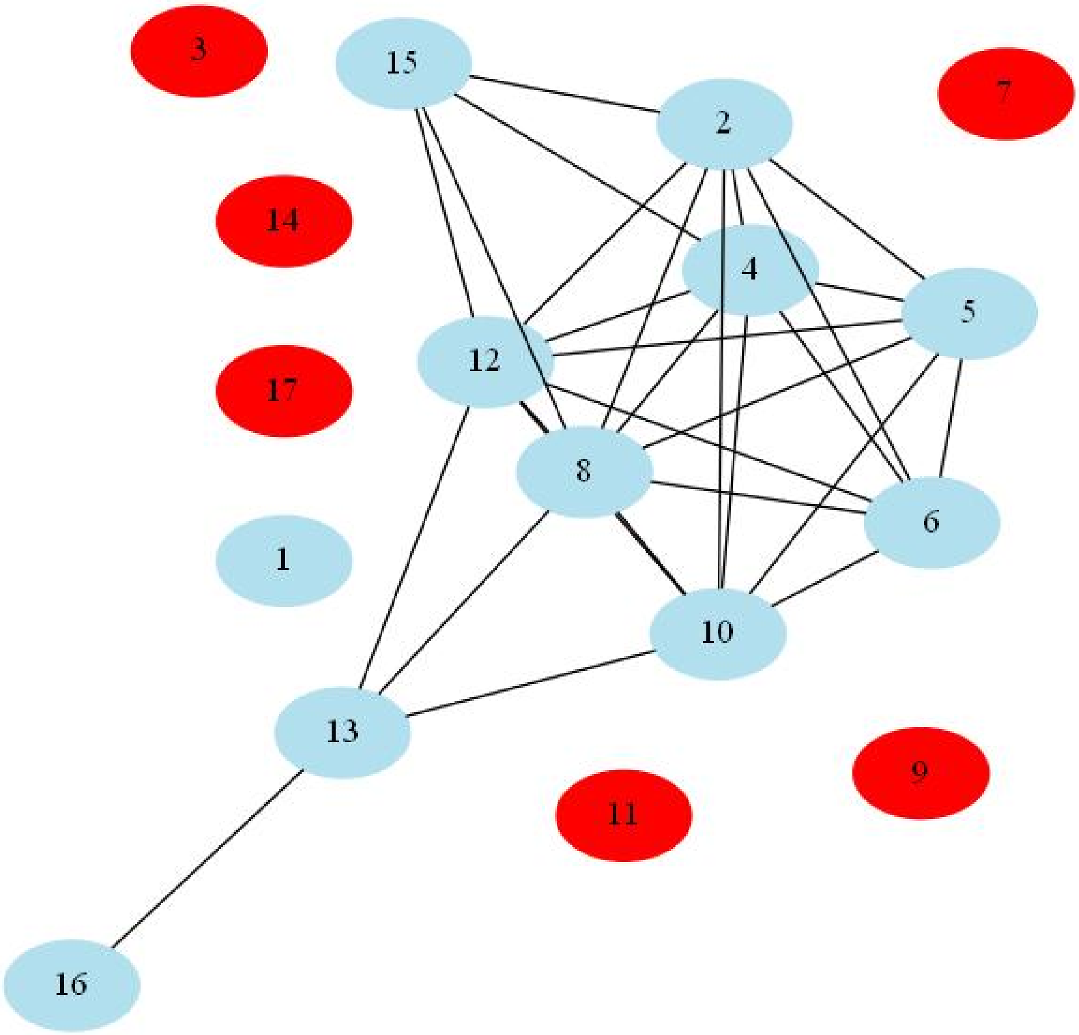


Figure 6: As Fig. 5, with the Manhattan distance

Trying to get a similar visualization with the Canberra distance, we were in for a surprise, as it can be seen in Fig. 7. After checking out for errors, and while still thinking of not including Fig. 7 here, we came up with a controversial claim, which we expose for debate:

In a (highly unlikely) question posed by a (highly unlikely) fictitious candidate reader of the novel, “*since I keep on hearing about the highly heterogeneous structure of the novel, it should be possible to read roughly half of the book and still be able to grasp the most out of it; now, which chapters should I read?*”, we claim that the connected chapters in Fig. 7 (i.e. the V. storyline chapters minus Chapter 14, framed by the first and the last of the Profane storyline chapters) constitute a possible valid answer.

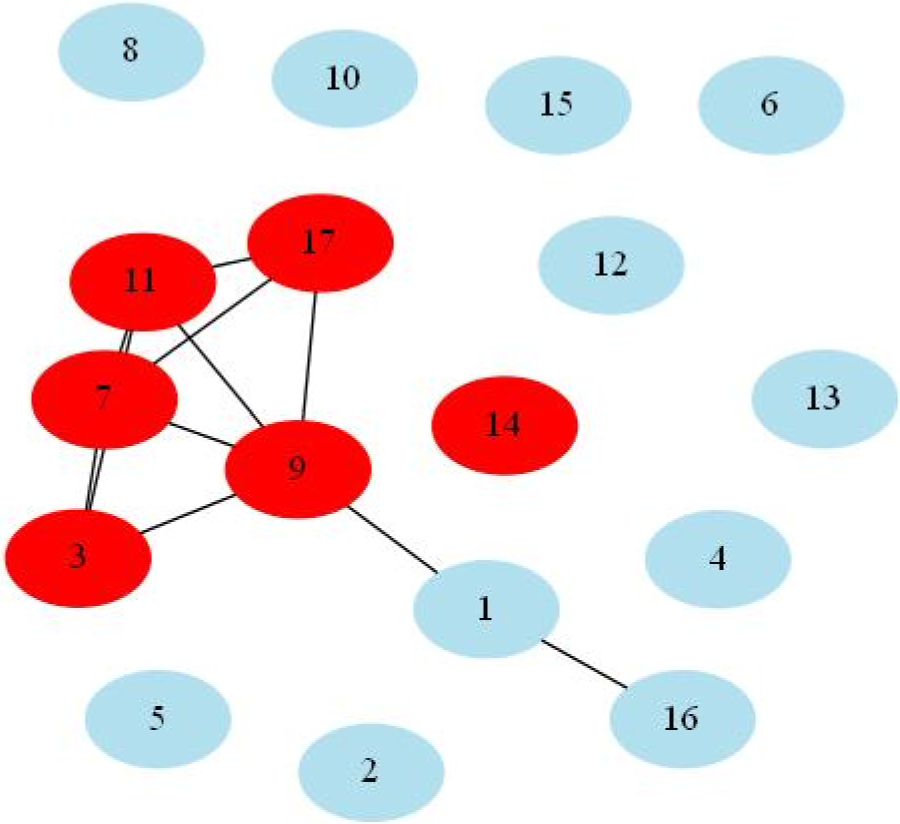


Figure 7: As Figs. 5 & 6, with the Canberra distance

6. Normalised compression distance

The normalised compression distance (NCD) is a relatively recent method, proposed by Cilibrasi and Vitányi, for computing the distance between generic data objects based on compression. The method has deep roots in information theory, particularly in the concept of Kolmogorov complexity.[41] Surprisingly enough, the method has yet to find its way into the standard text mining toolbox and it remains rather underexploited. An application to literature analysis was included already in the original NCD paper, where a perfect hierarchical clustering of five classic Russian authors (Dostoyevsky, Gogol, Turgenev, Tolstoy, and Bulgakov) is reported, based on three or four original texts per author;[42] interestingly enough, when fed with English translations of works by the same authors, the resulting clusters were biased by the respective translators.[43] In Cilibrasi and Vitányi's words,[44] "*it appears that the translator superimposes his characteristics on the texts, partially suppressing the characteristics of the original authors*", a rather well-known truth regarding literature translation, which nevertheless the method was able to independently re-discover, based on fairly simple quantitative measures.

The choice of the particular data compressor to be used is the only free parameter of the NCD method. Cebrián et al. have performed a thorough, independent performance test of the method using three different compressors, namely bzip2, gzip, and PPMZ, which in turn are example implementations of the three main types of compression algorithms, i.e. block-sorting, Lempel-Ziv, and statistical, respectively. Following their findings and recommendations, we do not use the gzip compressor here due to file size concerns; also, since PPMZ implementations are not common, we have used instead the LZMA compressor, which has an acceptable file size region[45] identical to that of the PPMZ and suitable for our data. The specific implementations employed are the `bz2` and `pylzma` Python libraries.

It should be stressed that, in stark contrast to all the other methods used in this paper, the NCD method does **not** rely on the bag-of-words assumption; also, the input text files are fed to the algorithm “as-is”, without any kind of preprocessing. That way, and by its nature, the NCD method is able to capture higher-order information included in the text, which by definition goes beyond the reach of all the other methods employed here.

Since the NCD is essentially a distance measure, it can itself be used for constructing hierarchical clusterings; indeed, this is the principal use of the method as suggested by its creators. Nevertheless, here we choose to use it in order to construct graph visualizations similar to those in Section 5 above. Figs. 8 and 9 depict such graphs, built using the LZMA and bzip2 compressors respectively. All the characteristics already met in Figs. 5 and 6 of Section 5 are again present here, and should by now look familiar: a main connected body consisting of the Profane storyline chapters; the V. storyline chapters as islands; the “gateway” function of Chapter 13; the “terminal” function of Chapter 16; and even the loose integration of Chapters 1 and 15 into the main cluster of the Profane storyline (Chapters 2, 4, 5, 6, 8, 10, and 12). We notice that the connection between Chapters 11 and 17 of the V. storyline shown in Fig. 9 is a meaningful one, as both chapters take place in Malta, with Fausto Maijstral as a central figure.

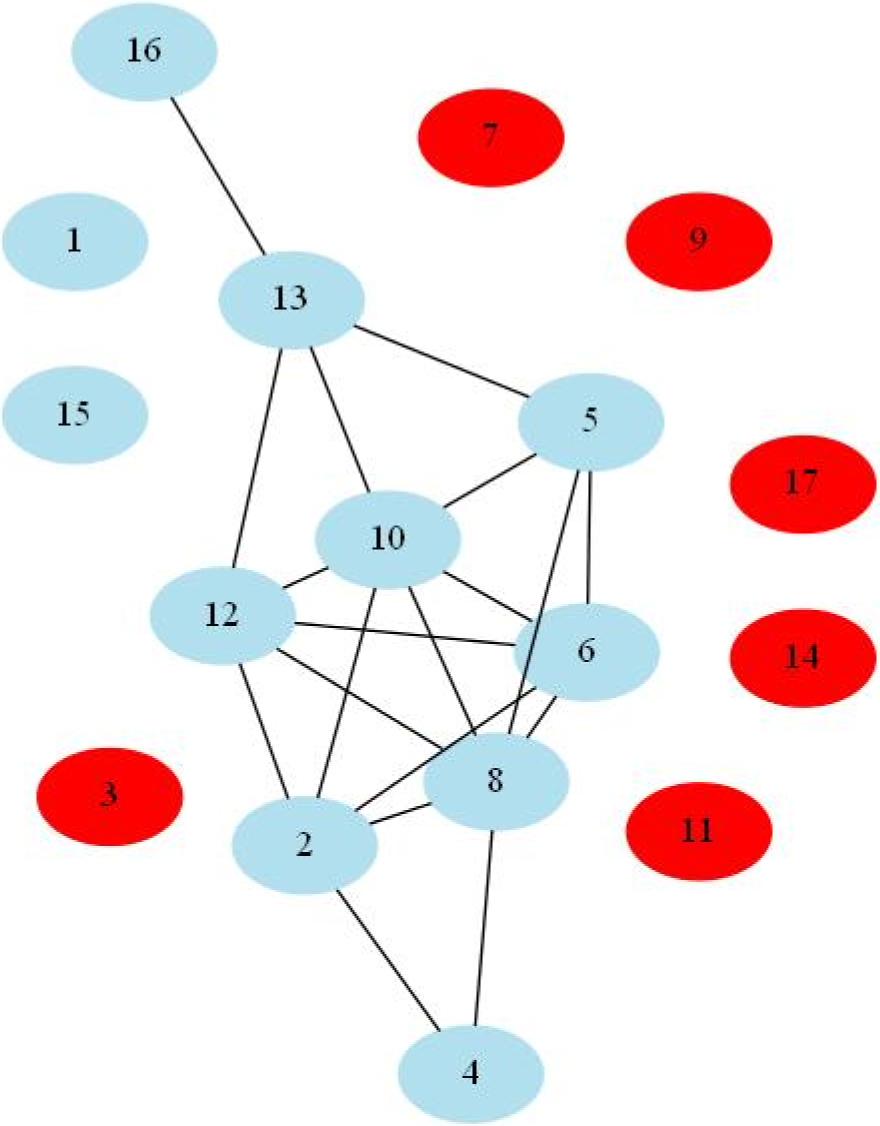


Figure 8: Graph visualization of the book chapters using the NCD distance computed with the LZMA compressor. Again, for convenience, the Profane storyline chapters are depicted in blue, and the V. storyline chapters in red.

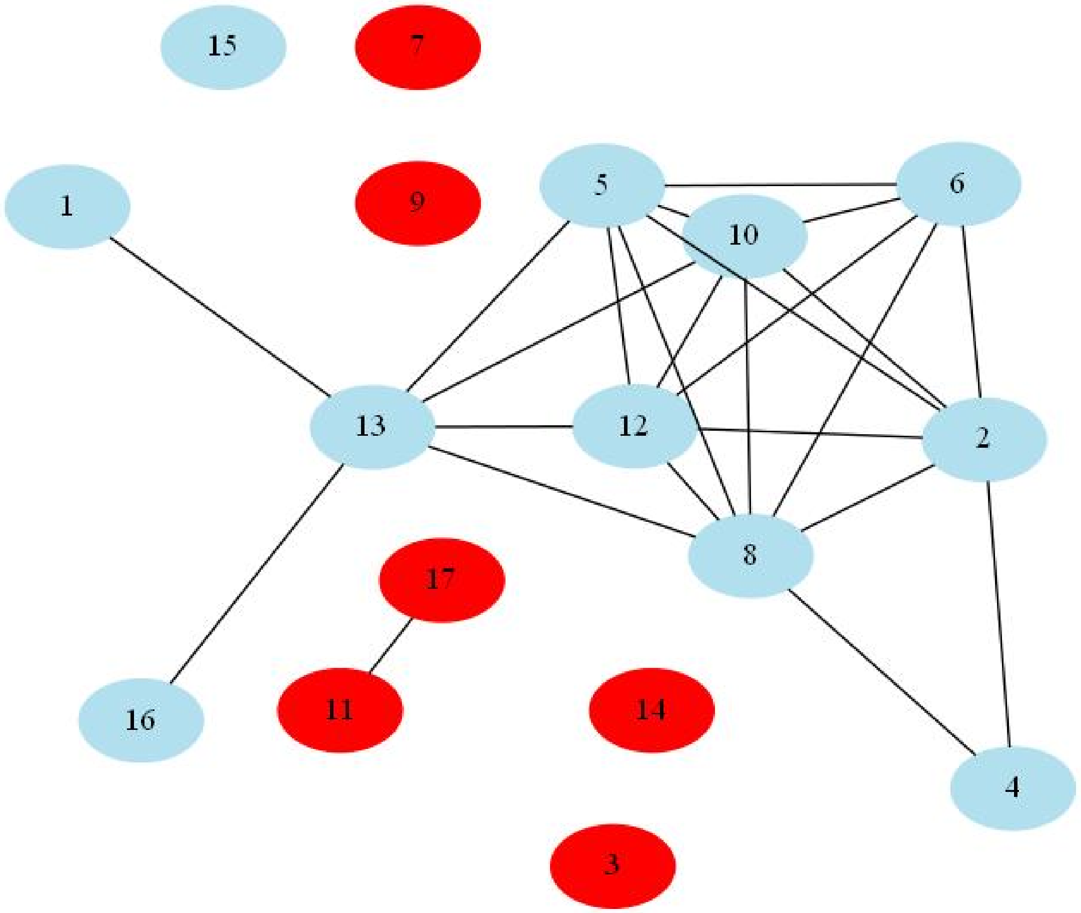


Figure 9: As Fig. 8, but with NCD computed using the bzip2 compressor. Notice the meaningful link between Chapters 11 and 17.

7. A simple topic model

Probabilistic topic modeling[46] is a family of algorithms that aims to automatically discover and extract thematic information from (usually large) corpora of text documents. Without going into the technical details here, for our purposes it suffices to say that, according to the topic modeling approach, each document in a collection consists of several topics in different proportions, whereas the topic set itself is common for the whole document collection. The method has found applications in the analysis of political texts,[47] as well as in meta-analyses of scientific papers published in academic journals, ranging from automatic tagging and labeling[48] to location and identification of specific research trends as they evolve in time.[49]

Here we will use one of the simplest and most basic approaches in topic modeling, namely the latent Dirichlet allocation (LDA),[50] as implemented in the R package `topicmodels`.[51] We will treat the whole book as our collection, with the documents being the individual chapters. Again, we aim for simplicity and clarity of demonstration rather than a rigorous treatment using more complex and sophisticated techniques. In what follows, the reader has to keep in mind that topic modeling, in contrast with the other techniques employed here, is a probabilistic method, and as such it is expected to give non-identical results for various runs of the algorithms with different initializations ("seeds") of the software random number generator involved.

In the elementary LDA approach, the number of topics one is looking for has to be predefined by the researcher. Following the suggestions of Grün and Hornik, we tried to determine the optimal number of topics by running the algorithm for a range of possible topic numbers and computing the log likelihood of each resulting model. Unfortunately this approach, when repeated for several different random seeds, gave a topics number in the range of 18-23. Clearly, the number of topics should not be greater or even equal to the number of documents (we did confirm that: running the algorithm for 18 topics gave the trivial result of assigning each chapter to one unique topic).

With the likelihood approach unsatisfactory, we tried to determine a reasonable number of topics ad hoc, based on our prior knowledge about the novel: we thought that this number should not be less than the number of the V. storyline chapters (6), and it should not be greater than roughly half the total number of chapters (8-9). That way, with some trial and error with numbers of topics between 6 and 9, we were able to come up with a fitted LDA model of seven (7) topics. Our results are shown in Table 4.

Topic #	**Chapters**
1	**9, 14**
2	**11, 17**
3	**3, 7**
4	2, 4
5	5*, 8, 10, 12, 13, 15*
6	1, 5*, 6
7	15*, 16

Table 4: Chapters assignment to topics, as produced by a 7-topic model. V. storyline chapters are denoted in bold. Asterisks denote chapters that were assigned to more than one topic with comparable probabilities.

Recall that in principle, according to the topic modeling approach: (i) a topic can be part of more than one document and (ii) a document can consist of one or more topics in some proportions. From Table 4, we can see that the topics discovered by the LDA algorithm are "pure" with regard to the two different storylines, i.e. there are no topics belonging to both. Moreover, the vast majority of our chapters are also “pure”, in the sense that they consist of a single topic, with the exception of Chapters 5 and 15, which consist of two topics each.

As already said, topic modeling is a probabilistic method, and the results shown in Table 4 are just the output of the algorithm for a specific random seed. But repeating the experiment 10 times with different random seeds, we kept on getting the same *qualitative* result, i.e. three topics assigned exclusively to the six V. storyline chapters, and four topics for the Profane chapters, with no mixing between the storylines, although the specific grouping of chapters to topics can be quite different.

For illustrative purposes, Table 5 shows the 10 most probable terms for each of the three topics of the V. storyline, as depicted in Table 4.

Topic #1	**Topic #2**	**Topic #3**
Mondaugen	Stencil	time
time	Fausto	Stencil
woman	time	girl
black	god	Victoria
eyes	Maijstral	Godolphin
night	street	father
found	children	english
girl	Malta	Vheissu
hair	priest	god
sun	night	world

Table 5: The 10 most probable terms for each of the three topics found in the V. storyline chapters, as shown in Table 4. Notice the presence of the most frequent terms, as shown in the wordcloud of Fig. 1 above. Uppercase letters have been manually restored where appropriate for the convenience of the reader.

Once again, the results prove to be rather robust and consistent, and not the outcome of some fine tuning: we performed some limited experiments with a number of 8 topics; most of the time, the results again were qualitatively similar to those in Table 5, but occasionally Chapter 16 alone of the Profane storyline would be grouped to the same topic with Chapter 17 of the V. storyline. This is rather justifiable, as both chapters take place in Malta with Fausto Maijstral as a central figure (recall from Table 3 above that Chapter 16 was again misgrouped in some of our clustering experiments).

8. Discussion and future work

It is a well-known fact among data mining practitioners[52] that unsupervised methods in general, and clustering in particular, can be like looking for patterns in the star-filled night sky: one will always be able to come up with some meaningful-looking ones, as the results of the ancient Greeks' vivid imagination still testify.[53]

Nevertheless, the convergence of the results produced by a number of different approaches provides a kind of safety against this mental trap, especially if the subject approaches are based on several non-overlapping assumptions and techniques.[54] And this is exactly what we report here: we have utilized a wide range of techniques and algorithms, both deterministic and probabilistic, including different term weighting schemes, different clustering methods and distance functions, varying parameterizations where applicable (e.g. for the Minkowski and NCD distances), ad hoc visualization techniques, with and without the bag-of-words assumption, and with several levels of text preprocessing, ranging from application of all standard preprocessing operators up to no preprocessing at all. Our results converge convincingly in revealing the heterogeneous structure of the novel at the chapter level.

It is not quite clear to us how (and if) such results can be of merit for the critic or the literary theorist. At the end of the day, we could easily imagine one arguing that we have just spent enormous amounts of human and computing power, just to reveal something that was rather known in the first place. Of course, such arguments cannot stand against any serious criticism: if we are to embark on any genuine journey towards the quantitative analysis of our literary heritage, we must first test our tools and methods, explore their range of applicability, and map their limitations; and there is hardly any better way of doing so, other that checking their outputs against already known facts, in order to gauge and calibrate their relevance and suitability. From this point of view, we consider the work exposed here as a successful demonstration.

There are several different ways and directions towards which the present study can be extended. Among the first, one could imagine dropping the bag-of-words assumption. There are already some relevant tools available: limiting the discussion to topic modeling, there have been proposed[55] extensions of the basic approach and hybrid models that can capture higher-order semantic structure and both short- and long-range dependencies between words in a document; some of these tools are also available as a free toolbox for Matlab.[56] Even the elementary LDA model is in principle readily applicable to more complex approaches, involving building blocks of n-grams or even paragraphs.[57]

An implicit characteristic of the present work is that it was implemented using just general purpose data analysis software, which, despite the functionality of some dedicated add-on packages, is perhaps still quite limited for this kind of study. We plan to undertake similar investigations in the future, utilizing freely available software that is dedicated to document analysis, such as MALLET.[58] In any case, however, the access to a considerable body of existing algorithms and the flexibility that are provided by a general purpose software tool such as R is extremely valuable and should not be underestimated.

By now, computer-assisted content analyses for literary works are not uncommon and, perhaps unsurprisingly, a good lot of them focus upon the Shakespearean corpus.[59] We choose to conclude the present study quoting Jonathan Hope, one of the pioneers in the field of digital scholarship on Shakespeare:

> We perform digital analysis on literary texts not to answer questions, but to generate questions. The questions digital analysis *can* answer are generally not ‘interesting’ in a humanist sense: but the questions digital analysis *provokes* often are. And these questions have to be answered by ‘traditional’ literary methods.[60]

Or, in the words of Stephen Ramsay:

> If text analysis is to participate in literary critical endeavor in some manner beyond fact-checking, it must endeavor to assist the critic in the unfolding of interpretive possibilities. We might say that its purpose should be to generate further “evidence,” though we do well to bracket the association that term holds in the context of less methodologically certain pursuits. The evidence we seek is not definitive, but suggestive of grander arguments and schemes.[61]

We would be very happy if the present work could serve as a trigger, in order to initiate more quantitative studies on the work of “*Thomas Pynchon, the greatest, wildest and most infuriating author of his generation*”.[62] In the meanwhile, we will delve further into the research paths proposed by Franco Moretti and Stephen Ramsay, trying to prepare ourselves against the day.-

Acknowledgments

We would like to thank Thanassis Paraskevas, for his assistance with some useful programming tips in Python; Martin Eve, for providing encouraging feedback to some early samples of the work presented herein; and Markus Hofmann, for fruitful remarks on a late draft of this work. Also, active support and encouragement by all colleagues of the 9th Training Group / 10th Training Cohort of the Supreme Joint War College is gratefully acknowledged.

End notes

1. Kirschenbaum.

2. Schreibman, Siemens, & Unsworth (eds).

3. Schreibman & Siemens (eds).

4. See, for example, Mimno’s work on computational historiography. See also Hagood and the references therein. Pointers to more references are given in Section 7.

5. The distinction between supervised and unsupervised methods is a standard one in the field of data mining. Unsupervised methods aim “to identify patterns in the data that extend our knowledge and understanding of the world that the data reflects”, without the existence of a “specific target variable that we are attempting to model” (Williams, p. 175); they are usually associated with what we call “descriptive” approaches, and they do not depend on any particular modeling input (hence “unsupervised”). In contrast, with the “predictive” approaches and the corresponding supervised methods, one tries to predict a specific target variable which has been previously defined as such in the modeling (hence “supervised”); an example of supervised methods in the quantitative analysis of text would be to try to assign a piece of text of unknown authorship to one of the authors in a predefined and limited list, once the algorithm has been previously “trained” with known texts of the subject authors (the target variable here being a “class label” attached to the text, with its author’s name). Only unsupervised methods are employed in the present study.

6. Stephen Ramsay comments on “quantitative analysis [as] chief among […] those activities that are usually seen as anathema to the essential goal of literary criticism” (Ramsay, p. 57).

7. Slade, p. 48.

8. Seed, p. 71.

9. Bloom, p. 47.

[10]. With the exception of Chapter 7, these two conditions are never in disagreement, i.e. there is no (other) chapter taking place at the novel's present without involving Benny Profane, or vice versa. Regarding Chapter 7, although it begins in the novel’s present, Profane is nowhere to be seen in it.

[11]. We will confess that, when commencing with the present study, we were erroneously *certain* that such an unambiguous mapping of chapters-to-storylines was already in place.

[12]. David Cowart, trying to construct a timeline-chronology of avatars and congeners of the *character* V., implicitly ends up with a collection of V. storyline chapters that is identical to ours, i.e. chapters 3, 7, 9, 11, 14, and the Epilogue (Cowart, pp. 41-42).

[13]. As David Seed notes, “There has been a tacit agreement among critics that the historical chapters tend to be richer and more varied than those set in 1956” (Seed, p. 72). He also comments on “the astonishing variety of tone and effects which Pynchon manages”, and “the local richness of these [historical] chapters” (Seed, p. 87).

[14]. “[In] the *bag of words model*, the exact ordering of the terms in a document is ignored but the number of occurrences of each term is material. We only retain information on the number of occurrences of each term. Thus, the document “Mary is quicker than John” is, in this view, identical to the document “John is quicker than Mary”. Nevertheless, it seems intuitive that two documents with similar bag of words representations are similar in content.” (Manning et al., p. 117, emphasis in the original).

[15]. In our (desperate) attempt not to get too technical, we choose to quote from a source that appeals to humanities readers rather than to quantitative scientists. Nevertheless, even in this case, it seems that we cannot avoid the explicit use of an equation… The framework of the following discussion (and the relevant document collection) is Virginia Woolf’s novel *The Waves*, and the assumed distinct “documents” in the collection are not the chapters (nonexistent here), but the *individual characters’ monologues*:

“Let *tf* equal the number of times a word occurs within a single document. So, for example, if the word “a” occurred 194 times in one of the monologues, the value of *tf* would be 194. A term frequency list is therefore the set of *tf* values for each term within that speaker’s vocabulary. Such lists are not without utility for certain applications […].

[If] we modulate the term frequency based on how ubiquitous the term is in the overall set of speakers, we can diminish the importance of terms that occur widely in the other speakers […] and raise the importance of terms that are peculiar to a speaker. *Tf-idf* accomplishes this using the notion of an inverse document frequency:

$$tf - idf = tf \times \left(\frac{N}{df} \right)$$

Let *N* equal the total number of documents and let *df* equal the number of documents in which the target term appears. We have six speakers. If the term occurs only in one speaker, we multiply *tf* by six over one; if it occurs in all speakers, we multiply it by six over six. Thus, a word that occurs 194 times, but in all documents, is multiplied by a factor of one (six over six). A word that occurs in one document, but nowhere else, is multiplied by a factor of six (six over one).” (Ramsay, p. 11 – the excerpt and the whole discussion can also be found online, in Chapter 26 of the *Companion to digital literary studies*, Schreibman & Siemens eds.).

For a more technical definition and discussion, see Manning et al. (pp. 117-119) or Rajaraman & Ullman, (p. 8), both freely available online.

[16]. Adapted from Manning et al., p. 119.

[17]. Technically speaking, a *vector*. See chapter 6 of Manning et al. for more technical details of this representation, which forms the basis for almost all quantitative analysis of texts.

[18]. Stephen Ramsay goes at length to argue that such text transformations, however distant they may initially seem from the scholar tradition of ‘close reading’, can be actually seen as a natural part of it: “Any reading of a text that is not a recapitulation of that text relies on a heuristic of radical transformation. The critic who endeavors to put forth a “reading,” puts forth not the text, but a new text in which the data has been paraphrased, elaborated, selected, truncated, and transduced. This basic property of critical methodology is evident not only in the act of "close reading," but in the more ambitious project of thematic exegesis. In the classroom, one encounters the professor instructing his or her students to turn to page 254, and then to page 16, and finally to page 400. They are told to consider just the male characters, or just the female ones, or to pay attention to the adjectives, the rhyme scheme, images of water, or the moment in which Nora Helmer confronts her husband. The interpreter will set a novel against the background of the Jacobite Rebellion, or a play amid the historical location of the theater. He or she will view the text through the lens of Marxism, or psychoanalysis, or existentialism, or postmodernism. In every case, what is being read is not the “original” text, but a text transformed and transduced into an alternative vision, in which, as Wittgenstein put it, we “see an aspect” that further enables discussion and debate.” (Ramsay, p. 16).

[19]. Tan et al., p. 66.

[20]. We encourage the reader to embrace and trust an intuitive approach here: documents that are “far apart” (i.e. higher distance) are thought of as more dissimilar than documents that are “close” together.

[21]. See Cha, for a comprehensive survey of about 45 different distance measures used in data mining and pattern recognition in general.

[22]. Tan et al., p. 69.

[23]. Manning et al. pp. 121-122, Tan et al., pp. 74-76.

[24]. Cha, p. 305, Fig. 2.

[25]. In text analysis jargon, “stop words” refer to extremely common words that are so frequently used that they become trivial and non-significant for the analysis. As Rajaraman & Ullman note: “Our first guess might be that the words appearing most frequently in a document are the most significant. However, that intuition is exactly opposite of the truth. The most frequent words will most surely be the common words such as “the” or “and”, which help build ideas but do not carry any significance themselves. In fact, the several hundred most common words in English (called *stop words*) are often removed from documents before any attempt to classify them.” (Rajaraman & Ullman, p. 8, emphasis in the original). See also Manning et al., p. 27.

[26]. Stemming refers to reducing different grammatical forms of word occurrences to a (hopefully) common root term. For example, under a stemming operation, words such as *organize*, *organizes*, and *organizing* would be all reduced to *organiz* [sic]. According to Manning et al., “The goal of [stemming] is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form. […] However [it] usually refers to *a crude heuristic process* that chops off the ends of the words *in the hope of* achieving this goal correctly *most of the time*, and often includes the removal of derivational affixes” (p. 32, emphases added). Manning et al. also comment on research demonstrating the poor results of stemming for most languages, including English (p. 46). As our text is an artistic one, where style does matter, we consider that we have one more reason for *not* applying word stemming in our analysis, on top of the crudeness of the approach itself and its poor results.

[27]. Jain & Dubes, p. 1, and Tan et al., p. 487.

[28]. See for example Hope & Witmore (2010) and Allison et al.

[29]. Tan et al., pp. 492 & 515 – “dendro” (“δένδρο”) being the (both ancient and modern) Greek word for “tree”.

[30]. Tan et al., p. 492.

[31]. Since the only hierarchical clustering approach we use here is the agglomerative one, we will drop the term “agglomerative” in what follows, keeping only the general term “hierarchical clustering”.

[32]. Tan et al., p. 517.

[33]. Jain & Dubes, p. 80.

[34]. Jain & Dubes, pp. 80-83, and Tan et al., p. 523.

[35]. As we hope it is clear from the discussion so far, these two choices (i.e. of a particular distance function and of a specific clustering method) are indeed independent between them; hence they can be combined in every desirable way.

[36]. See for example Hope & Witmore (2010) and Allison et al., which are rather typical cases.

[37]. See Cha, for the exact definitions of all distance functions used here.

[38]. Recall that the V. storyline chapters (“historical”) “tend to be richer and more varied than those set in 1956” (Seed, p. 72) – see also endnote 13 above.

[39]. We do not claim, of course, that the chapter ending is actually detected (let alone “proved”) in Fig. 5; we simply notice this as a rather playful, but worth-mentioning, coincidence.

[40]. Pynchon, p. 423.

[41]. See Li & Vitányi for an introductory treatment with applications.

[42]. Cilibrasi & Vitányi, p. 1540, Fig. 14.

[43]. Ibid, p. 1540, Fig. 15.

[44]. Ibid, p. 1539.

[45]. Cebrián et al., p. 382.

[46]. For a short and compact introduction, see Blei. Steyvers & Griffiths (2006) go into more details and examples, while maintaining an introductory viewpoint. Steyvers et al. provide a thorough introduction in a psychology context, including suggestions for possible links with the acquisition and application of semantic knowledge by humans.

[47]. See Grimmer. As a possible indication for the increasing significance of such quantitative methods in social sciences and the humanities, we notice that the paper by Grimmer was awarded the 2011 Warren Miller Prize for the best paper published in *Political Analysis* in 2010: http://www.oxfordjournals.org/our_journals/polana/awards_warrenmiller.html.

[48]. See Griffiths & Steyvers and Blei & Lafferty.

[49]. See Hall et al.

[50]. See Blei et al.

[51]. See Grün & Hornik.

[52]. See for example Tan et al., p. 532: “almost every clustering algorithm will find clusters in a data set, even if that data set has no natural cluster structure”. See also endnote 54 below.

[53]. Ironically enough, Pynchon himself seems to warn against such a stance; in the words of a character in the book, Dudley Eigenvalue: “In a world such as you inhabit, Mr. Stencil, any cluster of phenomena can be a conspiracy.” (Pynchon, p. 154).

[54]. In fact, Tan et al. demonstrate exactly this kind of cross-checking between the results of different clustering algorithms, as an example of good practice in evaluating the meaningfulness of the discovered clusters (pp. 532, 534).

[55]. Griffiths et al. and Wallach.

[56]. Steyvers & Griffiths (2005).

[57]. Blei et al., p. 995.

[58]. McCallum, Sutton.

[59]. See for example Hope & Witmore (2004), Hope & Witmore (2010), Allison et al.

[60]. Hope. Emphasis in the original.

[61]. Ramsay, p. 10.

[62]. Rankin.

References

Allison, Sarah, Heuser, Ryan, Jockers, Matthew, Moretti, Franco, & Witmore, Michael, "Quantitative formalism: an experiment", *Stanford Literary Lab Pamphlet* [1], 2011, http://litlab.stanford.edu/?page_id=255

Blei, David M. "Probabilistic Topic Models", *Communications of the ACM*, 55(4), 2012, http://dl.acm.org/citation.cfm?id=2133806.2133826, http://dx.doi.org/10.1145/2133806.2133826, pp. 77 - 84

Blei, David M., & Lafferty, John D. "A correlated topic model of *Science*", *The Annals of Applied Statistics*, 1(1), 2007, http://projecteuclid.org/euclid.aoas/1183143727, http://dx.doi.org/10.1214/07-AOAS114, pp. 17 - 35

Blei, David M., Ng, Andrew Y., & Jordan, Michael I. "Latent Dirichlet allocation", *Journal of Machine Learning Research*, 3, 2003, http://jmlr.org/papers/volume3/blei03a/blei03a.pdf, pp. 993 - 1022

Bloom, Harold, *Thomas Pynchon (Bloom's Major Novelists)* (New York: Chelsea House, 2003)

Cebrián, Manuel, Alfonseca, Manuel, & Ortega, Alfonso, "Common pitfalls using the normalized compression distance: what to watch out for in a compressor", *Communications in Information and Systems*, 5(4), 2005, pp. 367 - 384

Cha, Sung-Hyuk, "Comprehensive survey on distance/similarity measures between probability density functions", *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4), 2007, http://www.naun.org/multimedia/NAUN/m3as/mmmas-49.pdf, pp. 300 - 307

Cilibrasi, Rudi, & Vitányi, Paul M. B. "Clustering by compression", *IEEE Transactions on Information Theory*, 51(4), 2005, pp. 1523 - 1545

Cowart, David, *Thomas Pynchon & the Dark Passages of History* (Georgia: University of Georgia Press, 2012)

Griffiths, Thomas L., & Steyvers, Mark, "Finding scientific topics", *Proceedings of the National Academy of Sciences of the United States of America*, 101(suppl. 1), 2004, http://www.pnas.org/content/101/suppl.1/5228.full.pdf, http://dx.doi.org/10.1073/pnas.0307752101, pp. 5228 - 5235

Griffiths, Thomas L., Steyvers, Mark, Blei, David M., & Tenenbaum, Joshua B. "Integrating topics and syntax", in L.K. Saul, Y. Weiss, & L. Bottou (eds.), *Advances in Neural Information Processing Systems* (Cambridge, MA: MIT Press, 2004), http://psiexp.ss.uci.edu/research/papers/composite.pdf, pp. 537 - 543

Grimmer, Justin, "A Bayesian hierarchical topic model for political texts: measuring expressed agendas in Senate press releases", *Political Analysis*, 18(1), 2010, http://pan.oxfordjournals.org/content/18/1/1.full, http://dx.doi.org/10.1093/pan/mpp034, pp. 1 - 35

Grün, Bettina, & Hornik, Kurt, "topicmodels: an R package for fitting topic models", *Journal of Statistical Software*, 40(13), 2011, http://www.jstatsoft.org/v40/i13/, pp. 1 - 30

Hagood, Jonathan, "A brief introduction to data mining projects in the humanities", *Bulletin of the American Society for Information Science and Technology*, 38(4), 2012, http://dx.doi.org/10.1002/bult.2012.1720380406, pp. 20 - 23

Hall, David, Jurafsky, Daniel, & Manning, Christopher D. "Studying the history of ideas using topic models", *Proceedings of the Conference on Empirical*

Methods in Natural Language Processing (EMNLP), 2008, http://dl.acm.org/citation.cfm?id=1613763, pp. 363 - 371

Hope, Jonathan, "What happens in Hamlet?", *Wine Dark Sea Blog*, 17 August, 2012, http://winedarksea.org/?p=1596

Hope, Jonathan, & Witmore, Michael, "The very large textual object: a prosthetic reading of Shakespeare", *Early Modern Literary Studies*, 9(3), 2004, http://purl.oclc.org/emls/09-3/hopewhit.htm, pp. 6.1 - 6.36

Hope, Jonathan, & Witmore, Michael, "The hundredth psalm to the tune of 'Green Sleeves': digital approaches Shakespeare's language of genre", *Shakespeare Quarterly*, 61(3), 2010, http://mediacommons.futureofthebook.org/mcpress/ShakespeareQuarterly_NewMedia/hope-witmore-the-hundredth-psalm/, http://dx.doi.org/10.1353/shq.2010.0002, pp. 357 - 390

Jain, Anil K., & Dubes, Richard, *Algorithms for clustering data* (Englewood Cliffs, NJ: Prentice Hall, 1988)

Kirschenbaum, Matthew G. "What is digital humanities and what's it doing in English departments?", *Association of Departments of English Bulletin*, 150, 2010, http://dx.doi.org/10.1632/ade.150.55, pp. 55 - 61

Li, Ming, & Vitányi, Paul M. B. *An introduction to Kolmogorov complexity and its applications* (3rd ed.) (New York: Springer, 2008)

Manning, Christopher D., Raghavan, Prabhakar, & Schütze, Hinrich, *Introduction to information retrieval* (Cambridge: Cambridge UP, 2008), http://nlp.stanford.edu/IR-book/

McCallum, Andrew Kachites, *MALLET: A Machine Learning for Language Toolkit*, 2002, http://mallet.cs.umass.edu/.

Mimno, David, "Computational historiography: data mining in a century of classics journals", *ACM Journal on Computing and Cultural Heritage*, 5(1), 2012, http://dx.doi.org/10.1145/2160165.2160168, pp. 3:1 - 3:19

Moretti, Franco, *Graphs, maps, trees: abstract models of literary history* (London and New York: Verso, 2007)

Moretti, Franco, "Network theory, plot analysis", *Stanford Literary Lab Pamphlet* [2], 2011, http://litlab.stanford.edu/?page_id=255

Pynchon, Thomas, *V* (Philadelphia: Lippincott, 1963)

Rajaraman, Anand, & Ullman, Jeffrey David, *Mining of massive datasets* (Cambridge: Cambridge UP, 2011), http://infolab.stanford.edu/~ullman/mmds.html

Ramsay, Stephen, *Reading machines: toward an algorithmic criticism* (Urbana, IL: University of Illinois Press, 2011)

Rankin, Ian, "Reader Beware", *The Guardian*, 18 November, 2006, http://www.guardian.co.uk/books/2006/nov/18/fiction.ianrankin

Schreibman, Susan, Siemens, Ray, & Unsworth, John (eds.), *A companion to digital humanities* (Oxford: Blackwell, 2004), http://www.digitalhumanities.org/companion/

Schreibman, Susan, & Siemens, Ray (eds.), *A companion to digital literary studies* (Oxford: Blackwell, 2008), http://www.digitalhumanities.org/companionDLS/

Seed, David, *The fictional labyrinths of Thomas Pynchon* (Iowa City, IA: University of Iowa Press, 1988)

Slade, Joseph W. *Thomas Pynchon (Writers for the 70's)* (New York: Warner Paperback Library, 1974)

Steyvers, Mark, Griffiths, Thomas L., *Matlab Topic Modeling Toolbox*, 2005, http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm.

Steyvers, Mark, & Griffiths, Thomas L. "Probabilistic topic models", in T. Landauer, D. McNamara, S. Dennis, & S. Dennis (eds.), *Latent Semantic Analysis: A Road to Meaning* (Hillsdale, NJ: Lawrence Erlbaum, 2006)

Steyvers, Mark, Griffiths, Thomas L., & Tenenbaum, Joshua B. "Topics in semantic representation", *Psychological Review*, 114(2), 2007, http://dx.doi.org/10.1037/0033-295X.114.2.211, pp. 211 - 244

Sutton, Charles, *GRMM: GRaphical Models in MALLET*, 2006, http://mallet.cs.umass.edu/grmm/.

Tan, Pang-Ning, Steinbach, Michael, & Kumar, Vipin, *Introduction to data mining* (London: Pearson International Edition, 2006)

Wallach, Hannah M. "Topic modeling: beyond bag-of-words", *Proceedings of the 23rd International Conference on Machine Learning*, 2006, http://dx.doi.org/10.1145/1143844.1143967, pp. 977 - 984

Williams, Graham, *Data mining with Rattle and R: The art of excavating data for knowledge discovery* (New York: Springer, 2011)