Word counts and topic models: Automated text analysis methods for digital journalism research

Article · October 2015

DOI: 10.1080/21670811.2015.1093270

2 authors:

Elisabeth Günther
University of Münster
7 PUBLICATIONS **18** CITATIONS

SEE PROFILE

Thorsten Quandt
University of Münster
170 PUBLICATIONS **1,953** CITATIONS

SEE PROFILE

WORD COUNTS AND TOPIC MODELS
Automated text analysis methods for digital journalism research*

Elisabeth Günther & Thorsten Quandt

With digital journalism and social media producing huge amounts of digital content every day, journalism scholars are faced with new challenges to describe and analyze the wealth of information. Borrowing sophisticated tools and resources from computer science and computational linguistics, journalism scholars have started to gain insights into the constant information flow and made big data a regular feature of the scientific debate. Both deductive (manual and semi-automated) and inductive (fully automated) text analysis methods are part of this new toolset. In order to make the automated research process more tangible and provide an insight into the options available, we provide a roadmap of common (semi-)automated options for text analysis. We describe the assumptions and workflows of rule-based approaches, dictionaries, supervised machine learning, document clustering, and topic models. We show that automated methods have different strengths that provide different opportunities, enriching-but not replacing-the range of manual content analysis methods.

The Need for Automated Text Analysis Methods

Many changes have altered the face of journalism research in the past decades: as technological developments paved the way for digital journalism and social media, huge amounts of digital content are produced every day and can easily be accessed from locations all over the world. In our information society, "big data" offered the promise of a bright future for journalism research, with vastly improved possibilities for content analysis.

But does potentially easy access equal easy analysis? And does so-called "big data" lead to more insights into agenda-setting processes, news flows, and the portrayal of specific topics in the media? After the first excitement, there came disillusionment-the enormous amount of heterogeneous information calls for radically new approaches, new methods- and potentially new questions as well. To be able to describe and analyze this wealth of information, journalism researchers feel the need to adapt to the rapid advancements and turn to research methods that originated in unfamiliar disciplines such as computer science and computational linguistics. In this interdisciplinary research area, there is a variety of options available that can assist scholars to analyze hundreds of thousands

*The Version of Record of this manuscript has been published in 2016 and is available in *Digital Journalism, 4*(1), pp. 75-88. http://www.tandfonline.com/10.1080/21670811.2015.1093270

of documents efficiently. This article provides a practical guide, largely based on the authors' own experiences-and indeed, sometimes failures-of exploring and applying these new techniques.

Typically, we can divide text analysis methods into two groups: deductive methods that are based on a pre-defined codebook with a set of relevant categories, and inductive methods that share an explorative character aiming to identify certain attributes of the text content. Automated methods enrich our methods repertoire on both sides: previously established categories can be translated in a way that enables the computer to do (part of) the coding process for the researcher. Fully automated approaches, on the other hand, assist researchers to explore text collections that are beyond the capacity of a manual analysis. In the following, we go through the research process step by step, presenting common deductive and inductive automated text analysis methods. All approaches belong to the research area of natural language processing, which aims at the machine-based information extraction from human language. This information can be at various levels of granularity: methods target singular words or groups of words, relations between them or aim to reveal latent structures that connect words and documents invisibly. Given the challenge that document collections are often too big to read, these automated methods are a necessary extension to the traditional methods repertoire of journalism research and a logical first step-not only for fully automated analyses, but also for primarily manual analyses of textual content. Even in cases where the codebook is already set, inductive tools can be helpful to get to know the text corpus (i.e. the document collection) in case researchers have little prior knowledge about its content. In order to reduce manual workload or increase reliability, they also offer ways to subset the data by identifying relevant documents for a following (manual) in-depth analysis.

While automated methods minimize the costly manual coding phase or might even make it redundant, the research process also requires certain additional steps. Preprocessing, for example, is an essential foundation for the application of most natural language-processing algorithms and includes data cleaning and transformation-as computers process natural language in a fundamentally different way than human coders do, the computer science mantra "garbage in, garbage out" is key to avoid misleading results. In order to make the research process in this innovative area more tangible, we start with a description of common preprocessing tasks.

We proceed with a description of deductive text analysis methods that require the researcher to manually define the categories prior to the analysis (see Figure 1 for a roadmap). [1] Rule-based approaches are a toolset mostly used to extract text. With the help of logical operators and regular expressions (Friedl 2006), researchers can exploit recurring patterns within a text collection to retrieve relevant information, such as the name of the author or the title of a text. [2] If researchers can translate the respective categories directly into lists of elements that they want to identify within the text, hand-crafting a dictionary might be the appropriate solution. Sentiment analyses are a well-known example of this approach, evaluating the occurrence of positive and negative terms in a document based on a specific list (= dictionary). [3] Supervised machine learning also works with a built-in dictionary, but takes the demanding task of constructing it manually off the researchers' hands. Human coders only have to process a sample of the documents,

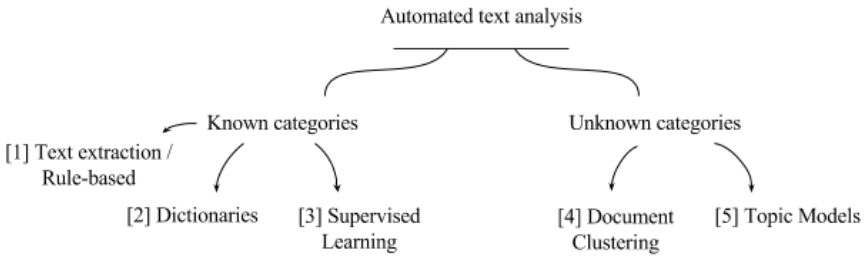

Figure 1: Overview of common text analysis methods

providing the computer with classification examples to derive machine-readable rules. By then automatically applying these rules to the additional documents, the workload of human coders can be reduced considerably.

Following these options that work in a deductive manner, we describe two fully automated approaches. These methods are especially useful when a researcher has little knowledge on the contents of a document collection. When crafting a codebook for a manual content analysis, setting up a dictionary, or specifying rules to extract text features are not feasible, fully automated approaches are a valuable aid to get to know more about the documents' contents. [4] Document clustering is one of the options to detect the topic structure of a document collection: basically also built on word frequencies, this method allocates documents into thematic groups by means of a cluster analysis. As there are no pre-defined categories, the researcher has to interpret the clusters to find the link between the grouped documents and make meaning of the results. [5] Topic models such as the Latent Dirichlet Allocation (LDA) also aim to discover links between the documents, but work with a statistically more sophisticated model. Based on a multi-level probability model, this approach describes each document as a mix of several latent topics and provides a thematic representation of the text collection.

Preparing the Data

Before turning to the actual analysis, researchers have to take a close look at the condition of their dataset. Just like in any traditional text analysis, the first step in the automated content analysis is to prepare the input material. Given that we might look at the analysis of hundreds of thousands of documents, this cannot be done by hand. In the following, we describe common techniques that can be applied to automatically clean and normalize the dataset before the actual analysis. Which steps are necessary depends on the condition the text is currently in (e.g. language and textual form) and the analysis strategy researchers plan to apply.

As a basic principle, natural language is not easy to analyze automatically: for a human coder, it is obvious that "write", "writes", and "written" refer to the same verb and activity, but a computer does not possess this knowledge by default. Words can

be inflected, in plural, contain special characters, be synonyms, or be ambiguous. Even deciding which part of the text is relevant for the analysis, and which is advertising, is a challenge. To ensure a high quality for the results of the automated text analysis methods, preparing the set of documents for the analysis is essential. It is not unusual that the preparation for the analysis demands most of the effort invested in the whole process (Witten, Eibe, and Hall 2011).

Removing Boilerplate

The first step in the preprocessing stage is to remove parts of the text that are irrelevant for its content. Online sources offer a wealth of information that is of interest to journalism scholars-very often, however, we get more than we wanted. When a Web crawler is used to retrieve the data, it does not distinguish between "good" and "bad" content, but by default simply downloads the complete Web page. This is due to the fact that, in contrast to human coders, a computer is not naturally equipped with the necessary reading literacy to identify relevant text parts (LaBerge and Samuels 1974). As a result, archives of online contents are often messy and contaminated with HTML markup, advertorial content, navigation elements, or user comments. This so-called boilerplate content is at best useless for the analysis, but increases computational costs for the following analyses and might even distort the results. Researchers can rely on several algorithms that remove the undesired data automatically (for an overview of state-of-the-art techniques, see Kohlschütter, Fankhauser, and Nejdl 2010).

Normalizing Text

The next step in the preprocessing is to normalize the data: when combining documents from different sources, different styles and formats have to be adjusted. News websites, for example, vary vastly in the way they manage their data and present their content. A major obstacle can arise from the fact that organizations choose different encodings to store text data on their local servers. If this is not taken into account when downloading online content, the text will be distorted with meaningless combinations of special characters. Websites with dynamic features can also pose a problem, as they reveal certain parts of the content only upon interaction with a user; due to the fact that, for example, a Web crawler does not interact with the website in the same way, this can create missing values. If metadata such as the date, title, or author of an article are of interest for the following analysis, this information can already be extracted when retrieving the documents; by accessing news articles via a website's RSS feed (Greer and Yan 2011), metadata can be accessed in a standardized format. Nonetheless, unexpected problems might arise, such as different time zones within the publication dates and times, and require careful inspection.

Normalization is also needed at the level of individual words. At this point, it is important to introduce the concept of terms: some words convey meaning only in combination with another ("United States") or share the same meaning with another ("U.S.A." and "USA"). Terms are the result of the normalization that is needed to preserve these units of meaning. Common steps in this process include the normalization of accents and diacritics ("Sao Paulo" and "São Paulo"), capitalization and case-folding (setting all words to

lower case; exceptions such as “Turkey” and “turkey” might apply), British and American spelling, and foreign names (“Cologne” and ”Köln”). Usually, numbers and punctuation are also removed at this step. Depending on the language of the text and the aim of the research project, additional processing steps might be necessary (Manning, Raghavan, and Schütze 2009, 26-30).

Removing Stopwords

In most text analysis methods, the importance of words in a given text differs greatly-not all are equally relevant for the analysis. To the contrary, removing non-selective function words (“do”, “with”, “until”) from the text is even a way to improve functionality and efficiency of many approaches. So-called stopwords are generally the most common words across the document collection and can therefore be filtered based on their frequency. Ready-made stopword lists are also available for many languages.

Stemming

Another common preprocessing step is to remove the inflectional endings from words, so that “laugh” and “laughs” are trimmed to the same word stem. This can be an appropriate way to remove information that is only necessary within a given sentence or document. Stemming efficiently reduces complexity in a large dataset, but comes at a price in individual cases: as stemmers are based on language-specific rules, they do not necessarily normalize synonyms (“woods” and “forest”) or homonyms (groups of words that share the same spelling but differ in meaning, such as “bank” as a financial institution and land alongside a river) correctly. If a stemmer acts too aggressively, it can furthermore be difficult to recognize the original meaning of overly truncated words.

Getting to Know the Data

Automated methods are especially useful when working with large datasets. While it is mostly not an option to look at every text in detail before the analysis, getting a basic understanding of the content of the collection is key to appropriately applying a text analysis method and to being able to interpret its outcome. To this means, simple frequency-based descriptions and visualizations are very effective. While some of the approaches presented in this section can also work as independent categories (e.g. readability parameters), they are included here as parts of a toolset to explore the context of a text collection when reading the documents is not an option. It is important to keep in mind that patterns found in a first exploratory step do not necessarily hold true in the following analyses; it is therefore recommended to carefully cross-validate before drawing conclusions. One goal of this step is to check for missing values or cases that have not yet been normalized. Especially when working with different sources, it can be difficult if not impossible to include all necessary preprocessing steps at the first attempt. Graphical visualizations such as a histogram for documents’ text lengths are great to identify outliers and preprocessing mistakes.

Text statistics

Text statistics are key parameters to display the characteristics of a document collection. These shallow text features are easily calculated, but, like a document's fingerprint, can provide background information and even reveal document authorship (Abbasi and Chen 2005). Lexical features such as word, sentence, and text length, for example, give indications on style, genre, and type of text, but can also show irregularities and problems within the dataset. Furthermore, they can be combined to form an indicator of text readability (DuBay 2004). As it is difficult for a computer to read and understand a text as it was intended by the author, complexity can be reduced by breaking it down to a list of words: the most frequent words, for example, can help to get an impression of the document's vocabulary. There are also several options to study keywords in context: instead of calculating the frequency of single words, the most frequent n-grams can show which two (bi-grams), three (tri-grams), or four words (four-grams) are most often jointly mentioned (so-called collocations). A concordance analysis is conceptually very similar, but requires a specific term as a starting point. For this term, surrounding text passages in all the documents are extracted to display the context in which it is mentioned. Co-occurrence analyses, on the other hand, do not require words to be mentioned in sequence; they are based on a co-occurrence matrix that denotes the number of times two words are mentioned in the same paragraph or document.

Visualizations

Next to text statistics, visualizations are a key to showcase patterns: basic text parameters can be plotted in order to get an overview of their dispersion across the document collection or over time. If metadata such as the date of the documents is available, the words with the highest frequency might be transferred into a word trend diagram to investigate changes in language over time. Even simple visualizations such as word clouds, a popular representation of word frequencies, can be helpful at this step.

Corpus Comparison

In order to explore a text collection, it is also worthwhile comparing the corpus against another one. The distinct features of the documents become apparent most easily when contrasted against those from another collection. While researchers are usually interested in the differences between the documents within a collection, this step aims to find common characteristics and derive knowledge about the data collection as a whole. Kilgarriff (2012) describes a simple yet efficient way to get to know a document collection by comparing a list of its top 100 keywords. Possible applications include comparing texts from different genres (commentary versus interview), types of text (tweets versus Facebook posts), sources (different countries or media outlets), or time periods (news articles before and after an election).

Automating the Analysis of Pre-defined Categories

When all necessary preprocessing steps have been conducted, researchers can finally turn to the actual analysis. In this section, we describe text analysis methods that are based on a pre-defined codebook; the main challenge therefore consists in the translation of the known categories into machine-readable tasks. While there are also other useful methods, we focus on three of the most common ones: rule-based methods, dictionaries, and supervised machine learning. Which one is best depends on the characteristics of the category and the task to be completed.

Rule-based Approach

Realizing a classification task with a rule-based approach is a very versatile option and commonly used for text extraction. As rules exploit recurring patterns, they are suitable for many variables that share a repetitive characteristic. To establish a rule set, researchers have to identify these patterns and translate them into machine-readable tasks. This can be thought of as creating a map for the computer that details starting point, junctions, distances, and destination-providing turn-by-turn instructions on how to navigate to the points of interest within a text. Both formal (retrieving the sixth word in the third line of every document) and textual properties (retrieving the words before every instance of "Obama") of a document can be helpful in this process and help to translate diverse interests of journalism research. Sjøvaag and Stavelin (2012), for example, use a rule-based approach in order to select and count metadata (publication date stamps, geographical location, news agency, and author) and Web-specific attributes (audio/video media players, flash games, polls, questionnaires, hyperlinks, and commentary sections) of online news. For each attribute, the authors set up a specific rule. Tailored to the given dataset and variables, rules are often domain- or even project-specific. For this reason, drafting them can be a trial-and-error process and might feel hacky, yet constitutes a valid option. There are also standards that can be applied without (much) adjustment. One of the classics in this section is the extraction of hyperlinks based on the source code of a website (which includes both the article text and the boilerplate content we discussed above): here, the HTML markup can be thought of as road signs within the text, guiding the Web browser to correctly display the content of a Web page. Every time there is a hyperlink, this is announced by the sign "href=", so that the Web browser can take the right turn to make the text clickable. Researchers who are interested in the hyperlinks within news articles can exploit this information by tracking all text parts that are preceded by the respective sign. Another popular application is the recognition of named entities, i.e. the names of persons, organizations, and geographic locations, within texts. As rules are, in this case, based on the sentence structure within a given text, they are language-specific and not universally applicable.

To allow for more complicated variables that include multiple conditions, the retrieved information can also be used as an input for decision trees: given an arbitrary category that aims to identify documents that include both a hyperlink to an official US government website and the name "Obama", a decision tree with four possible outcomes can be constructed. The classification is then executed based on a combination of the

above-mentioned text extraction tasks and the specified decision tree, and expressed as a dichotomous variable (for more information on decision trees, see Manning and Schütze 1999, 578).

In practice, the key to specify rules are regular expressions: as a part of many everyday applications (e.g. search engines, database queries) and programming languages (e.g. Python, Java), regular expressions are the basis to efficiently specify relevant text parts with a general pattern notation (Friedl 2006). Due to the universal nature of regular expressions, this approach is suitable for many categories that require text extraction of some sort. Depending on the level of standardization within the documents and the specifics of the task, applicability and preparation time can vary immensely.

Dictionaries

While rule-based approaches create a machine-readable map to the desired information within a text collection, dictionaries explicitly state which keywords to look for. For the automated classification, categories are described by lists of indicators that are then searched within the documents. When a text contains the terms "Merkel", "Obama", and "Jinping", for example, it is reasonable to assume that it is about politics. Sentiment analyses are a prominent example for this case (Pang and Lee 2008): based on a dictionary, the sentiment within a given document is evaluated by calculating the ratio of terms that are listed to indicate a positive versus a negative tone within the text.

Ready-made lists are available in some cases, such as for sentiment analyses. It is important to consider, however, that lists are often domain-specific and might not be applicable to the document collection at hand: Loughran and McDonald (2011) show that the prominent Harvard sentiment dictionary yields incorrect results when applied to another domain, such as financial texts. Almost three-quarters of the terms automatically classified as negative do not have this connotation in the financial context, but refer to company operations ("tax", "loss", "expense") or industry segments ("mine", "cancer", "tire"). For this reason, researchers have to be careful with external dictionaries or need extensive domain-specific knowledge to either adapt the dictionary or set up a new one by hand. The time and effort it takes to craft a reliable dictionary must not be underestimated. When the task is suitable, however, it is a computationally simple and conceptually straightforward solution to automate both formal and content-related categories.

Supervised Machine Learning

Supervised machine learning is a way to teach computers how to build a dictionary themselves. In this approach, documents are assigned to pre-defined categories based on labeled training data. In contrast to rule-based approaches and dictionaries, researchers do not have to specify the classification task themselves, but "teach" the computer by manually assigning some of the documents to the categories. The idea behind this is simple: in a first step, human coders conduct a manual content analysis by categorizing a random sample of the document collection based on a pre-defined codebook. In a second step, a learning classifier uses the category assignments (classes) in the training data to "learn" about the classification by deriving rules about the relationship between text features and

categories. Finally, these rules are applied to label the remaining test set.

Of course, the underlying model is more complex and depends on the classifier that is used. Intensive research on classifier effectiveness has produced a large body of literature and a fair number of classifiers to choose from, such as Naïve Bayes, Support Vector Machines, Neural Networks, and Random Forests-each with different qualities and assumptions about the dataset (see Sebastiani 2002). They can also be combined in an ensemble learning approach (Dietterich 2000) to improve the classification outcome.

In recent years, supervised machine learning techniques have been successfully applied to several domains of journalism research. In order to study issue attention and legislative trends, Hillard, Purpura, and Wilkerson (2008) conducted a supervised topic analysis of bill titles introduced in the US Congress, concluding that the combination of several algorithms increases both analysis accuracy and researchers' confidence. Also using an ensemble learning approach, Burscher, Vliegenthart, and de Vreese (2015) conducted several experiments to evaluate the automated coding of frames in Dutch newspaper articles. With high levels of observed classification performance, the authors recommend supervised learning as a promising analysis strategy for framing research. In another empirical evaluation, Scharkow (2013) applied a learning classifier to a content analysis of news values in German news articles. The author concludes that supervised classification is a viable option for the social sciences, but does not work well for all categories.

A main advantage of supervised machine learning is that researchers only have to provide explicit examples, but do not have to specify the rules themselves. This is typically done by conducting a manual content analysis with a part of the original dataset, but training data can also come from an external dataset that has already been annotated by a third party: Flaounas et al. (2013) base their analysis of 2.5 million news articles on the Reuters and New York Times corpora, two well-known datasets in natural language processing that come, among others, with manual annotations for a topic category.

The success of this approach depends on the quality and quantity of the training set. First, the reliability of the manual content analysis has to be guaranteed to make sure the learning classifier is not applied to a flawed training set. Second, it is important to pass a sufficient number of examples to the learning classifier for every category. This number might vary, however, depending on the number of categories and specifics of the dataset.

Categorizing Automatically

In the previous section, we presented three approaches where categories are known before the analysis, but require some sort of translation into machine-readable tasks to be suitable for an automated analysis. When categories are unknown and there is little knowledge about the document collection, fully automated methods are a way to systematically explore its content or assist in defining the categories.

Document Clustering

While the methods outlined above allow researchers to automate the classification of a broad range of possible categories, the purpose of document clustering is more narrow:

based on a cluster analysis, this approach is designed to reveal groups of documents with similar topics. Grimmer and King (2011), for example, perform a document clustering on US Senator Frank Lautenbach's press releases and find that they fulfill four basic functions: credit claiming, position taking, advertising, and partisan taunting. The authors apply the same method to George W. Bush's 2002 State of the Union Address to thematically group the then President's statements. As a bag-of-words approach, a term's syntactic properties and position within the article are not considered; the procedure is solely based on similarities between the documents' vocabularies, leaving out word order and syntax. Topics are then, in their most basic sense, simply defined as groups of documents with a similar term structure (Wartena and Brussee 2008).

The foundation for this procedure is the Vector Space Model: each document is represented as a vector in an n-dimensional space, where n is the total number of unique terms in all documents. As each dimension corresponds to one of the unique terms, a vector component's magnitude is defined by the number of times the respective term occurs within the given document. The basis for this is the so-called document-term matrix, which reports the frequencies of all n terms for each document. The Vector Space Model allows calculation of the semantic similarity between two documents based on their spatial proximity within the high-dimensional space (for more information, see Manning et al. 2009, 110).

To find groups of documents with a similar theme, several clustering algorithms can be used. Which one is best depends on the size of the document collection and researchers' interest: standard k-means is very popular due to its scalability, but requires researchers to define the desired number of clusters before the analysis. Hierarchical agglomerative clustering techniques, on the other hand, allow researchers to flexibly choose the best cluster solution after the analysis, but the respective algorithms are not efficient for large datasets. Combining the "best of both worlds", Steinbach, Karypis, and Kumar (2000) recommend the bisecting k-means algorithm, a divisive clustering algorithm with high performance efficiency, in a systematic evaluation of clustering techniques.

Even more than in a manual content analysis, a main part of the work begins after the analysis: while documents in the same clusters share a common ground based on their similar use of vocabulary, it is up to the researcher to make meaning of the results. To make the interpretation of the topic clusters easier, the most relevant words within each cluster are typically presented as cluster labels in the analysis output.

Topic Models

In recent years, topic models such as LDA have emerged as powerful tools to thematically organize content in large archives of digital texts (Blei and Lafferty 2009). Following the same logic as document clustering, they aim to uncover thematic patterns in document collections: in a fully automated classification, categories are estimated within the process, with documents simultaneously being assigned to these categories (Grimmer and Stewart 2013). The potential of topic models to work quickly through otherwise unstructured document collections offers new possibilities for social scientists: Quinn et al. (2010), for example, automatically examine the agenda of the US senate between 1997 and 2004 by fitting a topic model to more than 100,000 speeches. This way, they show how politi-

cal attention towards the inferred topics such as defense or abortion changes over time. Roberts et al. (2014) present a way to apply structural topic models to open-ended survey responses, making an otherwise time-consuming task both easier and more revealing.

Introduced by Blei, Ng, and Jordan in 2003, topic models infer the hidden semantic structure in a collection of documents based on a hierarchical Bayesian analysis (Blei and Lafferty 2009). As latent variables, topics only become apparent in similar patterns of words used across documents. Topic models all refer to the idea of an imaginary generative process, in which the distribution of topics in a document and the distribution of words in a topic are drawn from a probability distribution. By reversing this stochastic procedure, they infer the hidden structure based on the resulting high co-occurrence of groups of words. In this process, every word in every document of the collection is allocated to one of the topics, so that a document is represented as a mix of topics (or, algebraically, a topic weight vector). Accordingly, topics are formally defined as "distribution[s] over a fixed vocabulary" (Blei 2012, 78) that report the probability of each word within that topic.

In contrast to document clustering, LDA realistically models multiple topics per document and provides a topic model that can also be generalized to new documents. Some of the assumptions, however, such as not taking account of word and document order, are unrealistic and have been relaxed and extended in several new approaches: Rosen-Zvi et al. (2004) extend LDA to also include authorship information, so that personal writing style is considered as a feature in the analysis. To be able to account for word order, Wallach (2006) developed a topic model that incorporates the assumption that, in the generative process, words are drawn conditionally on the previous word. Hidden topic Markov models also consider word order by assuming that words in the same sentence have the same topic. In addition, consecutive sentences are more likely to have the same topic (Gruber, Rosen-Zvi, and Weiss 2007). With the introduction of dynamic topic models, Blei and Lafferty (2006) furthermore account for the fact that topics change over time and enable researchers to study the evolution of topics within a document collection. Finally, the Bayesian nonparametric topic model estimates the number of topics within the analysis (in LDA, this has to be specified a priori) and is able to identify new topics when applied to additional documents (Blei 2012). As it is (so far) not possible to include all extensions into one comprehensive topic model, researchers have to decide which feature is most important for their analysis when choosing the appropriate approach.

Recommendations and Discussion

In this article, we have presented a roadmap of text analysis methods that assist researchers to document collections that are too large to read. The approaches come with different assumptions, costs, and benefits: the main difference obviously lies in the fact that categories are known for rule-based approaches, dictionaries, and supervised learning, while document clustering and topic models are designed for exploration without prior knowledge. This translates into varying analysis costs: the first three are defined by high pre-analysis costs, as categories have to be conceptualized in hours of tedious work. While the latter have low costs at the first stage of the research process, the amount of time it

takes to interpret the results is disproportionately higher: post-analysis costs make up a substantial part of the research process for document clustering and topic models, due to the fact that the output is mostly not self-explaining and requires careful interpretation by a knowledgeable researcher. The choice of approach depends on the specifics of the document collection and, most importantly, the research interest. Benefits are often highest when approaches are combined: fully automated methods can also be used to get to know the dataset when preparing for a (manual) task that works with known categories. Before running a dictionary-based approach on a dataset, a simple co-occurrence analysis might also help to improve the dictionary by revealing search terms that researchers might have missed, and search terms that have to be excluded because they produce too many false-positives.

All methods that we presented are characterized by low costs for the analysis of large datasets: both by automating pre-existing categories as well as by categorizing automatically, researchers can save enormous amounts of time and effort when working with large datasets. The benefits of increased efficiency, reliability, and transparency, that are assets of the automated analysis, however, must not delude us that there are not also clear limitations: computers do not understand texts the way human coders can, and are only as good as the algorithms they perform. Furthermore, the application of methods and tools outlined in this article is not a trivial task. Implementing methods that originated in disciplines with significant differences to the social sciences therefore requires careful preparation. A solid understanding of the underlying statistical processes is needed to be able to assess how they will work on a given dataset and how their results can be interpreted. We need to be careful not to rely blindly on the power of algorithms to achieve large sample sizes-welcoming their benefits, but keeping their limitations in mind (for a discussion of limitations, see Mahrt and Scharkow 2013; Zamith and Louis 2015).

However, with open-source toolsets like Voyant Tools (voyant-tools.org) and vivid user communities for R and Python, there are many resources available for social scientists who aim to include automated text analysis methods into their projects. Using sophisticated tools and resources from disciplines such as computer science and computational linguistics, journalism scholars can gain insight into the constant information flow and make big data a regular feature in the scientific debate. As this innovative research area is constantly evolving, interdisciplinary collaborations are another great way to combine respective strengths and realize ambitious projects.

REFERENCES

Abbasi, Ahmed, and Hsinchun Chen. 2005. "Applying Authorship Analysis to Extremist Group Web Forum Messages." *Intelligent Systems* 20 (5): 67-75.

Blei, David, and John Lafferty. 2006. "Dynamic Topic Models." In *Proceedings of the 23rd International Conference on Machine Learning*, edited by William Cohen and Andrew Moore, 113-120. New York, NY: ACM.

Blei, David, and John Lafferty. 2009. "Topic Models." In *Text Mining: Classification, Clustering, and Applications*, edited by A. N. Srivastava and M. Sahami, 71-92. London: Chapman & Hall/CRC Press.

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

Blei, David M.. 2012. "Probabilistic Topic Models." *Communications of the ACM* 55 (4): 77-84.

Burscher, Bjö, Vliegenthart Rens, and de Vreese Claes H. 2015. "Using Supervised Machine Learning to Code Policy Issues: Can Classifiers Generalize across Contexts?" *The Annals of the American Academy of Political and Social Science* 659 (1): 122-131.

Dietterich, Thomas G. 2000. "Ensemble Methods in Machine Learning." In *Lecture Notes in Computer Science 1857: Multiple Classifier Systems*, edited by Josef Kittler and Fabio Roll, 1-15. Berlin: Springer.

DuBay, William H. 2004. *The Principles of Readability.* Costa Mesa, CA: Impact Information.

Friedl, Jeffrey E. F. 2006. *Mastering Regular Expressions.* Sebastopol, CA: O'Reilly.

Flaounas, Ilias, Omar Ali, Thomas Lansdall-Welfare, Tijl De Bie, Nick Mosdell, Justin Lewis, and Nello Cristianini. 2013. "Research Methods in the Age of Digital Journalism: Massive-Scale Automated Analysis of News-Content-Topics, Style and Gender." *Digital Journalism* 1 (1): 102-116.

Greer, Jennifer D., and Yan Yan. 2011. "Newspapers Connect with Readers through Multiple Digital Tools." *Newspaper Research Journal* 32 (4): 83-97.

Grimmer, Justin, and Gary King. 2011. "General Purpose Computer-Assisted Clustering and Conceptualization." *Proceedings of the National Academy of Sciences of the United States of America* 108 (7): 2643-2650.

Grimmer, Justin, and Brandon M. Stewart. 2013. "Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts." *Political Analysis* 21 (3): 267-297.

Gruber, Amit, Michal Rosen-Zvi, and Yair Weiss. 2007. "Hidden Topic Markov Models." *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics* 163-170.

Hillard, Dustin, Stephen Purpura, and John Wilkerson. 2008. "Computer-Assisted Topic Classification for Mixed-Methods Social Science Research." *Journal of Information Technology & Politics* 4 (4): 31-46.

Kilgarriff, Adam. 2012. "Getting to Know Your Corpus". *Text, Speech and Dialogue. Lecture Notes in Computer Science* 7499: 3-15.

Kohlschütter, Christian, Nejdl Wolfgang, and Fankhauser Peter. 2010. "Boilerplate Detection using Shallow Text Features." In *Proceedings of the third ACM international conference on Web Search and Data Mining*, 441-450. New York, NY: ACM.

LaBerge, David, and S. Jay Samuels. 1974. "Toward a Theory of Automatic Information Processing in Reading." *Cognitive Psychology* 6 (2): 293-323.

Loughran, Tim, and Bill McDonald. 2011. "When is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks." *The Journal of Finance* 66: 35-65.

Mahrt, Merja, and Michael Scharkow. 2013. "The Value of Big Data in Digital Media Research." *Journal of Broadcasting & Electronic Media* 57 (1): 20-33.

Manning, Christopher D., Raghavan, Prabhakar, and Schütze, Hinrich. 2009. *Introduction to information retrieval.* Cambridge: Cambridge University Press.

Quinn, Kevin M., Monrow, Burt L., Colaresi, Michael, Crespin, Michael H., and Radev Dragomir R. 2010. "How to Analyze Political Attention with Minimal Assumptions and Costs." *American Journal of Political Science* 54 (1): 209-228.

Pang, Bo, and Lillian Lee. 2008. "Opinion Mining and Sentiment Analysis." *Foundations and Trends in Information Retrieval* 2 (1-2): 91-231.

Roberts, Margaret E., M. Brandon, Dustin Tingley, Christopher Lucas, Jetson Leder-Luis, Shana Kushner Gadarian, Bethany Albertson, and David G. Rand. 2014. "Structural Topic Models for Open-Ended Survey Responses." *American Journal of Political Science* 58 (4): 1064-1082.

Rosen-Zvi, Michal, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. 2004. "The Authorship-Topic Model for Authors and Documents." In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, edited by Max Chickeringand Joseph Halpern, 487-494. Arlington: AUAI Press.

Scharkow, Michael. 2013. "Thematic Content Analysis Using Supervised Machine Learning: An Empirical Evaluation Using German Online News." *Quality and Quantity* 47 (2): 761-773.

Sebastiani, Fabrizio. 2002. "Machine Learning in Automated Text Categorization." *ACM Computing Surveys* 34 (1): 1-47.

Sjøvaag, Helle, and Stavelin, Eirik. 2012. "Web media and the quantitative content analysis: Methodological challenges in measuring online news content." *Convergence: The International Journal of Research into New Media Technologies* 18 (2): 215-229.

Michael, Steinbach, Karypis George, and Vipin Kumar. 2000. "A Comparison of Document Clustering Techniques." *KDD Workshop on Text Mining* 400 (1): 525-526.

Wallach, Hanna. 2006. "Topic Models: Beyond Bag-of-Words." In *Proceedings of the 23rd International Conference on Machine Learning*, edited by William Cohen, and Andrew Moore, 977-984. New York, NY: ACM.

Wartena, Christian, and Rogier Brussee. 2008. "Topic Detection by Clustering Keywords." In *Nineteenth International Workshop on Database and Expert Systems Application*, edited by A. Min, Tjoa , and Richard R. Wagner, 54-58. Los Alamitos, CA: IEEE.

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

Zamith, Rodrigo, and Seth Louis. 2015. "Content Analysis and the Algorithmic Coder: What Computational Social Science Means for Traditional Modes of Media Analysis." *The ANNALS of the American Academy of Political and Social Science* 659 (1): 307-318.