Information Sciences 563 (2021) 226–240

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Jointly modeling and simultaneously discovering topics and clusters in text corpora using word vectors

Gianni Costa, Riccardo Ortale *

ICAR-CNR, Via P. Bucci 8/9c, 87036 Rende (CS), Italy

ARTICLE INFO

Article history:
Received 6 May 2019
Received in revised form 19 November 2020
Accepted 11 January 2021
Available online 26 January 2021

Keywords:
Document clustering
Topic modeling
Word embeddings
Bayesian text analysis

ABSTRACT

An innovative model-based approach to coupling text clustering and topic modeling is introduced, in which the two tasks take advantage of each other. Specifically, the integration is enabled by a new generative model of text corpora. This explains topics, clusters and document content via a Bayesian generative process. In this process, documents include word vectors, to capture the (syntactic and semantic) regularities among words. Topics are multivariate Gaussian distributions on word vectors. Clusters are assigned corresponding topic distributions as their semantics. Content generation is ruled by text clusters and topics, which act as interacting latent factors. Documents are at first placed into respective clusters, then the semantics of these clusters is then repeatedly sampled to draw document topics, which are in turn sampled for word-vector generation.

Under the proposed model, collapsed Gibbs sampling is derived mathematically and implemented algorithmically with parameter estimation for the simultaneous inference of text clusters and topics.

A comparative assessment on real-world benchmark corpora demonstrates the effectiveness of this approach in clustering texts and uncovering their semantics. Intrinsic and extrinsic criteria are adopted to investigate its topic modeling performance, whose results are shown through a case study. Time efficiency and scalability are also studied.

1. Introduction

Topic modeling along with text clustering are key tasks in text mining [2], which can be unified to benefit mutually from each other [43,46]. In particular, topic modeling exposes the inherent semantics of a whole corpus of (not necessarily homogeneous) text documents. The uncovered semantics is a representation of the (meaning of) text documents as mixtures of topics, with topics being suitable word rankings. Performing topic modeling on a text corpus, while clustering its documents simultaneously, enables the summarization of clusters through corresponding distributions over the topics treated by the respective documents. Such cluster-specific topic distributions capture the underlying semantics of disjoint groups of homogeneous documents, thus providing a more detailed and coherent understanding of text. Symmetrically, text clustering uncovers patterns of homogeneity in text corpora. However, the semantic coherence of the discovered clusters can be penalized, if homogeneity only involves lexical regularities across text documents. The exploitation of the *bag-of-words* model for raw text processing likely worsens cluster quality. This is because of the resulting huge text representation (whose dimen-

* Corresponding author.
E-mail addresses: costa@icar.cnr.it (G. Costa), ortale@icar.cnr.it (R. Ortale).

https://doi.org/10.1016/j.ins.2021.01.019
0020-0255/

sionality amounts to vocabulary size) and its sparsity (which is especially challenging in the case of short texts). Clustering a text corpus, while simultaneously modeling the topics of its documents, allows for avoiding the foregoing limitations. Topic modeling provides a concise semantic representation of the text documents within a low-dimensional space of understandable topics. This permits a more effective partitioning of the text corpus into semantically-coherent and intelligible clusters.

Combining text clustering with topic modeling is challenging for the following reasons: foremost, the two tasks have to be suitably interpenetrated, in principle, both should operate in an interdependent manner, with each task acting as an enhancement of the other one; in addition, a synergic interaction between the two tasks should be ideally devised, so as to capture and suitably exploit the syntactic and also semantic relationships between words.

Combining text clustering with topic modeling is challenging for the following reasons: foremost, the two tasks have to be suitably interpenetrated, in principle, both should operate in an interdependent manner, with each task acting as an enhancement of the other one; in addition, a synergic interaction between the two tasks should be ideally devised, so as to capture and suitably exploit the syntactic and also semantic relationships between words.

In this article, a new approach to the seamless integration of text clustering with topic modeling is discussed. The proposed approach is grounded in a principled combination of solid foundations from several disciplines. These encompass probabilistic graphical modeling [27], Bayesian statistics [10,20,45], generative latent-factor modeling [6,34], text mining [2] and word vectors [4,32].

The intuition behind this approach consists in inferring the topics and cluster memberships of text documents from their contents. To this end, DISCOVER (*Document topIcS and Clusters from wOrd VEctoRs*) is developed, an innovative generative model of topics, text and clusters in document collections. Under DISCOVER, the input text documents are conceived as the observed outcome of an imaginary generative process. The latter is governed by clusters as well as topics, which operate as interacting latent factors. According to the generative semantics of DISCOVER, document clusters are endowed with respective multinomial distributions on the underlying topics. Basically, such topic distributions enforce intra-cluster coherence. A multinomial distribution is placed over clusters to pick document membership. Thus, the generic text document is generated in two steps. At first, the distribution over clusters is sampled to establish its membership, then, the topical distribution of the chosen cluster is repeatedly sampled to word the document content. Overall, there are three appealing features of the generative process modeled by DISCOVER: firstly, each textual document comprises word vectors, rather than discrete text units, this choice permits the syntactic and also semantic regularities across words to be taken suitably into account; secondly, the individual document clusters are explicitly assigned descriptive topic distributions as their respective semantics; thirdly, uncertainty is handled probabilistically according to the consolidated Bayesian treatment [6].

All observations and latent factors in the generative process of DISCOVER are characterized as random variables. In compliance with latent-factor modeling, random variables are distinguished into observed and unobserved. Specifically, the individual documents are characterized by means of observed random variables. The latter take on word vectors (drawn from the representation of topics below), rather than discrete words. The unobserved random variables are employed to characterize the latent factors since these are neither directly observable nor explicitly measurable. In more detail, topics are characterized as multivariate Gaussian distributions on the space of word vectors [4,32]. Their precision and mean are unobserved random variables, drawn from respective conjugate Gaussian-Wishart priors. Also, the cluster membership of each document is an unobserved random variable, sampled from the multinomial distribution over clusters. All multinomial distributions are drawn from corresponding conjugate Dirichlet priors. Yet, the conditional (in) dependencies among the aforementioned random variables are defined through the elegant formalism of probabilistic graphical modeling. In the context of the generative process of DISCOVER, such conditional (in) dependencies specify the interaction of text clustering with topic modeling, in addition to their influence on document wording.

Under DISCOVER, topic modeling and text clustering are performed simultaneously by Bayesian reasoning. The latter consists in learning the values of the latent random variables of DISCOVER via posterior inference [20,23,45] with parameter estimation. More precisely, collapsed Gibbs sampling is used for the *a posteriori inference of the assignments of documents to clusters. Parameter estimation is utilized for the distribution over cluster memberships as well as the topic distributions of clusters and documents.*

An extensive comparative experimentation over benchmark corpora demonstrates the effectiveness of DISCOVER in clustering texts and coherently uncovering their semantic topics. Notably, the experimentation of the topic modeling performance is articulated into a quantitative and qualitative assessment. The quantitative assessment accounts for intrinsic and extrinsic criteria corresponding to, respectively, the semantic coherence of the inferred topics and the classification effectiveness enabled by such topics. The qualitative assessment is a case study which elucidates the output of DISCOVER on real-world document corpora. It looks into the results from one of the chosen benchmark collections. Our experimentation also investigates the time efficiency and scalability of DISCOVER with the size of both the underlying text corpus and the word vectors.

The originality of DISCOVER lies in the Bayesian probabilistic formalization of an unprecedented and effective interplay between topic modeling in the space of word vectors and a particular instance of text clustering, which also involves the summarization/explanation of cluster semantics. The innovative contributions of this article are summarized below:

- The synergic pairing of text clustering with topic modeling is explored through an innovative approach.
- A Bayesian probabilistic generative model of text corpora, i.e., DISCOVER (*Document topIcS and Clusters from wOrd VEctoRs*), is developed.

- Under DISCOVER, text clustering is integrated with topic modeling as interacting latent factors, which influence content generation.
- Word vectors are suitably utilized under DISCOVER, in order to account for the syntactic and also semantic regularities across words.
- The inherent semantics of the uncovered clusters is clearly explained through intelligible topic distributions.
- The mathematical and algorithmic details of collapsed Gibbs sampling with parameter estimation are derived to perform text clustering jointly with topic modeling.
- An empirical comparative evaluation of DISCOVER is conducted over benchmark corpora.
- A new class of competitors is specifically introduced to contrast DISCOVER against pipelines of established approaches to text clustering as well as topic modeling.
- The results of this approach on real-word document corpora are elucidated through an explicative case study.

The rest of this article is structured as follows: notation and preliminary concepts are introduced in Section 2; the DISCOVER model is developed in Section 3; collapsed Gibbs sampling with parameter estimation is derived in Section 4; the empirical assessment of DISCOVER is presented in Section 5; a review of seminal related works is provided in Section 6; finally, conclusions are drawn in Section 7, where future research is also highlighted.

2. Preliminaries

Let $\boldsymbol{D}$ be a text corpus[1] on a vocabulary $\boldsymbol{V}$. $\boldsymbol{V}$ is a set comprising V words, i.e., $\boldsymbol{V} \triangleq \{w_1, \ldots, w_V\}$. $\boldsymbol{D}$ is a collection of D documents, i.e., $\boldsymbol{D} = \{\boldsymbol{d}_1, \ldots, \boldsymbol{d}_D\}$. Additionally, any document $\boldsymbol{d}$ of $\boldsymbol{D}$ actually comprises $n_{\boldsymbol{d}}$ lexical elements from $\boldsymbol{V}$, i.e., $\boldsymbol{d} \triangleq \{w_{\boldsymbol{d},1}, \ldots, w_{\boldsymbol{d},n_{\boldsymbol{d}}} | w_{\boldsymbol{d},n} \in \boldsymbol{V} \quad \text{with} \quad n = 1, \ldots, n_{\boldsymbol{d}}\}$. For the purpose of capturing the syntactic and also semantic regularities across words in $\boldsymbol{D}$, any document $\boldsymbol{d}$ of $\boldsymbol{D}$ is represented alternatively through word vectors [4,18,32]. Assume that $f : \boldsymbol{V} \mapsto \mathbb{R}^H$ is some suitable function, which embeds the individual words of $\boldsymbol{V}$ into a space of real-valued vectors having size H. $\boldsymbol{d}$ is formalized as made up of word vectors, i.e., $\boldsymbol{d} \triangleq \{\boldsymbol{w}_{\boldsymbol{d},1}, \ldots, \boldsymbol{w}_{\boldsymbol{d},n_{\boldsymbol{d}}} | \boldsymbol{w}_{\boldsymbol{d},n} = f(w_{\boldsymbol{d},n}) \quad \text{with} \quad w_{\boldsymbol{d},n} \in \boldsymbol{V} \quad \text{and} \quad n = 1, \ldots, n_{\boldsymbol{d}}\}$.

Two intrinsically characteristic properties of $\boldsymbol{D}$ are its topics and cluster structure. Both are *apriori* unknown.

Assume that $\boldsymbol{D}$ covers T topics: these are individually defined in Section 3 as multivariate Gaussian distributions on the above space of word vectors. In particular, for any $t = 1, \ldots, T$, the multivariate Gaussian distribution of the generic topic t has mean $\boldsymbol{\mu}_t \in \mathbb{R}^H$ and precision $\boldsymbol{\Lambda}_t \in \mathbb{R}^{H \times H}$. The precision and mean of all multivariate Gaussian distributions are jointly denoted as $\beta \triangleq \{\boldsymbol{\mu}_t, \boldsymbol{\Lambda}_t | t = 1, \ldots, T\}$.

The documents of $\boldsymbol{D}$ can be partitioned into K clusters $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_K$. The cluster membership of the generic document $\boldsymbol{d}$ is denoted by $c_{\boldsymbol{d}}$. For any $k = 1, \ldots, K$, it holds that $\Pr(c_{\boldsymbol{d}} = k) = \pi_k$, i.e., π_k is the probability that $\boldsymbol{d}$ belongs to $\boldsymbol{C}_k$. The partitioning of $\boldsymbol{D}$ is indicated as $\boldsymbol{C} \triangleq \{c_{\boldsymbol{d}} | \boldsymbol{d} \in \boldsymbol{D}\}$. Besides, $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_K\}$ represents the distribution of probability on clusters $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_K$.

Topics have a different relevance to the distinct clusters. In particular, for any $k = 1, \ldots, K$ and $t = 1, \ldots, T$, $\theta_{k,t}$ is the extent to which cluster $\boldsymbol{C}_k$ deals with topic t. Thus, the particular semantics of $\boldsymbol{C}_k$ can be characterized as a topic mixture $\boldsymbol{\theta}_k \triangleq \{\theta_{k,1}, \ldots, \theta_{k,T}\}$. $\boldsymbol{\Theta} \triangleq \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$ stands for the inherent semantics of all clusters $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_K$.

Inside clusters, documents are characterized by their semantics. The specific semantics of $\boldsymbol{d}$ is the topic mixture $\boldsymbol{\theta}_{\boldsymbol{d}} \triangleq \{\theta_{\boldsymbol{d},1}, \ldots, \theta_{\boldsymbol{d},T}\}$, with $\theta_{\boldsymbol{d},t}$ being the relevance of topic t in $\boldsymbol{d}$. $\boldsymbol{\Theta}^{(\boldsymbol{D})} \triangleq \{\theta_{\boldsymbol{d}_1}, \ldots, \theta_{\boldsymbol{d}_D}\}$ represents the intrinsic semantics of all documents.

Word vectors within documents are explained by the implicit contextualization of their respective topics. $\boldsymbol{z}_{\boldsymbol{d}}$ denotes the topics, which contextualize the word vectors of $\boldsymbol{d}$. More precisely, $\boldsymbol{z}_{\boldsymbol{d}} \triangleq \{z_{\boldsymbol{d},1} \ldots, z_{\boldsymbol{d},n_{\boldsymbol{d}}}\}$, where $z_{\boldsymbol{d},n}$ is one topic from the discrete interval $[1, T]$, which contextualizes the meaning of the corresponding word vector $\boldsymbol{w}_{\boldsymbol{d},n}$. Notation $\boldsymbol{Z} \triangleq \{\boldsymbol{z}_{\boldsymbol{d}} | \boldsymbol{d} \in \boldsymbol{D}\}$ represents the contextualization of the whole corpus $\boldsymbol{D}$.

2.1. Problem statement

Given a text corpus $\boldsymbol{D}$, the goal is to model jointly and perform simultaneously.

- *topic modeling*, i.e., learning $\beta, \boldsymbol{\Theta}, \boldsymbol{\Theta}^{(\boldsymbol{D})}$ and $\boldsymbol{Z}$;
- *text clustering*, i.e., learning $\boldsymbol{\pi}$ and $\boldsymbol{C}$.

The two tasks above are seamlessly integrated into a Bayesian probabilistic generative model of text corpora, which is presented in Section 3. This model incorporates both tasks as interacting factors, which latently influence document

[1] The notions of *text corpus*, *document collection* and *document corpus* are interchangeably used as synonyms.

wording. Posterior inference is derived along with parameter estimation in Section 4, in order to enable the discovery of the latent text clusters and topics, which led to the observation of the target corpus $\boldsymbol{D}$ according to the foregoing model.

3. The DISCOVER Model

DISCOVER (*Document topIcS and Clusters from wOrd VEctoRs*) is a generative latent-factor model of document collections with their respective clusters and topics. Under DISCOVER, any corpus $\boldsymbol{D}$ is conceived as the only observed result of a Bayesian probabilistic generative process. In this process, the constituting elements of $\boldsymbol{D}, \boldsymbol{\beta}, \boldsymbol{\Theta}, \boldsymbol{\Theta}^{(\boldsymbol{D})}, \boldsymbol{Z}, \boldsymbol{\pi}$ and $\boldsymbol{C}$ are random variables. The random variables in $\boldsymbol{\beta}, \boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\pi}$ and $\boldsymbol{C}$ are regarded as latent factors. These govern the generation of $\boldsymbol{D}$, although their values are neither directly observable nor explicitly measurable. The conditional (in) dependencies between the random variables under DISCOVER are shown in the directed graphical representation with plate notation of Fig. 1. Notice that, in Fig. 1, the shaded nodes are observed random variables, whereas the unshaded nodes are latent random variables.

The generative process hypothesized by DISCOVER performs the attainment of the observed random variables in $\boldsymbol{D}$, by fulfilling the conditional (in) dependencies of Fig. 1. This is accomplished as detailed in Fig. 2.

Initially, at step 1, topics are characterized as multivariate Gaussian distributions. In particular, for each topic $t = 1, \ldots, T$, the mean $\boldsymbol{\mu}_t$ and precision $\boldsymbol{\Lambda}_t$ of the corresponding multivariate Gaussian distribution are sampled from a conjugate Guassian-Wishart prior with a hyperparameter $\boldsymbol{\Xi} = \{\boldsymbol{\mu}_0, \boldsymbol{W}_0, \nu_0, \lambda_0\}$.

At step 2, the multinomial distribution $\boldsymbol{\pi}$ over the K clusters $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_K$ is drawn from a conjugate Dirichlet prior with hyperparameter $\boldsymbol{\tau}$.

At step 3, all clusters are assigned their respective semantics, that is, expressed through representative multinomial distributions over topics. For each cluster $\boldsymbol{C}_k$, the corresponding semantics $\boldsymbol{\theta}_k$ is drawn from a conjugate Dirichlet prior with hyperparameter $\boldsymbol{\gamma}$. The explicit association of clusters with their respective semantics is expected to be beneficial for enforcing intra-cluster coherence.

Lastly, at step 4, each document $\boldsymbol{d}$ of $\boldsymbol{D}$ is generated, by sampling the semantics $\boldsymbol{\theta}_{c_{\boldsymbol{d}}}$ of the corresponding cluster $c_{\boldsymbol{d}}$. At step 4(a), the cluster membership of $\boldsymbol{d}$ is established by sampling $c_{\boldsymbol{d}}$ from $\boldsymbol{\pi}$ and, accordingly, placing $\boldsymbol{d}$ inside cluster $c_{\boldsymbol{d}}$. With cluster membership in place, the content of $\boldsymbol{d}$ is worded, by exploiting $\boldsymbol{\theta}_{c_{\boldsymbol{d}}}$. Each word vector $\boldsymbol{w}_{\boldsymbol{d},n}$ of $\boldsymbol{d}$ is generated into two steps: at step 4(b) i, a topic $z_{\boldsymbol{d},n}$ is drawn from $\boldsymbol{\theta}_{c_{\boldsymbol{d}}}$; at step 4(b) ii, $\boldsymbol{w}_{\boldsymbol{d},n}$ is drawn from the respective topic $z_{\boldsymbol{d},n}$, i.e., the multivariate Gaussian distribution with mean $\boldsymbol{\mu}_{z_{\boldsymbol{d},n}}$ and precision $\boldsymbol{\Lambda}_{z_{\boldsymbol{d},n}}$.

The reliance of $z_{\boldsymbol{d},n}$ on $\boldsymbol{\theta}_{c_{\boldsymbol{d}}}$ through $c_{\boldsymbol{d}}$ allows for pairing text clustering with topic modeling under DISCOVER. It is worth noticing that the envisaged document generation entails the definition of document likelihood $\Pr(\boldsymbol{d}|\boldsymbol{z}_{\boldsymbol{d}}, \boldsymbol{\beta})$, that is formalized below

$$\Pr(\boldsymbol{d}|\boldsymbol{z}_{\boldsymbol{d}}, \boldsymbol{\beta}) \triangleq \prod_{n=1}^{n_{\boldsymbol{d}}} \mathcal{N}\left(\boldsymbol{w}_{\boldsymbol{d},n}|\boldsymbol{\mu}_{z_{\boldsymbol{d},n}}, \boldsymbol{\Lambda}_{z_{\boldsymbol{d},n}}\right) \tag{1}$$

4. Posterior inference

DISCOVER integrates text clustering with topic modeling by means of corresponding latent random variables, which interact in the Bayesian probabilistic generation of document corpora. Thus, given a corpus $\boldsymbol{D}$, both tasks are performed under DISCOVER through Bayesian reasoning. The latter involves the inference of a posterior distribution $\Pr(\boldsymbol{\beta}, \boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{C}|\boldsymbol{D})$, with which to trace back to the latent random variables in $\boldsymbol{\beta}, \boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\pi}$ and $\boldsymbol{C}$. This amounts implicitly to carrying out text clustering simultaneously with topic modeling.

As ordinarily experienced in practical applications of Bayesian models, exact posterior inference under DISCOVER is intractable. This leads to the choise of stochastic approximate inference, building on Gibbs sampling, an *MCMC* method [3] that often enables simple and intuitive inference algorithms, even if the number of random variables in the underlying model is potentially large [24]. An algorithm for collapsed Gibbs sampling with parameter estimation is designed. Its pseudo code is reported in Algorithm 1, where collapsed Gibbs sampling is carried out between lines 2 and 12 and parameter estimation is performed between lines 13 and 18.

Collapsed Gibbs sampling leverages the conjugacy between the multinomial and Dirichlet distributions, which allows for integrating out the multinomial random variables (or, also, parameters) in $\boldsymbol{\Theta}$ as well as $\boldsymbol{\pi}$. This is advantageous to expedite Gibbs sampling. The basic Gibbs sampling would operate by repeatedly drawing the values of all individual random variables in $\boldsymbol{\beta}, \boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\pi}$ and $\boldsymbol{C}$ from corresponding full conditionals in an iterative fashion. Owing to the foresaid type of conjugacy,

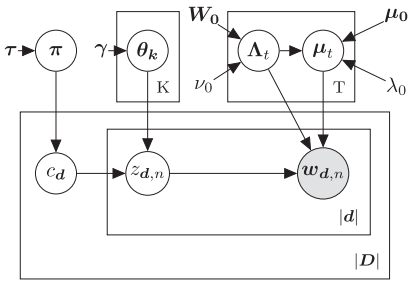

Fig. 1. Graphical representation of DISCOVER.

1. For each topic $t = 1, \ldots, T$, draw topic parameters $(\boldsymbol{\mu}_t, \boldsymbol{\Lambda}_t) \sim \mathcal{N}(\boldsymbol{\mu}_0, (\lambda_0 \boldsymbol{\Lambda_t})^{-1}) \mathcal{W}(\boldsymbol{W}_0, \nu_0)$.
2. Draw cluster probability distribution $\boldsymbol{\pi} \sim \textit{Dirichlet}(\boldsymbol{\tau})$.
3. For each $k = 1, \ldots, K$, draw the topic distribution associated with cluster $\boldsymbol{C}_k$, i.e., $\boldsymbol{\theta}_k \sim \textit{Dirichlet}(\boldsymbol{\gamma})$.
4. For each document $\boldsymbol{d}$ in $\boldsymbol{D}$
   (a) Draw cluster membership $c_{\boldsymbol{d}} \sim \textit{Discrete}(\boldsymbol{\pi})$;
   (b) For each $n = 1, \ldots, n_{\boldsymbol{d}}$
      i. draw the topic assignment for the n-th word vector, i.e., $z_{\boldsymbol{d},n} \sim \textit{Discrete}(\boldsymbol{\theta}_{c_{\boldsymbol{d}}})$;
      ii. draw the n-th word vector, i.e., $\boldsymbol{w}_{\boldsymbol{d},n} \sim \mathcal{N}(\boldsymbol{\mu}_{z_{\boldsymbol{d},n}}, \boldsymbol{\Lambda}^{(-1)}_{z_{\boldsymbol{d},n}})$.

Fig. 2. The generative process under DISCOVER.

collapsed Gibbs sampling focuses only on drawing the values of those random variables in $\boldsymbol{\beta}, \boldsymbol{Z}$ and $\boldsymbol{C}$. The values of the multinomial random variables of $\boldsymbol{\Theta}$ and $\boldsymbol{\pi}$ are simply obtained through parameter estimation. The full conditionals exploited in collapsed Gibbs sampling are mathematically detailed below.

Algorithm 1: Collapsed Gibbs sampling with parameter estimation.

Input: The document corpus $\boldsymbol{D}$; the number K of underlying clusters; the number T of latent topics; the number I of sampling iterations;
Output: the individual random variables in $\boldsymbol{Z}$ and $\boldsymbol{C}$.

```
randomly assign topics to word vectors and cluster memberships to documents;
for i = 1, ..., I do
    for t = 1, ..., T do
        sample (μ_t, Λ_t) through Equation 2 of Figure 3;
    end
    for d ∈ D do
        for n = 1, ..., n_d do
            sample z_{d,n} through Equation 3 of Figure 3;
        end
        sample c_d through Equation 4 of Figure 3;
    end
end
for k = 1, ..., K do
    estimate π_k through Equation 5 of Figure 4;
    for t = 1, ..., T do
        estimate θ_{k,t} through Equation 6 of Figure 4;
    end
end
```

The full conditional at line 4 of Algorithm 1 is formalized through Eq. (2) of Fig. 3 as a probability distribution on $\boldsymbol{\mu}_t$ and $\boldsymbol{\Lambda}_t$, given the remaining latent random variables and corpus $\boldsymbol{D}$. In particular, let $\boldsymbol{W}$ be the set of all word vectors, that are contextualized by topic t, i.e., $\boldsymbol{W} = \{\boldsymbol{w}_{\boldsymbol{d},n} | \boldsymbol{d} \in \boldsymbol{D} \text{ and } z_{\boldsymbol{d},n} = t\}$. The entities $\boldsymbol{\mu}_t^*, \boldsymbol{W}_t^*, \lambda_t^*$ and ν_t^* of Eq. (2) are reported next

$$\boldsymbol{\mu}_t^* = \frac{\lambda_0 \boldsymbol{\mu}_0 + |\boldsymbol{W}| \overline{\boldsymbol{T}}}{\lambda_0 + |\boldsymbol{W}|}; \qquad \lambda_t^* = \lambda_0 + |\boldsymbol{W}|; \qquad \nu_t^* = \nu_0 + |\boldsymbol{W}|$$
$$\left[\boldsymbol{W}_t^*\right]^{-1} = \boldsymbol{W}_0^{-1} + |\boldsymbol{W}| \overline{\boldsymbol{S}} + \frac{\lambda_0 |\boldsymbol{W}|}{\lambda_0 + |\boldsymbol{W}|} (\boldsymbol{\mu}_0 - \overline{\boldsymbol{T}}) (\boldsymbol{\mu}_0 - \overline{\boldsymbol{T}})'$$

with $\overline{\boldsymbol{T}}$ and $\overline{\boldsymbol{S}}$ being in turn defined as follows

$$\overline{\boldsymbol{T}} = \frac{1}{|\boldsymbol{W}|} \sum_{\boldsymbol{w}_{\boldsymbol{d},n} \in \boldsymbol{W}} \boldsymbol{w}_{\boldsymbol{d},n}; \quad \overline{\boldsymbol{S}} = \frac{1}{|\boldsymbol{W}|} \sum_{\boldsymbol{w}_{\boldsymbol{d},n} \in \boldsymbol{W}} \boldsymbol{w}_{\boldsymbol{d},n} \boldsymbol{w}_{\boldsymbol{d},n}'$$

The full conditional used at line 8 of Algorithm 1 is formalized through Eq. 3 of Fig. 3 as a probability distribution on $z_{\boldsymbol{d},n}$, given the remaining latent random variables in addition to corpus $\boldsymbol{D}$. Notation $\boldsymbol{Z}_{\neg(\boldsymbol{d},n)}$ represents all topics assigned to word vectors apart from the one attached to the n-th word vector of document $\boldsymbol{d}$. Moreover, $n_k^{(t)}$ is the count of how many times topic t has been treated within cluster $\boldsymbol{C}_k$. Yet, γ_t is a component of the hyperparameter $\boldsymbol{\gamma}$, which corresponds to topic t.

Lastly, the full conditional used at line 10 of Algorithm 1 is formalized through Eq. (4) of Fig. 3 as a probability distribution on $c_{\boldsymbol{d}}$, given the remaining latent random variables along with corpus $\boldsymbol{D}$. Notation $\boldsymbol{C}_{\neg \boldsymbol{d}}$ indicates the cluster memberships of all documents apart from document $\boldsymbol{d}$. Also, $\boldsymbol{n}_k \triangleq \left\{n_k^{(t)}\right\}_{t=1}^{T}$. $\Delta(\cdot)$ is the Dirichlet delta function [24]. $n^{(k)}$ is the count of how many times cluster $\boldsymbol{C}_k$ has been selected as document membership. τ_k is a component of the hyperparameter $\boldsymbol{\tau}$, which corresponds to cluster k.

Primarily, all counts $n_k^{(t)}$ and $n^{(k)}$ are readily updated as soon as samples are drawn. Across iterations, the full conditionals located at lines 8 and 10 of Algorithm 1 allow for alternating the sampling of the topic assignments given the current cluster memberships (among the other latent random variables) with the sampling of the cluster memberships given the current topic assignments (among the other latent random variables).

At the end of collapsed Gibbs sampling, the above counts $n_k^{(t)}$ and $n^{(k)}$ are exploited for parameter estimation. Under DISCOVER, it holds that $P(\boldsymbol{\pi}|\boldsymbol{\tau}, \boldsymbol{C}) \propto Dirichlet(\boldsymbol{\pi}|\boldsymbol{n} + \boldsymbol{\tau})$ and $P(\boldsymbol{\theta}_k|\boldsymbol{\gamma}, \boldsymbol{Z}, \boldsymbol{C}) \propto Dirichlet(\boldsymbol{\theta}_k|\boldsymbol{n}_k + \boldsymbol{\gamma})$. Accordingly, the latent random variables $\boldsymbol{\pi}_k$ and $\boldsymbol{\theta}_k$ are respectively estimated at lines 14 and 16 of Algorithm 1, via corresponding expectations of the Dirichlet distribution. These are formalized through Eqs. (5) and (6) of Fig. 4.

To conclude, it is worth noting that the semantics $\boldsymbol{\theta}_{\boldsymbol{d}}$ of the generic document $\boldsymbol{d}$ can be calculated as follows

$$\theta_{\boldsymbol{d},t} = \frac{n_{\boldsymbol{d}}^{(t)}}{\sum_{t'=1}^{T} n_{\boldsymbol{d}}^{(t')}} \quad \text{with} \quad t = 1, \ldots, T$$

with $n_{\boldsymbol{d}}^{(t)} = \sum_{n=1}^{|\boldsymbol{d}|} \delta(z_{\boldsymbol{d},n}, t)$ being the count of how many times topic t is treated within $\boldsymbol{d}$.

5. Empirical evaluation

A thorough experimentation of our approach was carried out on real-world benchmark text corpora. The pursued purposes are manifold, i.e.:

- evaluating its effectiveness in clustering corpora of texts and uncovering their topics;
- assessing whether the integration of the two tasks is actually more effective than each task in isolation;
- assessing whether the integration of the two tasks is actually more effective than suitably pipelining both tasks through a trivial sequential arrangement.
- studying its time efficiency and scalability with the size of both the underlying text corpus and the word vectors.

5.1. Text corpora, preprocessing and word vectors

All tests were conducted on two real-world text corpora, i.e., *20-Newsgroups* in addition to *Reuters-21578*. Both are benchmarks for topic modeling as well as text classification, which also contain ground-truth categories for the evaluation of text clustering.

20-Newsgroups[2] contains 11,268 text documents, that are divided into 20 groups.

Reuters-21578[3] consists of 21,578 text documents, which are grouped in 90 imbalanced categories. The individual documents can be found inside multiple categories. Because of category imbalance and overlap, we preprocessed *Reuters-21578* in compliance with common practice. This involves sampling the text corpus, with the aim to preserve only those documents,

[2] http://qwone.com/jason/20Newsgroups/.
[3] http://www.daviddlewis.com/resources/testcollections/reuters21578/.

$$P(\boldsymbol{\mu}_t, \boldsymbol{\Lambda}_t | \boldsymbol{\beta}_{\neg t}, \boldsymbol{C}, \boldsymbol{D}, \boldsymbol{Z}, \boldsymbol{\tau}, \boldsymbol{\gamma}, \boldsymbol{\mu}_0, \boldsymbol{W}_0, \nu_0, \lambda_0) \propto \mathcal{N}(\boldsymbol{\mu}_t | \boldsymbol{\mu}_t^*, (\lambda_t^* \boldsymbol{\Lambda}_t)^{-1}) \mathcal{W}(\boldsymbol{\Lambda}_t | \boldsymbol{W}_t^*, \nu_t^*) \tag{2}$$

$$P(z_{\boldsymbol{d},n} | \boldsymbol{Z}_{\neg(\boldsymbol{d},n)}, \boldsymbol{C}, \boldsymbol{D}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\gamma}, \boldsymbol{\mu}_0, \boldsymbol{W}_0, \nu_0, \lambda_0) \propto \frac{1}{\sqrt{(2\pi)^H |\boldsymbol{\Lambda}_t|^{-1}}} e^{-\frac{1}{2}(\boldsymbol{w}_{\boldsymbol{d},n} - \boldsymbol{\mu}_t)^T \boldsymbol{\Lambda}_t (\boldsymbol{w}_{\boldsymbol{d},n} - \boldsymbol{\mu}_t)}. \tag{3}$$

$$P(c_{\boldsymbol{d}} | \boldsymbol{C}_{\neg \boldsymbol{d}}, \boldsymbol{Z}, \boldsymbol{D}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\gamma}, \boldsymbol{\mu}_0, \boldsymbol{W}_0, \nu_0, \lambda_0) \propto \frac{\Delta(\boldsymbol{n}_k + \boldsymbol{\gamma})}{\Delta(\boldsymbol{\gamma})} \cdot \frac{n^{(k)} - 1 + \tau_k}{\left(\sum_{k'=1}^{K} n^{(k')} + \tau_{k'}\right) - 1} \tag{4}$$

Fig. 3. The full conditionals of the collapsed Gibbs sampling in Algorithm 1.

$$\boldsymbol{\pi}_k = \frac{n^{(k)} + \tau_k}{\sum_{k'=1}^{K} n^{(k')} + \tau_{k'}} \tag{5}$$

$$\boldsymbol{\theta}_{\boldsymbol{k},\boldsymbol{t}} = \frac{n_k^{(t)} + \gamma_t}{\sum_{t'=1}^{T} n_k^{(t')} + \gamma_{t'}} \tag{6}$$

Fig. 4. Equations for parameter estimation.

which are originally placed inside one of the biggest-in-size categories. We drew a sample of 7,674 text documents, which are individually located inside one of the 8 biggest-in-size categories.

The above document corpora were suitably cleaned. The non-alphabetic characters were removed along with stop words; the alphabetic characters were lower-cased; all words consisting of less than 3 alphabetic characters were discarded; words were neglected, if their document occurrences are less than 5 in *Reuters-21578* and 10 in *20-Newsgroups*.

Word vectors were learnt on an auxiliary large-scale corpus from English Wikipedia by means of *word2vec*[4] [32]. The size H of word vectors was uniformly set to 50 for all tests and involved competitors.

5.2. Competitors

This approach was compared against a broad selection of state-of-the-art competitors. These are organized into four baseline groups.

The first group includes three approaches to text clustering, i.e., *k*-means, HAC as well as NMF [1]. *k*-means is a well-known technique for partitional clustering. HAC is the agglomerative variant of the hierarchical clustering, which has been extensively considered for its proven effectiveness. NMF is a feature-transformation method, which allows for document clustering via non-negative matrix factorization.

The second group involves two topic models, i.e., LDA as well as Gaussian LDA. Under LDA, documents are mixtures of topics, while topics correspond to word rankings [9]. Gaussian LDA [18] extends LDA by modeling topics as multivariate Gaussian distributions on the space of word vectors.

The third group consists of several pipelines of competitors from the previous two groups. There are two types of these pipelines. The first type is meant to cluster a semantic representation of text corpora, in which documents are characterized as their corresponding topic mixtures. All pipelines of the first type are detailed in Table 1. The second type is conceived to unveil the latent semantics of document clusters, which are previously identified by suitably grouping text corpora. All pipelines of the second type are detailed in Table 2.

[4] https://code.google.com/p/word2vec/.

Table 1
Pipelines designed for performing text clustering on a preliminary topic modeling.

Pipeline	Description
LDA → HAC	HAC clusters documents by their topics, as captured through LDA
LDA → *k*-means	*k*-means clusters documents by their topics, as captured through LDA
Gaussian LDA → HAC	HAC clusters documents by their topics, as captured through Gaussian LDA
Gaussian LDA → *k*-means	*k*-means clusters documents by their topics, as captured through Gaussian LDA

Table 2
Pipelines designed for performing topic modeling on a preliminary text clustering.

Pipeline	Description
HAC → LDA	LDA discovers topics within each cluster formed by HAC
HAC → Gaussian LDA	Gaussian LDA discovers topics within each cluster formed by HAC
k-means → LDA	LDA discovers topics within each cluster formed by *k*-means
k-means → Gaussian LDA	Gaussian LDA discovers topics within each cluster formed by *k*-means
NMF → LDA	LDA discovers topics within each cluster formed by NMF
NMF → Gaussian LDA	Gaussian LDA discovers topics within each cluster formed by NMF

The fourth group contains MGCTM [46], i.e., a cutting-edge approach to the integration of text clustering with topic modeling.

The above four groups of competitors are functional to the pursuit of the evaluation goals, which are listed at the beginning of Section 5. In particular, the comparison of our approach against the baselines of the first two groups substantiates whether coupling the two tasks is more effective than each task alone. Also, the relative assessment against the baselines of the third group corroborates whether coupling the two tasks is more effective with respect to arranging both in a pipeline. The comparison with MGCTM sheds light on the rationality of the modeling choices behind DISCOVER, in addition to clarifying whether these lead to a competitive gain with respect to the current state of the art.

5.3. Text clustering

The performance of competitors in clustering *20-Newsgroups* and *Reuters-21578* were studied. More precisely, their effectiveness at capturing the ground-truth categories underlying the selected text corpora were assessed.

Competitors looked for as many document clusters in the chosen text corpora as the respective ground-truth categories. NMF, LDA and Gaussian LDA are not explicitly conceived for text clustering; nonetheless, these competitors were still used for clustering the chosen corpora. To this end, LDA and Gaussian LDA were evaluated alone, while NMF was evaluated both alone and as a component of the pipelines NMF → LDA as well as NMF → Gaussian LDA. In these experiments, the individual latent topics[5] were viewed as clusters and, consequently, their number was set equal to the number of ground-truth categories. Each text document was placed in the cluster corresponding to the most relevant topic in its semantics, as caught by NMF, LDA and Gaussian LDA [30].

LDA and Gaussian LDA were also naturally used for topic modeling, either alone or as components of the pipelines in Table 1 and Table 2. Following [46], in these tests, the number of latent topics was set to 120 on *20-Newsgroups* and 60 on *Reuters-21578*. The same settings were also retained to fix the number of latent topics under DISCOVER.

As far as MGCTM is concerned, all of its parameters were set on the chosen text corpora as described in [46].

The effectiveness of all competitors was measured through two widely adopted metrics, i.e., *Accuracy* and *NMI* (Normalized Mutual Information) [12,13,46,48]. The values of both metrics are in the range $[0, 1]$, with larger values being indicative

[5] In the application of NMF to text data, the basis vectors correspond to topics.

Table 3
Clustering accuracy of all competitors.

Corpus	Approach	*Accuracy*
20-Newsgroups	HAC	0.2580
	k-means	0.2662
	NMF	0.2474
	LDA	0.3036
	Gaussian LDA	0.3618
	LDA → HAC	0.3243
	LDA → *k*-means	0.3793
	Gaussian LDA → HAC	0.3731
	Gaussian LDA → *k*-means	0.3859
	DISCOVER	0.4732
	MGCTM	0.4089
Reuters-21578	HAC	0.3927
	k-means	0.3715
	NMF	0.5007
	LDA	0.5311
	Gaussian LDA	0.5427
	LDA → HAC	0.5532
	LDA → *k*-means	0.3264
	Gaussian LDA → HAC	0.5592
	Gaussian LDA → *k*-means	0.3416
	DISCOVER	0.5983
	MGCTM	0.5641

of better document partitions. Table 3 and Table 4 summarize the observed *Accuracy* and *NMI* for all competitors on *Reuters-21578* as well as *20-Newsgroups*.

HAC and *k*-means are less effective than MGCTM and DISCOVER. This substantiates the improvement of text clustering when the latter is seamlessly paired to topic modeling. MGCTM and DISCOVER are aware of document semantics and, hence, more effective in document partitioning.

Word vectors refine the effectiveness of the pipelines for text clustering. Gaussian LDA → HAC and Gaussian LDA → *k*-means are more effective in comparison to their counterparts LDA → HAC and LDA → *k*-means, which are instead unaware of word vectors.

LDA → HAC, LDA → *k*-means, Gaussian LDA → HAC and Gaussian LDA → *k*-means benefit of topic modeling. Nonetheless, such competitors are not so effective as DISCOVER and MGCTM. Even the incorporation of word vectors into Gaussian LDA → HAC and Gaussian LDA → *k*-means does not compensate for the loss in effectiveness with respect to DISCOVER and MGCTM. This finding demonstrates the usefulness of devising a synergy between text clustering as well as topic modeling, rather than trivially pipelining both without a mutuality. By looking at Table 3 and Table 4, one can notice that pipelining does not necessarily imply a gain in effectiveness with respect to text clustering alone. The absence of a mutuality may be harmful to clustering effectiveness.

The effectiveness of NMF, LDA as well as Gaussian LDA is lower than the effectiveness of MGCTM and DISCOVER. This finding is due to two limitations. Firstly, when performing text clustering, NMF, LDA as well as Gaussian LDA interpret document semantics through a number of topics, which equals the number of ground-truth categories. This is likely to negatively affect the effectiveness of text clustering. An underestimation/overestimation of the number of topics determines an unrealistically limited/inflated interpretation of document semantics. Secondly, under NMF, LDA as well as Gaussian LDA, each text document is assigned to one cluster. The latter simplistically corresponds to the most important topic of the particular document, rather than reflecting its cross-topic similarity to the other documents of the same corpus. Noticeably, even the incorporation of word vectors into Gaussian LDA does not cope with the above two limitations.

DISCOVER is the most effective on *20-Newsgroups* as well as *Reuters-21578* among all competitors. DISCOVER overcomes MGCTM in effectiveness. This evidence corroborates the rationality of the design of DISCOVER along with its underlying ideas (i.e., the exploitation of word vectors, the explicit assignment of semantic topics to clusters as well as the devised pairing of text clustering with topic modeling).

Table 4
Clustering NMI of all competitors.

Corpus	Approach	*NMI*
20-Newsgroups	HAC	0.2330
	k-means	0.2412
	NMF	0.2060
	LDA	0.2997
	Gaussian LDA	0.3316
	LDA → HAC	0.3388
	LDA → *k*-means	0.3993
	Gaussian LDA → HAC	0.3821
	Gaussian LDA → *k*-means	0.4032
	DISCOVER	0.4814
	MGCTM	0.4112
Reuters-21578	HAC	0.1282
	k-means	0.3762
	NMF	0.3215
	LDA	0.4174
	Gaussian LDA	0.4252
	LDA → HAC	0.2915
	LDA → *k*-means	0.3464
	Gaussian LDA → HAC	0.4361
	Gaussian LDA → *k*-means	0.3528
	DISCOVER	0.5139
	MGCTM	0.4568

5.4. Topic modeling

The evaluation of our approach in topic modeling was quantitative as well as qualitative.

5.4.1. Quantitative evaluation

The assessment of topic models involves choosing suitable evaluation criteria. These divide into intrinsic and extrinsic. Among the quantitative intrinsic ones, held-out likelihood and perplexity are often adopted for this purpose [44]. These measures are not necessarily good predictors of human judgment [16]. The latter was explicitly involved in the assessment of topic coherence in [46]. The extrinsic criteria encompass exploiting the uncovered topics for performing an external task [33].

In this article, we evaluated and ranked competitor performance by resorting to both intrinsic and extrinsic criteria. The intrinsic criterion is semantic coherence [37,39]. The extrinsic criterion involves the exploitation of topics in the context of a supervised classification task. The assessment of competitor performance according to both criteria is detailed below.

In our tests, semantic coherence (SC) was computed as the mean of the coherence of all inferred topics, namely $SC = \frac{1}{T}\sum_{t=1}^{T} SC^{(t)}$. In turn, under LDA, Gaussian LDA, MGCTM and DISCOVER, the coherence $SC^{(t)}$ of any topic t was quantified with the metric presented in [33]. This metric is a well-known quantitative intrinsic criterion, that still satisfactorily agrees with human assessments of coherence. Formally, assume that $\boldsymbol{w}_{t,1}, \ldots, \boldsymbol{w}_{t,R}$ are the top-R most probable words of topic t. The semantic coherence $SC^{(t)}$ of topic t is defined beneath

$$SC^{(t)} = \sum_{r=2}^{R}\sum_{p=1}^{r-1} log\frac{f(\boldsymbol{w}_{t,r}, \boldsymbol{w}_{t,p}) + 1}{f(\boldsymbol{w}_{t,p})}$$

with $f(\boldsymbol{w}_{t,p})$ being the document frequency of word $\boldsymbol{w}_{t,p}$ and $f(\boldsymbol{w}_{t,r}, \boldsymbol{w}_{t,p})$ the co-document frequency of words $\boldsymbol{w}_{t,r}$ and $\boldsymbol{w}_{t,p}$ [33].

The topic coherence above required an adjustment under the pipelines of Table 2. These pipelines repeatedly infer the individual latent topics from each input cluster; therefore, we adapted $SC^{(t)}$ as reported below

Table 5
Semantic coherence of all competitors.

Corpus	Approach	*SC*
20-Newsgroups	LDA	−207.62
	Gaussian LDA	−202.18
	HAC → LDA	−205.22
	HAC → Gaussian LDA	−201.63
	k-means → LDA	−206.51
	k-means → Gaussian LDA	−201.88
	NMF → LDA	−205.93
	NMF → Gaussian LDA	−202.04
	MGCTM	−200.46
	DISCOVER	−196.82
Reuters-21578	LDA	−181.41
	Gaussian LDA	−175.83
	HAC → LDA	−178.25
	HAC → Gaussian LDA	−174.37
	k-means → LDA	−179.44
	k-means → Gaussian LDA	−175.71
	NMF → LDA	−178.91
	NMF → Gaussian LDA	−176.15
	MGCTM	−172.52
	DISCOVER	−169.14

$$SC^{(t)} = \frac{1}{K}\sum_{k=1}^{K}\sum_{r=2}^{R}\sum_{p=1}^{r-1} log\frac{f\left(\boldsymbol{w}_{t^{(k)},r}, \boldsymbol{w}_{t^{(k)},p}\right)+1}{f\left(\boldsymbol{w}_{t^{(k)},p}\right)}$$

where $t^{(k)}$ is the topic t inferred from the input cluster k.

Larger values of *SC* correspond to more semantically coherent topics. The *SC* scores of all competitors were calculated, by maintaining the parameter settings detailed in Section 5.3. In particular, the coherence of the inferred topics was computed by taking into account their top-15 most probable words. Table 5 summarizes the observed *SC* scores for all competitors.

MGCTM and DISCOVER infer the most semantically coherent topics among all competitors.

The higher semantic coherence attained by MGCTM and DISCOVER in comparison with LDA as well as Gaussian LDA substantiates the benefit of pairing topic modeling to text clustering, instead of performing topic modeling alone. The incorporation of word vectors into Gaussian LDA improves its semantic coherence with respect to LDA. Despite this, under Gaussian LDA, the awareness of word vectors cannot counterbalance the absence of any interaction with text clustering. The latter task is found to improve sensibly the semantic coherence of MGCTM and DISCOVER.

All pipelines for topic modeling based on word vectors (i.e., HAC → Gaussian LDA, *k*-means → Gaussian LDA and NMF → Gaussian LDA) infer more semantically-coherent topics than their counterparts devoid of word vectors (i.e., HAC → LDA, *k*-

Table 6
Classification effectiveness enabled by the inferred topics.

Corpus	Model	Precision	Recall	F-Measure
20-Newsgroups	TFIDF	0.6223	0.6022	0.6120
	LDA	0.6539	0.6520	0.6525
	DISCOVER	0.6728	0.6618	0.6672
Reuters-21578	TFIDF	0.7985	0.7322	0.7639
	LDA	0.8114	0.7845	0.7977
	DISCOVER	0.8062	0.7926	0.7993

Table 7
Excerpt of output inspection on *20-Newsgroups*.

Cluster 1		Cluster 2	
Topic 1	Topic 2	Topic 3	Topic 4
church	god	war	military
catholic	bible	attack	armenian
holy	jesus	russian	extermination
spirit	christian	government	turkey
father	life	peace	people
paul	believe	soviet	genocide
pope	lord	soldiers	azerbaijan
revelation	faith	army	captain
orthodox	love	troops	weapon
theology	sin	villages	civilians

means → LDA as well as NMF → LDA). Although aware of word vectors, the foresaid pipelines infer less semantically-coherent topics compared to MGCTM and DISCOVER. This is due to the lack of interaction between topic modeling and text clustering. More precisely, text clustering ignores topic modeling; hence, text documents are not suitably clustered by their respective semantics, which eventually makes the above pipelines infer less semantically coherent topics with respect to MGCTM and DISCOVER. This finding demonstrates that a seamless integration of topic modeling and text clustering is advantageous compared with naively pipelining both. In accordance with the results of Table 3 and Table 4, pipelining was again found to be not necessarily beneficial. In this regard, Table 5 shows that performing topic modeling on text clustering does not necessarily imply a gain in semantic coherence with respect to topic modeling alone. The lack of a mutuality can be harmful to semantic coherence.

DISCOVER achieves the highest semantic coherence with respect to all competitors. Mirroring the results of Table 3 and Table 4, DISCOVER overcomes MGCTM in sematic coherence. This confirms the rationality of the design of DISCOVER along with the underlying modeling choices.

In order to supplement the findings in Table 5, the classification performance enabled by the inferred topics was further investigated with the aim of evaluating the effectiveness of an SVM classifier at labeling the text documents of the chosen text corpora with their respective ground-truth categories. Accordingly, three experiments were run. In two out of the three experiments, the SVM classifier was trained over the topic distributions of the individual text documents, labeled by the respective ground-truth categories. These distributions were previously inferred through DISCOVER and LDA, respectively. For both competitors, the number of topics was set to 120 on *20-Newsgroups* and 60 on *Reuters-21578* [46]. The number of clusters under DISCOVER was set to the number of ground-truth categories in *20-Newsgroups* and *Reuters-21578*. In the third experiment, the SVM classifier was trained over the *bag-of-words* representation of the individual text documents (according to common practice in supervised document classification), labeled by the respective ground-truth categories. The popular *TFIDF* weighting scheme was adopted to capture word saliency in the foresaid *bag-of-words* representations.

Table 6 reports the results of the three classification tests in terms of precision, recall and F-measure. These measures were averaged across the ten-fold cross validation of the SVM classifier effectiveness. Higher values of the averaged precision, recall and F-measure indicate a higher effectiveness of the SVM classifier. On the chosen text corpora, LDA enables a more effective classification compared to TFIDF, since the topic distributions better capture text semantics with respect to the TFIDF weighting of raw text. DISCOVER allows for the most effective classification performance, since the topic distributions under DISCOVER capture the particular semantics of the disjoint groups of homogeneous documents in the underlying text corpora. This proves the benefit of simultaneously uncovering text topics and clusters in an interdependent manner.

5.4.2. Qualitative evaluation

The output of our approach on real-world document corpora is elucidated by inspecting the results observed on *20-Newsgroups*. Table 7 presents an insightful explanation of the discovered clusters along with their semantics and individual topics. For convenience of presentation, in Table 7, two are the inspected clusters. Their semantics is characterized through the top-2 most relevant topics. Also, these topics are narrowed to their top-10 most representative words. The rankings of these words were obtained by means of the posterior topic means and precisions of Eq. (2).

DISCOVER finds clusters with an easily comprehensible semantics. This stems from the intelligibility of topics, the clarity and specificity of their words, and the coherence of these words in the respective topics.

Table 7 also confirms the intra-cluster coherence enforced by DISCOVER. Each cluster is semantically discriminated by a corresponding subset of closely-related topics. The meaning of *Cluster 1* is mostly imputable to *Topic 4* and *Topic 13*. These are well-assorted topics, according to which *Cluster 1* can be intuitively interpreted as basically devoted to *Christianity*. *Topic 97* and *Topic 98* intuitively explain *Cluster 2* as mainly devoted to the *Turkish-Armenian War*.

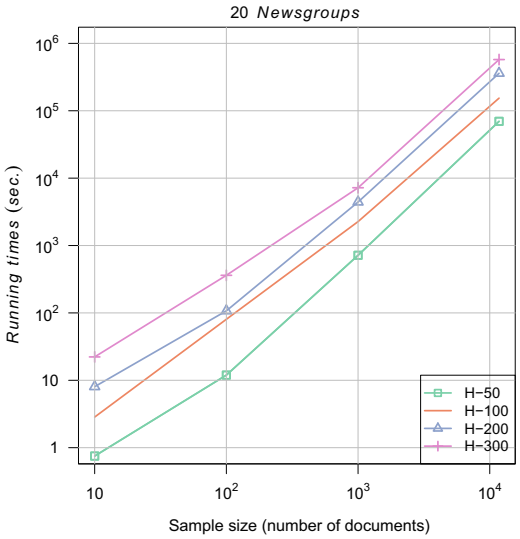

Fig. 5. Empirical runtime analysis.

Table 8
Main differences between DISCOVER and MGCTM.

	MGCTM [46]	DISCOVER
Task	integration of text clustering and topic modeling	integration of text clustering and topic modeling with word vectors
Topic types	local and global [46]	one undiversified set
Word vectors	✗	✓
Cluster distribution type	multinomial distribution over clusters	multinomial distribution over clusters
Conjugate prior for cluster distribution	✗	✓(Dirichlet distribution)
Topic characterization	multinomial distributions over discrete words	multivariate Gaussian distributions over the space of real-valued word vectors
Conjugate prior for topic distributions	✗	✓(Gaussian-Wishart prior)
Summarization/explanation of cluster semantics	✗	✓
Posterior inference	variational inference	collapsed Gibbs sampling
Hyperparameters	different local hyperparameters plus one global hyperparameter	one hyperparameter

5.5. Time efficiency and scalability

The time efficiency and scalability of our approach with the size of both the underlying corpus and the word vectors were thoroughly investigated. To this end, the runtime for posterior inference under DISCOVER was evaluated over increasingly larger.

- samples of *20-Newsgroups*, whose size ranges from 10 to 11,268 text documents;
- word vectors, whose dimensionality varies from 50 to 300.

Fig. 5 shows the resulting scalability on a Linux machine, equipped with an Intel Xeon (Eight-Core) E5 processor and 32 GB RAM. The observed scalability is mainly due to the fact that modeling topics as multivariate Gaussian distributions on the space of word vectors implies the computation of $\boldsymbol{\mu}_t^*, \lambda_t^*, \nu_t^*$ and $\left[\boldsymbol{W}_t^*\right]^{-1}$, which is partly mitigated by resorting to Cholesky decomposition [18,28]. Besides, Gibbs sampling implies a higher computational cost on larger corpora.

6. Related works

Topic models are meant to represent and uncover the themes of a text corpus [7,41]. The spectrum of topic model applications is very wide, encompassing information retrieval, natural language processing, computer vision, relevance judgments, social media analysis, sentiment analysis as well as geographic topic modeling [9,22,19,15,26,31,11,17]. There are two broad families of topic models: *traditional* and *enhanced*. Traditional topic models, such as [25,9,8,42,38], do not capture the syntactic and semantic relationships between words. Instead, enhanced topic models, such as [18,29,49], rely on word vectors to explicitly consider word regularities. DISCOVER shares the exploitation of word vectors with [18]; nonetheless, DISCOVER is substantially different from both families of topic models since these are not conceived to be synergically paired to text clustering.

Text clustering [5] is generally exploited with the aim to organize, browse, summarize, classify and visualize document corpora [1,2]. Document partitioning has been implemented by means of various techniques, including spectral methods [35], hierarchical methods [40], partitional methods [14] and matrix factorization [48,47]. DISCOVER differs from such approaches to text clustering, since these are not conceived to be synergically paired to topic modeling.

Topic models are used for clustering purposes in [30,50,36]. In these studies, topics are interpreted as clusters and, thus, documents are assigned to their most relevant semantic topic. Unlike [30,50,36], DISCOVER seamlessly integrates topic modeling and text clustering.

Text clustering is paired with topic modeling also in MGCTM [46]. The devised approach is substantially different from MGCTM [46], as emphasized in Table 8. Regarding model design, MGCTM explicitly deals with the text units of documents, which hinders the identification of the syntactic and also semantic relationships between words. Under MGCTM, clusters are not assigned semantic descriptions, thus not being immediately intelligible. Also, MGCTM uses local as well as global topics. This requires determining the number of topics of the two types, in addition to a demanding hyperparameter tuning. The distribution over clusters under MGCTM lacks a prior. On the contrary, the devised approach conceives documents as made up of word vectors. Consequently, topics are characterized as multivariate Gaussian distributions on the space of word vectors. This choice allows for the awareness of the syntactic and also semantic relationships between words. Furthermore, one undistinguished type of topics is exploited, to characterize the inherent semantics of clusters, so that their coherence is enforced. There is only one hyperparameter for the topic distributions of the individual clusters, which simplifies both tuning and inference. Yet, the multinomial probability distribution over clusters is drawn from a Dirichlet prior. Lastly, as far as inference is concerned, MGCTM adopts variational inference. Instead, our approach relies on collapsed Gibbs sampling with parameter estimation.

7. Conclusions

A new model-based approach is presented to combining text clustering with topic modeling. Both were seamlessly and synergically integrated under DISCOVER, a Bayesian probabilistic generative model of text collections. Under DISCOVER, documents comprise word vectors in order for the syntactic and also semantic regularities across words to be captured. Document clusters and their topics are intended as interacting latent factors, which rule content generation. These latent factors are unveiled via posterior inference. The latter implicitly amounts to performing text clustering simultaneously with topic modeling. An algorithm was designed for approximate posterior inference, which implements the derived mathematical details of collapsed Gibbs sampling with parameter estimation.

An extensive comparative experimentation over real-world benchmark text corpora revealed the effectiveness of DISCOVER in clustering text corpora and discovering their topics. The topic modeling performance was studied according to intrinsic and extrinsic criteria corresponding to, respectively, semantic coherence and effectiveness of document classification by semantics. A case study was developed to demonstrate and discuss the results of topic modeling. The time efficiency and scalability of DISCOVER with the size of both the text corpus and the word vectors were also investigated.

Future research mainly aims at the incorporation of topic correlations [49] into DISCOVER. Finally, an effort of great practical interest is the design of new joint models of text topics and clusters, in which Bayesian nonparametrics [21] enables the automatic tuning of the number of components of either type. This would avoid the need for corresponding input values, which are in general hardly determined beforehand.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Gianni Costa: Conceptualization, Methodology, Software, Writing - original draft, Visualization, Investigation, Supervision, Software, Validation, Writing - review & editing. **Riccardo Ortale:** Conceptualization, Methodology, Software, Writing - original draft, Visualization, Investigation, Supervision, Software, Validation, Writing - review & editing.

References

[1] C. Aggarwal, C. Zhai, A survey of text clustering algorithms, in: C. Aggarwal, C. Zhai (Eds.), Mining Text Data, Springer, Boston, MA, 2012, pp. 77–128.
[2] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E.D. Trippe, J.B. Gutierrez, K. Kochut, A brief survey of text mining: classification, clustering and extraction techniques. In arXiv preprint arXiv:1707.02919, 2017. .
[3] C. Andrieu, N. De Freitas, A. Doucet, M.I. Jordan, An introduction to mcmc for machine learning, Mach. Learn. 50 (1–2) (2003) 5–43.
[4] Y. Bengio, R. Ducharme, P. Vincent, C. Janvin, A neural probabilistic language model, J. Mach. Learn. Res. 3 (2003) 1137–1155.
[5] P. Berkhin, Grouping Multidimensional Data, chapter A Survey of Clustering Data Mining Techniques, Springer, Berlin, Heidelberg, 2006, pp. 25–71.
[6] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[7] D. Blei, J. Lafferty, Text Mining: Classification, Clustering, and Applications, chapter Topic Models, Chapman & Hall/CRC Data Mining and Knowledge Discovery Series (2009) 71–94.
[8] D.M. Blei, J.D. Lafferty, Correlated topic models, in: Proc. of Advances in Neural Information Processing Systems, 2005, pp. 147–154. .
[9] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.
[10] G.E.P. Box, G.C. Tiao, Bayesian Inference in Statistical Analysis, Wiley-Interscience, 1992.
[11] J. Boyd-Graber, Y. Hu, D. Mimno, Applications of topic models, Found. Trends Inf. Retrieval 11 (2–3) (2017) 143–296.
[12] D. Cai, X. He, J. Han, Document clustering using locality preserving indexing, IEEE Trans. Knowl. Data Eng. 17 (12) (2005) 1624–1637.
[13] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, IEEE Trans. Knowl. Data Eng. 23 (6) (2011) 902–913.
[14] M.E. Celebi, editor. Partitional Clustering Algorithms, Springer International Publishing, 2015.
[15] Y. Cha, J. Cho, Social-network analysis using topic models, in: Proc. of Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, 2012, pp. 565–574. .
[16] J. Chang, J. Boyd-Graber, S. Gerrish, C. Wang, D.M. Blei, Reading tea leaves: how humans interpret topic models, in: Proc. of Int. Conf. on Neural Information Processing Systems, 2009, pp. 288–296.
[17] G. Costa, R. Ortale, Marrying community discovery and role analysis in social media via topic modeling, Proc. of Pacific-Asia Conference on Knowledge Discovery and Data Mining (2018).
[18] R. Das, M. Zaheer, C. Dyer, Gaussian lda for topic models with word embeddings, in: Proc. of the Meeting of the Association for Computational Linguistics, 2015, pp. 795–804.
[19] L. Dietz, S. Bickel, T. Scheffer, Unsupervised prediction of citation influences, in: Proc. of Int. Conf. on Machine learning, 2007, pp. 233–240. .
[20] A. Gelman, J.B. Carlin, H.S. Stern, D.B. Rubin, D.B. Dunson, Bayesian Data Analysis, Chapman and Hall/CRC, 2013.
[21] S.J. Gershman, D.M. Blei, A tutorial on bayesian nonparametric models, J. Math. Psychol. 56 (1) (2012) 1–12.
[22] T.L. Griffiths, M. Steyvers, Finding scientific topics, in: Proc. of the National Academy of Sciences of the United States of America, 2004, pp. 5228–5235. .
[23] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer-Verlag, New York, 2009.
[24] G. Heinrich. Parameter estimation for text analysis. Technical report, University of Leipzig, 2008. Available at http://www.arbylon.net/publications/text-est.pdf. .
[25] T. Hofmann, Probabilistic latent semantic indexing, in: Proc. of Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, 1999, pp. 50–57. .
[26] L. Hong, A. Ahmed, S. Gurumurthy, A.J. Smola, K. Tsioutsiouliklis, Discovering geographical topics in the twitter stream, in: Proc. of Int. Conf. on World Wide Web, 2012, pp. 769–778. .
[27] D. Koller, N. Friedman, Probabilistic Graphical Models. Principles and Techniques, The MIT Press, 2009.
[28] A. Krishnamoorthy, D. Menon, Matrix inversion using cholesky decomposition, in: Proc. of IEEE Int. Conf. on Signal Processing: Algorithms, Architectures, Arrangements, and Applications, 2013, pp. 70–72. .
[29] S. Li, T.-S. Chua, J. Zhu, C. Miao, Generative topic embedding: a continuous representation of documents, in: Proc. of the Meeting of the Association for Computational Linguistics, 2016, pp. 666–675.
[30] Y. Lu, Q. Mei, C. Zhai, Investigating task performance of probabilistic topic models: an empirical study of plsa and lda, Inf. Retrieval 14 (2) (2011) 178–203.
[31] W. Luo, B. Stenger, X. Zhao, T.-K. Kim, Automatic topic discovery for multi-object tracking, in: Proc. of AAAI Conf. on Artificial Intelligence, 2015, pp. 3820–3826. .
[32] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proc. of Int. Conf. on Neural Information Processing Systems, 2013, pp. 3111–3119.
[33] D. Mimno, H.M. Wallach, E. Talley, M. Leenders, A. McCallum, Optimizing semantic coherence in topic models, in: Proc. of Conf. on Empirical Methods in Natural Language Processing, 2011, pp. 262–272. .
[34] K.P. Murphy, Machine Learning: A Probabilistic Perspective, The MIT Press, 2012.
[35] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: Proc. of Advances in, 2001, pp. 849–856. .
[36] D.Q. Nguyen, R. Billingsley, L. Du, M. Johnson, Improving topic models with latent feature word representations, Trans. Assoc. Comput. Linguist. 3 (2015) 299–313.
[37] M. Röder, A. Both, A. Hinneburg, Exploring the space of topic coherence measures, in: Proc. of ACM Int. Conf. on Web Search and Data Mining, 2015, pp. 399–408. .
[38] M. Rosen-Zvi, C. Chemudugunta, T. Griffiths, P. Smyth, M. Steyvers, Learning author-topic models from text corpora, ACM Trans. Inf. Syst. 28(1) (2010) 4:1 – 4:38. .
[39] F. Rosner, A. Hinneburg, M. Röder, M. Nettling, A. Both, Evaluating topic coherence measures. In arXiv:1403.6397. .
[40] N. Sahoo, J. Callan, R. Krishnan, G. Duncan, R. Padman, Incremental hierarchical clustering of text documents, in: Proc. of ACM Int. Conf. on Information and Knowledge Management, 2006, pp. 357–366.
[41] M. Steyvers, T. Griffiths, Latent Semantic Analysis: A Road to Meaning, chapter Probabilistic Topic Models. Lawrence Erlbaum, 2007, pp. 427–448. .
[42] H.M. Wallach, Topic modeling: beyond bag-of-words, in: Proc. of Int. Conf. on Machine Learning, 2006, pp. 977–984. .
[43] H.M. Wallach. Structured Topic Models for Language (Ph.D. thesis), University of Cambridge, 2008. .
[44] H.M. Wallach, I. Murray, R. Salakhutdinov, D. Mimno, Evaluation methods for topic models, in: Proc. of Int. Conf. on Machine Learning, 2009, pp. 1105–1112. .
[45] R. Winkler, An Introduction to Bayesian Inference and Decision, Probabilistic Publishing (2003).
[46] P. Xie and E.P. Xing. Integrating document clustering and topic modeling. In Proc. of Int. Conf. on Uncertainty in Artificial Intelligence, pages 694–703, 2013. .
[47] W. Xu, Y. Gong, Document clustering by concept factorization, in: Proc. of Int. ACM SIGIR Conf. on Research and Development in Informaion Retrieval, 2004, pp. 202–209. .
[48] W. Xu, X. Liu, Y. Gong, Document clustering based on non-negative matrix factorization, in: Proc. of Int. ACM SIGIR Conf. on Research and Development in Informaion Retrieval, 2003, pp. 267–273. .
[49] G. Xun, Y. Li, W.X. Zhao, J. Gao, A. Zhang, A correlated topic model using word embeddings. In Proc. of Int. Joint Conf. on Artificial Intelligence, 2017, pp. 4207–4213. .
[50] X. Yan, J. Guo, Y. Lan, X. Cheng, A biterm topic model for short texts, in: Proc. of Int. Conf. on World Wide Web, 2013, pp. 1445–1456. .