[*JCH* 3.1-2 (2016) 93-118]
https://doi.org/10.1558/jch.31662

JCH (print) ISSN 2051-9672
JCH (online) ISSN 2051-9680

Mining the Past – Data-Intensive Knowledge Discovery in the Study of Historical Textual Traditions

Kristoffer L. Nielbo[1]
Aarhus University, Denmark
Email: kln@cas.au.dk

Ryan Nichols[2]
California State University, USA
Email: rnicholas@fullerton.edu

Edward Slingerland[3]
University of British Columbia, Canada
Email: edward.slingerland@ubc.ca

Abstract: Text-heavy and unstructured data constitute the primary source materials for many historical reconstructions. In history and the history of religion, text analysis has typically been conducted by systematically selecting a small sample of texts and subjecting it to highly detailed reading and mental synthesis. But two interrelated technological developments have rendered a new data-intensive paradigm – one that can usefully supplement qualitative analysis – possible in the study of historical textual traditions. First, the availability of significant computing power has made it possible to run algorithms for automated text analysis on most personal computers. Second, the rapid increase in full text digital databases relevant to the study of religion has considerably reduced costs related to data acquisition and digitization. However, a limited understanding of the scope, advantages, and limitations of data-intensive methods have created real obstacles to the implementation of this paradigm in historical research. This is unfortunate, because history offers a rich and uncharted field for data-intensive knowledge discovery, and historians already have the much sought after and necessary domain expertise. In this article we seek to remove obstacles to the data intensive paradigm by presenting its methods and models for handling text-heavy data.

Keywords: Text mining; information retrieval; historical research; methodology.

> "Increasingly, scientific breakthroughs will be powered by advanced computing capabilities that help researchers manipulate and explore massive datasets".
> (Hey *et al.,* 2009)

1. Introduction

High performance computing (HPC) and the massive increase and systemization of data available in digital databases ("big data") have revolutionized the sciences. This revolution, which has been dubbed the fourth paradigm[4] or, more neutrally, data intensive science, has impacted the humanities and arts differentially. Data intensive science has given new life to methods and research practice in a few areas like archaeology (Cooper and Green 2015). Other areas, such as linguistics and literary studies, have included data intensive methods in existing subareas (Gerhard *et al.* 2012; Jockers 2013). In some cases wholly new areas have emerged, as we are currently seeing with humanities data and digital humanities at large (Arnold and Tilton 2015; Schreibman *et al.* 2008). Yet the majority of humanities research remains untouched by the fourth paradigm: the vast majority of humanities scholars continue to perform theory building, empirical investigation and mental synthesis without the use of HPC, and digital databases are primarily relevant for handling research literature in a "one article at a time" mode.

With a focus on detailed descriptions and close reading of primary sources, history and history of religion as academic disciplines (historical research henceforth) belong to this majority. This is a shame, not only because these disciplines deny themselves the benefits of new data-intensive techniques, but because such techniques are particularly helpful if one endeavours to explore the human mind through historical evidence. Data mining – an interdisciplinary research field that combines HPC and database in search for meaningful patterns in data (Witten *et al.,* 2011) – offers a range of techniques for modelling and testing claims about history. The application of data mining to textual data (i.e. text mining) is especially relevant because historical texts reflect linguistic encoding of both explicit and implicit cognition (Slingerland and Chaduk 2011). While archaeology has an established tradition for applying data mining techniques to material culture, historical research lacks the necessary competences for applying text mining to written historical sources. Without these, historical research is not fully able to *mine* historical traditions that have relied on written media for cultural transmission. While supplying these competences is beyond the scope of a single article – or a journal issue for that matter – the current article will provide a map of text mining techniques and methodological principles that can serve as an initial guide.

In a discussion of knowledge discovery and pattern identification in databases it is necessary to introduce a somewhat artificial distinction from the outset. Data can be considered as either structured or unstructured (Witten *et al.,* 2011). This means that the data either have or do not have a machine readable model associated with them. The data model of structured data defines fixed fields of the stored data (e.g. name, location, date) and their restrictions (e.g. name in roman letters, location in latitude and longitude, date in numerals according to the Gregorian calendar). From the perspective of data mining, structured data are easy to store, query and analyse with a computer exactly because of their data model. We know this from spreadsheets where simple row-column combinations are enough to access specific instances of variables of the entire dataset. For these reasons, many historical databases use structured data (e.g. the Database of Religious History [DRH] or Puloto).

Data in natural languages, such as primary historical sources, lack such fixed, machine-readable fields, and are hence unstructured. Unstructured data demand considerable preprocessing before they can be systematically queried, quantified, and analysed with data mining techniques. Full text databases primarily contain unstructured data, for example the Chinese (ctext.org) or the Sacred Text Archive (sacred-texts.com). If we therefore want to model historical phenomena, we must familiarize ourselves with text mining – that is, the data mining techniques for modelling unstructured text heavy data. The remainder of this article concerns such techniques exclusively.

The qualitative humanist could argue that natural language is a type of data that does not follow mathematical laws. This, however, is incorrect (Banchs 2013). When, for instance, a word occurs in a text, it is more likely to occur again in relative close proximity of the first occurrence (i.e. words appear in bursts, figure 1a and 1b) (Katz 1996). For any given text dataset, a word's frequency is inversely proportional to its rank (i.e. Zipf's Law, figure 1c) (Zipf 1935), which means that the most frequent word is twice as frequent as the second most frequent word, three times as frequent as the third most frequent word, and so forth. A corollary to this is that the relationship between vocabulary size (i.e. the number of unique words or types) and the number of words (i.e. the number of tokens) in a dataset is such that the vocabulary increases as a function of the number of texts, but this increase diminishes as more texts are included in the dataset (i.e. Heaps' Law, figure 1d) (Heaps 1978). The quantitative or mixed-methods humanist could then argue that language has structural elements that are amenable to statistical analysis, and that historical research already uses variations in these rules to analyse sources and make inferences about the writer's mental states.

Text mining techniques are, however, more explicit and systematic in applications of such statistical rules than historical research and, more importantly, capable of applying them to collections of texts at a different scale of magnitude than any human researcher could ever read in a lifetime.

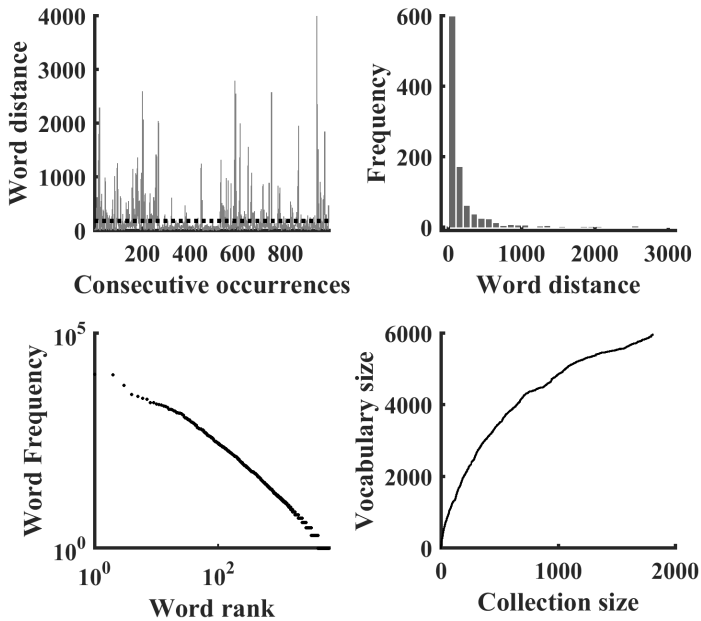

Figure 1: Fundamental properties of natural language (clockwise from upper left corner): a-b) Word burstiness. (a) A time series showing the word distance between consecutive occurrences (proximity) of "Jesus" in the KJV New Testament of the Bible (NT). The average distance separating two occurrences of "Jesus" is 184 words (dotted line); (b) The distribution of word distances shows that more than half of the distances are below 75 words or less than five sentences; (c) Zipf's Law. Word frequency rank (descending order) plotted against word frequency for NT in logarithmic space; (d) Heaps' Law. Number of unique words (types) in NT plotted as a function of total words (tokens) in the collection of NT books (i.e. NT type–token relation).

Text mining is a heterogeneous field that spans many different research areas (e.g. Natural Language Processing, Information Retrieval, Web Mining, and Machine Learning) (Miner 2012). Natural Language Processing (NLP), which combines computer science and computational linguistics in the study language for human communication, develops many of the preprocessing tools and language models applied in text mining (Jurafsky and Martin 2008). Information Retrieval applies NLP to extract structured information from unstructured data (Manning *et al.*, 2008). When

information needs to be extracted from the web, which is often the case in more contemporary research, Web Mining offers a range of techniques for accessing web activity and server logs (Liu 2011). Finally, Machine Learning (ML), which used to be a peripheral subfield of computer science that studied pattern recognition and artificial intelligence, has during the last decade become essential to any area engaged with data-intensive methods (Bishop 2008; Hastie *et al.*, 2009).

Apart from a focus on unstructured data, the various applications of text mining share a common methodology that revolves around text selection and cleaning on the one hand, and quantitative modelling and evaluation on the other (Figure 2). This common methodology underlies several industrial standards, most notable KDD (Knowledge Discovery in Databases), SEMMA (Sample-Explore-Modify-Model-Assess), and CRISP-DM (CRoss-Industry Standard Process for Data Mining) that all map the workflow of data mining (Azevedo 2008; Usama *et al.*, 1996). While the application of industrial standards might seem to be antithetical to the core task of humanistic analysis, they can serve as helpful supplements, and even correctives, to traditional qualitative analysis. Moreover, the difficulty of separating out qualitative and quantitative components of a research project becomes clear when one considers how important cultural and linguistic domain expertise is for valid selection, interpretation, and evaluation of historical and contemporary text contents.

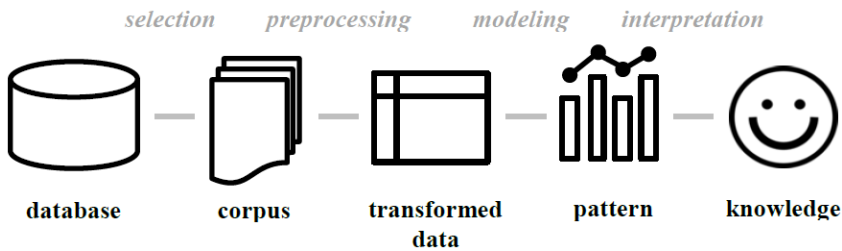

Figure 2: Common Methodology. Illustration of the different elements and steps of a text mining workflow.

2. Selection

The first step in the text mining workflow is identification of a full-text database that is relevant to the research question at hand. Research libraries and the internet offer almost unlimited primary sources for contemporary projects, but an actual database might not exist, or existing databases might lack sufficient resources. Research questions that engage historical

issues further complicate matters because the relevant texts are likely not to be digitized, and in many cases manuscripts exist in non-standard forms that render optical character recognition (OCR) of such a poor quality that considerable human effort must be invested in correcting them.

Humanities research projects therefore often start by building a database from scratch or extending existing databases. This will typically involve identification of content and in many cases also digitization of new texts. In any case, there is a trade-off between working with existing suboptimal databases and building a new database: on the one hand the Garbage In, Garbage Out principle should guide any text mining model. That is, the quality of the output depends critically on the quality of the input, but on the other hand, projects risk depleting their resources by very technical and expensive development processes. Luckily, time is our ally, because of the massive increase in the high quality full-text database we are currently witnessing.

As a side note, it is worth noting that, in recent decades, enormous amounts of effort and funding have been dedicated to building a massive database of historical texts, such as the todai database. These represent resources of historically unprecedented power and scope for the study of cultural history. Too often, however, the term "digital humanities" is used exclusively with reference to these sorts of database building efforts, without further thought being given to how texts in this new medium might be analysed in novel ways. Million-dollar textual databases built at great effort over the course of decades are frequently employed only as faster versions of resources we already had, such as concordances. Gaining familiarity with techniques such as data mining allows scholars to fully leverage the potential of these new massive textual databases by analysing them in ways that would have previously been impossible in the pre-digital age.

To return to data-mining techniques, having identified or built a database, the second step is selecting a subset of the texts in the database that can constitute a research corpus. Distinguishing between database and research corpus can be confusing because in many cases, especially when the research questions require construction of a new database, the two are identical. However, the availability of large-scale databases makes it necessary to distinguish between, on the one hand, collections of full-text data that can be used to solve a range of research problems (i.e. a database) and, on the other hand, a set of target data that is selected on the basis of specific research questions (i.e. a research corpus). Selection of a research corpus can apply both probability and nonprobability sampling (e.g. Jockers and Mimno 2013; Nichols *et al.,* forthcoming; Slingerland and Chaduk 2011). Regarding generalization, it is important to notice that, although HPC

removes any practical hindrance for modelling every text in the research relevant population of texts, we have to treat the research corpus as a sample – that is, a subset of larger set of data. This results from the fact that it is rarely possible to show that the research corpus actually covers the entire population of texts. In most cases, as is obvious with historical texts, it is highly likely that other relevant texts exist or have existed.

A central decision when constructing a research corpus is setting a document size. Continuing with the example from figure 1, if we want to model a topic space of the books of the New Testament (NT),[5] the document size is books, and the project will have 27 documents.[6] If, instead, we simply want to count the number of keywords in the NT, the document size is a collection of books and the project will have one document. Other document sizes could be sentences, verses or some external formal criteria (e.g. strings of 1000 characters or four word segments). The important point is that a research corpus has many possible document sizes, but the selected size determines the level of granularity at which the research question will be answered. A research question can of course entail several document sizes, but this implies parallel preprocessing and eventually different models.

One final thing to consider is the available metadata for the research corpus. Metadata are data about data, more exactly data that describe or summarize general features of the data. These might be date, location, authorship, or translation, to mention a few. Author demographics such as age, gender, ethnicity and religion might be particularly interesting in a reconstruction of specific historical persons (e.g. Baunvig and Nielbo 2017). Demographic features can be used in models for numerical prediction (e.g. predicting lexical variety as a function of author age) and classification (e.g. classifying gender on the basis of word frequency). In many cases, general metadata will be available from the database and, if the database uses a text encoding standard, have dedicated structured fields in the text. With more specific queries it will often be necessary to collect metadata from secondary sources.

3. Preprocessing

To initiate preprocessing it is necessary to segment the string of text in each document into the constituent parts or tokens. This process is called tokenization, and essentially consists of identifying the smallest units of analysis in the text mining project (Weiss, Indurkhya and Tong 2010). In many cases, tokens will be identical to single words, but some applications combine word-level tokens with continuous sequences of text or so-called *n*-grams (e.g. Grant and Walsh 2015). If for instance word sequences such

as "Jesus Christ" and "Son of God" are relevant to the research question, it becomes necessary to include both bigrams and trigrams in the model. More recently, projects applying character level n-grams have been quite successful in modelling relational meaning (Klein *et al.* 2003; Zhang *et al.*, 2015).

To return to our New Testament example, the books of the NT corpus comprise 27 documents and have almost 160,000 word-level tokens distributed over approximately 14,000 unique words or types. A simple word by document representation of word frequencies therefore results in a document-term matrix with dimensions 27 rows × 14,000 columns, which has almost 400,000 entries. Most of these entries do not convey any information relevant to the research question. Since words do not occur with equal probability (c.f., figure 1b) the matrix representation is sparse – that is, the majority of entries contain a zero (only 11% of the matrix contains non-zero entries), indicating that a given word (e.g. "wine") does not occur in a given text (e.g. 1 Corinthians). The goal of preprocessing is to transform a high dimensional and noisy dataset into a lower dimensional and cleaner dataset. By removing irrelevant information, standardizing and annotating linguistic forms it becomes possible to construct a numerical representation of the corpus that is tailored to the specific research project.

A widely used set of simple transformations consists of punctuation and number removal, lower case conversion, and stop word filtering (Banchs 2013). Removing punctuation and numbers as well as transforming all letters to lower case reduces the number of types in the NT corpus to 5,975, in other words, an almost 60% reduction. Stop words are very frequent words in any given language (English: "the", "is", "at", "which", "on" and so on) that do not convey any discriminatory information, because they occur equally distributed across all the documents (assuming that the documents are length normalized). There is no general agreement on content of the word list for a stop word filter, which reflects the fact that filters should always be set up in accordance with specific research questions. Many text mining packages and tools (e.g. NLTK for Python, *tm* for R, RapidMiner, and SAS) do, however, offer standardized filters for a wide range of contemporary languages. Working with historical and non-western texts can present a particular challenge, because available filters might be very limited and data insufficient to create a new filter. If, however, the research corpus is built from a larger database, one can simply use the most frequent words from the database. The stop word filter applied to the NT corpus removes an additional 93 types based on a list of 174 English stop words. This difference of 81 stop words exactly reflects the effect of applying a contemporary filter to an historical text (in this case the 17th century translation).

Complex transformations exist at any level of specificity, from completely domain-general transformations of word form to corpus specific standardizations of document length. Since documents will use various grammatical forms of the same word, it is often necessary to reduce these forms to a common base form (Bird *et al.,* 2009; Manning *et al.,* 2008). Stemming is a set of transformations that reduce a word to its stem by simply removing its ending. Applying the widely used Porter's Stemming Algorithm (Porter 1980) to the NT corpus reduces words like "pray", "prayed" and "praying" to "pray", but "prayeth" remains. Since Porter's algorithm lack a rule of the form "-eth →", the archaic third-person indicative form of pray remains. To transform archaic language and irregular forms requires morphological analysis and the use of specific vocabularies, which is where lemmatization comes in (Manning *et al.,* 2008). Lemmatization reduces various linguistic forms of a word to their common canonical form (i.e. the lemma) such that both "prayeth" and "prayest" are included in the "pray" type. A somewhat related transformation is the use of thesauri and lexical databases (e.g. WordNet) for synonym detection. In this case the goal is to replace synonymous words with a basic name form. In the NT corpus it can be relevant to replace "Christ", "Son of Man", and "Son of God" with the common denominator "Jesus".

In the majority of cases, text mining projects targeting the content of text disregard syntactic information. There can be many reasons for this, but many models rely on a bag-of-words assumption (Banchs 2013). A bag-of-words model of language essentially disregards word order. Instead, a document is treated as a bag containing all its words without sequential position, thus making word frequency central. If, however, syntactic information is relevant, there exists a range of tools and packages for conducting syntactic analysis that can, for instance, include word class in the model. Parts-Of-Speech tagging is a set of techniques for grammatical annotation that adds POS tags (e.g. NN: noun, singular; NNS: noun, plural; and VB: verb, base) to every token in a corpus (Bird *et al.,* 2009). POS tagging is often used to preprocess a corpus for a text mining technique called Named Entity Recognition, which extracts specific entities such as locations, dates, names. Importantly, POS tagging is hindered by the use of the transformations mentioned above. If both POS-tags and these transformations are necessary, the POS tagger should generally precede all other transformations.

Having preprocessed the corpus adequately, it is possible to use the transformed data to construct a reasonably simple numerical representation of the corpus. There are a range of mathematical models of language that can be used to construct this representation, the exposition of which is beyond the scope of this article. It is nonetheless important to distinguish

between two general types of mathematical models of language: statistical models and geometrical models (Banchs 2013). Common to both types of models are the use of word frequencies to represent documents. Word frequencies are often weighted to take into account the dispersion of words and the overall size of the corpus (e.g. Term Frequency-Inverse Document Frequency weighting). Statistical models use word frequencies to compute probabilities for the occurrences of and the dependencies between word-, word sequence-, and document-level units. The bag-of-words model mentioned above is actually a set of statistical models, which differ in terms of mathematical and linguistic assumptions (Banchs 2013). Both *n*-gram models and the equally popular topic models, which we will encounter in section 4.3, belong to this set of bag-of-words models.

Geometrical models, on the other hand, represent documents using a vector space (i.e. arrays of numbers, which allow for algebraic operations) and use basic geometry, such as distance and angle, to estimate document (dis-)similarity (Banchs 2013). A document in a geometrical model is represented as a document vector that describes the word frequencies of the entire corpus in the specific document. The fully preprocessed NT-corpus was represented by a 27 rows × 4,101 columns matrix, where each row is a document vector. The similarity between the books of the NT can then be compared by measuring the distance between their vectors. The Euclidean distance between the (length normalized) document vectors of Mark and Matthew is shorter (0.31) than Mark and Luke (0.34), indicating that Mark is more similar to Matthew than Luke in terms of their word content.

The final preprocessing step that is often applied to document-term matrices – that is after the numerical representation is constructed – is reduction of sparsity. Sparsity reduction is a purely mathematical operation that removes words (columns) that are sparse (i.e. dominated by zero entries). Sparsity can be problematic in the subsequent text mining for formal reasons, but conceptually this preprocessing step removes words that are only present in very few documents and therefore only describes their uniqueness. Applying sparsity reduction to the NT-corpus reduces the matrix columns to 1,537 words, which is only about 10% of the initial vocabulary! From a humanistic point of view, removing rare words on formal criteria can be problematic, because it uniformly standardizes one's corpus. Such "blind" standardization might run counter to our particular interests: for instance, we might want to show that a document is an outlier within a tradition. As with all of these techniques, sparsity reduction is optional, and should only be employed if it makes sense in terms of the research question being explored.

4. Modelling

With preprocessing in place, the transformed dataset (i.e. the numerical representation of the research corpus) can now be subjected to a text mining technique. The goal of this step on the text mining workflow is to model the data and extract a general pattern. While NLP and Information Retrieval deliver the resources for transforming unstructured data into a numerical representation, text mining at its core consists of applying a range of data mining methods (i.e. techniques for extracting patterns from quantitative data) and algorithms (i.e. formal machine readable procedures for applying methods to data).

For a humanist, one of the primary text mining challenges lies in identifying a specific technique that can answer the research question. One way of approaching this issue is by answering two questions. Firstly, what level of analysis are you interested in: words, words or *n*-gram-level relations, or documents-level relations? Secondly, how many documents does your research corpus contain: 1s, 10s, 100s or 1000s?[7] While the first question will tell which set of techniques to choose, the second points to method restrictions on the number of target documents. Machine learning for all levels of analysis techniques requires a large number of documents, but even simple correlation estimates of word-level similarity need several documents in order to be reliable.

The text mining techniques in this section follow an ascending order for required number of target documents. Techniques for word counting can be applied to only one document, while many techniques for modelling relations between words require more documents, and document-level modelling require many documents. Importantly, the levels of analysis are hierarchically embedded such that word counting can be meaningfully applied to 100s and even 1,000s of documents and word relations to 1,000s of documents.

4.1. Word Counting

Count-based evaluation methods have a long tradition in the humanities, as indicated by the amount of effort invested in pre-computer age concordances. In its most rudimentary form, these methods compute an absolute frequency (i.e. number of times a word occurs) of every word for every document in the corpus and arrange them in a searchable list.

One of several weighting schemes is typically applied to the word frequencies. Relative frequency weighting is common, which normalizes the frequency by the total number of words in the document or corpus. This weighting scheme can be used to estimate the relative importance

of a word under non-uniform document lengths. Term frequency-inverse document frequency (tf-idf) weighting is another widespread weighting scheme (Manning *et al.,* 2008). Tf-idf weighting solves the problem that absolute and relative frequencies result in all words being given equal importance.

All words, however, are not equally good at discriminating between documents. Analyses show that optimal term discrimination is obtained by words with high frequency within a document, but low overall corpus frequency (Salton and Buckley 1988). "God" for instance has low term discrimination in the NT corpus, because it occurs in every document, while "Mary" has good term discrimination because it does not occur in any of the epistles (with the exception of a bit of noise in Romans 16:6).[8] The keyword "Mary" can, in other words, be used to identify a class of books that is dominated by biographical and historical content, while "God" cannot. Weighting term frequency by the inverse document frequency removes words with high overall corpus frequency from the model.

Computing word frequencies can be used for a range of analyses or serve as an input for more advanced text mining techniques. Distribution of keywords in documents can be a useful heuristic for exploring presence/absence of central characters and concepts in sources (figure 3a). Several statistics that summarize a corpus can be calculated directly from word frequencies (Jockers 2014). One example is the Type-Token Ratio (TTR) statistic (figure 3c), which measures vocabulary variation as the number of unique words divided by the total number of words, and can be used to estimate lexical diversity (e.g. Jockers 2007). TTR, and similar statistics such as hapax legomena, are, however, sensitive to document length (i.e. they are negatively correlated), and hence length normalization should be considered.

Applying a POS tagger to the research corpus makes it possible to count classes of words instead of keywords. While syntactic word classes in some cases can be relevant to historical research, the above mentioned application of POS taggers for Named Entity Recognition (NER) seems much more relevant. NER extracts particular entities like persons and locations from documents, making it possible to identify specific groups and estimate the relative importance of these entities in and across documents (figure 3b). NER should be used with caution when mining historical texts, because tagging resources are typically developed for contemporary languages and entity identifiers can have changed over time (e.g. "Organization" in figure 3b).

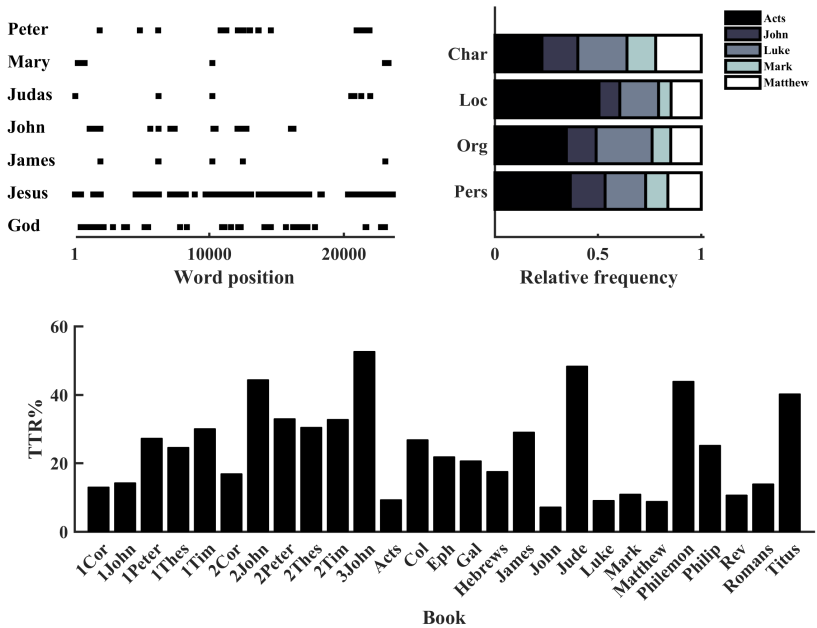

Figure 3: Word Counting (clockwise from upper left corner):(a) Word position of central characters in the Gospel of Matthew. "Jesus" and "God" are represented throughout the narrative, while "Peter", "Mary",[8] "Judas", "John" and "James" seem to have more specific functions in the plot. (b) Name Entity Recognition for entities Person (Pers), Organization (Org), and Location (Loc). Relative to document length (Char) Acts, which account the history of several persons and their travels, has more mentions of Persons and Locations than any other book in the NT. Organization here is misleading because it primarily captures Person entities. (c) Type-token ratios (TTR) for each book of the NT. Notice that the Gospels, which are among the longest documents of the NT, have the lowest TTR.

4.2. Relations between Words

Instead of counting frequencies of words and word classes, it is possible to estimate how related or, more accurately, how associated two or more words are in a research corpus (Tan *et al.*, 2005). For example, "Jesus" occurs in 26 books and "said" (past tense for "say") in 14 books out of 27 books of the NT. At mere chance level they would occur together in almost 50% of the books, while in actuality they occur together in 52% of the books. This does not mean that "Jesus" is always the subject for "said", because the association model only estimates whether or not the words occur together somewhere in each book of the NT. Document size therefore becomes extremely important for association mining, because it determines the unit of comparison. The answer to how strongly "Jesus" and "said" are associated will

look different for a verse-level model compared to the book-level model ("said" is actually more strongly associated with "Jesus" at a verse-level). There are many techniques for conducting association mining, but we will focus on examples of probabilistic and geometric approaches, respectively. Both approaches estimate association strength between words in a collection of documents, but differ in terms of mathematical concepts.

Probabilistic approaches to association mining often represent word relations as a co-occurrence matrix (or co-occurrence distribution) over a collection of documents (Banchs 2013). A co-occurrence matrix is a square matrix that has all the types in a corpus vocabulary as its rows and columns. Each entry in the co-occurrence matrix represents the number of documents in which two words co-occur, and the main diagonal represents the number of documents in which a specific word occurs. The co-occurrence matrix of the preprocessed NT corpus will have 1,537 rows and columns as shown in Figure 4.

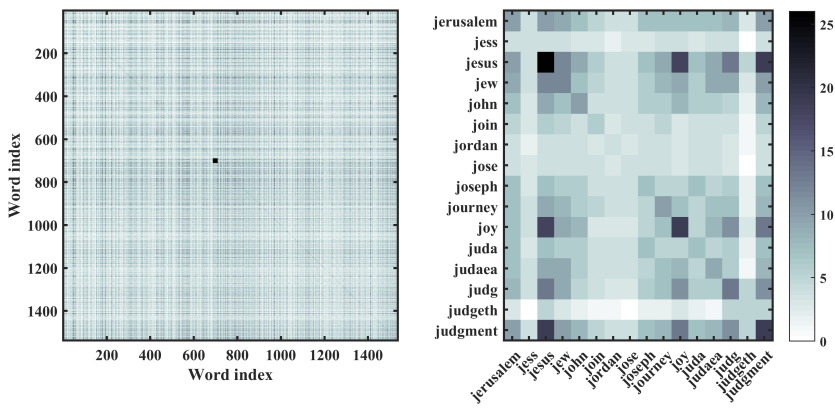

Figure 4: A co-occurrence (left to right): (a) the full co-occurrence matrix of the preprocessed NT corpus. Each entry represents the number of documents $word_i$ and $word_j$ co-occurs in (white equals zero document co-occurrence and black equals 27 documents co-occurrences). (b) a closer look at 16 word co-occurrences in the matrix (small square in 4a at indices 691 to 706). "Jesus" is present in 26 documents (dark square on main diagonal) and co-occurs in many documents with "joy" (18 documents) and "judgment" (19 documents).

The row and column number for "Jesus" is 693 and the entry for row 693 and column 693 shows that "Jesus" occurs in 26 documents ("Jesus" is absent in the Third Epistle of John). Entry combinations 693 ("Jesus") and 1,099 ("said") show that "Jesus" and "said" co-occur in 14 documents. Using the co-occurrence matrix it is possible to calculate several probabilistic

association measures such as conditional probability and mutual information (e.g. Michelbacher *et al.,* 2007). The probability of encountering "Jesus" in the NT books when encountering "said" is 1, $P(\text{Jesus}|\text{said}) = 1$. The reverse however does not hold, because "Jesus" occurs in documents where "said" does not, $P(\text{said}|\text{Jesus}) = 0.54$. Compared to other word associations, the mutual dependence between "Jesus" and "said" is not particularly strong in a book-level model. In contrast some formulaic relations (e.g. "Jesus" and "Christ") and character relations (e.g. "Jesus" and "father") always occur in the same documents and therefore have very high mutual information.

Geometric approaches apply general distance measures from geometry to estimate association strength (Banchs 2013). Two words that are located close to each other in a high dimensional space are therefore treated as similar. A distance measure is typically applied to the columns of a document-term matrix, and therefore measures the distance between vectors that represent word frequencies over the research corpus. Widely used measures are correlation and cosine distance between any two word vectors (Banchs 2013; Manning *et al.,* 2008). Applying a distance measure to all columns in a term-document matrix results in a word distance matrix. The dimensionality of the word distance matrix is similar to the co-occurrence matrix (i.e. square matrix with word types in rows and column). Each entry in the matrix contains a distance between two words, but the main diagonal is always 0 (i.e. a word has zero distance from itself). Both correlation[9] (0.14) and cosine distance (0.11) between "Jesus" and "said" in the document-term matrix of the NT books are quite short and statistically reliable ($p < .0001$) indicating that the association is not a coincidence. In contrast to the co-occurrence matrix, distance measures include word frequencies as part of association strength. Because "Jesus" often does occur as the subject of "said", their frequencies tend to covary in the documents, which is reflected in the magnitude of the association measure.

Some text mining techniques compare documents by estimating word relations as a function of a shared relation to theoretical constructs. Such constructs can vary in their level of specificity, ranging from sentiment analysis of very general subjective qualities (positive/neutral/negative) to highly detailed dictionaries of cognition, moral values, and personality (Graham *et al.,* 2009; Pang and Lee 2008; Tausczik and Pennebaker 2010). Common to these techniques is a procedure for determining how often words (or sequences of words) that are related to a given construct occur in the document, and by extension the document's construct score. Although simple in its construct (e.g. positive/negative or happy/sad), sentiment analysis that is used to estimate consumer attitudes and opinions is typically implemented with supervised machine learning algorithms (Pang and Lee

2008). Advanced rule-based dictionaries that subsume extensive word lists under multiple constructs are, on the other hand, just an extension of word counting in terms of computations (Tausczik and Pennebaker 2010).

4.3. Relations between Documents

The real benefit of data intensive methods for historical research is their capacity to handle thousands and even millions of documents. While word counts and association mining provide an elevated perspective on a collection of documents, techniques for modelling relations between documents in large-scale databases presents a true bird's-eye view (Moretti 2013). In many cases the collection of documents can be so large that it is unfeasible to read through a representative sample manually (e.g. Tangherlini and Leonard 2013). Instead, algorithms for grouping and categorizing data can be used to discover document similarities and relate these similarities to available metadata.

Currently the field of Machine Learning (ML) is the primary source of such algorithms. The sheer volume of digital data, and the velocity with which they accumulate, results in a growing need for intelligent algorithms that can learn from data. ML is experiencing a popularity explosion[10] because it is the field, or subfield, of computer science that develops algorithms for pattern recognition and statistical learning (Bishop 2008; Hastie *et al.,* 2009). The importance of ML to text mining cannot be overstated, because it develops an increasing amount of advanced techniques for text mining (Baharudin *et al.,* 2010). In this section, we will cover basic solutions to two common tasks: document clustering and document classification. Both tasks are extremely relevant to historical research, and at the same time represent two generic learning tasks in ML, namely unsupervised and supervised learning, respectively.

4.3.1. Document Clustering

The basic task of a clustering algorithm is to group collection of data into clusters (or subgroups) based on object similarity. Clustering can be used to generate clusters that are meaningful or simply useful in some way (Pan-Nang *et al.,* 2005). In document clustering one is typically interested in meaningful clusters, that is, clusters of documents that share some semantic or stylistic features that are conceptually relevant (Manning *et al.,* 2008). Document clustering can also be used to "compress" a document space without regard to content. In this case clustering is done for utility and can be used to prepare a corpus for further analysis (Andrews and Fox 2007). The task of document clustering is solved by using an unsupervised

learning algorithm – that is, an algorithm that learns the underlying grouping structure (in this case, clusters) of a dataset without the use of preexisting class values that label this structure.

To illustrate document clustering, we can apply a clustering algorithm to the books of the NT research corpus. Traditionally the books of the NT are grouped using the following classes: Gospels, Acts, Pauline Epistles, Non-Pauline Epistles and Revelation (Spivey and Smith 1994). If these class values are withheld from the learning task (i.e. unsupervised learning), it is interesting to see what groups the algorithm will find. Instead of applying an algorithm to the distance matrix of words (i.e. columns) in the documents term matrix as in the previous section, a document clustering algorithm is applied to distance matrix of the documents (columns) in the document-term matrix (Figure 5). The algorithm is therefore comparing similarity between documents (in this case distributions of words between documents).

Document clustering distinguishes between two types of clustering, flat and hierarchical clustering, respectively (Manning *et al.,* 2008). In flat clustering, the algorithm partitions the documents into a set of non-overlapping clusters, but it does not compute any inter-cluster relational information (i.e. the cluster structure is flat). In contrast, hierarchical clustering results in a group of clusters that are hierarchically embedded in a nested tree structure. Applying a flat clustering algorithm (i.e. k-means) to the NT corpus results in a model with three clusters that approximate with the traditional scheme (figure 5a). The model includes some Non-Pauline (James, 1Peter, Hebrews and James) in the Pauline group, and groups Revelation and Acts with the Gospels. This last group is interesting because the model seems to be tracking shared narrative features. A model based on a hierarchical clustering algorithm (i.e. nearest-neighbour linkage) is similar, but results in four general clusters (figure 5b). The nested structure show that although Revelation is included in the "Narrative" cluster, it is dissimilar from the remaining members (i.e. it branches away at the highest level). Additionally, the model identifies the unity of the Synoptic Gospels by nesting them closest together in the Narrative cluster. Because these results cohere with established findings in genre analysis, they may not be particularly surprising to New Testament scholars; what these results do provide is a quantitative confirmation of findings that have otherwise been achieved through qualitative analyses. In fields where the concept of genre is unproblematic these findings can serve as a benchmark for the algorithm (i.e. the algorithm can reproduce common knowledge or ground truth). In some fields, however, genre is a more controversial concept (Underwood 2016), in which case clustering can be used in a confirmatory sense.

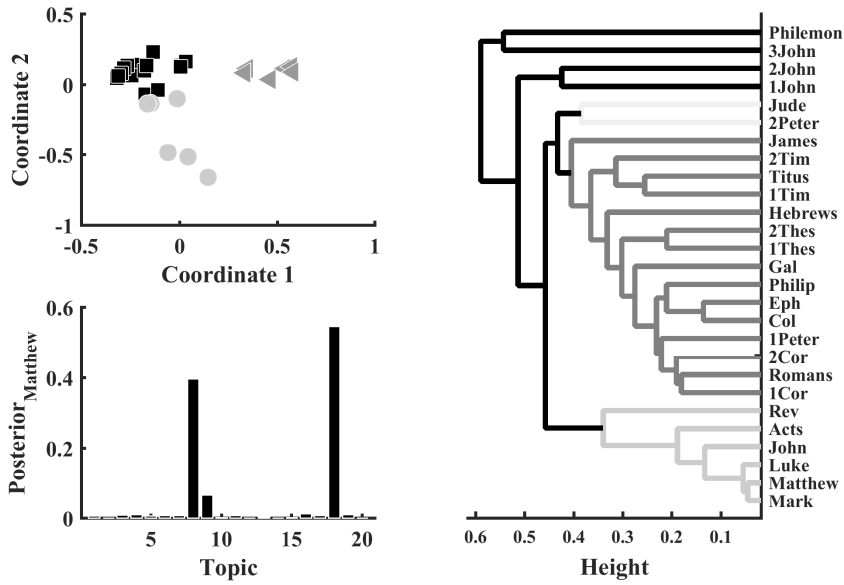

Figure 5: Document clustering (clockwise from upper left corner): (a) Three cluster in the NT identified with a flat and hard clustering algorithm. The clusters roughly align with the Gospels, Acts, Revelations (triangles), Pauline Epistles (squares), and Non-Pauline Epistles (circles). (b) A dendrogram based on a hierarchical hard clustering algorithm that identifies four clusters of books within the NT (light grey to black). The more similar two documents are the closer their branches are. (c) Topic distribution for the Gospel of Matthew in topic model of NT. Algorithms used for topic modelling (e.g. VEM) perform soft clustering.

Another useful distinction in document clustering is hard versus soft clustering (Manning *et al.*, 2008). In hard clustering, documents are assigned to one cluster only, that is, cluster membership is exclusive (Manning *et al.*, 2008). Soft clustering, on the other hand, assigns cluster membership by calculating a document's distribution over all clusters. The flat clustering algorithm applied to the NT books is hard, that is, each book is only a member of one cluster. The hierarchical cluster algorithm allows for overlapping clusters (e.g. Synaptic cluster within the Narrative cluster), but the model is not soft, because a document is not treated as a distribution over all clusters, and for model interpretation we cut the tree at some level that define exclusive cluster membership (e.g. at the level of four NT clusters).

True soft clustering brings us to topic models, which have become increasingly popular in the computationally-informed humanities. Topic models are a set of probabilistic models that performs soft clustering of documents (Blei *et al.*, 2003; Miner *et al.*, 2009). Algorithms for topic modelling are generative,

that is, they try to infer the hidden classes (i.e. topics) that have generated a collection of documents (Blei 2012). A topic model can be used to cluster both words and documents, because a topic is a cluster of co-occurring words (i.e. a distribution over words) and a document is a cluster of topics (i.e. a distribution over topics). Because topic modelling is a soft clustering technique, it captures the intuition in topic-oriented text analysis (e.g. discourse analysis) that a document is a mixture of several topics.

To illustrate topic modelling, we ran a variational expectation-maximization algorithm on the books of the NT in order to estimate 20 topics in a Latent-Dirichlet Allocation model, which is a simple type of topic model (Blei *et al.,* 2003). For the sake of simplicity, this illustration focuses on the Book of Matthew (figure 5c). Matthew is a discrete probability distribution over all topics, but most topics are only marginally present. The posterior probabilities of topics 8, 9 and 18 are, however, considerably different from zero. Common to these three topics are keywords such as "Jesus", "say/said", and "man". In contrast, topics 11 and 15 dominate Revelation, which are characterized by "God", "angel", "beast", "earth" and "heaven". Comparing the topic spaces of the two documents show that they do not cluster in terms of topics. Because the model is sensitive to word co-occurrences, it tracks Matthew's biographical content and Revelation's apocalyptic content. In contrast, both Mark and Luke overlap considerably with Matthew in topics 8 and 9. Further analysis actually revealed that topic 18 tracks features that are unique to Matthew, topic 9 tracks features that Matthew shares with Mark, and topic 8 consists of features common to Matthew, Mark, and Luke.

4.3.2. Document Classification

Techniques for classifying documents apply supervised learning algorithms to a research corpus with the purpose of building a classifier that can predict the class of new documents and test competing classification schemes (Baharudin *et al.,* 2010). While an unsupervised learning algorithm searches for documents groups within the research corpus without the use of class values (e.g. three or four clusters among the books of the NT), a supervised learning algorithm learns to map a collection of documents (e.g. the books of the NT) onto a categorical class values or labels (e.g. "Narrative", "Pauline", and "Non-Pauline"). When the classifier is trained – that is, when the algorithm has learned the mapping – it can be applied to other documents (e.g. the Gospel of Thomas) to determine their class value or, alternatively, several classifiers can be trained on different classification schemes (three classes vs. four classes) and their performance compared to estimate the best fit.

For document classification, the research corpus is divided into a *training set*, which the model uses to learn the mapping, and a *test set*, which is used to evaluate the classifier. Both sets consist of input data and class values. The input data can be any numerical representation of a document collection (e.g. a document-term matrix, a distance matrix, or even the weights of a topic model) and the class values are a set of categorical labels. During training, the classifier learns the correct document-class mapping through iterated cycles of input presentation, class output, and finally adjustment of model according to correct class value. The concept of supervised learning stems from this procedure, where an external supervisor corrects the model's output. When the classifier has learned to map document-class mapping satisfactorily, a set of test documents is used to estimate how well the classifier performs on unseen documents.

Instead of comparing the three books of NT clusters to the traditional classification scheme, it is possible to train a classifier using "Narrative", "Pauline Epistles" and "Non-Pauline Epistles" class values and inspect its performance. In this example, we use a gradient descent algorithm to train a multi-layered feed-forward neural network, which is a type of classifier that is loosely inspired by the computational properties of the human brain (Gurney 1997; Hagan *et al.,* 2002). To make the classification task a little more interesting, every book of the NT was segmented into slices of 100-words resulting in a total of 1,821 slices. The slices were preprocessed and transformed into a document-term matrix, which was subjected to 95% sparsity reduction (only words that occurred in 5% or more of the slices remained in the matrix). This resulted in a document-term matrix with dimension 1,821 rows × 221 columns. Sixty percent of the slices belonged to the Narrative class, 24% to the Pauline class, and 16% to the Non-Pauline class. The supervised task consisted of learning the correct mapping between every 100-word slice and the NT book class it belonged to. The data was divided into random samples with a training set of 85% of the slices and the remaining 15% were set aside for testing.

Results of a classifier can be reported in several ways. Here we will use a confusion matrix that compares the classifier's output class to the correct target class (Figure 6). The main diagonal of the confusion matrix shows how many slices were correctly classified. Each entry outside the main diagonal shows how many slices from a target class (x-axis) were incorrectly assigned to another output class (y-axis). In general, the classifier performed quite well with 1,640 slices out of 1,821 correctly classified, which results in an overall predictive accuracy of 90%. This should be compared to a baseline accuracy of 60% that the classifier could obtain by predicting Narrative for every slice. Another interesting detail is the Non-Pauline class, which is the

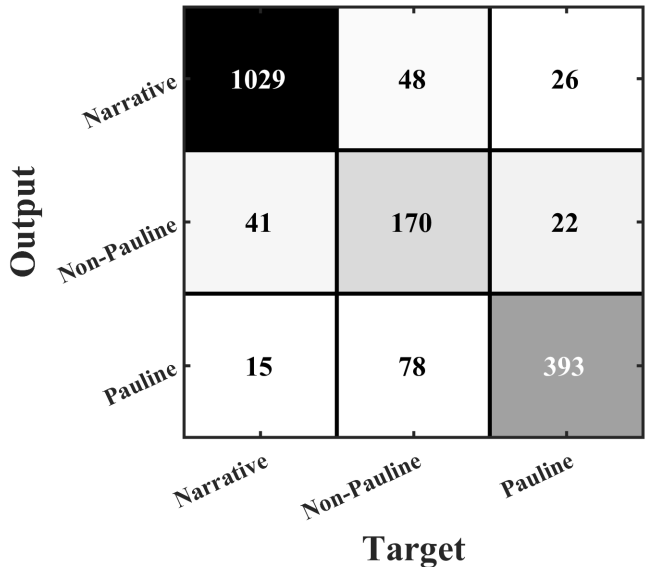

Figure 6: Confusion matrix. Estimation of the NT classifier's performance. Target classes are those true classes of the "supervisor", while output classes are the classifier's predictions. Results are collapsed across training and testing to cover the entire set of slices.

most problematic class because the classifier only has 66% accuracy for the Non-Pauline slices (195 of 296 slices). In the clustering models we found that a subset of the Non-Pauline Epistles were grouped "incorrectly" with the Pauline Epistles due to their similarity in word distributions. It is these similarities in content that are responsible for the classifier's incorrect class assignments. Finally, we segmented the Gospel of Thomas in 100 word-slices (a total of 53 slices in the Lambdin translation) and applied the classifier to the slices. The classifier correctly identified the Gospel-like character of Thomas by assigning 94% of the slices to the Narrative class.

5. Evaluation and Interpretation

The value of an extracted pattern is in itself limited, because any research corpus lends itself to numerous patterns. In order to convert a pattern into actionable knowledge (i.e. knowledge that makes a difference), it is necessary to evaluate the model behind the pattern and interpret the pattern within its context.

Model evaluation is concerned with the quality of the extracted pattern and the suitability of the model (Witten *et al.*, 2011). Results of word

counting and association mining are often used as input for numerical prediction in which case standard statistical procedures for evaluating model fit and generalizability can be applied (e.g. Baunvig and Nielbo 2017; Slingerland and Chudek 2011). As an example, we can model how often "Jesus" and "said" occurs in Gospels compared to the Epistles of the NT. In that case, the model to be evaluated is a linear model that predicts relative joined word frequency ("Jesus" and "said") on document class (Narrative, Pauline, Non-Pauline) and statistical hypothesis testing can be applied to the model. The "Jesus-said" model is statistically reliable: $F(1, 24) = 14.8$, $p < .0001$); and we can use an effect size, such as η^2 (Richardson 2011), to estimate how well the model explains the data. In this case, 55% of the variation is explained by the model. The model does, in other words, find strong support in the data, and the claim that "Jesus said" is a function of document class in the NT is therefore substantiated.

Since models of relations between documents rely heavily on machine learning, their evaluation procedures are typically also derived from that field. Machine learning offers a range of evaluation procedures, e.g. estimator score methods, internal scoring strategies, and functions for assessing performance metric (Bishop 2007). One way to evaluate a document clustering model is to look at how much variation in the document-term matrix the cluster model explains compared to treating the documents as homogenous (i.e. as belonging to one global cluster). For the hard cluster models in the previous section this evaluation measure lies around 30%, which is acceptable when you remember that each document is represented by more than 1,500 features (i.e. unique words in columns of the document-term matrix). Allowing for more cluster would increase this evaluation measure, but at the cost of model interpretability. In the extreme, a model with 27 clusters would simply result in the research corpus that the model was intended to explain.

A confusion matrix is used to evaluate a classifier's performance (Kohavi and Provost 1998). Predictive *accuracy* is one such performance measure, which in the books of the NT example was around 90% (that is, for 90% of the NT slices the classifier accurately predicted their class value). Two widely used performance measures are *precision,* which measures the number of selected documents that are relevant (i.e. how certain are we that a document is correctly classified), and *recall,* which measures the number of relevant documents that are selected (i.e. how good is the classifier at detecting documents within a given category) (Witten *et al.,* 2011). Both measures have a range from 0 (worst performance) to 1 (best performance). Evaluating the books of the NT classifier for precision and recall further support that it performs very well: precision = .88; recall = .97.

The goal of formal evaluation is to support robust interpretation of the model results. Most text mining endeavours have the ambition of discovering an entirely new piece of information, but in many cases they have to settle for supporting existing theories. This, conversely, can be thought of as an informal validation of the model. If text mining techniques can reproduce previous findings in new ways, they are likely to convince even the sceptics. Once this sort of validation step has been performed, the same mining techniques can be applied to entirely new problems, or to critique or amend current views in a given field, with enhanced confidence and authority. For instance, a recent topic modelling study of an ancient Chinese text of disputed date, the *Book of Documents* or *Shujing* 書經 (Nichols *et al.*, forthcoming), successfully reproduced the basic scholarly consensus concerning the dating of individual chapters of the text (demonstrating the reliability of the technique), but suggested areas in which the consensus might be wrong (adding to our scholarly knowledge).

As in any sort of historical research, the domain expertise of the humanities researchers is invaluable for interpreting and contextualizing the results. As text mining becomes more widespread, it is very likely to create entirely new areas for applications for humanities domain expertise. Without such expertise, the results of text mining historical and cultural data remain superficial and abstract.

6. Conclusion

In this article we have introduced different elements of the text mining workflow with the hope that more historians can see the potential for text mining for their research. Many things have been left out in order to present an accessible and coherent picture within the limits of an article. Current developments in character-level models, word embedding and deep structured learning are changing many aspects of text mining as we write. The general workflow nevertheless remains the same: text selection, preprocessing, modelling and interpretation. Because the knowledge that this workflow produces is critically dependent on humanities domain expertise, it is of utmost importance that historical and cultural researchers can be found who are willing and able to participate in text mining projects. Given the proliferation of massive, digital corpora of historical texts, various forms of text mining provide us with tools to take advantage of these entirely new scholarly resources.

That said, it is important to emphasize that text mining is not meant to replace traditional qualitative methods in the humanities. On the contrary, detailed readings of carefully selected text can and should complement

large-scale quantitative analysis. Projects that rely primarily on document-level text mining need to acquire a basic understanding of the data through qualitative assessment. Classical humanities disciplines have centuries of experience with qualitative interpretation of patterns and critical insights. Finally, humanities research constitutes a rich theoretical resource for mapping the interplay between text internal and text external factors. For humanities scholars, text mining represents a new and exciting tool in our arsenal. Moreover, the participation of humanities scholars in text mining projects has the potential to show the scientific as well as societal relevance of humanities research in a positive new light.

Endnotes

1. Kristoffer L. Nielbo is Associate Professor at Interacting Minds Centre, School of Culture and Society, Aarhus University, Denmark.
2. Ryan Nichols, is Associate Professor at the Department of Philosophy, College of Humanities and Social Sciences, California State University, Fullerton, USA.
3. Edward Slingerland is Distinguished University Scholar and Professor of Asian Studies at the University of British Columbia, Vancouver, BC. He is also Director, Cultural Evolution of Religion Research Consortium, Director, Database of Religious History, and Co-Director, Centre for the Study of Human Evolution, Cognition, and Culture.
4. For discussions of the paradigms of scientific research see (Hey *et al.*, 2009).
5. The New Testament from the King James Version of the Bible is used throughout the article: https://www.ebible.org/kjv/kjvtxt.zip. For the sake of simplicity we use this small research corpus for all examples. Importantly, this is a dummy dataset, we are not making any empirical claims about the New Testament. *The only function of the dataset is to exemplify the methods*. Ideally we would have used a common benchmark dataset, but we have not been able to find one within historical research. With certain linguistic limitations, the methods do scale to any corpus.
6. Code for all examples is available at: https://github.com/digitaltxtlab/mining_history.
7. Matthew Jockers makes a similar distinction between micro-, meso-, and macro-level of analysis (Jockers 2014).
8. The unigram "Mary" introduces noise in a plot analysis of Matthew because it designates more than one person. To correct for that it is necessary to use higher order n-grams (e.g. "Mary Magdalene").
9. Distance measures are 1 minus the correlation coefficient (or the cosine of the angle between points), so correlation distance 0.14 is equivalent to Pearson's r = .86.
10. In 2015, ML peaked in Gartner's Hype Cycle for Emerging Technologies, thereby replacing Big Data from the previous year: http://www.gartner.com/newsroom/id/3114217 (12 January 2016).

References

Andrews, Nicholas O., and Edward A. Fox. 2007. "Recent Developments in Document Clustering". 2015. Available at https://vtechworks.lib.vt.edu/handle/10919/19473

Arnold, Taylor, and Lauren Tilton. 2015. *Humanities Data in R: Exploring Networks, Geospatial Data, Images, and Text. 1st ed.*. New York: Springer. https://doi.org/10.1007/978-3-319-20702-5

Azevedo, Ana Isabel Rojão Lourenço. 2008. "KDD, SEMMA and CRISP-DM: A Parallel Overview", available at http://recipp.ipp.pt/handle/10400.22/136

Baharudin, Baharum, Lam Hong Lee and Khairullah Khan. 2010. "A Review of Machine Learning Algorithms for Text-Documents Classification". *Journal of Advances in Information Technology* 1(1): 4–20. https://doi.org/10.4304/jait.1.1.4-20

Banchs, Rafael E. 2013. *Text Mining with MATLAB*. New York: Springer. https://doi.org/10.1007/978-1-4614-4151-9

Baunvig, Katrine F., and Kristoffer L. Nielbo. 2017. "Kan man validere et selvopgør?". Proceedings from Nordiskt Nätverk för Editionsfilologer 2015. Skrifter 12: 45–67.

Bird, Steven, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python. 1st Edition*. Cambridge, MA: O'Reilly Media.

Blei, David M. 2012. "Probabilistic Topic Models". *Communications of the ACM* 55(4): 77–84. https://doi.org/10.1145/2133806.2133826

Blei, David M., Andrew Y. Ng and Michael I. Jordan. 2003. "Latent Dirichlet Allocation". *The Journal of Machine Learning Research* 3: 993–1022.

Cooper, Anwen, and Chris Green. 2015. "Embracing the Complexities of 'Big Data' in Archaeology: The Case of the English Landscape and Identities Project". *Journal of Archaeological Method and Theory* 23(1): 271–304. https://doi.org/10.1007/s10816-015-9240-4

Fayyad, Usama, Gregory Piatetsky-Shapiro and Padhraic Smyth. 1996. "From Data Mining to Knowledge Discovery in Databases". *AI Magazine* 17(3): 37.

Grant, Will J., and Erin Walsh. 2015. "Social Evidence of a Changing Climate: Google Ngram Data Points to Early Climate Change Impact on Human Society". *Weather* 70(7): 195–97. https://doi.org/10.1002/wea.2504

Hastie, Trevor, Robert Tibshirani and Jerome Friedman. 2011. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. New York: Springer.

Heaps, Harold S. 1978. *Information Retrieval, Computational and Theoretical Aspects*. Orlando, FL: Academic Press Inc.

Hey, Tony, Stewart Tansley and Kristin Tolle, eds. 2009. *The Fourth Paradigm: Data-Intensive Scientific Discovery. 1st edition*. Redmond, WA: Microsoft Research.

Jockers, Matthew L. 2013. *Macroanalysis: Digital Methods and Literary History. 1st Edition*. Urbana, IL: University of Illinois Press.

– 2014. *Text Analysis with R for Students of Literature*. New York: Springer.

Jockers, Matthew L., and David Mimno. 2013. "Significant Themes in 19th-Century Literature". *Poetics* 41(6): 750–69. https://doi.org/10.1016/j.poetic.2013.08.005

Jurafsky, Daniel, and James Martin. 2008. *Speech and Language Processing, 2nd Edition*. Upper Saddle River, NJ: Prentice Hall.

Katz, Slava M. 1996. "Distribution of Content Words and Phrases in Text and Language Modelling". *Natural Language Engineering* 2(1): 15–59. https://doi.org/10.1017/S1351324996001246

Klein, Dan, Joseph Smarr, Huy Nguyen and Christopher D. Manning. 2003. "Named Entity Recognition with Character-Level Models". In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 – Volume 4, 180–83. CONLL 2003. Stroudsburg, PA, USA: Association for Computational Linguistics. https://doi.org/10.3115/1119176.1119204

Kohavi, Ron, and Foster Provost. 1998. "Glossary of Terms". *Machine Learning* 30: 271–74. https://doi.org/10.1023/A:1017181826899

Liu, Bing. 2011. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. 2nd edn. New York: Springer. https://doi.org/10.1007/978-3-642-19460-3

Manning, Christopher, Prabhakar Raghavan and Hinrich Schütze. 2008. *Introduction to Information Retrieval. 1st edition*. New York: Cambridge University Press. https://doi.org/10.1017/CBO9780511809071

Michelbacher, Lukas, Stefan Evert and Hinrich Schütze. 2007. "Asymmetric Association Measures". Proceedings of the Recent Advances in Natural Language Processing (RANLP 2007). (15 January 2016). Available at http://www.stefan-evert.de/PUB/MichelbacherEtc2007.pdf

Miner, Gary. 2012. *Practical Text Mining and Statistical Analysis for Non-Structured Text Data Applications*. Waltham, MA: Academic Press.

Moretti, Franco. 2013. *Distant Reading*. 1st edition. London & New York: Verso.

Nichols, Ryan, Kristoffer L. Nielbo, Edward Slingerland, Uffe Bergeton, Carson Logan and Scott Kleinman. forthcoming. Modeling the Contested Relationship between Analects, Mencius, and Xunzi: Preliminary Evidence from a Machine-Learning Approach. *Journal of Asian Studies*.

Porter, M. F. 2006. "An Algorithm for Suffix Stripping". *Program: Electronic Library and Information Systems* 40(3): 211–18. https://doi.org/10.1108/00330330610681286

Richardson, John T. E. 2011. "Eta Squared and Partial Eta Squared as Measures of Effect Size in Educational Research". *Educational Research Review* 6(2): 135–47. https://doi.org/10.1016/j.edurev.2010.12.001

Schreibman, Susan, Ray Siemens and John Unsworth. 2008. "The Digital Humanities and Humanities Computing". In *A Companion to Digital Humanities*, Susan Schreibman, Ray Siemens and John Unsworth. Oxford: Blackwell.

Slingerland, Edward, and Maciej Chudek. 2011. "The Prevalence of Mind-Body Dualism in Early China". *Cognitive Science* 35(5): 997–1007. https://doi.org/10.1111/j.1551-6709.2011.01186.x

Spivey, R. A., and D. M. Smith. 1994. *Anatomy of the New Testament: A Guide to Its Structure and Meaning* (5th edition). Englewood Cliffs, NJ: Prentice Hall.

Tangherlini, Timothy R., and Peter Leonard. 2013. "Trawling in the Sea of the Great Unread: Sub-Corpus Topic Modeling and Humanities Research". *Poetics* 41(6): 725–49. https://doi.org/10.1016/j.poetic.2013.08.002

Tan, Pang-Nang, Michael Steinbach and Vipin Kumar. 2005. *Introduction to Data Mining*. 1st edition. Boston, MA: Pearson.

Tausczik, Y. R., and J. W. Pennebaker. 2010. "The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods". *Journal of Language and Social Psychology* 29(1): 24–54.

Underwood, T. 2016. The Life Cycles of Genres. Journal of Culture Analytics. Retrieved from: http://culturalanalytics.org/2016/05/the-life-cycles-of-genres/

Weikum, Gerhard, Johannes Hoffart, Ndapandula Nakashole, Marc Spaniol, Fabian M. Suchanek and Mohamed Amir Yosef. 2012. "Big Data Methods for Computational Linguistics". *IEEE Data Eng. Bull.* 35(3): 46–64.

Weiss, Sholom M., Nitin Indurkhya and Tong Zhang. 2010. *Fundamentals of Predictive Text Mining*. New York: Springer. https://doi.org/10.1007/978-1-84996-226-1

Witten, Ian H., Eibe Frank and Mark A. Hall. 2011. D*ata Mining: Practical Machine Learning Tools and Techniques, Third Edition*. Burlington, MA: Morgan Kaufmann.

Zhang, Xiang, Junbo Zhao and Yann LeCun. 2015. "Character-Level Convolutional Networks for Text Classification". In *Advances in Neural Information Processing Systems*, 649–57.

Zipf, George K. 1935. *The Psycho-Biology of Language: An Introduction to Dynamic Philology*. 1st edition. Cambridge, MA: M.I.T. Press.