Communications of the Association for Information Systems

Tutorial ISSN: 1529-3181

Text Mining For Information Systems Researchers: An Annotated Topic Modeling Tutorial

Stefan Debortoli
University of Liechtenstein, Institute of Information Systems, Vaduz, Liechtenstein, stefan.debortoli@uni.li

Oliver Müller
IT University of Copenhagen, Copenhagen, Denmark

Iris Junglas
Florida State University, College of Business, Tallahassee, FL, USA

Jan vom Brocke
University of Liechtenstein, Institute of Information Systems, Vaduz, Liechtenstein

Abstract:

It is estimated that more than 80 percent of today's data is stored in unstructured form (e.g., text, audio, image, video); and much of it is expressed in rich and ambiguous natural language. Traditionally, the analysis of natural language has prompted the use of qualitative data analysis approaches, such as manual coding. Yet, the size of text data sets obtained from the Internet makes manual analysis virtually impossible. In this tutorial, we discuss the challenges encountered when applying automated text-mining techniques in information systems research. In particular, we showcase the use of probabilistic topic modeling via Latent Dirichlet Allocation, an unsupervised text mining technique, in combination with a LASSO multinomial logistic regression to explain user satisfaction with an IT artifact by automatically analyzing more than 12,000 online customer reviews. For fellow information systems researchers, this tutorial provides some guidance for conducting text mining studies on their own and for evaluating the quality of others.

Keywords: Text Mining, Topic Modeling, Latent Dirichlet Allocation, Online Customer Reviews, User Satisfaction

1 Introduction

With the emergence of the Web 2.0 and social media, the amount of unstructured, textual data on the Internet has grown tremendously, especially at the micro level (Gopal, Marsden, & Vanthienen, 2011). For example, at the time of writing, Amazon.com alone offered more than 140 million customer reviews, about more than 9 million products, written by millions of Amazon users, and spanning a time frame of almost 20 years (McAuley, Pandey, & Leskovec, 2015; McAuley, Targett, Shi, & van den Hengel, 2015). And the over 300 million active Twitter users had generated an average of 500 million Tweets per day (Twitter, 2015). This abundance of publicly available data creates new opportunities for both qualitative and quantitative Information Systems (IS) researchers.

Traditionally, the analysis of natural language data has prompted the use of qualitative data analysis approaches, such as manual coding (Quinn, Monroe, Colaresi, Crespin, & Radev, 2010). Yet, the size of textual data sets available from the Internet exceeds the information processing capacities of single researchers, and even that of research teams. And despite methodological guidelines on how to improve the validity and reliability when analyzing qualitative data with multiple coders (Saldaña, 2012), the biases arising from researchers' subjective interpretations of the data cannot be completely mitigated (Indulska, Hovorka, & Recker, 2012).

Text mining techniques allow to automatically extract implicit, previously unknown, and potentially useful knowledge from large amounts of unstructured textual data in a scalable and repeatable way (Fan, Wallace, Rich, & Zhang, 2006; Frawley, Piatetsky-Shapiro, & Matheus, 1992). Although the automated computational analysis of text only scratches the surface of a natural language's semantics, it has proven to be a reliable tool when fed with sufficiently large data sets (Halevy, Norvig, & Pereira, 2009). Against this background, text mining offers an interesting and complimentary strategy of inquiry for IS research that can be well combined with other data analysis methods (e.g., regression analysis) or used to triangulate research results gained from more traditional data collection and analysis methods. In particular, automated text mining allows IS researchers to (1) overcome the limitations of manual approaches to qualitative data analysis, and (2) yield insights that could otherwise not be found. The following two examples shall serve as an illustration.

In a study that has received much public and scholarly attention, Michel et al. (2011) investigated cultural trends by computing the yearly relative frequency of words appearing in Google Books. This simple statistical analysis, applied to more than five million digitized books, produced some interesting insights. The study found, for instance, that the diffusion of innovations, measured by word frequencies corresponding to certain technologies (e.g., radio, telephone) over time, is accelerating at an increasing rate. While at the beginning of the 19th century it took an average of 66 years from invention to widespread adoption of a technology, the average time-to-adoption dropped to 27 years around 1900.

Another illustrative example comes from the field of social psychology. Pennebaker and colleagues (Pennebaker, 2011; Tausczik & Pennebaker, 2010) developed the Linguistic Inquiry and Word Count (LIWC) tool that allows to automatically quantify the linguistic style of texts by counting the use of different function words (e.g., pronouns, articles, prepositions). Function words are frequent in natural language, but readers – and coders – usually do not consciously pay attention to them, but focus on content words (e.g., nouns, verbs) instead. Yet, it turns out that subtle differences in usage patterns of function words are important predictors for numerous psychological states. Researchers have used LIWC, for instance, to detect deception in online customer reviews, as reviewers who are lying tend to use more personal pronouns and "I" words and less concrete terms (e.g., numbers) than truthful reviewers (Ott, Choi, Cardie, & Hancock, 2011).

In this tutorial, we discuss the challenges encountered when applying automated text-mining techniques in information systems research. Applying text mining requires the skill sets of a diverse set of fields, including computer science and linguistics; and not every IS researcher is familiar with the concepts and methods of these fields. While there exists a host of technical literature on the ideas and methods underlying specific text mining algorithms, such as topic modeling (Blei, 2012) or sentiment analysis (Pang & Lee, 2008), these publications rarely touch upon the "how-to" aspects of applying text mining as a strategy of inquiry for (information systems) research. We particularly focus on probabilistic topic modeling as a technique for inductively discovering topics running through a large collection of texts (corpus), such as user-generated content from the web. In addition to outlining the foundations of topic modeling, we

illustrate its concrete use by presenting available software tools and showcasing their application with the help of an integrated example from the area of online customer reviews.

The remainder of this paper is structured as follows. First, we provide an overview of approaches for analyzing large text corpora in general, and subsequently delve into probabilistic topic modeling in particular. Second, we discuss typical challenges encountered in topic modeling studies and outline potential ways for overcoming them. Third, we introduce tools for applying topic modeling and illustrate their application with the help of an integrated example. Finally, we conclude by discussing limitations of the presented methods.

2 Background

2.1 Analyzing Large Text Corpora

One of the most fundamental tasks in text analysis, both manual and automated, is text categorization, that is, the task of assigning chunks of texts (e.g., e-mails, social media comments, news) to one or more categories (e.g., spam or no spam, positive or negative sentiment, business or politics or sports news). Different methods of text categorization are available, and each of them is associated with certain assumptions and costs (see Table 1).

The traditional approach used for categorizing text in social science research is manual coding (Berg & Lune, 2011). Coding aims at differentiating and combining the original data into categories in order to capture the essential meaning of a chunk of data (Miles & Huberman, 1994). Various coding techniques exist (see, e.g., Saldaña, 2012); however, at the most basic level a distinction between bottom-up and top-down approaches can be made (Urquhart, 2012). As part of bottom-up coding, codes are suggested by the data (i.e., words and phrases)—irrespective of extant theory (Urquhart, 2012). The coder is expected to approach the analysis task with an open mind and to not impose preconceptions on the data. In contrast, for top-down coding coders use a predefined coding schema derived from literature and assign the data to these codes (Urquhart, 2012). This latter style of coding is sometimes also used in combination with counting instances of codes, for example when conducting systematic content analysis.

Manual coding has many strengths, such as human's unrivaled capacity to understand the meaning of natural language or the possibility for highly complex and contingent mappings between text features and categories (Quinn et al., 2010). Yet, it also suffers from a number of limitations. First, it is prone to human subjectivity and, hence, different coders may end up with different results (Urquhart, 2001). In order to overcome these threats to validity and reliability, various strategies for achieving intersubjective verifiability have been proposed, including the use of codebooks, having multiple coders, or conducting inter-coder reliability tests (Indulska et al., 2012). Yet, the applicability of these strategies is restricted by a second limitation of manual coding. Manual coding is costly in terms of needed person-hours and requires substantive domain knowledge (Quinn et al., 2010). To overcome these limitations, researchers have developed computer-aided approaches for text analysis by applying dictionary-based or machine learning algorithms.

A dictionary-based text categorization relies on experts assembling lists of words and phrases that are likely to indicate the membership of a chunk of text in a particular category (Quinn et al., 2010). Using this dictionary, a computer can then automatically parse through large amounts of texts and determine the classification of a unit of text. A dictionary-based categorization is only applicable if categories are predefined, and the mapping between text features (i.e., words and phrases) and categories is known in advance and can be codified. In other words, dictionary-based approaches can only be applied to automate top-down manual coding. Many sentiment analysis methods, which classify texts into positive or negative categories, such as the popular SentiStrength (Thelwall, Buckley, Paltoglou, Cai, & Kappas, 2010), use dictionary-based text categorization methods.

A second approach of automating top-down manual coding entails the use of supervised learning methods. Like before, categories are known and predefined, however the mapping between text features and categories is not explicitly known (Quinn et al., 2010). Using a set of manually classified documents as training examples, supervised machine learning algorithms can then be applied to automatically detect a relationship between the usage of a word and its category assignments. The learned patterns can then be used to classify new or unseen texts. E-mail spam filtering is a classic example for the effective use of supervised learning methods for text categorization, for example, by picking out words in financial advertisements such as "$$$," "credit," and "f r e e."

Finally, unsupervised machine learning methods for text categorization attempt to find hidden structures in texts for which no predefined categorization exists (Quinn et al., 2010). Unsupervised learning methods (e.g., clustering, dimensionality reduction) use features of texts to inductively discover latent categories and assign units of texts to those categories. This inductive approach is comparable to manual bottom-up coding, or open coding as known from the grounded theory method (Berente & Seidel, 2014). Unsupervised text categorization approaches have some distinct advantages over manual coding: (1) they require only little human intervention and substantive knowledge in the pre-analysis and analysis phase, (2) they generate reproducible results since they are not subject to the human subjectivity bias, and (3) today's algorithms and computing systems can cope with ample volumes of texts that would be impossible to analyze even with large coding teams. On the down side, unsupervised methods require an extensive post-analysis phase that is typically very time consuming as a researcher has to make sense of the automatically generated inductive categorizations.

Table 1. Assumptions and Costs of Different Text Categorization Methods (Adapted and Extended from Quinn et al., 2010)

	Manual Coding (Bottom up)	Manual Coding (Top down)	Dictionaries	Supervised Machine Learning	Unsupervised Machine Learning
Assumptions					
Categories are predefined	No	Yes	Yes	Yes	No
Relevant text features are known	Yes	Yes	Yes	Yes	Yes
Mapping between text features and categories is known	No	No	Yes	No	No
Costs					
Pre-analysis costs					
Person-hours spent conceptualizing	Low	High	High	High	Low
Level of substantive knowledge	Low	High	High	High	Low
Analysis costs					
Person-hours spent per text	High	High	Low	Low	Low
Level of substantive knowledge	Moderate	Moderate	Low	Low	Low
Post-analysis costs					
Person-hours spent interpreting	Moderate	Low	Low	Low	Moderate
Level of substantive knowledge	High	High	High	High	High

2.2 Probabilistic Topic Modeling

In the following, we will discuss probabilistic topic modeling, an unsupervised machine learning method, in more detail. Unsupervised machine learning methods rely only on a few assumptions in terms of the underlying text data and require minimal costs for data analysis which enables researchers to apply them on a broad variety of sources and large volumes.

The underlying idea of many unsupervised learning methods for text categorization is rooted in the distributional hypothesis of linguistics (Firth, 1957; Harris, 1954), referring to the observation that "words that occur in the same contexts tend to have similar meanings" (Turney & Pantel, 2010, p. 142). For example, co-occurring words such as "goal," "ball," "striker," and "foul" in newspaper articles could be interpreted as markers for a common category, namely "football," and used to group articles accordingly.

Several distributional methods for unsupervised text categorization have been developed and extended over the last decades. Among the most frequently used approaches in IS research are Latent Semantic Analysis (Landauer, Foltz, & Laham, 1998), Latent Dirichlet Allocation (Blei, Ng, & Jordan, 2003), and Leximancer (Smith & Humphreys, 2006). Latent Semantic Analysis (LSA) extracts distributional word usage patterns through reducing the dimensionality of a term-document matrix by applying a singular value decomposition. The resulting latent semantic factors, which share many similarities with the outputs of factor analysis or principal components analysis, are often interpreted as topics (Landauer et al., 1998). LSA has been a groundbreaking development in the field of computational linguistics, but suffers from interpretability issues as the computed factor loadings often have no clear interpretation. In order to overcome these shortcomings, probabilistic LSA (pLSA) (Hofmann, 1999) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003; Blei, 2012) have been developed as extensions to the classic LSA idea. In both methods, the associations between documents and topics as well as between topics and words are represented as probability distributions that can be used for further statistical analyses. For example, the estimated probability distributions can be grouped and aggregated by document metadata, or be used as predictors in regression or classification models. Also, various commercial tools exist that apply a distributional hypothesis. Leximancer (http://www.leximancer.com), for example, combines unsupervised extraction of word co-occurrence patterns with concept mapping and intuitive visualizations (Smith & Humphreys, 2006). However, Leximancer's algorithms and data structures are patented and, hence, only scarcely documented.

In the following, we will describe the application of probabilistic topic modeling with LDA in detail. We chose LDA for three reasons: (1) LDA is an evolution of the seminal LSA idea and both methods have been used extensively in academic research[1], (2) numerous free and open source LDA software libraries exist for most statistical programming languages (including R, Python, Java), and (3) LDA's capability of extracting semantically meaningful topics from texts and categorizing texts according to these topics has been validated in several empirical studies (e.g., Boyd-Graber, Mimno, & Newman, 2014; Chang, Boyd-Graber, Gerrish, Wang, & Blei, 2009; Lau, Newman, & Baldwin, 2014; Mimno, Wallach, Talley, Leenders, & McCallum, 2011).

The core idea behind LDA, first proposed by Blei et al. (2003), is an imaginary generative process that assumes that authors compose *D* documents by first choosing a discrete distribution of *T* topics to write about, and then drawing *W* words from a discrete distribution of words that are typical for each topic (see Figure 1). In other words, a document is defined by a probability distribution over a fixed set of topics, and each topic, in turn, is defined by a probability distribution over a confined vocabulary of words. While all documents are assumed to be generated from the same fixed set of topics, each document exhibits these topics in different proportions, possibly ranging from 0% (if a document fails to talk about a topic entirely) to 100% (if a document talks about a topic exclusively). The computational task of the LDA algorithm is to estimate the hidden topic and word distributions, given the observed per-document word occurrences. This estimation can be done either via sampling approaches (e.g., Gibbs sampling) or optimization approaches (e.g., Variational Bayes).

[1] At the time of writing, a search on Google Scholar for „latent semantic analysis" produced over 32,000 hits, and a search for „latent dirichlet allocation" over 19,000 hits.

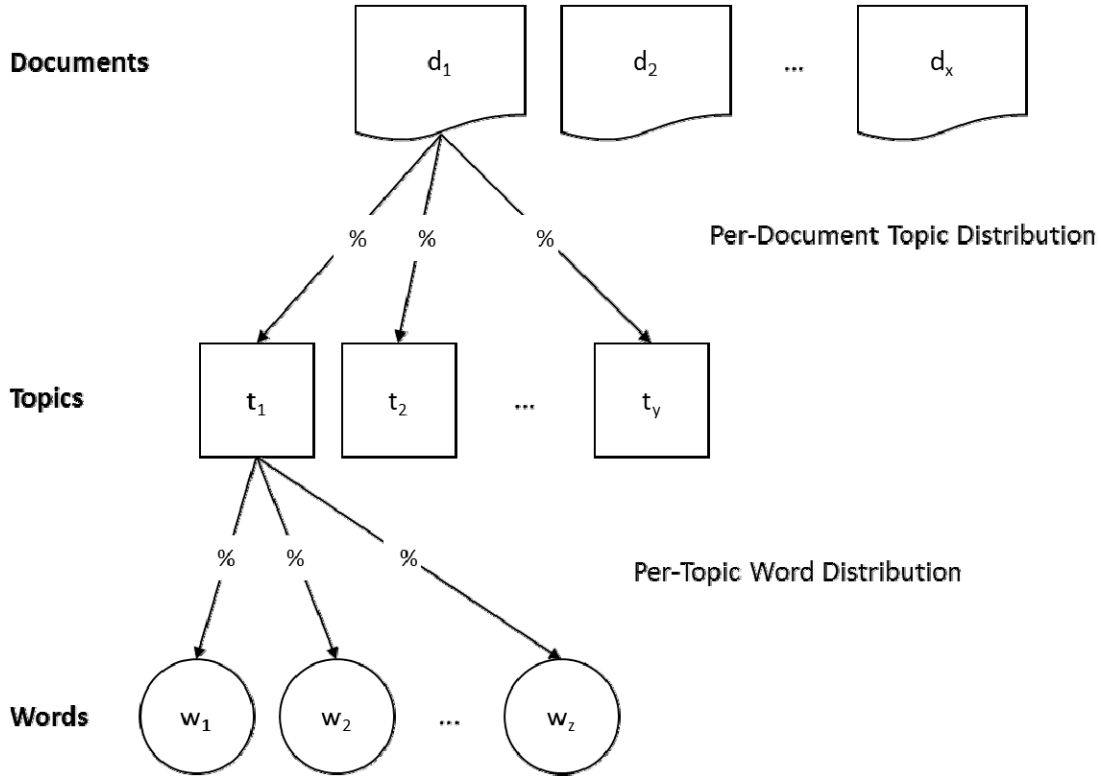

Figure 1. Schematic Overview of LDA

Figure 2 illustrates the basic idea behind LDA using an exemplary online customer review about a "Fitbit Flex"[2] device and its topic distribution, as well as six topics and their word distributions. The exemplary review covers three topics to different degrees, namely Topic 3 (55%), Topic 2 (35%), and Topic 6 (10%); other topics are not present (0%). Each topic, in turn, is represented by a distribution over words. Topic 3, for example, assigns high likelihood for words like "weight" (8%), "loss" (5%), and "pounds" (4%), indicating that the topic covers weight loss as an effect of using the Fitbit device. Topic 2, on the other hand, has highly probable words like "gift" (10%), "love" (7%), or "christmas" (7%), signaling that the Fitbit device has been given or received as a present. Finally, the most probable words for Topic 6 are "app" (12%), "iphone" (8%), and "sync" (3%), referring to the synchronization between the Fitbit and the corresponding iPhone app.

[2] The Fitbit Flex is a wearable technology to track and analyze personal health and fitness data around the clock.

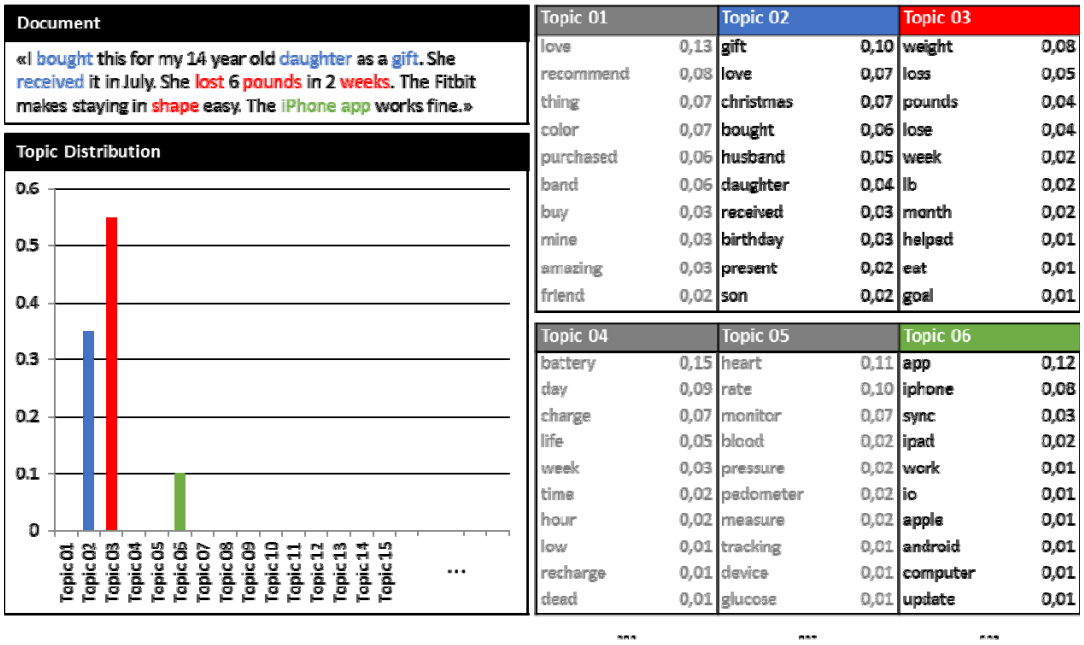

Figure 2. Illustrative Example of LDA

3 Practical Challenges of Applying Topic Modeling

We now turn to the practical challenges faced when applying topic modeling as a method for automated analysis of large sets of qualitative data. These challenges roughly mirror the phase of a typical research process.

3.1 Challenge #1: Obtaining Data From The Web

As topic modeling produces valid results only when fed with a sufficiently large data set (n>1,000), it is typically not used for analyzing data collected by a researcher herself (e.g., interview transcripts, field notes), but for analyzing texts produced by a large group of people (e.g., user-generated content originating from social media websites, research articles written by a scientific community) and available as Internet resources. In broad terms, there are three ways to extract text data from Internet sources: (1) via Application Programming Interfaces (APIs), (2) via web crawlers, or (3) via file downloads.

APIs are programmatic data access points made available by data providers to offer reusable content in a controlled way (e.g., by restricting the scope and amount of data that can be accessed) and in a structured format (e.g., by using markup languages like XML or JSON). While extracting data via APIs usually ensures high levels of data quality, it is rather rare for providers to expose the full breadth and depth of their data via APIs—in the best of cases, they require users to pay for such data requests. A popular API used in text mining studies is the Twitter API, as it is subject to very few restrictions. While the APIs to social networks like Facebook, LinkedIn or Google+ only permit access to data about "friends," the Twitter API provides access to data about all members of the network.

Web crawlers offer another way to extract data from the web. These automated programs traverse the web's topology and download relevant pages and hyperlinks (Liu, 2011). They are not operated by data providers, but by data consumers or intermediaries. Web crawlers parse, or "scrape," a page's content using simple natural language processing heuristics (e.g., regular expressions). As web pages typically contain lots of noise (e.g., HTML tags, advertisement banners), many web crawlers try to filter out irrelevant elements; with more or less success. In addition to extracting content, web crawlers are also able to capture the web's underlying linkages and social structures by building a graph of interconnected actors (e.g., web pages, users). Overall, web crawlers provide researchers with lots of flexibility. For example, a researcher can develop a crawler that targets specific topics by initializing it with a set of

search terms or seed URLs. Such flexibility, however, comes at a price. Crawlers often require in-depth programming, and the quality of data gathered might not be up to par to what is required for analysis.

Finally, researchers can also use open data repositories that are downloadable from the web. Open data comprises data that can be freely used, reused, and redistributed by anyone, subject only to the requirement of attribution (OKF, 2012). Over the last years, governments (e.g., http://www.data.gov/), not-for-profit organizations (e.g., http://en.wikipedia.org/wiki/Wikipedia:Database_download), research institutions (e.g., https://snap.stanford.edu/data/), and private organizations (e.g., http://www.yelp.com/dataset_challenge) have established open data repositories that contain large collections of text data that can be of interest to IS researchers. Most of these data sets are integrated and curated, which eases access and ensures data quality.

In choosing a data collection approach, one has to consider the time frame that can be captured. Data snapshots are the easiest to accomplish and are supported by most APIs and web crawlers. Also, many open data sets are the result of cross-sectional surveys and, hence, represent snapshots. Collecting longitudinal data are more problematic. While some APIs and open data files provide historical data, web crawling is done in periodic batch runs that are unable to capture the full volatility of web pages. Finally, the most difficult time frame to capture is data in real time. Only so-called streaming APIs, such as Twitter's Firehose API that allows real-time access to the complete stream of Tweets (currently about 4,000 Tweets per second), are able to provide this capability. Firehose access, however, is restricted to selected partner organizations only.

3.2 Challenge #2: Getting Data Ready For Analysis

Natural language data are characterized by a lack of well-defined structures and a high proportion of noise. Hence, in almost all cases the data needs to undergo an extensive preprocessing phase before it can be statistically analyzed through topic models. Although the data preparation steps are rarely highlighted in the presentation of research results, they typically require 45 to 60 percent of the overall effort (Kurgan & Musilek, 2006).

As a first step, a high-level exploratory data analysis (EDA) should be performed in order to get an initial feeling for the data set and to identify potential data quality problems. Besides the computation of summary statistics (e.g., number of documents in the data set, average number of words per document), researchers should use visualizations. For example, word-frequency plots provide valuable information about required data cleaning and natural language processing steps. Likewise, plotting timestamps of documents on a timeline can quickly reveal missing data, potentially pointing to errors during data collection, or temporal trends and seasonal patterns. If the obtained text documents contain numerical information that will be used in later analyses, for example as independent or dependent variables in a regression analysis, they should also be plotted in order to visualize their distributions and identify potential anomalies.

After having explored the overall data set, the obtained texts need to be inspected and preprocessed at the document level. Typical preparation steps include: data cleaning, data construction, data formatting, and natural language processing.

Data cleaning is one of the fundamental steps in getting natural language data ready for analysis by removing duplicates and noise. As many data sets used in text mining studies constitute secondary data, the chances that they contain "unclean" data is rather high. For example, posts on online social networks like Twitter might contain duplicate records (retweets, spam), and data collected by web crawlers might be full of noise in the form of HTML tags. Duplicates and noise, if left unattended, may not only lead to biased but also to incorrect results.

Data construction entails deriving new attributes and/or records. Examples of derived attributes are computations involving multiple attributes (e.g., calculating the longevity of an online review by subtracting the date of creation from the current date) or single attribute transformations (e.g., tagging reviews with geographic locations). The necessity for the creation of new data attributes is highly dependent on the subsequent data analysis procedure. To ensure transparency, exact formulas for the derivation of new attributes should be provided.

After these initial steps, most documents need to be (re-)formatted to allow for processing with particular analysis tools or methods. Re-formatting can range from simple changes of individual values (e.g., removing illegal characters or changing character encodings) to complex data model transformations. For

example, data extracted via APIs, Web crawlers, or downloads are mostly represented in flat files (e.g., CSV) or hierarchical data models (e.g., XML, JSON); for analysis as well as storage, it might be useful to convert such data into a relational (e.g., for SQL databases) or key-value data model (e.g., for NoSQL databases). Ideally, the original data model with its various sources as well as the final data model that is used for analytical purposes is illustrated in detail and sufficiently documented.

After document-level preprocessing, the set of individual documents undergoes a number of low-level natural language processing (NLP) steps, such as tokenization (i.e., splitting up documents into sentences and sentences into words), n-gram creation (i.e., the creation of n consecutive words: 1-grams are, e.g., "fast," "food," or "chain;" 2-grams are the concatenation of two 1-grams, e.g., "fast food;" and 3-grams consist of three 1-grams, e.g., "fast food chain"), stopping (i.e., removal of common or uninformative words), part-of-speech filtering (i.e., identifying and filtering words by their part of speech), lemmatizing (i.e., reducing a word into its dictionary form, e.g., plural to singular for nouns, verbs to the simple present tense), stemming (i.e., reducing a word to its stem), and the creation of a structured numerical representation of the document collection (e.g., creating a vector or matrix representation) (Miner et al., 2012). The common objective of these transformations is to remove noise and to gradually turn qualitative textual data into a numerical representation that is amenable to latter statistical analysis. Unfortunately, there is no easy recipe for selecting the appropriate combination of natural language preprocessing steps. Much of it is determined by the study's goal and its underlying dataset. However, some strategies can be applied in terms of stop-word removal, text normalization, and collocation-discovery (Boyd-Graber et al., 2014) that alleviate the dilemma to some extent.

In order to identify stop words, generating word frequency lists (i.e., counts of the number of occurrences of every word in a text corpus) is a useful approach. For example, when studying online customer reviews about the Apple iPhone, the terms "Apple" and "iPhone" will have high frequency counts, but do not add particular value to the analysis and, therefore, may be removed. Other approaches, such as tf-idf weighting of word counts, may also be applied to automatically filter uninformative terms (Salton & McGill, 1983).

Text normalization typically includes the conversion of all characters to lower-case as well as lemmatizing every word. For example, the words "dog," "Dog," "dogs," and "Dogs" will all be converted to "dog," resulting in just one word instead of four different tokens. The concept of text normalization can even be pushed further by applying stemming. For example, the words "analyze" and "analysis" will be reduced to "analy" in stemming. This reduction in word, however, might lead to another problem (Evangelopoulos, Zhang, & Prybutok, 2012)—it will make it impossible to differentiate whether "analy" is referring to a noun or verb in a given context.

Finally, discovering collocations of words, or multi-word expressions (i.e., n-grams | n > 1), can be helpful to find the correct meaning of words. For example, the word "house" means one thing in a given context, but the word "white house" has, in the majority of cases, an entirely different meaning. Therefore, performing an n-gram analysis with n > 1 is recommended, particular in cases where the results will be interpreted by humans later on.

3.3 Challenge #3: Fitting and Validating a Topic Model

Fitting a topic model to a collection of documents can be challenging. The LDA algorithm is sensible to changes in its parameters and variations in input data, introduced, for example, through different data preparation procedures.

The most crucial LDA parameter is the number of topics to be extracted (Blei et al., 2003; Boyd-Graber et al., 2014). When choosing too many topics, the algorithm might unearth a plethora of topics that are only minimally distinct (e.g., topics differ in writing style, but not in content), and choosing too few topics might unnecessarily constrain the exploratory potential of topic modeling. Best practice is therefore to vary the number of topics and to evaluate the quality of the resulting models based on the goal of the study. If the research goal is the creation of a topic model that can be interpreted by humans by producing a quantitative representation of a large collection of texts, then the number of topics to be chosen is typically low and might range between 10 and 50. If, in contrast, the topic model is meant to serve as input for another statistical model (e.g., regression, classification, clustering) and human comprehensibility is not an important factor, the number of topics to be chosen is determined by the model fit and less by its interpretability; here, the number of topics might range between 30 and 100, or even higher.

Another set of parameters that have to be chosen as part of the LDA setup are the hyperparameters α and β, which control the shape of the per-document topic distribution and per-topic word distribution, respectively. A large α leads to topic distributions that are broad (i.e., documents contain many topics), and a large β leads to word distributions that are broad (i.e., topics contain many words). In contrast, small values for α and β lead to more sparse distributions, i.e., documents are assumed to contain only few topics, and topics are assumed to contain only few words. Although most topic modeling tools allow users to define α and β explicitly, common practice is to use established standard values (e.g., one divided by the number of topics), or to rely on optimization techniques, as described by Wallach, Mimno, & McCallum (2009), to automatically determine appropriate values.

Once the topic model has been calculated, researchers will have to interpret the results. For presentation purposes, the LDA results are often displayed in form of lists that show the top-n most likely words per topic (Ramage, Rosen, Chuang, Manning, & McFarland, 2009). While this is an intuitive way of presentation, it can bias the investigator as each topic is actually a distribution over the full vocabulary found within the corpus. Therefore, when interpreting the meaning of a topic, a researcher is advised to inspect the actual word probabilities (and not only their rankings), as well as the documents that are strongly associated with each topic (which can be obtained through the per-document topic distribution). Often, researchers then assign descriptive labels to topics in order to assist readers in the interpretation of topics. As with the manual coding of texts, it is recommended to have the interpretation and labeling of topics conducted by at least two independent researchers.

Validating topic models can be difficult. Due to its unsupervised nature, there are no ground rules or gold standards (as of yet) on how topic modeling results can be assessed. In the computer science community, topic models are often evaluated by either measuring their performance for a subsequent task (e.g., information retrieval, regression, classification), or by measuring how well a model trained on a given corpus fits an unseen, or held-out, text (for an overview see Wallach, Murray, Salakhutdinov, & Mimno, 2009). Both approaches assume that the topic model is used by another algorithm. Yet, experiments have shown that topic models with high predictive accuracy do not necessarily possess good human interpretability (Chang et al., 2009).

For topic models intended to be interpreted by humans, Boyd-Graber et al. (2014) propose two guiding questions in order to evaluate their semantic qualities:

1. Are individual topics meaningful, interpretable, coherent, and useful?
2. Are assignments of topics to documents meaningful, appropriate, and useful?

Common threats to the interpretability of individual topics (Question 1) are multi-fold (Boyd-Graber et al., 2014). Too many common words—or alternatively, too many specific words (e.g., names, numbers)—that either cause topics to be too broad or too specific, can prevent a researcher from gaining a deeper understanding of the corpus. Adjusting the list of stop words and re-running the analysis might help to resolve these issues.

Another reason for low quality topics are so-called mixed topics. While the words do not make sense when taken together, they contain subsets of words, which—when taken together—make perfect sense. In other words, mixed topics contain more than one topic and should be split. The opposite is true for identical topics where the algorithm proposes two topics that are semantically equivalent. Both, mixed and identical topics, can be avoided by either increasing or decreasing the number of topics to be extracted.

Finally, there is always the possibility of encountering a nonsensical topic. Such a topic can occur, for example, if the documents exhibit a particular structural pattern and/or have a common writing style and vocabulary. For example, a set of research papers that frequently contain the words “figure” and “table” might cause an algorithm to generate a topic based on those words. While adding these words to the algorithm’s stoplist seems a viable option, it would most likely compromise the quality of other topics. Hence, excluding the topics from further analysis is typically the best solution.

Only recently, researchers have started to develop some quantitative criteria to evaluate the semantic quality of individual topics by comparing the algorithm’s word assignments with that of human users (Ramage et al., 2009) or by measuring the statistical properties of topics (Boyd-Graber et al., 2014).

For instance, the word intrusion task, as introduced by Chang et al. (2009), aims at quantifying the semantic coherence of topics. In this task, six randomly ordered words are presented to human evaluators. Five words are drawn from the most probable words of a given topic and one word—the

intruder—is randomly chosen from the vocabulary of the corpus. The idea is that for a topic to be semantically coherent, a human judge should be able to easily spot the intruder. For example, most people would identify the word "apple" as the intruder in the topic defined by the words {dog, cat, horse, apple, pig, cow} (Boyd-Graber et al., 2014). In contrast, the word "coffee" would be difficult to identify as the intruder in the following topic defined by a semantically incoherent word list {table, sky, apple, yellow, city, coffee}.

Instead of humans, the measurement of topic coherence can also take place in an automated fashion (Lau et al., 2014; Mimno et al., 2011; Newman, Lau, Grieser, & Baldwin, 2010). Most automated approaches compare the most frequently used words of a topic with texts that are known to have a high semantic coherence, such as Wikipedia or newspaper articles. The idea is that words of highly coherent topics (e.g., {dog, cat, horse, apple, pig, cow}) should frequently co-occur within a reference text (e.g., a Wikipedia article about animals); if they don't, it indicates low semantic coherence.

A similar logic can be applied to validate the assignments of topics to documents (Question 2). Using a topic intrusion task where a random document is presented to human evaluators by offering four topic choices, each represented by its top-n most likely words, the validity of topic-to-document assignments can be assessed. Three of the topics are topics that exhibit a high likelihood for the document under question, and one topic is randomly selected (Chang et al., 2009). Measuring how well human coders can identify the intruder topic provides an indication of the quality of the document-topic assignments made by the LDA algorithm.

3.4 Challenge #4: Going Beyond Description

Topic models are, by default, descriptive in nature, i.e., they represent quantitative summaries of large document collections. Particularly for exploratory studies, descriptive models are often the main objective of a study (e.g., the competency taxonomy derived from job advertisements by Debortoli et al. (2014)). Because all associations in topic models are represented as probabilities, a researcher not only can present relevant topics along with selected word and document distributions, but he can also group and aggregate topic probabilities by different document meta data (e.g., by author, geography, time). This, for example, allows topics to be ranked by prevalence, to compare their prevalence for specific sub-groups, or to track the evolution of topics over time (see Grimmer & Stewart (2013) for examples).

Apart from descriptive purposes, topic models can also be used for explanatory or predictive purposes (Blei et al., 2003). For that, the estimated per-document topic probabilities are used as independent variables or predictors in a regression or classification model. Müller et al. (2016), for example, use the probabilistic topic assignments of more than 1 million online customer reviews about video games to build a statistical model that is able to predict the helpfulness of a new or unseen review.

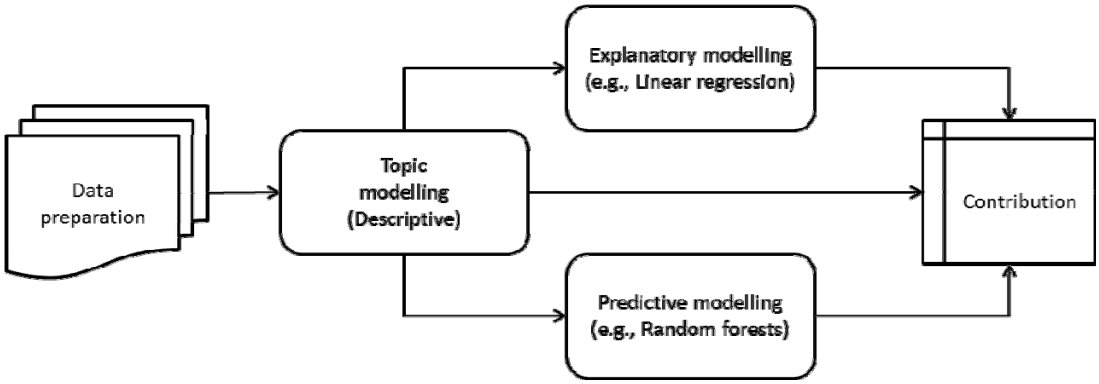

Figure 3. Relationship Between Topic Modeling and Explanatory and Predictive Modeling

As mentioned before, the objective of a study (i.e., description, explanation, prediction) has important implications for topic model fitting. Researchers that aim at description, or intend to feed a topic model into a subsequent explanatory model for hypothesis testing, tend to apply more granular topic models (e.g., 10-50 topics) in order to be able to present their results in full length and in a comprehensible way. In

contrast, when the objective of the study is prediction—and the comprehensibility of the process and results are less important—more high-dimensional representations of documents (e.g., 100+ topics) in combination with non-linear regression or classification techniques (e.g., highly accurate but otherwise "black box" random forests models) have shown to produce the most accurate results. Yet, these models and techniques are difficult—if not impossible—to understand by humans and are hence less useful for description and explanation purposes (see, e.g., Martens and Provost (2014) for a more detailed discussion).

Finally, engaging with existing theories and literature is crucial in order to go beyond a pure quantitative description of a given corpus. One way to do this is to try to map the automatically identified topics with known theoretical constructs in order to place them in their nomological network. Similar to the topic labeling approach, multiple researchers should engage in this topic-construct mapping task. For this, a deep understanding of the domain of interest as well as its theoretical foundation is of utmost importance in order to draw valid conclusions. It may also be helpful to provide a list of definitions of the theoretical constructs that are likely to be discovered to all participating coders to establish a common understanding. In case a topic does not correspond to an existing construct, a researcher may want to theorize about its ontology.

4 An Illustrative Topic Modeling Study

In this section, we illustrate the practical application of topic modeling in combination with explanatory regression analysis, using online customer reviews as an exemplary data source. The presentation of the illustrative example is loosely structured according to the CRISP-DM (Cross-Industry Process for Data Mining) framework, which comprises the phases business understanding (which we renamed into research question), data understanding, data preparation, modeling, evaluation, and deployment (which we renamed into interpretation) (Shearer, 2000).

4.1 Research Question

The goal of our illustrative text mining study is to explain users' satisfaction with a consumer electronics product, as defined by its star rating, by mining the textual and unstructured parts of a review. Our approach is driven by the intuition that the appearance of certain topics in online customer reviews has a significant impact on the corresponding star rating. As an exemplary product we have chosen the "Fitbit Flex Wireless Activity & Sleep Wristband" (https://www.fitbit.com/flex), one of the early wearable technologies to track and analyze personal health and fitness data around the clock.

4.2 Data Understanding

Online customer reviews are defined as "peer-generated product evaluations posted on company or third party web-sites" (Mudambi & Schuff, 2010). Besides freeform text comments, reviews typically contain a numerical product rating (often on a scale from 1 to 5 stars) as well as additional metadata (e.g., reviewer name, date of review, helpfulness votes). Amazon, the largest Internet-based retailer in the world, is also one of the largest sources of online customer reviews (Business Wire, 2010). For the "Fitbit Flex" device more than 12,900 customer reviews are available on Amazon (as of May 2015), spanning more than two years of customer feedback about the product.

Since most e-commerce platforms do not offer APIs to access customer reviews, collecting reviews via web crawling is often the method of choice. For the purpose of this tutorial, we have developed a web crawler that captures all historical product reviews of the "Fitbit Flex" on Amazon. We used the Python package "Beautiful Soup" (http://www.crummy.com/software/BeautifulSoup/), which is designed for extracting data out of HTML files. After downloading the reviews from Amazon, we formatted them as a list of JSON (JavaScript Object Notation) objects to be compatible with the text mining tool we used for topic modeling. Figure 4 shows an exemplary customer review in JSON format. Besides the textual comments, it contains additional metadata, such as the star rating (between 1 and 5), the author (anonymized), and the review date. In total, we have crawled 12,910 reviews, spanning a timeframe of more than three years between March 2012 and May 2015.

```
{
        "rating": 2.0,
        "author": "Anonymous",
        "text": "Disappointed. It came with only one band and it said that it would come with two, both the large and small.",
        "date": "2015-04-25"
}
```


Figure 4. Example of an Online Customer Review in JSON Format

As a next step, we performed an exploratory data analysis. Calculating and plotting descriptive statistics, such as the number of reviews (12,910), number of words (457,239), number of unique words (4,556) and overall word frequencies, provided a first overview of the data set. For instance, an initial word frequency plot showed (Figure 5) that function words like articles and pronouns dominated the corpus. As these words bear little meaning, we decided to remove them in the subsequent data preparation phase.

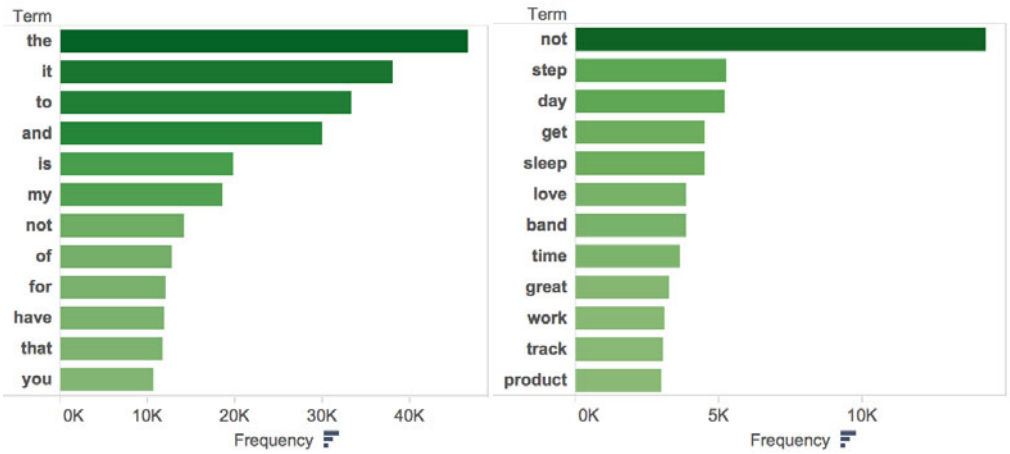

Figure 5. Word-Frequency Plots Before (left) and After (right) data preparation

The existence of metadata for each customer review allowed us to plot the distribution of reviews along a temporal dimension (Figure 6). An interesting observation is that the number of reviews spikes in the last week of December and in the middle of February 2015, which might indicate that the “Fitbit Flex” devices were popular Christmas and Valentine’s Day gifts.

Figure 6. Number of Reviews Over Time

Graphing the average star rating over time supports the assumption that users were continuously satisfied with the device (Figure 7), averaging 3.64 (out of 5 stars). A histogram shows a J-shaped distribution for the star rating (Figure 8), which is a common phenomenon for online customer reviews, caused by purchasing and under-representation biases (Hu, Zhang, & Pavlou, 2009).

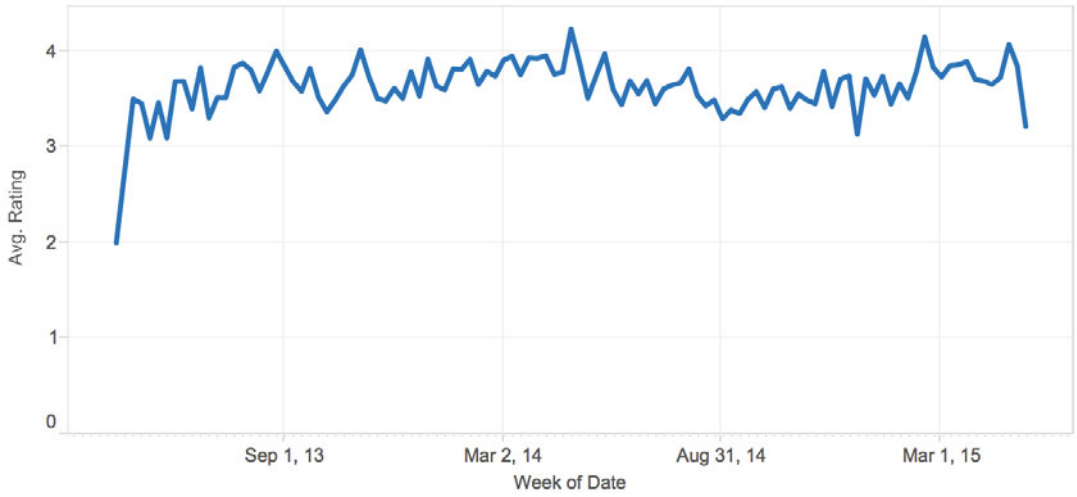

Figure 7. Average Star Rating Over Time

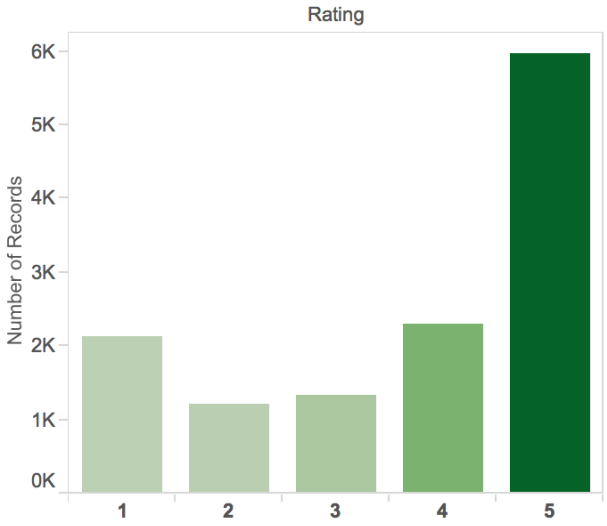

Figure 8. J-shaped Distribution of Star Rating

4.3 Data Preparation, Modeling, and Evaluation

As previously discussed, data preparation procedures can have a substantial influence on the quality of topic modeling results, and going back and forth between data preparation, modeling and evaluation is widely common in text mining studies. Therefore, the three steps are reported in combination in the following. All natural language processing and topic modeling steps were performed with the cloud-based tool MineMyText.com, and the results can be publicly accessed at https://app.minemytext.com/fitbit.

We first performed a number of preparation-modeling-evaluation cycles to determine an appropriate number of topics to be extracted from the document collection. We tested different alternatives ranging

between 20 and 100 topics (in steps of 10) and qualitatively evaluated the cohesiveness of the resulting topics. We determined 50 topics to be the best solution, as more fine-grained topic models (between 50 and 100 topics) produced a growing number of near-duplicate topics, and more coarse-grained models (between 20 and 50 topics) failed to clearly discriminate between topics.

After setting the number of topics to 50, we cleaned the reviews from as much noise as possible. This included:

1. n-gram tokenizing, i.e., splitting documents into single words (i.e., 1-grams: e.g., “product,” “love”), groups of two successive words (i.e., 2-grams: e.g., “highly recommended,” “app store”), or groups of three successive words (i.e., 3-grams: e.g., “heart rate monitor”),
2. removing uninformative but frequent stop words (e.g., “the,” “and”),
3. part of speech (POS) filtering (i.e., removing words based on their part of speech, such as noun, verb, adjective, or adverb),
4. lemmatizing (i.e., reducing words to their dictionary form, e.g., “reviews” and “reviewing” to “review”),
5. removing numbers (e.g., “2014”), and
6. removing HTML tags and other technical symbols, which may stem from web scraping activity.

Table 2 shows an excerpt of the results of the initial LDA analysis. It displays the most probable words for eight selected topics, revealing a number of data quality issues that impaired a proper interpretation of topics. For example:

1. The most probable terms in Topic 7 were “device” and “devices” and in Topic 15 “band” and “bands.” In order to harmonize these terms, we added a lemmatization step to our preprocessing pipeline.
2. Many topics contained the words “fitbit,” “flex,” “fit,” and “bit” among the Top-10 most probable words (see, e.g., Topic 29). This is not surprising, as all reviews were about the Fitbit device. From a text-mining perspective these words do not add new information to the reviews; on the contrary, they might even bias the statistical analysis or hinder the interpretation of results. Therefore, we eliminated those terms by adding them to the custom stop-word list.
3. Engaging with the vocabulary of the domain of interest (e.g., the features of a product under review) is crucial for interpreting and making sense of the results of any text mining study. For example, we spotted that Topic 28 concerned the “silent alarm” function of the device. Unfortunately, the terms “alarm” and “silent” were treated as independent terms by the LDA algorithm. By modifying our model to include n-grams, we forced the algorithm to create a new composite term “silent_alarm,” which helped to better understand the topic. The same problem was observed for Topic 41 (“heart,” “rate,” and “monitor” → “heart_rate_monitor”; “blood” and “pressure” → “blood_pressure”).
4. Some customers provided lots of details in their reviews, for example, the month and year of their purchase. As this information was already captured by the metadata (date field), we chose to remove number words.

Table 2. Exemplary Topics of Initial Topic Model (before data preparation)

Topic ID	Most probable words
T1	activity life device active time people long make lifestyle thought
T2	app device web site iphone android data good apps interface
T7	device devices tracking day data similar monitoring account people point
T15	band bands wear easy love color wrist small comfortable large
T28	alarm silent wake sleep set feature alarms vibrating clock morning
T29	bit fit love body bodymedia bought great media features thought
T31	2014 purchased bought charge 2013 july received june months week

Table 2. Exemplary Topics of Initial Topic Model (before data preparation)

T41	heart rate monitor blood pressure track measure activity things sleep

In order to fine-tune the topic model, we experimented with different data preparation options, re-ran the LDA algorithm, and used an automated approach to evaluate the quality of the resulting descriptive topic model (see variations listed in Table 3). We applied Lau et al.'s (2014) approach to automatically evaluate the semantic coherence of a topic model by calculating how often pairs of terms from the top-n words of a topic co-occur within a narrow window (e.g., 10 words) sliding over a reference corpus (for detailed information about the technique, see Lau et al. (2014) and Newman et al. (2010)). In experiments the resulting normalized pointwise mutual information (NPMI) metric, which can range between -1 (worst) and +1 (best), showed a high correlation with human judgments of semantic coherence (Pearson correlation between 0.84 and 0.98) (Lau et al., 2014). We calculated the NPMI score for different sets of preprocessing options, using the original corpus of reviews as a reference corpus. The results indicate that configuration #5 in Table 3 (i.e., 3-gram tokenization, removal of standard stop words, removal of numbers, lemmatization, POS tagging (nouns, verbs, adjectives), and a small list of custom stop words (fitbit, flex, fit, bit)) produced the topic model with the best interpretability.

Table 3. Different Data Preparation Options and Their Effect on Semantic Coherence

#	Tokenization	Standard stop words	Removing numbers	Lemmatization	POS filtering	Custom stop words	Semantic coherence (NPMI)
1	1-gram						0.1281
2	1-gram	Yes					0.1615
3	1-gram	Yes				fitbit, flex	0.1872
4	3-gram	Yes	Yes	Yes		fitbit, flex, fit, bit	0.2390
5	3-gram	Yes	Yes	Yes	N, V, ADJ	fitbit, flex, fit, bit	0.2826
6	3-gram	Yes	Yes	Yes	N, V, ADJ, ADV	fitbit, flex, fit, bit	0.2760

Table 4 summarizes the final topic model by showing the top-10 most probable words for each of the 50 topics, and Figure 9 visualizes the overall distribution of topics across the corpus (i.e., the higher the probability of a topic, the more reviews talk about the topic).

Table 4. Topics of Final Topic Model

Topic	Most probable words
T1	day step week work walk time steps_day walking couple end
T2	wear shower water time band love charge comfortable wear_shower swimming
T3	weight lost pound lose loss week lb lost_pounds month weight_loss
T4	minute active activity walking step mile running active_minutes run track
T5	wrist wearing wear time zip pedometer put thing lost clip
T6	heart rate heart_rate monitor rate_monitor heart_rate_monitor blood pressure blood_pressure pedometer
T7	sleep tracking time night step sleep_tracking day pattern feature hour
T8	instruction work set site website find web user time figure
T9	gift love christmas bought husband daughter received birthday gave present
T10	motivated move love day walk make step moving motivate keeps_motivated
T11	product great recommend great_product love recommend_product good excellent not_recommend good_product

Table 4. Topics of Final Topic Model

T12	charge charger charging hold month unit battery issue hold_charge problem
T13	sleep mode sleep_mode put time tap tapping put_sleep forget turn
T14	track sleep step keep_track track_steps love activity track_sleep great keeps_track
T15	stair ultra count track climbed step big flight deal floor
T16	make time made long thing life love make_sure aware change
T17	band wrist difficult clasp put hard wristband snap time bracelet
T18	return amazon day product item week charge worked purchased happy
T19	work great works_great not_work item advertised fine love idea not_great
T20	working stopped month stopped_working week worked charging bought quit stopped_charging
T21	accurate step hand pedometer wear stride setting distance dominant wrist
T22	lost band clasp wrist fell time design wristband fall secure
T23	calorie burned calories_burned track burn step day many_calories eat weight
T24	month year mine broke le bought strap issue lasted warranty
T25	app iphone sync ipad iphone_app work io apple android computer
T26	light force display time band step progress watch dot show
T27	step count arm movement accurate hand counting walking count_steps moving
T28	activity sleep monitor level daily activity_level activity_sleep day aware daily_activity
T29	phone sync computer app device android syncing bluetooth not_sync work
T30	wife bought love scale aria wife_loves bought_wife gift aria_scale
T31	jawbone nike band fuel app fuelband nike_fuel accurate fuel_band wanted
T32	goal day step daily reach love set meet hit progress
T33	tool fitness great health goal great_tool program tracking feature activity
T34	good thing work bad idea device make give price tracking
T35	friend fun love great family challenge compete step competition lot
T36	tracker sleep_tracker sleep activity fitness great activity_tracker step fitness_tracker accurate
T37	product review star buy give people good thing read problem
T38	battery day charge life battery_life week charged time hour low
T39	money worth waste time waste_money not_worth piece pedometer product buy
T40	step mile day walked many_steps walk number distance see_many steps_take
T41	great stay motivator moving motivated great_motivator love active keep_moving track
T42	band wristband wrist month broke large week small replacement wrist_band
T43	customer service customer_service support email day replacement contacted problem product
T44	fitness pal fitness_pal myfitnesspal app syncs sync love apps mfp
T45	habit active sleeping healthy helped change pattern medium sleep care
T46	love recommend thing color recommend_anyone day purchased band love_love
T47	food log intake calorie activity water sleep track exercise app
T48	alarm silent wake silent_alarm set feature sleep vibrating clock morning
T49	easy easy_use set love wear comfortable great easy_set app super
T50	device data activity information tracking software make give time interface

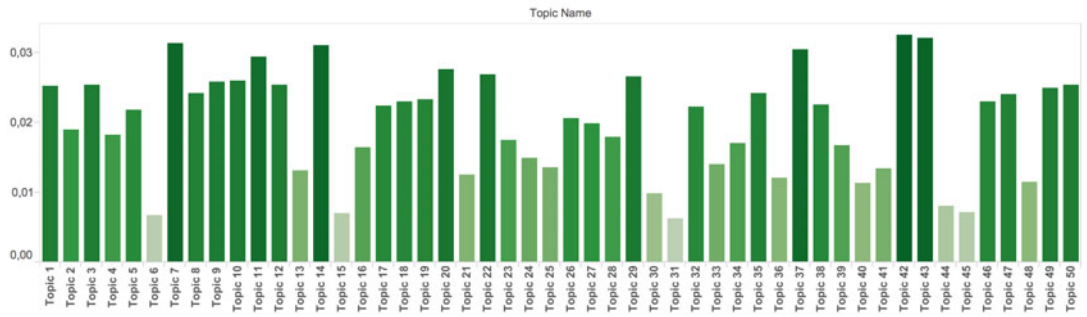

Figure 9. Overall Topic Distribution

The final step of the modeling phase was to quantify the influence of the identified topics (independent variables) on user satisfaction (dependent variable). For this, different regression analysis techniques can be applied. The most common choice would be using a linear ordinary least squares (OLS) regression; however, the star rating is measured on an ordinal, not on a continuous scale. Consequently, ordered logistic regression would be a better choice. Yet, testing the proportional odds assumption of ordered logistic regression against our dataset showed that the topics' influence on star rating varied between levels of star rating—a consequence of the J-shaped distribution of user satisfaction. Hence, we decided to use multinomial logistic regression, which treats the different levels of our dependent variable (i.e., 1, 2, 3, 4, 5 stars) as unordered categories. Consequently, it produces separate coefficients for each level of the dependent variable; in our example 5 coefficients for each of the 50 topics (i.e., 250 coefficients). In order to manage the complexity of the resulting model, and to increase its interpretability, we chose LASSO (Least Absolute Shrinkage and Selection Operator) to fit the model to the data. LASSO is a linear regression method that performs variable selection by shrinking the coefficients of uninfluential independent variables to exactly zero, thereby producing a model that only includes the most important independent variables for explaining the dependent variable (Hastie, Tibshirani, & Friedman, 2013).

Figure 10 visualizes the coefficients of the LASSO regression model.[3] For example, the analysis revealed that the top-5 topics associated with a 5-star rating include: Topic 46 ("recommend to others"), Topic 10 ("motivation to move"), Topic 3 ("losing weight"), Topic 35 ("competing with friends"), and Topic 49 ("easy to use"). In contrast, the top-5 topics associated with a 1-star rating include: Topic 39 ("negative cost/benefit ratio"), Topic 18 ("Amazon's product return policy"), Topic 20 ("malfunction"), Topic 43 ("customer service"), and Topic 8 ("operating instructions"). The goodness-of-fit of the estimated model, as measured by the fraction of deviance explained by the model, amounts to 0.26 and its classification accuracy to 0.57.

[3] In order to select the most appropriate lambda parameter for the lasso penalty, we have performed a grid search using 10-fold cross validation resulting in λ=0.00021.

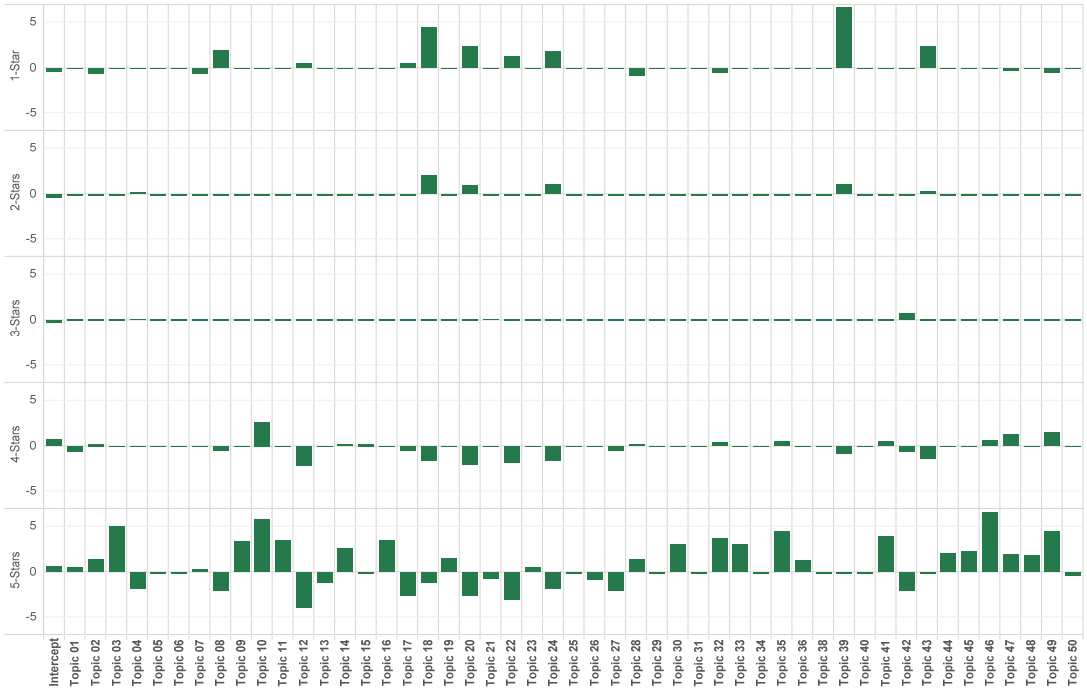

Figure 10. Coefficients of LASSO Multinomial Logistic Regression

4.4 Interpretation

The last step consisted of understanding and making sense of the discovered topics and their influence on user satisfaction. The meaning of a topic can be uncovered by analyzing its most probable terms in combination with the associated most probable documents. Figure 11 shows a bubble chart of the word distribution of Topic 3. The size and color of the bubbles both represent the probability of a given term in a given topic. A first labeling of this topic may yield “losing weight.” However, in order to verify whether the initial interpretation based on word probabilities makes sense, a thorough investigation of the associated documents is recommended. Table 5 confirms that customers are happily reporting their success stories about losing weight with the help of their Fitbit device. Overall, two researchers independently interpreted and labeled all 50 topics and, apart from minor wording differences, reached an inter-coder agreement of 84 percent.

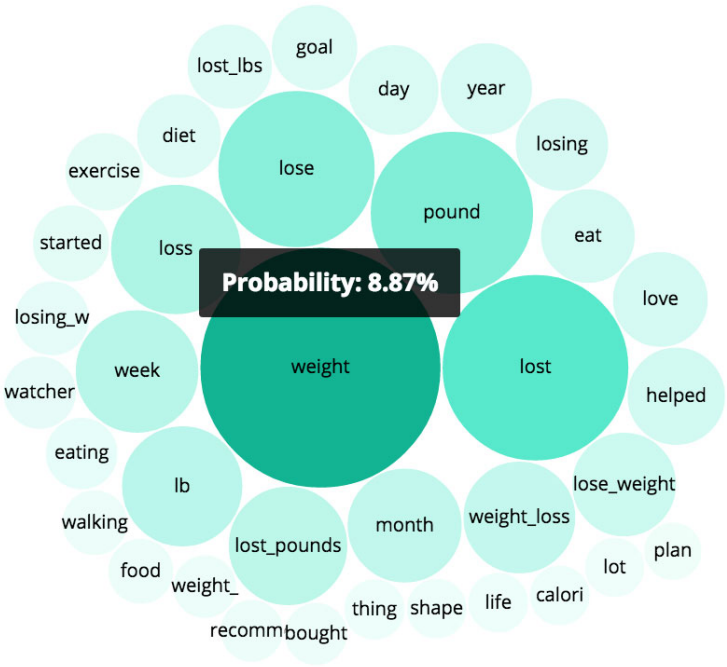

Figure 11. Most Probable Words of Topic 3 ("losing weight")

Table 5. Reviews Most Strongly Associated with Topic 3 ("losing weight")

Probability	Text	Star Rating
0.81	Helped me reach my weight loss goals and maintain my weight for 6 weeks now. Has become part of me 24/7.	5
0.78	The FitBit has helped my make a total lifestyle change. I've lost 27 pounds so far, and counting.	5
0.78	I AM TRYING TO LOSE WEIGHT FOR 3 EVENTS THIS SUMMER. THE FITBIT HAS HELPED WITH MY MOTIVATION A LOT. I HAVE NOT LOST A LOT OF WEIGHT. BUT WHEN SOMEONE SEES ME AND THEY HAVE NOT SEEN ME FOR A WHILE THEY SAY "YOUR GETTING SKINNY". I WOULD RECOMEND THIS ITEM TO ANYONE WANTING TO GET MOTIVATION TO MOVE AND GET FIT.	5
0.76	I love mine. It helped me lose 20 lbs.	5
0.72	I bought this at the end of July 2014. I wanted to track what I was eating, loose about 15 pounds, and increase my general fitness. I will be 60 years old in May. I have changed my lifestyle, I eat less food and things that are more healthy. I have lost about 33 pounds as I just kept going. I feel better than I have in 20 years, my general fitness is outstanding now since I have paid attention. The Fitbit Flex has been an integral part of my program. It's an outstanding value to accomplish all of this for me. The only problem, I need a lot of new clothes, my waist size decreased 4 inches, in a little over 3 months. I walk very briskly twice a day for my exercise. If you will learn how to use the Fitbit, it is really quite simple and very worthwhile.	5

In order make sense of the discovered topics against the background of existing theory, we tried to map the interpreted topics to theoretical constructs of theories from the field of technology acceptance, in particular the Technology Acceptance Model (TAM) (Venkatesh & Davis, 2000; Venkatesh, Morris, Davis, & Davis, 2003) and the IS Success Model (e.g., DeLone & McLean, 2003). Similar to the interpretation of topics through labeling, two researchers performed a mapping between topic and construct independently based upon a list of theoretical definitions (Table 7). The coders reached an inter-coder agreement of 86 percent.

Table 6. Definitions of Constructs Related to User Satisfaction

Construct	Sub-construct	Definition	Source
System Quality		Desirable characteristics of an information system.	Petter, DeLone and McLean 2013
	Reliability	Dependability of the system operation	Wixom and Todd 2005
	Flexibility	The way the system adapts to changing demands of the user	Wixom and Todd 2005
	Integration	The way the system allows data to be integrated from various sources	Wixom and Todd 2005
	Accessibility	The ease with which information can be accessed or extracted from the system	Wixom and Todd 2005
	Timeliness	Degree to which the system offers timely responses to requests for information or action	Wixom and Todd 2005
Information Quality		Desirable characteristics of the system outputs (content, reports, dashboards).	Petter, DeLone and McLean 2013
	Completeness	Degree to which all possible states relevant to the user population are represented in the stored information	Nelson, Todd, and Wixom 2005
	Accuracy	Degree to which information is correct, unambiguous, meaningful, believable, and consistent	Nelson, Todd, and Wixom 2005
	Format	Degree to which information is presented in a manner that is understandable and interpretable to the user and thus aids in the completion of a task	Nelson, Todd, and Wixom 2005
	Currency	Degree to which information is up-to-date, or the degree to which the information precisely reflects the current state of the world that it represents	Nelson, Todd, and Wixom 2005
Service Quality		The quality of support that system users receive from the IS department and IT support	Petter, DeLone and McLean 2013
Net Benefits		Extent to which an information system is contributing to the success of individuals, groups, organizations, industries, and nations.	Petter, DeLone and McLean 2013
Usefulness		Degree to which an individual thinks that using a particular system would enhance his or her job performance	Davis, Bagozzi and Warshaw 1989
Ease of Use		Degree to which an individual things that using a particular system would be free of effort	Davis, Bagozzi and Warshaw 1989
User Satisfaction		Users' level of satisfaction with the information system	Petter, DeLone and McLean 2013

Table 6 shows a sample result of the mapping process. Most of the topics were unambiguously mapped with extant constructs. For example, "losing weight" (Topic 3) was mapped with the construct of net benefits, as losing weight seems to be an indirect consequence—through increased physical activity—of using the Fitbit Flex device. And customer comments about the "accuracy of activity tracking" (Topic 4) were mapped with the construct of accuracy, a sub-construct of information quality construct in the IS Success Model, and "Amazon's product return policy" (Topic 18) with service quality. With regards to TAM, we were able to map Topic 49 "easy to use" with the ease of use construct, and the Topic 1 "step tracking" with usefulness, as it represents one of the core features of the device.

Overall, of the 50 topics identified, we mapped 14 with system quality, 12 with usefulness, 6 with net benefits, 3 with information quality, 2 with service quality, and 2 with ease of use—all being antecedents of user satisfaction. Some of the remaining topics were classified as indictors, rather than antecedents, of user satisfaction (Topics 11, 33, and 46).[4] In addition, we discovered eight topics that neither corresponded to constructs of the IS Success Model nor TAM. For instance Topic 39 "negative cost/benefit" or Topic 31 "comparison with competitor products" do not have a theoretical equivalent in either of the two models. This may give rise to extend the existing theories or to develop entirely new theories—two goals that are beyond the purpose of this tutorial.

Table 7. Definitions of Constructs Related to User Satisfaction

Topic	Most probable terms	Exemplary highly associated review sentences	Label	Mapping to existing constructs
T1	day step week work walk time steps_day walking couple end	I'm averaging about 12,350 steps a day. I look forward to the day when I can use it on my early and late night runs. When I first got it I was lucky to get 4000 steps a day because most of my job is at a desk. I had to really work to get to the 10k target. I've had it now for a couple of months and have increased my target to 12k daily.	Step tracking	Usefulness (Venkatesh & Davis, 2000)
T3	weight lost pound lose loss week lb lost_pounds month weight_loss lose_weight helped love eat losing year day goal lost_lbs	Helped me reach my weight loss goals and maintain my weight for 6 weeks now. Has become part of me 24/7. The FitBit has helped my make a total lifestyle change. I've lost 27 pounds so far, and counting.	Losing weight	Net Benefits (DeLone & McLean, 2003)
T4	minute active activity walking step mile running active_minutes run track exercise accurate record bike register weight measure treadmill hour	I like the fitbit and enjoy seeing my steps on the rise. However, I often use the elliptical or the bicycle and it does not record that activity. Only works with walking. That's disappointing to me. Just today walked with fit bit on a GPS measured 3 mile trail. Took me 48 minutes. Fit bit registered 2.5 miles and 23 minutes of activity. So it's ok if 50% accuracy is acceptable to you.	Accuracy of activity tracking	Information Quality -> Accuracy (DeLone & McLean, 2003)
T9	gift love christmas bought husband daughter received birthday gave present son day christmas_gift purchased year bought_husband sister mother loved	I bought it as a Christmas present for my brother in law and he loves it. I gave the item as a gift. I think she likes it as much as I do mine, that I received as a gift.	Fitbit as a gift	No corresponding IS construct identified

[4] Removing these three topics from the explanatory regression model only slightly reduced its goodness-of-fit and predictive accuracy (fraction of deviance explained: 0.2440, classification accuracy: 0.5596).

Table 7. Definitions of Constructs Related to User Satisfaction

T10	motivated move love day walk make step moving motivate keeps_motivated	It has made me much more aware that I need to move more during the day. It has helped me get more fit. Love it! Really motivates you to get up and get moving! Looking forward to getting a lot of use out of it!	Motivation to move	Net Benefits (DeLone & McLean, 2003)
T12	charge charger charging hold month unit battery issue hold_charge problem not_charge not_hold time charged usb work not_hold_charge light contact	Worked for about six months before battery refused to take a charge. After many emails, the mfr did send a new one. Now, six months later, the same thing - battery will not charge. Love my Fitbit. However, 2.5 month in and I'm having major battery charging issues. Hoping to get resolved ASAP.	Battery charging issues	System Quality -> Reliability (DeLone & McLean, 2003)
T18	return amazon day product item week charge worked purchased happy refund returned disappointed policy replacement bought purchase buy warranty	The fitbit won't charge and amazon won't accept returns after 30 days. This is the second fitbit received after the first one also didn't charge. This item only worked for less than 90 days, want to return to amazon for replacement, not allowing a return.	Amazon's product return policy	Service Quality (DeLone & McLean, 2003)
T31	jawbone nike band fuel app fuelband nike_fuel accurate fuel_band wanted	Bought this and Jawbone simultaneously. This is much better on performance than Jawbone.Its app is better and the blue tooth connectivity helps. I've owned a Nike+ band and wore it for almost a year. The flex is smaller, has interchangeable bands (if you can find them) and has a ton of more features over Nike's band.	Comparison to competitor products	No corresponding IS construct identified
T39	money worth waste time waste_money not_worth piece pedometer product buy worth_money thing save work expensive junk not_waste spent disappointed	I repeat do not waste your money :(Do not waste your money. Unless you have money to throw away. way too expensive for what it can do and for how inaccurate it is. I can get the same thing for free or real cheap	Negative cost/benefit ratio	No corresponding IS construct identified
T43	customer service customer_service support email day replacement contacted problem product	Customer service is awful. Defective product, and the Fit Bit company makes you jump through so many hoops to repair or replace a $100 product, that it hopes you just give up. I still do not have a resolution to my complaint over the defective product. Awful customer service and experience. Good luck getting a refund or replacement.	Customer service	Service Quality (DeLone & McLean, 2003)

Table 7. Definitions of Constructs Related to User Satisfaction

T49	easy easy_use set love wear comfortable great easy_set app super accurate simple setup easy_wear make dashboard work comfortable_wear band	Great for accountability. Easy to set up and use. I got it for my bestfriend and she loves it! She said it was extremely easy to set up and wears well with just about anything.	Easy to use	Ease of Use (Venkatesh & Davis, 2000)

4.5 Summary

Our illustrative topic modeling study showcases how open and naturally occurring text data can be used to explain customer satisfaction of a given product in a fully data-driven, inductive and largely automated manner. We collected more than 12,900 online customer reviews about the "Fitbit Flex" wearable technology from Amazon and applied the LDA topic modeling algorithm to extract independent variables for building an explanatory statistical model of user satisfaction. We were able to map numerous inductively identified topics to existing theoretical constructs and put them in a nomological network, which we then analyzed with a LASSO multinomial logistic regression. The results revealed that aspects of net benefits and perceived usefulness have the strongest influence on positive user satisfaction (4 and 5 stars), whereas poor system and service quality have the strongest influence on negative user satisfaction (1-star ratings). Furthermore, we have identified explanatory factors, such as "negative cost/benefit ratio," which are not part of existing IS theories on technology acceptance.

5 Conclusion

In this tutorial, we have discussed challenges of text mining, in particular topic modeling, and showcased its application by means of an illustrative example. Fellow researchers may use this tutorial as a blueprint and example for their own topic modeling studies, or to judge the quality of others.

Text mining methods provide a wide range of tools for analyzing large amounts of diverse texts with reasonable assumptions and costs, thereby allowing IS researchers to tap into new data sources, which were largely inaccessible before. However, despite all advances that have been made in natural language processing and machine learning over the last decade, these statistical techniques make use of simplified models in order to handle the complexity of natural language, and are far from replicating the process of how humans assign meaning to language. For example, most topic models treat texts as unordered sets of words, completely ignoring word order or sentence structure. Furthermore, just because it has been demonstrated that topic modeling delivers high quality results on some data sets, does not automatically mean that it performs well on every data set. If, for example, the overall text collection is small (e.g., open-ended questions from a survey), very broad in scope (e.g., e-mails), noisy (e.g., texts scrapped from websites), or the individual documents are quite short (e.g. Tweets), topic modeling may fail to produce insightful results. Hence, evaluating the validity of topic modeling results through experimentation and triangulation is essential. After all, text mining methods like topic modeling cannot replace human analysis, but only augment it.

In this tutorial, we have introduced only one text mining technique: topic modeling. Depending on the research goal, applying other techniques may be more suitable. Due to its unsupervised nature, topic modeling is especially suited to inductively discover patterns in large text collections. Particularly for exploratory studies in fields that are scarce of constructs and theory, or for extending existing theory, this approach might be useful. If, in contrast, the object of the study is rather confirmatory, dictionary-based methods may be more suitable. With dictionary-based methods, a researcher can carefully generate dictionaries and rules in order to fit a model to a set of predefined testable hypotheses; their exploratory potential, however, is very limited.

Finally, in this tutorial we tried to present two complex statistical methods (i.e., LDA and LASSO) in an easy-to-understand way for a broad audience. Researchers interested in the application of LDA or LASSO are advised to thoroughly work through the original literature in order to gain a deeper understanding of the methods before interpreting their outputs.

References

Berente, N., & Seidel, S. (2014). Big Data & Inductive Theory Development: Towards Computational Grounded Theory? In Proceedings of the 20th Americas Conference on Information Systems (pp. 1–11). Savannah.

Berg, B. L., & Lune, H. (2011). Qualitative Research Methods for the Social Sciences. Boston: Pearson.

Blei, D. (2012). Probabilistic topic models. Communications of the ACM, 55(4), 77–84.

Blei, D., Ng, A., & Jordan, M. (2003). Latent dirichlet allocation. Journal of Machine Learning Research, 3(1), 993–1022.

Boyd-Graber, J., Mimno, D., & Newman, D. (2014). Care and Feeding of Topic Models: Problems, Diagnostics, and Improvements. In E. M. Airoldi, D. Blei, E. A. Erosheva, & S. E. Fienberg (Eds.), Handbook of Mixed Membership Models and Their Applications (pp. 3–34). Boca Raton: CRC Press.

Business Wire. (2010). 2010 Social Shopping Study Reveals Changes in Consumers' Online Shopping Habits and Usage of Customer Reviews. Retrieved from http://www.businesswire.com/news/home/20100503005110/en/2010-Social-Shopping-Study-Reveals-Consumers'-Online

Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C., & Blei, D. (2009). Reading tea leaves: How humans interpret topic models. In Proceedings of the Advances in Neural Information Processing Systems Conference (pp. 1–9). Vancouver.

DeLone, W. H., & McLean, E. R. (2003). The DeLone and McLean Model of Information Systems Success : A Ten-Year Update. Journal of Management Information Systems, 19(4), 9–30.

Evangelopoulos, N., Zhang, X., & Prybutok, V. R. (2012). Latent Semantic Analysis: five methodological recommendations. European Journal of Information Systems, 21(1), 70–86.

Fan, W., Wallace, L., Rich, S., & Zhang, Z. (2006). Tapping the power of text mining. Communications of the ACM, 49(9), 76–82.

Firth, J. R. (1957). A synopsis of linguistic theory 1930-1955. In Studies in Linguistic Analysis (pp. 1–32). Oxford: Philological Society.

Frawley, W., Piatetsky-Shapiro, G., & Matheus, C. (1992). Knowledge discovery in databases: An overview. AI Magazine, 13(3), 57–70.

Gopal, R., Marsden, J. R., & Vanthienen, J. (2011). Information mining - Reflections on recent advancements and the road ahead in data, text, and media mining. Decision Support Systems, 51(4), 727–731.

Grimmer, J., & Stewart, B. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. Political Analysis, 21(3), 1–31.

Halevy, A., Norvig, P., & Pereira, F. (2009). The unreasonable effectiveness of data. IEEE Intelligent Systems, 24(2), 8–12.

Harris, Z. (1954). Distributional Structure. Word, 10(23), 146–162.

Hastie, T., Tibshirani, R., & Friedman, J. (2013). The elements of statistical learning. New York: Springer.

Hofmann, T. (1999). Probabilistic Latent Semantic Indexing. In Proceedings of the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 50–57). Berkeley.

Hu, N., Zhang, J., & Pavlou, P. a. (2009). Overcoming the J-shaped distribution of product reviews. Communications of the ACM, 52(10), 144-147.

Indulska, M., Hovorka, D. S., & Recker, J. (2012). Quantitative approaches to content analysis: identifying conceptual drift across publication outlets. European Journal of Information Systems, 21(1), 49–69.

Kurgan, L. A., & Musilek, P. (2006). A survey of Knowledge Discovery and Data Mining process models. The Knowledge Engineering Review, 21(1), 1–24.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. Discourse Processes, 25(2-3), 259–284.

Lau, J. H., Newman, D., & Baldwin, T. (2014). Machine Reading Tea Leaves: Automatically Evaluating Topic Coherence and Topic Model Quality. In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, 530–539.

Liu, B. (2011). Web Data Mining. Berlin: Springer.

Martens, D., & Provost, F. (2014). Explaining data-driven document classifications. MIS Quarterly, 38(1), 73–99.

McAuley, J., Pandey, R., & Leskovec, J. (2015). Inferring Networks of Substitutable and Complementary Products. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining Pages. Sydney.

McAuley, J., Targett, C., Shi, Q., & van den Hengel, A. (2015). Image-based Recommendations on Styles and Substitutes. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. Santiago.

Michel, J.-B., Shen, Y. K., Aiden, A. P., Veres, A., Gray, M. K., Pickett, J. P., … Aiden, E. L. (2011). Quantitative analysis of culture using millions of digitized books. Science, 331(6014), 176–182.

Miles, M., & Huberman, A. (1994). Qualitative data analysis: An expanded sourcebook. Thousand Oaks: Sage Publications, Inc.

Mimno, D., Wallach, H., Talley, E., Leenders, M., & McCallum, A. (2011). Optimizing semantic coherence in topic models. In Proceedings of the Conference on Empirical Methods in Natural Language Processing. Stroudsburg.

Miner, G., Elder, J., Hill, T., Nisbet, R., Delen, D., & Fast, A. (2012). Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications. Waltham: Academic Press.

Mudambi, S. M., & Schuff, D. (2010). What Makes a Helpful Online Review? A Study of Customer Reviews on Amazon.com. MIS Quarterly, 34(1), 185–200.

Müller, O., Junglas, I., vom Brocke, J., & Debortoli, S. (2016). Utilizing Big Data Analytics for Information Systems Research: Challenges, Promises and Guidelines. European Journal of Information Systems, forthcoming.

Newman, D., Lau, J., Grieser, K., & Baldwin, T. (2010). Automatic evaluation of topic coherence. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics. Los Angeles.

OKF. (2012). Open Data Handbook Documentation. Retrieved from http://opendatahandbook.org/pdf/OpenDataHandbook.pdf

Ott, M., Choi, Y., Cardie, C., & Hancock, J. T. (2011). Finding Deceptive Opinion Spam by Any Stretch of the Imagination. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, 2(1-2), 1–135.

Pennebaker, J. W. (2011). The Secret Life of Pronouns: What Our Words Say About Us. New York: Bloomsbury Press.

Quinn, K. M., Monroe, B. L., Colaresi, M., Crespin, M. H., & Radev, D. R. (2010). How to analyze political attention with minimal assumptions and costs. American Journal of Political Science, 54(1), 209–228.

Ramage, D., Rosen, E., Chuang, J., Manning, C. D., & McFarland, D. A. (2009). Topic Modeling for the Social Sciences. In Proceedings of the Workshop on Applications for Topic Models. Whistler.

Saldaña, J. (2012). The Coding Manual for Qualitative Researchers. London: Sage Publications, Inc.

Salton, G., & McGill, M. (1983). Introduction to Modern Information Retrieval. New York: McGraw-Hill.

Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. Journal of Data Warehousing, 5(4), 13–22.

Smith, A. E., & Humphreys, M. S. (2006). Evaluation of unsupervised semantic mapping of natural language with Leximancer concept mapping. Behavior Research Methods, 38(2), 262–279.

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

Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. (2010). Sentiment in short strength detection informal text. Journal of the American Society for Information Science and Technology, 61(12), 2544–2558.

Turney, P., & Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. Journal of Artificial Intelligence Research, 37(1), 141–188.

Twitter. (2015). Twitter Usage and Company Facts. Retrieved October 30, 2015, from https://about.twitter.com/company

Urquhart, C. (2001). An Encounter with Grounded Theory: Tackling the Practical and Philosophical Issues. In E. M. Trauth (Ed.), Qualitative Research in Information Systems: Issues and Trends (pp. 104–140). Hershey: Idea Group Publishing.

Urquhart, C. (2012). Grounded theory for qualitative research: A practical guide. Sage Publications, Inc.

Venkatesh, V., & Davis, F. D. (2000). A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies. Management Science, 46(2), 186–204.

Venkatesh, V., Morris, M., Davis, G., & Davis, F. (2003). User acceptance of information technology: Toward a unified view. MIS Quarterly, 27(3), 425–478.

Wallach, H., Mimno, D., & McCallum, A. (2009). Rethinking LDA: Why priors matter. In Proceedings of the Conference on Neural Information Processing Systems. Vancouver.

Wallach, H., Murray, I., Salakhutdinov, R., & Mimno, D. (2009). Evaluation methods for topic models. In Proceedings of the International Conference on Machine Learning. Montreal.

About the Authors

Stefan Debortoli is a research assistant and Ph.D. candidate at the Institute of Information Systems at the University of Liechtenstein. His doctoral studies focus on applying Big Data Analytics as a new strategy of inquiry in Information Systems Research. Within the field of Big Data Analytics, he focuses on the application of text mining techniques for research purposes. His work has been published in the European Journal of Information Systems, Communications of the Association for Information Systems, and Business & Information Systems Engineering. Stefan has several years of working experience in the field of software engineering and IT project management across various industries.

Oliver Müller is an Associate Professor at the IT University of Copenhagen. He holds a BSc and MSc in Information Systems and a Ph.D. in Economics from the University of Münster, Germany. The goal of Oliver's research is to help organizations and individuals to create value through big data and analytics. At this, he particularly focuses on extracting knowledge from large amounts of unstructured text data, from both the Internet and enterprise-internal sources. His research has been published in the European Journal of Information Systems, Journal of the Association for Information Systems, Communications of the Association for Information Systems, Computers & Education, and others.

Iris Junglas is an Associate Professor for Information Systems at Florida State University. Her research interest captures a broad spectrum of topics, most prominent are the areas of E-, M- and U-Commerce, health-care information systems, the consumerization of IT and business analytics. Her research has been published in the European Journal of Information Systems, Information Systems Journal, Journal of the Association of Information Systems, Management Information Systems Quarterly, Journal of Strategic Information Systems, and various others. She serves on the editorial board of the Management Information Systems Quarterly Executive and the Journal of Strategic Information Systems and is also a senior associate editor for the European Journal of Information Systems.

Jan vom Brocke is Professor for Information Systems at the University of Liechtenstein. He is the Hilti Endowed Chair of Business Process Management, Director of the Institute of Information Systems, Co-Director of the International Master Program in Information Systems, Director of the Ph.D. Program in Business Economics, and Vice-President Research and Innovation at the University of Liechtenstein. In his research he focuses on digital innovation and transformation capturing business process management, design science research, Green IS, and Neuro IS, in particular. His research has been published in Management Information Systems Quarterly, Journal of Management Information Systems, Business & Information Systems Engineering, Communications of the Association for Information Systems, Information & Management and others. He is author and editor of seminal books, including the International Handbook on Business Process Management as well as the book BPM – Driving Innovation in a Digital World. He has held various editorial roles and leadership positions in Information Systems research and education.