High-Reproducibility and High-Accuracy Method for Automated Topic Classification

Andrea Lancichinetti,[1,2] M. Irmak Sirer,[2] Jane X. Wang,[3] Daniel Acuna,[4]
Konrad Körding,[4] and Luís A. Nunes Amaral[1,2,5,6]

[1]*Howard Hughes Medical Institute (HHMI), Northwestern University, Evanston, Illinois 60208, USA*
[2]*Department of Chemical and Biological Engineering, Northwestern University, Evanston, Illinois 60208, USA*
[3]*Department of Medical Social Sciences, Northwestern University, Feinberg School of Medicine, Chicago, Illinois 60611, USA*
[4]*Department of Physical Medicine and Rehabilitation, Rehabilitation Institute of Chicago, Northwestern University, Chicago, Illinois 60611, USA*
[5]*Department of Physics and Astronomy, Northwestern University, Evanston, Illinois 60208, USA*
[6]*Northwestern Institute on Complex Systems, Northwestern University, Evanston, Illinois 60208, USA*

(Received 21 May 2014; published 29 January 2015)

Much of human knowledge sits in large databases of unstructured text. Leveraging this knowledge requires algorithms that extract and record metadata on unstructured text documents. Assigning topics to documents will enable intelligent searching, statistical characterization, and meaningful classification. Latent Dirichlet allocation (LDA) is the state of the art in topic modeling. Here, we perform a systematic theoretical and numerical analysis that demonstrates that current optimization techniques for LDA often yield results that are not accurate in inferring the most suitable model parameters. Adapting approaches from community detection in networks, we propose a new algorithm that displays high reproducibility and high accuracy and also has high computational efficiency. We apply it to a large set of documents in the English Wikipedia and reveal its hierarchical structure.

DOI: 10.1103/PhysRevX.5.011007 Subject Areas: Interdisciplinary Physics

I. INTRODUCTION

The amount of data that we currently collect and store is unprecedented. A challenge for its analysis is that a significant fraction of these data is in the form of unstructured text. One of the central challenges in the field of natural language processing is bridging the gap between information in text databases and their organization within structured topics. Topic-classification algorithms are key to closing this gap.

Topic models take as input a set of text documents (the corpus) and return a set of topics that can be used to describe each document in the corpus. Topic models set the foundation for text-recommendation systems [1,2], digital image processing [3,4], computational biology analyses [5], spam filtering [6], and countless other modern-day digital applications. Because of their importance, there has been an extraordinary amount of research and a number of different implementations of topic-model algorithms [7–14].

At the core of every topic-model algorithm is the requirement to find the global maximum of a likelihood function characterized by numerous local maxima. This optimisation problem is also the challenge at the core of the study of disordered systems in physics [15,16]. Additionally, topic modeling is closely related to the problem of fitting stochastic block models to complex networks [17–25].

Surprisingly, even though it is well established that the problem of fitting topic models is computationally hard, little is known about how the vastness and roughness of the likelihood landscape impact algorithm performance in practice. In order to get a grasp on the magnitude of this challenge, we conduct a controlled analysis of topic-model algorithms for highly specified sets of synthetic data. This high degree of control allows us to tease apart the theoretical limitations of the algorithms from other sources of error that would remain uncontrolled with real-world data sets [26–28]. Our analyses reveal that standard techniques for likelihood optimization are significantly hindered by the roughness of the likelihood-function landscape, even for very simple cases. Significantly, we show that the limitations of the current implementations of topic-model algorithms can easily be overcome by using a network approach to topic modeling.

Our manuscript is organized as follows. In Sec. II, we present some background on topic models. Section II is followed, in Sec. III, by an investigation of the performance of topic models on an elementary test case. In Sec. IV, we introduce our new algorithm, which we denote as TopicMapping, and in Sec. V, we systematically compare

the performance of the state-of-the-art topic-model algorithms against that of TopicMapping.

II. BACKGROUND

The state-of-the-art methods in topic modeling attempt to fit the values of the parameters of a generative model of the documents in the corpus. The first major attempt to develop a generative model was probabilistic latent semantic analysis (PLSA) [9]. The current gold standard for the field is latent Dirichlet allocation (LDA) [10,29,30]. For the first time, topic models offered a principled approach for clustering text documents, with a well-defined set of assumptions. This approach spurred a consistent body of research aimed at generalizing the models and relaxing their assumptions.

The generative models underlying the PLSA and LDA algorithms assume that each topic is characterized by a specific word-usage probability distribution and that every document in the corpus is a generated from a mixture of topics. As an example, consider a corpus of documents generated from two topics, mathematics and biology (Fig. 1). Each document in the corpus will draw from the set of topics with idiosyncratic probabilities. For instance, a document d_{bio} drawing mostly from the biology topic might have $p(\text{topic} = \text{biology}|d_{\text{bio}}) = 0.9$ and $p(\text{topic} = \text{math}|d_{\text{bio}}) = 0.1$.

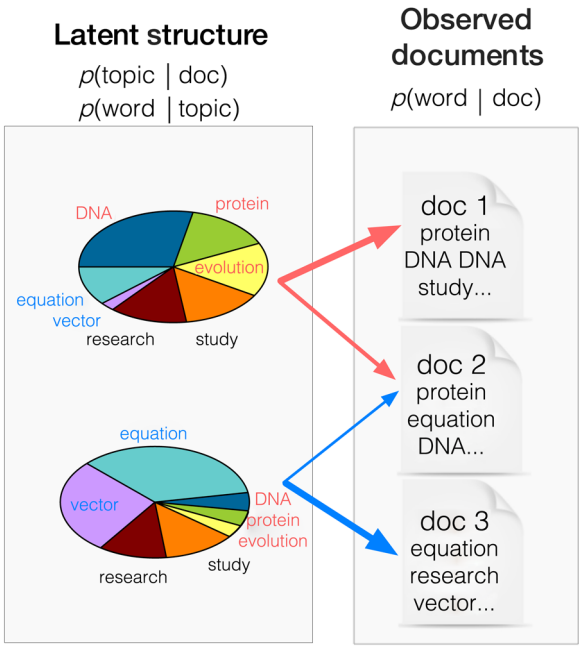

FIG. 1. Generative model for documents in the corpus. Documents are assumed to be a mixture of topics. The topic structure is latent, meaning that one cannot access the "true" set of topics used to generate the documents in the corpus. However, the set of topics can be estimated using a topic-model algorithm. To this end, one calculates the word frequencies in each document and models them as mixtures of different topics, math and biology in this example.

Documents with different topic mixtures will use words differently because the probability of using a given word depends on the topic. Importantly, it is assumed that some words will be strongly associated with a single topic; otherwise, it would be impossible to fit the model. For example, words such as "DNA" or "protein" will primarily be used in biology-focused documents because $p(\text{word} = \text{DNA}|\text{topic} = \text{biology}) \gg p(\text{word} = \text{DNA}|\text{topic} = \text{math})$. In contrast, words such as "tensor" or "equation" will primarily be used in a math-focused document because $p(\text{word} = \text{tensor}|\text{topic} = \text{biology}) \ll p(\text{word} = \text{tensor}|\text{topic} = \text{math})$. There will, however, be other words, such as "research" or "study," that are generic and will be used nearly equally by both topics. In practice, one only has access to the word counts in each document, while the actual topic structure is unobservable, that is, latent. The challenge is thus to estimate the topic structure, which is defined by the set of probabilities $p(\text{topic}|\text{doc})$ and $p(\text{word}|\text{topic})$.

For concreteness, let us assume that a corpus comprised of N documents is generated from K topics using N_w distinct words. Then, one needs to estimate $N \times K$ probabilities $p(\text{topic}|\text{doc})$ and $K \times N_w$ probabilities $p(\text{word}|\text{topic})$. PLSA and LDA both aim to estimate the values of these $K \times (N + N_w)$ probabilities that have the highest likelihood of having generated the data [9,10,31,32]. Thus, both PLSA and LDA rely on maximization of a likelihood that depends nonlinearly on a large number of variables, a non-deterministic polynomial-time hard problem [33].

A major difference between the two models is that for PLSA, the $N \times K$ probabilities $p(\text{topic}|\text{doc})$ are free parameters that must be estimated directly from the data, whereas LDA assumes that the set of probabilities $p(\text{topic}|\text{doc})$ is random variables that have been drawn from a Dirichlet distribution [34]. Thus, for LDA, one only needs to estimate K parameters (one per topic) $\{\alpha_1, \alpha_2, \ldots, \alpha_K\}$. These α's are called concentration parameters or hyperpriors. The number of parameters for LDA is often reduced further, by assuming that all hyperpriors take the same value, typically denoted by α. The "version" of LDA that assumes a single value of the hyperprior is called symmetric LDA, whereas the full model with the K hyperpriors is called asymmetric LDA [35].

III. EVALUATION OF STATE-OF-THE-ART TOPIC-MODEL ALGORITHMS

A. Theoretical limits on the performance of topic-model algorithms on an elementary test case

Sophisticated practitioners know that a very large number of topic models can fit the same data almost equally well. This model competition poses a serious challenge for algorithmic stability. We begin our investigation of this

issue by considering a well-defined elementary test case [36], which we denote as the language corpus. Here, topics are fully unmixed languages—that is, no word is used by more than one language—and each document is written entirely in a single language. In order to match this test corpus to the assumed LDA generative model, we use a two-step process to create synthetic documents. In the first step, we select a language with probability $p(\text{language})$, which, in practice, is equivalent to a Dirichlet distribution with a very small hyperprior parameter (see the Supplemental Material [37]). Given the language, in the second step, we randomly sample L_d words from that language's vocabulary into the document. For the sake of simplicity, we restrict the vocabulary of each language to a set of N_w unique equiprobable words. Thus, an "English" document in the language corpus is just a "bag" of English words.

For concreteness, consider a language corpus generated from three languages and a distinct number of documents in each language. Consider also that the number of documents in English more than exceeds the sum of the number of documents in the other two languages. Because of stochastic fluctuations in how words are assigned to documents generated from a single language, an implementation of a topic-model algorithm could correctly infer the three languages as topics, but it could also split English into two or more "dialects" and merge the two other languages into a single topic [Fig. 2(a)]. The latter, alternative model is wrong on two counts: It splits English-topic documents into two topics, and it assigns documents in the two smaller languages into a single topic. Naïvely, one would expect the incorrect alternative model to have a smaller likelihood than the correct generative model. However, this supposition is not always fulfilled for PLSA [9] or for symmetric LDA.

In fact, dividing, that is, overfitting, the English documents in the corpus yields an increase of the likelihood. As we show in the Supplemental Material [37], the log-likelihood of the alternative model increases by as much as $\log^2 2$ per English document. Similarly, merging, that is, underfitting, the "French" and "Spanish" documents, results in a decrease of the log-likelihood of $L_d \log 2$ per French and Spanish document, where L_d is the average length of the documents. By comparing these two opposing changes, one can identify a critical fraction of English documents above which the alternative model will have a greater likelihood than the correct generative model [Fig. 2(b)]. Moreover, numerical simulations demonstrate that topic-model algorithms fail before one hits the theoretical limit set by the critical fraction of English documents [Figs. 2(c) and 2(d)].

The theoretical limit for using maximization of a likelihood function in order to infer the correct generative model is not limited to topic modeling. Of particular significance, this limit also holds for non-negative matrix factorization [8] with Kullback-Leibler divergence, which is equivalent to PLSA [38]. (Non-negative matrix factorization is a popular low-rank-matrix-approximation algorithm that has found countless applications, for example, in face recognition, text mining, and many other high-dimensional data-analysis applications.)

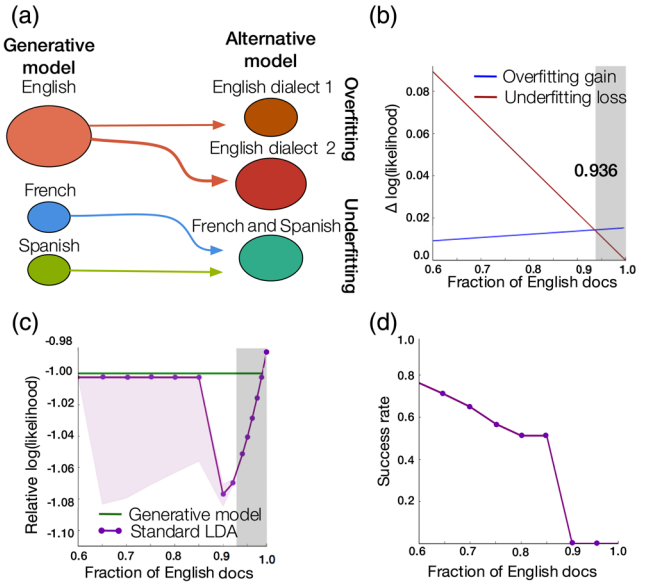

FIG. 2. The language corpus. (a) We generate synthetic documents for a corpus where each document is a collection of words from one of three languages or topics: English, French, and Spanish. Even though the corpus is generated from three topics, if one of the topics (in this example, English) is much more highly represented in the corpus than the others, a typical topic-model algorithm might assign English documents to two topics, while French and Spanish documents may be assigned to a single topic. (b) Consider the case where each language has a vocabulary of 20 unique equiprobable words and each document comprises 10 words. For these parameters' values, the alternative model has a larger log-likelihood (for PLSA or symmetric LDA) if the fraction of English documents in the corpus is greater than 0.936. (c) Performance of the symmetric LDA algorithm, assuming that $K = 3$ topics are used to generate the documents in the corpus. The solid line and points indicate the median change in log-likelihood of the model inferred by the algorithm, and the shaded area delimits the 25th and 75th percentiles. Note that, in practice, the LDA algorithm does not infer the correct generative model (green curve) prior to the theoretical limit (gray shaded area). (d) Probability that symmetric LDA infers the correct generative model when setting $K = 3$.

More generally, the critical threshold for the fraction of documents belonging to the underfitted topic depends on the typical number of words L_d in the documents comprising the corpus. Specifically, the critical threshold decreases as $1/L_d$. In fact, by increasing the typical length of the documents or using asymmetric LDA [35], one can show, for the language corpus, that the generative model always has a larger likelihood than the alternative model (see the Supplemental Material [37]). The ratio of the log-likelihood of the alternative model and the generative model can be expressed as

$$\frac{\langle \log \mathcal{L}_{\rm alt} \rangle}{\langle \log \mathcal{L}_{\rm true} \rangle} \simeq 1 - \frac{f_U \log 2}{\log N_w}, \quad (1)$$

where $\mathcal{L}_{\rm alt}$ and $\mathcal{L}_{\rm true}$ are the likelihoods of the alternative and generative models, respectively, and f_U is the fraction of documents in the corpus belonging to the underfitted topic.

Even though the generative model has a larger likelihood, the ratio on the left-hand side of Eq. (1) can become arbitrarily close to 1. The reason is that the ratio is independent of the number of documents in the corpus and of the length of the documents. Thus, even with an infinite number of infinitely long documents, the generative model does not "tower" above other models in the likelihood landscape. The consequences of this fact are important because the number of alternative latent models that can be defined is extremely large—with a vocabulary of 1000 words per language, the number of alternative models is on the order of 10^{300} (see the Supplemental Material [37]). Thus, while our analysis shows that asymmetric LDA [35] assigns the largest likelihood to the correct generative model regardless of the documents' lengths, this result is countered by the fact that there is an extremely large number of incorrect models that will have likelihoods extremely close to that of the correct model.

Unlike symmetric LDA, asymmetric LDA does not assume that the hyperpriors all have the same values. The assumption of equal hyperpriors results, however, in a bias of the algorithm toward solutions in which all topics "want" to contain the same number of documents. While this bias vanishes for all methods (symmetric and asymmetric LDA as well as PLSA) if the documents contain a sufficiently large number of words, the problem is that the differences in likelihood remain very small, making the task of finding the global maximum extremely challenging.

B. Numerical analysis of the performance of topic-model algorithms on an elementary test case

Although the language corpus is a highly idealized case, it provides a clear example of the challenges posed by the existence of many competing models with nearly identical likelihoods. Indeed, due to the high degeneracy of the likelihood landscape, standard optimization techniques are unlikely to infer the model with the highest likelihood even in such simple cases and will likely infer different models for different optimization runs, as has been previously reported [11,35]. Moreover, because topics comprising a small fraction of documents are the hardest to resolve (see the Supplemental Material, Sec. 1.6 [37]), standard algorithms will require one to assume that there is an unrealistically large number of topics giving rise to the corpus because "extra topics" are needed in order to "resolve" topics with small fractions of documents.

We next test these hypotheses numerically on two synthetic language corpora. We denote the first corpus as egalitarian. This corpus is generated from ten languages, and each language is used to generate the same number of documents. We denote the second corpus as oligarchic. Again, this corpus is generated from ten languages, but now, two large topics are used to generate 30% of the documents each, while the other eight small topics are used to generate 5% of the documents each. For both corpora, we use the real-world word frequencies [39] of the languages.

In order to determine the validity of the models inferred by the algorithms under study, we calculate both the accuracy and the reproducibility of the algorithms' outputs (Fig. 3). We use a measure of normalized similarity (see Sec. VII) to compare the inferred model to the generative model (accuracy) and to compare the inferred models obtained from different optimization runs of the algorithm (reproducibility).

Our theoretical analysis shows that PLSA and symmetric LDA are unable to "detect" the existence of topics comprising a small fraction of documents. In the synthetic corpora that we now consider, the fraction of documents in every topic is outside the critical region. Thus, for these corpora, the generative model has the greatest likelihood with both PLSA and symmetric LDA. In order to provide a best-case scenario for their performances, we run the standard algorithms [9,10] with the number of topics in the generative model (as we show in the Supplemental Material [37], estimating that the number of topics via model selection would lead to a dramatic overestimation of the number of topics).

We find that PLSA and the standard optimization algorithm implemented with LDA (variational inference) [10] are systematically unable to find the global maximum of the likelihood landscape. As shown in Fig. 4, these algorithms have surprisingly low accuracy and reproducibility, especially when topic sizes are unequal [40]. Taken together, the results in this section clearly demonstrate that the approach taken by standard topic-model algorithms for

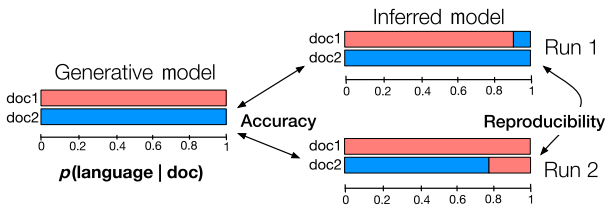

FIG. 3. Performance of an algorithm on the analysis of synthetic corpora. We define accuracy as the best-match similarity (see Sec. VII) among the fitted model and the generative model. Reproducibility is the similarity among fitted models obtained in different runs.

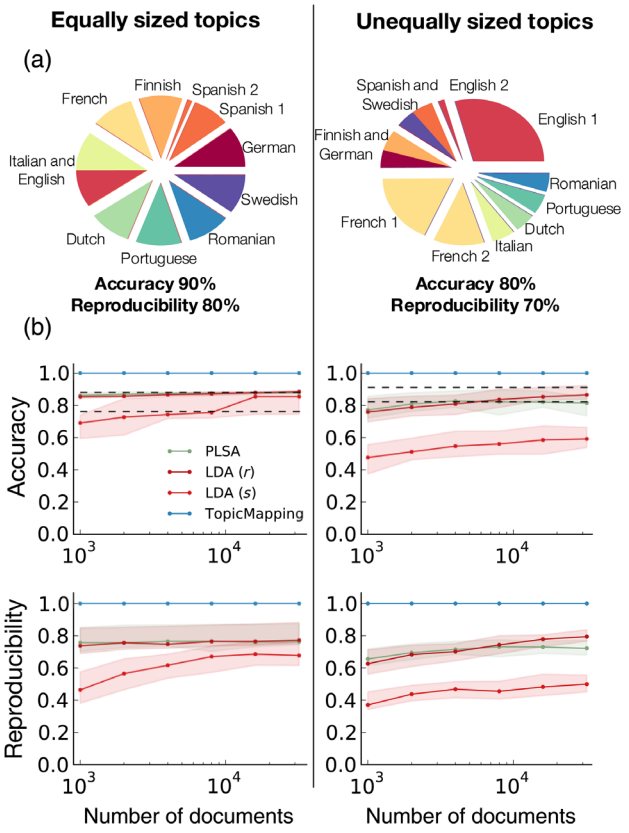

FIG. 4. Numerical evaluation of algorithm performance. We generate documents for the corpora according to a two-step process. First, we assign a language to the document. Next, we draw with replacement 100 words with the probability corresponding to their frequency in the chosen language. For simplicity, we use a vocabulary limited to the 1000 most frequently used words in the language and exclude words that are not unique to the language. When using LDA and PLSA algorithms, we assume that we already know the correct number of topics in the corpora. (a) Illustration of inferred topics using LDA standard optimization for corpora with the equally and unequally sized topics. Each "slice" in the pie charts represents the topic inferred by LDA for a set of documents. Different colors indicate the languages assigned to the documents in the generative model. In the equal-sized topics, English and Italian documents are assigned a single topic and Spanish documents are assigned to two different topics. In the unequal-sized topics, English and French documents are split into two topics each, whereas German and Finnish documents are assigned a single topic, as are Swedish and Spanish documents. (b) Reproducibility and accuracy of four topic-modeling algorithms for the language corpora. The dashed lines indicate the expected accuracy when overfitting one language and underfitting two other languages (top line) or when overfitting two languages and underfitting four (bottom line). The full lines show median values, and the shaded regions denote the 25th to 75th percentiles. LDA (r) and LDA (s) refer, respectively, to random and seeded initializations for the optimization technique.

exploring the likelihood landscape is extremely inefficient, whether one starts from random initial conditions or by randomly seeding the topics using a sample of documents (Fig. 4).

IV. A NETWORK APPROACH TO TOPIC MODELING

One can take a corpus and construct a bipartite network of words and documents, where a word and document are connected if the word appears in the document [41]. This bipartite network can be projected onto a unipartite network of words by connecting words that coappear in a document [42]. In the language corpora, separating documents using distinct languages is as trivial as finding the connected components of word network. For general corpora, however, inferring topics will be more difficult because words will likely be shared by multiple topics.

In order to tackle the greater difficulty in inferring topics when some words are shared by multiple topics, we propose a new approach involving four steps, which we denote as TopicMapping (Fig. 5). As we will see, the first two steps have the single purpose to denoise the word network [43].

Step 1: Preprocessing.—Many words in the English language stem from the same root. Without preprocessing, words such as "star" and "stars" would be viewed by an algorithm as distinct, as would different tenses of the same verb. In order to make the analysis more robust, we start by preprocessing the documents in a corpus using a stemming algorithm that replaces words by their stem [45]. Additionally, we remove a standard list of so-called "stop words," that is, words such as "the," "we," "from," "to," and so on that will not provide useful topic information.

Step 2: Pruning of connections.—We calculate the dot-product similarity [46] of each pair of words that coappear in at least one document and then compare it against the expectation for a null model where words are randomly shuffled across documents. We find that the distribution of

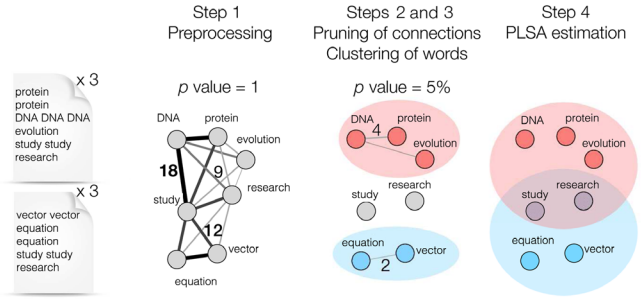

FIG. 5. Illustration of the TopicMapping algorithm. Step 1: Consider a corpus comprising six documents, three in the topic biology and three in the topic math. We exclude stop words from those documents and stem words in order to denoise the data. We then build a network connecting words with weights equal to their dot-product similarity. Steps 2 and 3: We filter nonsignificant weights, using a p value of 5%, and we run Infomap [44] to obtain the community structure of the network. In this case, we find two clusters and two isolated words (study and research). Step 4: We refine the word clusters using a topic model: The two isolated words are now assigned to both topics.

dot-product similarities of pairs of words for the null model is well approximated by a Poisson distribution whose average depends on the frequencies of the words in the pair (see the Supplemental Material [37]). We set a p value of 5% for determining whether the co-occurrence of the pair of words can be explained by the null model and retain only connections between pairs of words that appear more frequently than would be expected from the null model.

Step 3: Clustering of words.—We make the assumption that topics in the corpus will give rise to communities of words in the pruned unipartite word network. Under this assumption, one can use one of the many well-performing algorithms for community detection reported in the literature [25,47,48]. We choose here to use the Infomap algorithm developed by Rosvall and Bergstrom [44]. In contrast to the standard topic-modeling algorithms, community-detection algorithms determine the number of communities in the network in an unsupervised manner; that is, they do not require the user to guess the number of topics present in the corpus. We take the communities identified by Infomap as a guess for the number of topics and word composition of each of the topics used to generate the corpus.

Step 4: Topic-model estimation.—Because Infomap is an exclusive clustering algorithm—that is, words can belong to a single topic—we refine this first guess using one of the latent topic models that allow for nonexclusivity. For example, we locally optimize a PLSA-like likelihood in order to obtain our final estimate of model probabilities (see the Supplemental Material for more information [37]). One may potentially refine the estimation of the topic model for the corpus using asymmetric LDA likelihood optimization [10] and taking as the initial guess of the parameters the probabilities found in step 3 or the parameter estimates obtained with PLSA. In practice, we find that if the topics are not too heterogeneously distributed, the asymmetric LDA algorithm converges after only a few iterations, as our parameter estimation using PLSA is generally very close to a LDA likelihood maximum.

The numerical results displayed in Fig. 4 demonstrate that TopicMapping performs perfectly on the language test. This result comes as no surprise since the words belonging to different languages will never co-occur in a document, and thus, the nodes in the word network will organize into disconnected clusters. Clearly, this neat separation will not happen in more realistic cases. When projecting the bipartite network of documents and words onto a unipartite network of words, we will be throwing away information contained in the corpus. This projection will create particular difficulties when considering generic words, that is, words that appear in all documents regardless of topic, or words that are used by multiple topics. Generic words are easily handled by the first two steps of the TopicMapping pipeline. Words used by multiple topics will either be assigned to a single topic or be identified as separate communities by Infomap. However, using PLSA or LDA to refine model estimation enables us to recover that information. Finally, while Infomap is intrinsically stochastic, it converges to extremely similar solutions upon different runs [49]; thus, the TopicMapping algorithm yields extremely reproducible results.

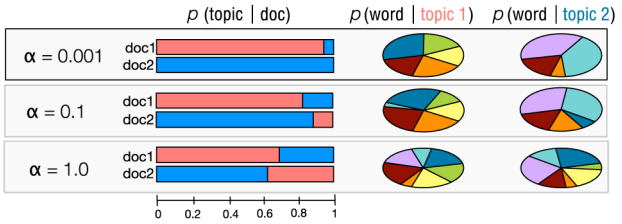

FIG. 6. Creating synthetic corpora using the generative model. For each document, $p(\text{topic}|\text{doc})$ is sampled from a Dirichlet distribution whose hyperparameters are defined as $\alpha_{\text{topic}} = K \times p(\text{topic}) \times \alpha$, where K is the number of topics, $p(\text{topic})$ is the probability (i.e., the size) of the word topic, and α is a parameter that tunes how mixed documents are: Smaller values of α yield a simpler model where documents make use of fewer topics. We also have a parameter to fix the fraction of generic words, and we implement a similar method for deciding $p(\text{word}|\text{doc})$ for specific and generic words (see Sec. VII). Once the latent topic structure is chosen, we create a corpus drawing words with probabilities given by the mixture of topics.

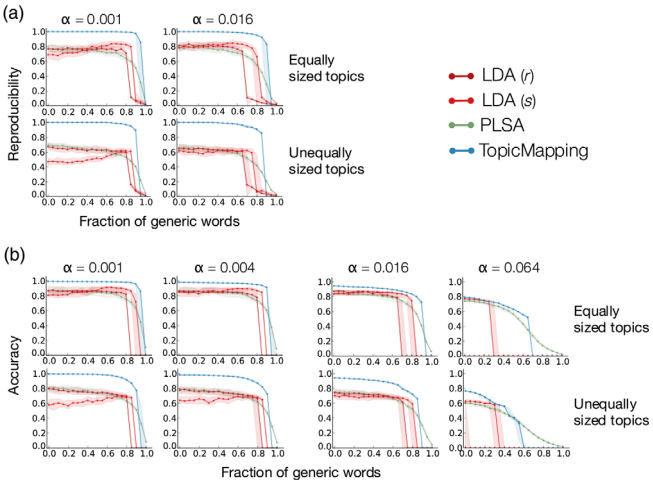

FIG. 7. Performance of topic-modeling algorithms on synthetic corpora. We plot the (a) reproducibility and (b) accuracy of the different algorithms. In all our tests, we generate a corpus of 1000 documents, of 50 words each, and our vocabulary is made of 2000 unique equiprobable words. We set the number of topics $K = 20$, and we input this number in LDA and PLSA. "Equally sized" means all the topics have equal probability $p(\text{topic}) = 5\%$, while in the "unequally sized" case, four large topics have probability 15% each, while the other 16 topics have probability 2.5%. LDA (s) and LDA (r) refer to seeded and random initializations for LDA (variational inference). The plots show the median values as well as the 25th and 75th percentiles.

V. RESULTS

We will now systematically test the validity of the TopicMapping algorithm and compare its performance against that of standard LDA optimization methods.

A. Synthetic corpora

In order to systematically evaluate the accuracy and reproducibility of the different algorithms, we must first develop a validation system. To this end, we implement a comprehensive generative model based on the assumptions behind LDA (Fig. 6). Specifically, we generate documents and assign them topics drawn from a Dirichlet topic distribution. We tune the difficulty in separating topics within the corpora by setting (1) the value of a parameter α that determines both the extent to which documents mix topics and the extent to which words are significantly used by different topics and (2) the fraction of words that are generic, that is, contain no information about the topics (see Sec. VII).

Figure 7 presents results for a large number of synthetic corpora. We calculate both accuracy and reproducibility of the algorithms for several parameters' values (see also the Supplemental Material [37]). Our results make it plainly obvious that LDA has low reproducibility and low accuracy even for corpora that are generated according to its assumed generative model. The reason for the low validity of LDA for these corpora is the same as for the language test: While the generative model has the highest likelihood if topics are sufficiently equal in size, the sheer number of overfitting models is so large, and they can have likelihoods so close to the global maximum that the optimization procedure is almost guaranteed to yield an incorrect estimation of the parameters.

B. Corpus of scientific publications

As a second test, we use a real-world corpus for which one has a good *a priori* understanding of the topics. Specifically, we collect a corpus of 23 838 documents from Web of Science: Each document contains the title and

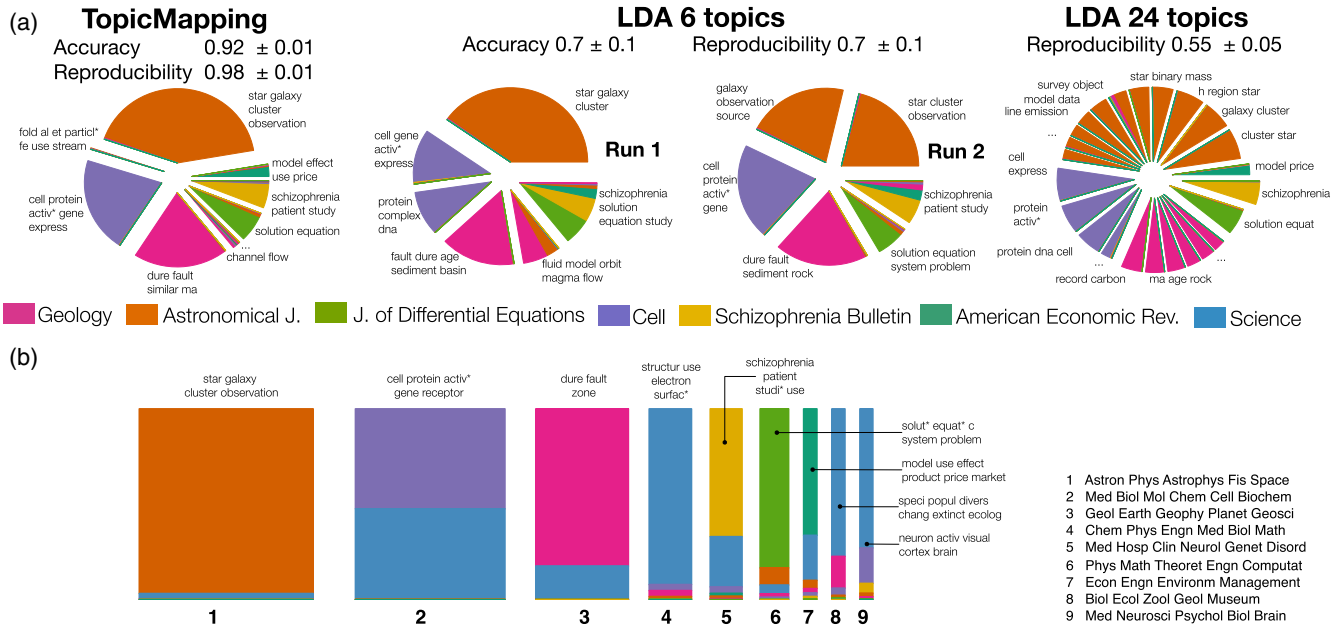

FIG. 8. (a) Performance of topic-model algorithms on an *a priori* well-characterized corpus of scientific publications. We represent the topic model inferred by each algorithm as a pie diagram. Each slice of the pie indicates a single topic. Different colors correspond to different journals, and the area taken by a given journal (color) in a given topic (slice) is proportional to the probability of the corresponding journal given that topic: $p(\text{journal}|\text{topic}) = \sum_{\text{doc}} p(\text{journal}|\text{doc}) \times p(\text{doc}|\text{topic})$. We identify as topic keywords the most frequent words for documents inferred to have been generated by the topic. The * symbol indicates that the word shown is the stem of a number of words [45]. The TopicMapping algorithm is nearly perfect in its ability to identify the source of the publication. The second and third pies show the performance of standard LDA when we use the number of journals as our estimate of the number of topics. As we find for the synthetic corpora, publications from large journals are split and publications from small journals are merged. The fourth pie shows the performance of the standard LDA algorithm when we estimate the number of topics using model selection. Small topics are now resolved, but big topics are split so that each topic is comparable in size. As we learned from the analysis of synthetic corpora, we find a large decrease in reproducibility. (b) Effect of adding publications from the multidisciplinary journal Science. Many of the publications in Science are assigned to the same topics as publications from the disciplinary journals. However, some of the publications in Science are assigned to new topics. The total number of topics found is 19, but only topics with probability bigger than 2% are shown in the figure (nine topics). In order to validate these results, we present both the keywords identified for each topic and the departments most highly represented for the affiliations in the author lists of the publications assigned to the topic.

the abstract of a paper published in one of six top journals from different disciplines (geology, astronomy, mathematics, biology, psychology, and economics). Preprocessing yields 106 143 unique words and approximately 8.7×10^6 connections.

We surmise a generative model in which each journal defines a topic and in which each document is assigned exclusively to the topic defined by the journal in which it was published. We then compare the topics inferred by symmetric LDA (variational inference) and TopicMapping with the surmised generative model. It is visually apparent that TopicMapping has nearly perfect accuracy and reproducibility, whereas standard LDA optimization using the known number of topics has a significantly poorer performance [Fig. 8(a)]. When using held-out likelihood, the recommended approach for estimating the number of topics for the LDA algorithm, the results become dramatically worse. Held-out likelihood maximization for different numbers of topics estimates that the corpus comprises 20 to 30 topics (see the Supplemental Material [37]). Even if the estimated number of topics were correct, the validity of the LDA algorithm would be extremely low, since reproducibility across runs is only 55%.

In order to further test the validity of these algorithms, we add 16 688 documents from the multidisciplinary journal Science [Fig. 8(b)]. Since Science regularly publishes papers in geology, astronomy, biology, and psychology, we expect many of the papers to be assigned to the topics defined by the disciplinary journals. However, Science also publishes papers in other disciplines, including chemistry, physics, and neuroscience; thus, we predict that new topics will emerge. Indeed, the keywords for topic 4 are consistent with papers in chemistry and physics, and those are the departments most highly represented in the affiliations of the author lists of those papers [Fig. 8(b)]. Similarly, the keywords for topic 9 are consistent with papers in neuroscience, and the departments most highly represented in the affiliations of the author lists of those papers are medicine and neuroscience [Fig. 8(b)].

When considering the likelihood of the estimated models, TopicMapping yields a slightly larger likelihood than standard LDA optimization when considering only models with the same effective number of topics. This slight improvement is not surprising since overfitting will yield larger likelihoods. However, the small difference in likelihood between an algorithm with 0.7 accuracy and reproducibility and one with 0.92 accuracy and 0.98 reproducibility raises clear theoretical concerns about trusting likelihood as the only measure of performance of an algorithm. (See the Supplemental Material for a more detailed discussion of this matter [37].)

C. The English Wikipedia corpus

One of the most valued characteristics of the LDA algorithm is its relatively low computational demand, which enables the study of very large corpora. Because TopicMapping involves an additional step (pruning of connections) not required by LDA, it is important to determine whether its computational requirements become impractical when considering large corpora.

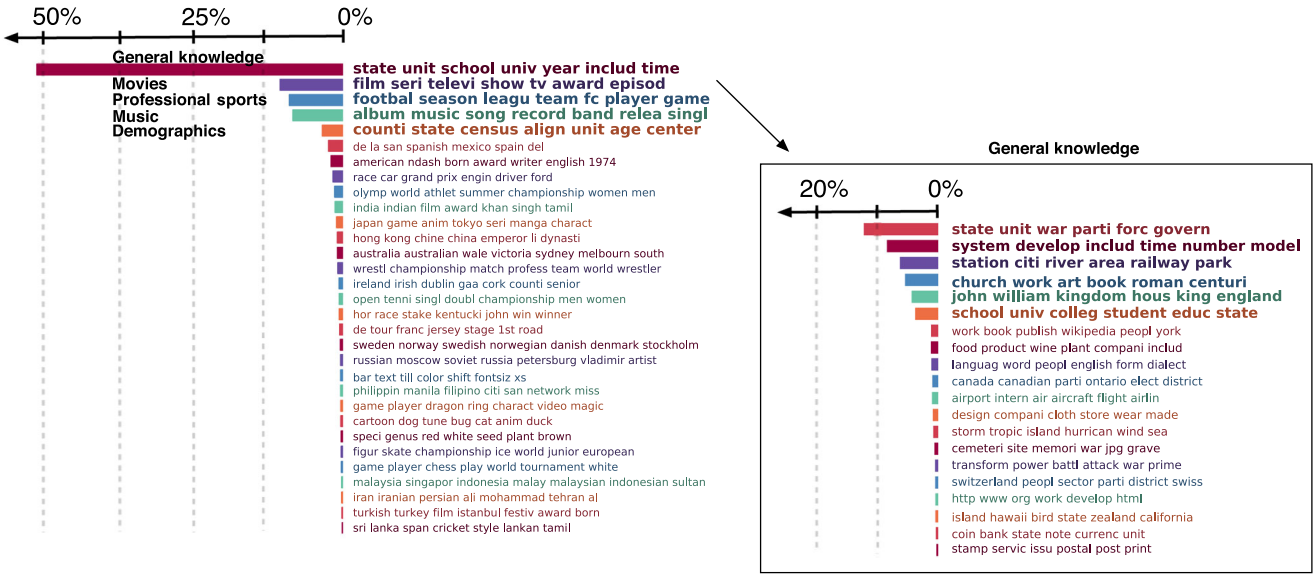

FIG. 9. Topic model of the English Wikipedia corpus obtained with TopicMapping. We show model estimates after one single iteration of LDA refining. Further iterations result in a much greater number of topics and low reproducibility of the results (see the Supplemental Material, Secs. 1.6 and 10 [37]). We highlight the top five topics by size, which together account for 80% of all documents. Four of the five topics are very easy to identify. However, the largest topic, which we denote as "general knowledge," is harder to grasp. However, fitting of a (sub)topic model to the documents in the general knowledge topic yields a set of (sub)topics that are again quite straightforward to interpret.

In order to evaluate computational requirements for large corpora, we apply TopicMapping to a sample of the English Wikipedia acquired in May 2013. The whole English Wikipedia comprises more than 4×10^6 articles, but most are very short (stubs). We restrict our attention to those articles with at least five inlinks, five outlinks, and 100 unique words. Additionally, we prune any words that appear in fewer than 100 articles because these words may be unusual given names or locations. Our English Wikipedia corpus comprises 1 294 860 articles and approximately 800×10^6 words (118 599 unique words, after stemming and removing stop words).

The most time-consuming step in the TopicMapping pipeline is step 2, the pruning of the connections between pairs of words. Fortunately, this step can be easily parallelized. Specifically, we use nine threads and assign to each one a fraction of the total word pairs we had to consider. Doing so, we are able to construct the pruned network of words in roughly 12 h using our computing cluster. The next step, clustering of the words using Infomap, is extremely fast: Each run of the algorithm takes about 1 h, and we run it 10 times. After that, we run the PLSA estimation algorithm with a single thread, taking less than 1 day.

Running our English Wikipedia corpus through LDA turns out to be significantly slower. Thus, we also parallelize the LDA optimization on about 50 threads. This parallelisation reduces the computing time needed to complete one iteration of the LDA algorithm to about 1 h.

The results after one single LDA refinement are shown in Fig. 9. As mentioned above, for this system, the method finds a very heterogeneous topic distribution, which the full LDA optimization would change significantly (see the Supplemental Material [37]). Thus, we decide to show the results before the refinement and to find the subtopics of the largest topic, running the algorithm on the subcorpus of words assigned to it.

VI. CONCLUSIONS

Ten years since its introduction, there has been surprisingly little research on the validity of LDA optimization algorithms for inferring topic models [35]. Our systematic analysis clearly demonstrates that current implementations of LDA have low validity. Moreover, we show that algorithms developed for community detection in networks can be modified for topic modeling with remarkable improvements in validity. Specifically, community-detection algorithms yield an educated guess of the parameter values in the latent generative model. Interestingly, TopicMapping provides only slight improvements in terms of likelihood but yields greatly improved accuracy and reproducibility.

While topic modeling is likely a new and novel area of interest for physicists, we believe that physics approaches hold tremendous potential for advancing our understanding of topic models and other "big data" algorithms. In particular, in the area of community detection, a substantial amount of work has recently been done on stochastic block models, which, similarly in spirit to LDA, try to fit a generative model of the network [18,19]. We would not be surprised if similar techniques would offer new insights into topic modeling.

VII. METHODS

A. Comparing models

Here, we describe the algorithm for measuring the similarity between two models p and q. Both topic models are described by two sets of probability distributions: $p(\text{topic}|\text{doc})$ and $p(\text{word}|\text{topic})$. Given a document, we would like to compare two distributions: $p(t'|\text{doc})$ and $p(t''|\text{doc})$. The problem is not trivial because the topics are not labeled: The numbers we use to identify the topics in each model are just one of the $K!$ possible permutations of their labels. Instead, documents have, of course, the same labels. For this reason, it is easy to quantify the similarity of topics t' and t'' from different models, if we look at which documents are in these topics: We can use Bayes's theorem to compute $p(\text{doc}|t')$ and $q(\text{doc}|t'')$ and compare these two probability distributions. We propose to measure the distance between $p(\text{doc}|t')$ and $q(\text{doc}|t'')$ as the one-norm (or Manhattan distance): $\|p(\text{doc}|t') - q(\text{doc}|t'')\|_1 = \sum_{\text{doc}} |p(\text{doc}|t') - q(\text{doc}|t'')|$. Since we are dealing with probability distributions, $\|p - q\|_1 \leq 2$. We can then define the normalized similarity between topics t' and t'' as $s(t', t'') = 1 - \frac{1}{2}\|p(\text{doc}|t') - q(\text{doc}|t'')\|_1$.

To get a global measure of how similar one model is with respect to the other, we compare each topic t' with all topics t'' and we pick the topic that is most similar to t'. Thus, the similarity we get best matching model p versus q is : $\text{BM}(p \to q) = \sum_{t'} p(t') \max_{t''} s(t', t'')$, where BM stands for best match and the arrow indicates that each topic in p looks for the best-matching topic in q. Of course, we can make this similarity symmetric, averaging the measure with $\text{BM}(p \leftarrow q) = \sum_{t''} q(t'') \max_{t'} s(t', t'')$: $\text{BM}(p, q) = \frac{1}{2}[\text{BM}(p \to q) + \text{BM}(p \leftarrow q)]$.

Although this similarity is normalized between 0 and 1, it does not inform us about how similar the two models are compared to what we could get with random topic assignments. For this reason, we also compute the average similarity $\text{BM}(p \to q_s)$, where we randomly shuffle the document labels in model q. Our null-model similarity is then defined as $\text{BM}_{\text{rand}} = \frac{1}{2}[\text{BM}(p \to q_s) + \text{BM}(p_s \leftarrow q)]$.

Eventually, we can define our measure of normalized similarity between the two models as

$$\text{BM}_n = \frac{\text{BM} - \text{BM}_{\text{rand}}}{1 - \text{BM}_{\text{rand}}}. \tag{2}$$

An analogous similarity score can be defined for words using $p(\text{word}|\text{topic})$ instead of $p(\text{doc}|\text{topic})$.

B. Generating synthetic corpora

The algorithm we use to generate synthetic data sets relies on the generative model assumed by LDA. First, we specify the number of documents and the number of words in each document L_d. For simplicity, we set the same number of words for each document $L_d = L$. Next, we set the number of topics K and the probability distribution of each topic $p(\text{topic})$. Finally, we specify the number of words in our vocabulary N_w and the probability distribution of each word $p(\text{word})$. For the sake of simplicity, we use uniform probabilities for $p(\text{word})$, although the same model can be used for arbitrary probability distributions. All these parameters define the size of the corpus; the other aspect to consider is how mixed documents are across topics and how mixed topics are across words: This mixing can be specified by one hyperparameter α, whose use will be made clear in the following. The algorithm works in the following steps.

(1) For each document "doc," we decide the probability this document will make use of each topic $p(\text{topic}|\text{doc})$. These probabilities are sampled from the Dirichlet distribution with parameters $\alpha_{\text{topic}} = K \times p(\text{topic}) \times \alpha$. The definition is such that "topic" will be used in the overall corpus with probability $p(\text{topic})$, while the factor K is a normalization that assures that we get $\alpha_{\text{topic}} = \alpha$ for equiprobable topics. In this particular case, $\alpha = 1$ means that documents are assigned to topics drawing the probabilities uniformly at random. (See the Supplemental Material for more on the Dirichlet distribution [37].)

(2) For each topic, we need to define a probability distribution over words: $p(\text{word}|\text{topic})$. For this purpose, we first compute $p(\text{topic}|\text{word})$ for each word, sampling the same Dirichlet distribution as before $[\alpha_{\text{topic}} = K \times p(\text{topic}) \times \alpha]$. Second, we get $p(\text{word}|\text{topic})$ from Bayes's theorem: $p(\text{word}|\text{topic}) \propto p(\text{topic}|\text{word}) \times p(\text{word})$.

(3) We now have all we need to generate the corpus. Every "word" in document "doc" can be drawn, first by selecting "topic" with probability $p(\text{topic}|\text{doc})$ and second by choosing "word" with probability $p(\text{word}|\text{topic})$.

Small values of the parameter α will yield "easy" corpora where documents are mostly about one single topic and words are specific to a single topic (Fig. 7). For simplicity, we keep α constant for all documents and words. However, it is highly unrealistic that all words are mostly used in a single topic, since every realistic corpus contains generic words. To account for these generic words, we divide the words into two classes, specific and generic words: For the former class, we use the same α as above, while for generic words, we set $\alpha = 1$. The fraction of generic words is a second parameter we set.

ACKNOWLEDGMENTS

We thank Xiaohan Zeng, David Mertens, and Adam Hockenberry for discussions. L. A. N. A. gratefully acknowledges the Army Research Office (ARO) for the support from Grant No. W911NF-14-1-0259.

[1] X. Jin, Y. Zhou, and B. Mobasher, in *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, KDD 2005* (ACM, New York, 2005), p. 612–617.

[2] R. Krestel, P. Fankhauser, and W. Nejdl, in *Proceedings of the Third ACM Conference on Recommender Systems, RecSys 2009* (ACM, New York, 2009), p. 61–68.

[3] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, in *IEEE International Conference on Computer Vision* (IEEE, New York, 2005), p. 1331–1338.

[4] J. C. Niebles, H. Wang, and L. Fei-fei, *Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words*, Int. J. Comput. Vis. **79**, 299 (2008).

[5] B. Liu, L. Liu, A. Tsykin, G. J. Goodall, J. E. Green, M. Zhu, C. H. Kim, and J. Li, *Identifying Functional miRNA-mRNA Regulatory Modules with Correspondence Latent Dirichlet Allocation*, Bioinformatics **26**, 3105 (2010).

[6] I. Bíró, J. Szabó, and A. A. Benczúr, in *Proceedings of the 4th International Workshop on Adversarial Information Retrieval on the Web, AIRWeb 2008* (ACM, New York, 2008), p. 29–32.

[7] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, *Indexing by Latent Semantic Analysis*, J. Am. Soc. Inf. Sci. **41**, 391 (1990).

[8] D. D. Lee and H. S. Seung, *Learning the Parts of Objects by Non-negative Matrix Factorization*, Nature (London) **401**, 788 (1999).

[9] T. Hofmann, in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, UAI, 1999* (Morgan Kaufmann, San Francisco, 1999), p. 289–296.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan, *Latent Dirichlet Allocation*, J. Mach. Learn. Res. **3**, 993 (2003).

[11] M. Steyvers and T. Griffiths, *Probabilistic Topic Models*, Handbook Latent Semantic Anal. **427**, 424 (2007).

[12] D. Mimno, M. Hoffman, and D. Blei, in *Proceedings of the 29th International Conference on Machine Learning (ICML-12)* (2012), p. 1599–1606, http://icml.cc/2012.

[13] A. Anandkumar, D. Hsu, F. Huang, and S. M. Kakade, in *Advances in Neural Information Processing Systems* (2012), p. 917–925, http://papers.nips.cc/book/advances-in-neural-information-processing-systems-25-2012.

[14] S. Arora, R. Ge, Y. Halpern, D. Mimno, A. Moitra, D. Sontag, Y. Wu, and M. Zhu, in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)* (2013), p. 280–288, http://icml.cc/2013.

[15] A. Bunde and S. Havlin, *Fractals and Disordered Systems* (Springer-Verlag, New York, 1991).

[16] O. C. Martin, R. Monasson, and R. Zecchina, *Statistical Mechanics Methods and Phase Transitions in Optimization Problems*, Theor. Comput. Sci. **265**, 3 (2001).

[17] B. Karrer and M. E. J. Newman, *Stochastic Block Models and Community Structure in Networks*, Phys. Rev. E **83**, 016107 (2011).
[18] T. P. Peixoto, *Hierarchical Block Structures and High-Resolution Model Selection in Large Networks*, Phys. Rev. X **4**, 011047 (2014).
[19] D. B. Larremore, A. Clauset, and A. Z. Jacobs, *Efficiently Inferring Community Structure in Bipartite Networks*, Phys. Rev. E **90**, 012805 (2014).
[20] X. Yan, C. Shalizi, J. E. Jensen, F. Krzakala, C. Moore, L. Zdeborová, P. Zhang, and Y. Zhu, *Model Selection for Degree-Corrected Block Models*, J. Stat. Mech. (2014) P05007.
[21] R. Guimerà and M. Sales-Pardo, *Missing and Spurious Interactions and the Reconstruction of Complex Networks*, Proc. Natl. Acad. Sci. U.S.A. **106**, 22073 (2009).
[22] R. Guimera, M. Sales-Pardo, and L. A. N. Amaral, *Modularity from Fluctuations in Random Graphs and Complex Networks*, Phys. Rev. E **70**, 025101 (2004).
[23] R. Guimera and L. A. N. Amaral, *Cartography of Complex Networks: Modules and Universal Roles*, J. Stat. Mech. (2005) P02001.
[24] R. Guimera and L. A. N. Amaral, *Functional Cartography of Complex Metabolic Networks*, Nature (London) **433**, 895 (2005).
[25] S. Fortunato, *Community Detection in Graphs*, Phys. Rep. **486**, 75 (2010).
[26] M. Girvan and M. E. J. Newman, *Community Structure in Social and Biological Networks*, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002).
[27] A. Lancichinetti, S. Fortunato, and F. Radicchi, *Benchmark Graphs for Testing Community Detection Algorithms*, Phys. Rev. E **78**, 046110 (2008).
[28] A. Lancichinetti and S. Fortunato, *Community Detection Algorithms: A Comparative Analysis*, Phys. Rev. E **80**, 056117 (2009).
[29] D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum, *Advances in Neural Information Processing Systems* (MIT Press, Massachusetts, 2003).
[30] D. M. Blei and J. D. Lafferty, *A Correlated Topic Model of Science*, Am. Astron. Soc. **1**, 17 (2007).
[31] T. L. Griffiths and M. Steyvers, *Finding Scientific Topics*, Proc. Natl. Acad. Sci. U.S.A. **101**, 5228 (2004).
[32] R. Nallapati, W. Cohen, and J. Lafferty, in *Proceedings of the Seventh IEEE International Conference on Data Mining Workshops, ICDMW, 2007* (IEEE Computer Society, Washington, DC, 2007), p. 349–354.
[33] D. Sontag and D. Roy, *Complexity of Inference in Latent Dirichlet Allocation*, Adv. Neural Inf. Process. Syst. **24**, 1008 (2011).
[34] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Models and Applications*, Continuous Multivariate Distributions Vol. 59 (Wiley, New York, 2002).
[35] H. Wallach, D. Mimno, and A. McCallum, *Rethinking LDA: Why Priors Matter*, Adv. Neural Inf. Process. Syst. **22**, 1973 (2009).
[36] "Toy" models are helpful because they can be analytically treated and provide insights into more realistic cases. While this approach is standard in physics, surprisingly, it has not been explored in the topic-model literature.
[37] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevX.5.011007 for TopicMapping software, data sets, and related codes.
[38] E. Gaussier and C. Goutte, in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (ACM, New York, 2005), p. 601–602.
[39] Invoke IT Blog, http://invokeit.wordpress.com/frequency-word lists.
[40] In the Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevX.5.011007, we also show the results for asymmetric LDA implementing Gibbs sampling [35], which, again, performs better in the egalitarian case.
[41] I. S. Dhillon, in *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, New York, 2001), p. 269–274.
[42] T. Zhou, J. Ren, M. Medo, and Y.-C. Zhang, *Bipartite Network Projection and Personal Recommendation*, Phys. Rev. E **76**, 046115 (2007).
[43] Specifically, the fact that the words "the" and "to" appear in every document does not provide any information about the topic. Similarly, if two very frequent words appear together in a single document, that is to be expected.
[44] M. Rosvall and C. T. Bergstrom, *Maps of Random Walks on Complex Networks Reveal Community Structure*, Proc. Natl. Acad. Sci. U.S.A. **105**, 1118 (2008).
[45] The English (Porter2) Stemming Algorithm, http://snowball.tartarus.org/algorithms/english/stemmer.html.
[46] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, 1st ed. (Addison-Wesley, Boston, 2005).
[47] R. Guimerà, M. Sales-Pardo, and L. A. N. Amaral, *Module Identification in Bipartite and Directed Networks*, Phys. Rev. E **76**, 036102 (2007).
[48] M. Sales-Pardo, R. Guimera, A. A. Moreira, and L. A. N. Amaral, *Extracting the Hierarchical Organization of Complex Systems*, Proc. Natl. Acad. Sci. U.S.A. **104**, 15224 (2007).
[49] D. Hric, R. K. Darst, and S. Fortunato, *Community Detection in Networks: Structural Clusters Versus Ground Truth*, Phys. Rev. E **90**, 062805 (2014).