Topic modeling algorithms and applications: A survey

Article *in* Information Systems · October 2022

DOI: 10.1016/j.is.2022.102131

5 authors, including:

Aly Abdelrazek
Nile University
3 PUBLICATIONS **93** CITATIONS
SEE PROFILE

Yomna Eid
Nile University
8 PUBLICATIONS **108** CITATIONS
SEE PROFILE

Walaa Medhat
Nile University
50 PUBLICATIONS **3,331** CITATIONS
SEE PROFILE

Ahmed Hassan Yousef
Egypt University of Informatics
108 PUBLICATIONS **3,850** CITATIONS
SEE PROFILE

Topic modeling algorithms and applications: A survey

Aly Abdelrazek[a,*], Yomna Eid[a], Eman Gawish[a], Walaa Medhat[a], Ahmed Hassan[a]

[a]*Information Technology and Computer Science, CIS, Nile University, Giza, Egypt.*

Abstract

Topic modeling is used in information retrieval to infer the hidden themes in a collection of documents and thus provides an automatic means to organize, understand and summarize large collections of textual information. Topic models also offer an interpretable representation of documents used in several downstream Natural Language Processing (NLP) tasks. Modeling techniques vary from probabilistic graphical models to the more recent neural models. This paper surveys topic models from four aspects. The first aspect categorizes different topic modeling techniques into four categories: algebraic, fuzzy, probabilistic, and neural. We review the wide variety of available models from each category, highlight differences and similarities between models and model categories using a unified perspective, investigate these models' characteristics and limitations, and discuss their proper use cases. The second aspect illustrates six criteria for proper evaluation of topic models, from modeling quality to interpretability, stability, efficiency, and beyond. Topic modeling has found applications in various disciplines, owing to its interpretability. We examine these applications along with some popular software tools which provide an implementation of some models. The fourth aspect reviews available datasets and benchmarks. Using two benchmark datasets, we conducted experiments to compare seven topic models along the proposed metrics. The discussion highlights the differences between the models and their relative suitability for various applications. It notes the relationship between evaluation metrics and proposes four key aspects to help decide which model to use for an application. Our discussion also shows that the research trends move towards developing and tuning neural topic models and leveraging the power of pre-trained language models. Finally, it highlights research gaps in developing unified benchmarks and evaluation metrics.

Keywords: Topic Modeling, Neural, Probabilistic, Evaluation, LDA

1. Introduction

Topic modeling (TM) has been used successfully in mining large text corpora where a topic model takes a collection of documents as an input and then attempts, without supervision, to uncover the underlying topics in this collection [1]. Each topic describes a human-interpretable semantic concept. TM then provides a latent interpretable representation of documents according to the conceived topics [2].

Topic modeling is applied in different fields. Natural Language Processing (NLP) tasks have leveraged topic modeling, like text summarization and sentiment analysis. Moreover, the powerfulness of topic modeling is extended to other fields, like bioinformatics and economics. TM is also used with

*Corresponding author
Email address: al.abdelrazek@nu.edu.eg (Aly Abdelrazek)

other sets of data where the concept of words and documents is replaced with similar entities of similar structure. The entities could be items in online shops, segments in images, or genes in gene sets.

Since 1990 [3] with the introduction of Latent Semantic Indexing (LSI), researchers have developed several techniques for topic modeling that vary in the modeling capacity. Moreover, different models make different assumptions about the corpus, the documents' representation, and the topics. The developed topic models occupy a broad spectrum for different use cases. They also exhibit different evaluation metrics scores. More models continue to be developed till now. This continuous development tries to address the current research gaps and extend the TM applications' scope.

According to their underlying modeling techniques, four categories of topic models are recognized: algebraic, fuzzy, probabilistic, and neural. Algebraic topic models were developed first during the 1990s [3]. Then, following the advent of LDA in 2003, most of the developed topic models were Bayesian Probabilistic Topic Models (BPTMs) such as [1][4][5] . These BPTMs have proof of efficiency and dominated the research venues till 2015. The reason is that they are usually simple to deploy, easy to interpret, and modular enough to extend into more complex models. They also are adequately computationally efficient since they have relatively few parameters [1]. Recently, topic models increasingly use neural components [6]. Examples of neural topic models (NTMs) are [7] and [8]. Some NTMs use contextualized representation to represent the documents instead of the usual bag of words [9] [10] [11]. They can use pre-trained models on a large corpus, such as BERT variants [12]. In general, NTM offers flexibility and scalability. It is important to note that different TM categories offer different advantages and are better suited for a particular setting. Therefore, they co-exist side by side.

Due to a large number of available models, a need arises to summarize them. There have been attempts at summarizing the wide range of available topic models in the literature. In [13], two categories were presented. The first category classifies the models into four subgroups, Latent semantic indexing (LSI), Probabilistic latent semantic indexing (PLSI), Latent Dirichlet allocation (LDA), and Correlated topic model (CTM). The second category discusses different topic evolution models over time. In [14], the survey proposed hierarchical categorization criteria where models were classified according to being LDA and non-LDA based, bag-of-words or sequence-of-words approach, and finally, unsupervised or supervised learning models. In [15], different models were discussed in terms of their features and limitations. In [16], the authors categorized topic models into 3 classes. The first discusses traditional topic models like LSI, PLSI, and LDA. The second class focuses on the topic evolution models. Finally, the third class discusses some joint techniques of topic modeling and other algorithms, including LDA-VSM+K-means and LDA-Word2vec+SVM. In [17], the authors have investigated the research work done on LDA topic modeling. A detailed survey on topic modeling for long and short text in the last decade was introduced in [18] whereas the specific category of neural topic models was surveyed in [19].

This paper takes a holistic approach to survey TM. It discusses the four categories of TM and presents the wide variety of available models in a structured perspective. The goal is to identify research gaps and highlight similarities and differences between models and model categories. Furthermore, since evaluation metrics vary in different papers, we propose a comprehensive set of evaluation metrics that can be applied to all models to ensure consistency in models' evaluation. Additionally, available benchmark datasets in topic modeling are reviewed, followed by an overview of different tools and applications of topic modeling. We then experiment with various models from different categories and compare them along the proposed evaluation criteria using two common benchmark datasets. This perspective considers the latest advancements in different topic modeling techniques. Hopefully, it will ignite interest in TM and help the research community develop improved models.

The rest of this paper is organized as follows. Section 2 provides background information and establishes the definitions and notation whereas section 3 provides the methodology. The categorization of models is outlined in section 4 followed by a survey of several models from the categories. Different evaluation criteria of topic models are discussed in section 5. Further, section 6 highlights some applications

of various topic models and introduces popular software tools that implement several models. It also briefly shows the available datasets and benchmarks for topic models. In section 7, we compare various models from different categories along the proposed metrics, and we discuss the results in section 8, where we also note the research trends and highlight the research gaps. Finally, the paper is concluded in section 9.

2. Background and definitions

Topic models attempt to model three entities: constructs, collections, and topics. The constructs are the elements that come together to make a collection. In textual data, constructs are usually words that are grouped to constitute a document or a collection of words. A topic is a cluster of constructs that together describe a pure semantic meaning. It is an idealized notion of a document that is as pure as possible. Also, it is a homogeneous collection whose constructs have so much in common, usually semantically. Mathematically, a topic is described as a probability distribution over the constructs [20].

Most topic models work by observing the co-occurrence of constructs in the collections. In linguistics, this aligns with the distributional hypothesis [21] [22] which states that the distributional properties of tokens dictate their semantic meaning. In other words, a word is characterized by the context in which it appears. Similarly, the co-occurrence of pixels in images defines segments (topics) due to the spatially local correlations between pixels.

TM can be applied to several data forms, but we will only consider their application in text throughout the following sections. In textual data, documents are presumed to be a heterogeneous collection of constructs that span more than one topic. Topic sparsity is also presumed following the heuristic that any one document will usually be about a small number of topics.

Topic models take a corpus D as input and attempt to obtain two sets of distributions: The first set T is for topics: K distributions over V constructs (tokens) where K is the number of topics, usually a hyperparameter, and V is the vocabulary size in the corpus. The second set of distributions Z is for documents, and it is a distribution for each document in corpus over K topics, where for each document, z_k indicates the weight of the k^{th} topic. This second distribution provides an interpretable latent representation of documents (Similar to disentangled Variational Auto Encoder, VAE [23], [24]). These latent representations and T (TM parameters) are learned to best explain the observed documents. This learning is called the inference process. The generative process generates a document from a presumed Z and T.

Note that there are several ways in which collections can be decomposed into topics, leading to unstable outcomes where each model run results in a separate set of topics. Accordingly, model regularization is necessary, and it is discussed in section 5.

TM can uncover polysemy, where a single construct has several semantic meanings [25]. TM captures this by having the same construct showing up in several topics. Synonymity, where two different constructs have similar semantic meaning, can also be captured by TM when different constructs appear together in the same topic in more than one distribution of T.

The representation of documents is different for different models. Some models consider the collection as an unordered set of its constructs. In texts, these models are known as the bag of words models [22]. The input to the topic model, in this case, is a vector of word counts.

$$\boldsymbol{b} \in \mathbf{Z}^{V}_{>=0} \tag{1}$$

Where b vector is sparse, b_v is the number of occurrences of the token (word) $v \in \{1, \ldots, V\}$ in the document. On the other hand, some other models consider the sequence of constructs, which

acknowledges the dependence between the constructs. In texts, an input document to a topic model will thus be represented as a dense vector S of length L.

$$s \in \mathbf{N}^L \tag{2}$$

Where $s_j \in \{1, \ldots, V\}$ is the index of the vocabulary for the j^{th} token (word), $j \in \{1, \ldots, L\}$, L is the length of the document.

Another document representation is the contextual representation that is becoming popular with the ubiquity of transformer-based language models such as BERT, where the document is represented as an embedding vector. BERT models capture contextual word embeddings better than global embeddings as they learn different representations for polysemous words.

3. Methodology

Models are categorized into four categories: algebraic, fuzzy, Bayesian probabilistic, and finally, neural topic models. This categorization is shown in figure 1.

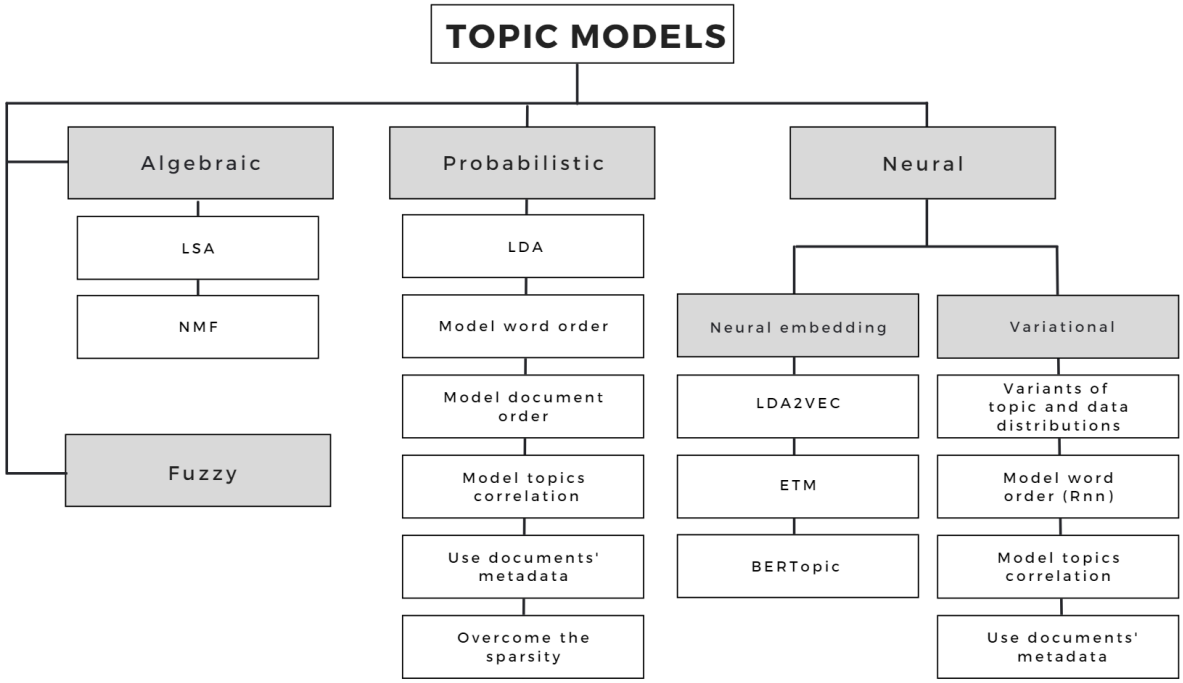


Figure 1: Topic models categorization according to the used technique

The first TM category is the algebraic models. They are simple but do not have a solid statistical foundation, nor do they define a proper generative data model [20]. This linear algebra approach was developed first in [3]. There are more recent approaches, too, such as [26] that use spectral decomposition for inference. Algebraic models are simple and relatively efficient. On the other hand, fuzzy topic models are clustering techniques that target assigning words to associated topics. They have proved their effectiveness in short text documents where they overcome the sparsity problem.

Coming to the Bayesian probabilistic models, they were dominant before 2015, when the research efforts shifted to developing neural topic models (NTMs). Bayesian models are intuitive and extendable.

They define a fictitious generative process about how the observed documents come to be. Then, they work backward to infer the topics that could have created the documents. Bayesian models' parameters tend to have more concrete meaning and thus are easily interpreted. This interpretability of parameters is helpful to troubleshoot a model, validate it, and interpret the results, especially in cross-disciplinary use cases. It is also helpful whenever we want to generate more data. On the other hand, the inference problem in this category of models becomes more complicated with more complex modeling.

Neural Topic Models (NTMs) are more flexible and scalable because, in NTMs, the inference problem is an optimization problem. However, their parameters may not have a meaningful interpretation. It is usually hard to investigate why the model does or does not work. NTMs train at different objective functions, for example, minimizing the reconstruction error of documents or optimizing the model's predictive accuracy. This optimization goal does not necessarily result in high-quality interpretable topics. A comparison between different modeling techniques is presented in table 1.

Table 1: comparison between topic models' categories.

Category	Description	Pros	Cons
Algebraic	Decompose the document-term matrix, then find a low-rank approximation to it.	Simple, intuitive, and computationally relatively efficient. Some adaptations can handle short text documents.	Provides no solid statistical foundation. And does not define a generative data model.
Fuzzy	Extract topics from the documents, then assign the words to these topics/ clusters using a degree of truth instead of Boolean logic "1 or 0".	It can handle sparsity in short text documents (for example, tweets).	Most of the applications focus on medical data.
Bayesian probabilistic	Define a generative process through a Bayesian graphical model (Directed acyclic graph). Then work backward for inference.	Simple, intuitive, extensible, and interpretable.	inference becomes complicated with increased model complexity.
Neural Topic models	Substitute the intractable posterior inference with optimization.	Flexibility of joint training, optimizing for topic coherence, attaining complex models, scalability.	interpretability of model parameters. Also, it generally cannot handle sparsity.

4. Topic Models

The different model categories shall be discussed in detail in the following subsections.

4.1. Algebraic Models

The first model to develop was Latent Semantic Allocation (LSI) [3]. LSI is a BOW model that represents the corpus as a document term matrix (DTM). Then it decomposes the DTM using singular value decomposition into its constituents. It considers the largest singular values in each document – corresponding to the most evident topics. LSI uncovers patterns between terms of the corpus, such as synonymity. It can be used to compare documents and perform queries for information retrieval tasks, whereas the interpretability of LSI is somehow limited. This limitation is common with several topic models that generate limited human-interpretable documents. Being a BOW model, it disregards order and retains multiplicity. LSI implicitly assumes a probabilistic generative model where words and documents are jointly Gaussian distributed. This assumption does not match reality. Probabilistic LSI addresses this with a multinomial generative model [20].

Non-negative matrix factorization (NMF) [27] is another algebraic model used extensively in NLP. It assumes that a given higher dimension vector can be factorized into a lower dimension representation, given that all the resultant representations are non-negative. In this respect, a model called SeaNMF was proposed in [28] to discover latent topics in short texts using a semantics-assisted NMF. It exploits the word-context semantic associations among the words by applying neural word embeddings. This approach has proved to boost the model performance. It leverages the word-context semantic relationships learned from skip-gram. The extensive evaluations performed over real-world short datasets demonstrate the superior performance of the presented models in terms of topic coherence and accuracy. Topic coherence is a metric that evaluates a topic modeling technique according to its human interpretability, which we will discuss in detail in section 5.

4.2. Probabilistic Models

Topic models in this category consider the probabilistic nature of the modeled entities. They map the relationship between different model random variables through a graph, usually a directed acyclic graph (DAG), and thus called Bayesian models. Note that some models such as [29] proposed an undirected graph. In the DAG, nodes represent random variables. These random variables are either observed, such as the constructs or words. They can be latent or hidden, such as the topic distribution. The existence of an edge indicates probabilistic dependence between any two random variables.

4.2.1. Latent Dirichlet Allocation

This is the simplest topic model in this category which was later extended in several ways. It is a bag of words model that represents a document as a vector of length L. In LDA, document generation process is defined in probabilistic terms [1] and [30]. It assumes that each document i , is generated, one word at a time, through sampling a topic from the distribution of topics for this document, θ_i and then sampling a word w from this topic (a topic is a distribution over words).

The topic assignment document i is denoted z_i . The distribution of topics itself θ_i is drawn from a Dirichlet distribution where each component -topic- in the sampled mixture is independent of the other. Incorporating Dirichlet distribution models the topic sparsity per document; confirmatory to the observations in real documents. This generative process defines a joint probability distribution over the observed documents and the hidden topic structure.

$$p(\theta, z, w|\alpha, \beta) \tag{3}$$

Where both α and β are Dirichlet priors and are model hyperparameters. Then to infer the hidden topic structure, we use the joint distribution to compute the conditional distribution of the topic

structure given the observed documents. The inference is calculated in equation 4.

$$p(\theta, z|w, \alpha, \beta) = \frac{p(\theta, z, w|\alpha, \beta)}{p(w|\alpha, \beta)} \tag{4}$$

The denominator in the last equation is the marginal probability of the observations, which can be computed by summing the joint distribution over every possible hidden topic structure. The number of possible topic structures is so large that this summation is intractable to compute. Thus, approximations via variational inference or Markov Chain Monte Carlo (MCMC) are used for inference.

However, the results are not very stable. The model suffers from the order effect, where the order of the fed documents during training can change the generated topics. It also requires the explicit setting of the number of topics, which may be challenging to determine. Moreover, LDA is not very robust against changes to Dirichlet priors and other model hyperparameters. The search space of all possible hyperparameters is big. An optimization technique such as genetic algorithms or differential evolution [31] could be used to reduce the search space. LDA was extended into various models to overcome its shortcomings and to better adapt to different corpora properties.

4.2.2. Extending LDA

One way to extend LDA is to relax the bag of words assumption, where the model does not consider the order of words in a document. This assumption is a reasonable assumption to make to uncover the topics in a corpus. For better language modeling, this assumption is relaxed [32]. In sentence LDA, the words in one sentence are assumed to have been generated by the same topic [33]. Sentence LDA models the co-occurrence of words in finer granularity than LDA. The order of documents is also not considered by LDA. In the case of corpora that span several years, topics are likely to change over time, and the order of documents is thus relevant. An example is in [5], where a topic is not a single distribution but a sequence of distributions over words.

Another LDA variant observes that the topics per document are usually correlated. The occurrence of a topic in one document correlates with the occurrence of other 'related topics. LDA does not model the correlation between topics and considers them independent. So, when inferring the latent topics of a document, one may end up with disparate topics. The correlated topic model (CTM) [4] is one example of this. CTM utilizes the logistic normal distribution to model topics' correlation instead of the Dirichlet distribution in LDA. Furthermore, LDA uses standard variational inference. However, CTM uses the mean-field variational inference to approximate the posterior inference. Another example is the Pachinko allocation model [34], which models word correlations in addition to the correlation between topics.

Many documents contain metadata about, for example, the author, the title, links, publication date, and other data. Some LDA extensions incorporate these metadata, such as [35] which considers the author, and [36] which considers the links in documents. Other LDA variants can infer the number of topics hyperparameter from the data, such as the Bayesian non-parametric Hierarchical Dirichlet Processes (HDP) model proposed in [37].

Short text documents such as on social websites and microblogs pose a challenge due to the sparsity of words. Another generative probabilistic model, the biterm topic model (BTM) [38], considers each co-occurring word pair as a single term. The model is built upon a biterm set instead of documents.

4.3. Fuzzy Models

There are two approaches for any clustering task: hard and fuzzy. In the hard clustering approach, an item is forced to belong to one cluster. However, it can belong to multiple clusters in the case of fuzzy clustering approach with different membership degrees [39].

Fuzzy set theory is selected among other topic models because other models suffer from computational complexity. Other models also require probabilistic inference like Gibbs sampling [40]. The authors of [41] have mentioned that applying fuzzy representation to LDA enhances the topic's coherence compared to regular LDA and term-weighted LDA. In this work, the authors implemented LDA using fuzzy document representation.

This fuzzy representation maps each document to a vector of a keyword. Each keyword belongs to each document with some fuzzy membership value, calculated using the semantic similarity between each keyword and all the other words. This approach enhances the quality of generated topics by assigning weights to each keyword according to its importance. It also overcomes the sparsity problem. Moreover, it counts the semantic association among words.

Fuzzy topic modeling is applied to different fields, like text mining and medical and educational fields. The authors of [42] have proposed a novel fuzzy topic modeling approach that tries to solve the problems of noise and sparsity in short texts. The three essential components of this approach are BOW, Principal Component Analysis (PCA), and fuzzy c-means algorithm.

This approach leverages BOW in generating both local and global frequencies. The PCA dimension reduction algorithm is used with a global term to remove the negative impact of high dimensionality. Lastly, fuzzy-c-means assigns each point to a different cluster with varying membership functions degrees for each cluster. The fuzzy topic modeling technique has enhanced the classification and clustering tasks' accuracy compared to the baseline methodologies.

Fuzzy topic modeling can also be applied in personalized web searching to get user-oriented results. The authors of [43] have presented a personalization fuzzy topic modeling technique. Potential personalization is estimated to enhance the effectiveness of the search system. The fuzzy logic is used to handle the uncertainty of each document's topics. On the other hand, the fuzzy C-Means algorithm is used to retrieve the relevant topics. The ranked results from the fuzzy topic modeling technique were compared to those from LDA techniques. Finally, the proposed experiments have shown that the fuzzy topic modeling technique has overcome the LDA.

Coming to the medical field, the authors of [44] aimed at modeling medical documents with lower computational cost and higher performance accuracy. This technique uses the fuzzy set theory and word weighting techniques (the importance of a word in a document in a corpus) for feature transformation (representing words into different formats like numbers for better modeling) in the preprocessing stage. This technique models medical topics in 3 basic steps: local term weighting, then global term weighting (with five different methodologies), and finally, fuzzy clustering.

For dimension reduction, the authors did not use PCA like in the previous work. However, they have developed a different technique. Each clustering step input is the step that precedes it. So, if we have an nxm matrix with 10 clusters in the first step, then the next step will be $nx10$ matrix with 9 clusters. The number of column dimensions will decrease until the minimum number of clusters are reached.

In the educational field, and due to the increasing number of available online lecture videos, many video lectures have multiple labels as they contain overlapped subjects. This fact hinders the efficiency of video search and the quality of retrieved results. In [45], a lecture-content-based clustering is applied to overcome the previous challenges. This technique leverages the transcripts of the available videos to get the content of the videos. For each cluster representation, they have relied on Wikipedia documents. Then, a similarity value is calculated between the extracted video topics and the representative docu-

ments for each cluster. Based on this similarity value, a fuzzy clustering method is applied to cluster the content of the videos.

4.4. Neural Models

The first subcategory of these models uses neural components such as the models that use embeddings [8], [46], [47], [48]. LDA2VEC [47] is a model that uses Word2Vec along with LDA to discover the topics behind a set of documents. The word2vec model [49] initially assigns vectors to words randomly and then changes these vectors to optimize (maximize) an objective function.

In word2vec [49], we start by specifying a window size and then slide this window over the words in the document. A pivot is first chosen within the window, and all other words inside the window are considered the context. The objective is to develop a vector representation that maximizes the probability of the context words given the pivot. This is called the skip-gram word2vec.

In LDA2VEC [47], we add a document vector to the pivot word, both used to obtain the context vector. This context vector is then used to predict context words. The obtained vector is dense, making it hard to interpret its meaning and ultimately make decisions off it. LDA2VEC thus decomposes the document vector into two components: topic mixture weights and a topic matrix. The mixture weights are sparse vectors that can be interpreted. This is achieved by enforcing Dirichlet priors on the vector weights to regularize the model. The document context makes better predictions than a model that only considers local context. It also can learn better word vector representations of a special lexicon.

Another model that uses embeddings is ETM [8], which is considered a variant of LDA that incorporates neural components. In ETM, the generated topic mixture is pulled from a logistic normal distribution, and words per topic are drawn by projecting the topic vector onto the vocabulary vector. It performs state-of-the-art results and provides more interpretable topics. ETM does not require the removal of stop words and rare words, unlike LSI, LDA, and CTM. It performs well as the corpus vocabulary increases, especially with rare and stop words left in the vocabulary.

Another utilization of word embeddings was illustrated in [50]. This work focuses on the effect of auxiliary word embedding to enrich topics modeling in the context of short text. As previously mentioned, the short texts suffer from the sparsity problem. This problem was tackled by exploiting the semantic relationships between the words during topic inference.

In [51], a new framework that leverages pre-trained word embeddings were introduced. This framework first obtains text embeddings and then clusters these embeddings into semantically similar texts. Words are then extracted from each cluster to represent the topic. Each of these steps is independent, which allows for flexible modeling. In [7], the text embeddings can be obtained using any pre-trained language model such as Bidirectional Encoder Representations from Transformers or BERT [12]. The resulting embeddings are clustered using Hierarchical Density-Based Spatial Clustering of Applications with Noise or HDBSCAN [52] to obtain a semantically coherent set of documents in each cluster that represent a topic. The author then proposes using a class-based Term Frequency-Inverse Document Frequency or cTF-IDF to obtain the words representing each topic.

Incorporating neural components is usually advantageous. For example, the previously discussed probabilistic correlated topic models suffer from high computational costs and inefficient scaling. In light of these challenges, an efficient correlated topic model with topic embeddings was developed [53] where topics are represented as vectors, and the model is trained to learn the topics embeddings and correlations in terms of the vector's closeness.

4.4.1. Variational Autoencoders-Neural Topic Models

The most popular framework for NTMs, however, is the models that extend the generative process of probabilistic models through Variational Auto Encoders (VAEs) [54] and [55]. In the remaining of this section, we use NTMs to refer to this subcategory. NTMs offer several advantages over Bayesian probabilistic models. Documents can be represented as either probability vectors that offer an easier interpretation or as embedding vectors that are easier for a network to handle.

The main advantage of NTM is that the inference problem is formulated as an optimization problem. This formulation makes it easier to optimize for specific aspects such as coherence [56]. It also allows for joint training with other DNNs. Moreover, this allows NTMs to scale to bigger corpora and more complex models. In Bayesian probabilistic models, inference algorithms must be customized. They tend to get more complicated as models get complex. In NTM, however, the inference problem is easier to compute and parallelize.

Encoder and decoder networks, respectively model

The inference and generative processes in NTMs. The encoder learns θ where $z = \theta(b)$; the projection parameters from the observed document b to its latent representation z. The decoder learns ϕ where $b = \phi(z, T)$; the projection parameters from the latent document representation to the observed document representation, T is the collection of word distributions of all the topics.

The learning process optimizes the ELBO, which is a lower bound on the log-marginal likelihood of the observed document representation. This optimizatin is equivalent to finding an approximate solution to the posterior probability distribution.

Different NTMs have a different configuration of the prior distribution of z, the posterior distribution, and the observed data distribution. As with the case of probabilistic models, some NTMs consider the sequence of words and the correlation between topics. Some incorporate documents' metadata, while others address short text documents' sparsity. For example, [9], uses Dirichlet as a prior distribution for z. To compute the gradient of the expectation in ELBO, the authors apply the Laplace approximation, where Dirichlet samples are approximated by samples from a logistic normal distribution whose mean and co-variance are specifically configured. They used their inference method in ProdLDA; a topic model in which the distribution over individual words is a product of experts rather than the mixture model used in LDA.

5. Evaluation metrics

Topic models can be applied in various application domains. So, they can be evaluated extrinsically according to how they perform in the domain where they were applied [57] and [58]. They can also be evaluated intrinsically by considering the generated topics themselves. Intrinsic evaluation is independent of any specific domain and is thus more general. The different models differ in simplicity, computation efficiency, and modeling assumptions. They accordingly differ in how they perform on different corpora and different applications.

There is little consensus on the aspects of topic model evaluation. There have also been different methods to evaluate a specific aspect. This paper reviews a comprehensive distilled set of evaluation criteria and their methods. The evaluation criteria discussed are quality, interpretability, stability, topic diversity, efficiency, and flexibility (figure 2). Note that preprocessing stages before feeding the corpus to the topic model impact the outcome. For example, it is common to remove words that have little topical content, such as "the", "and", and "if".

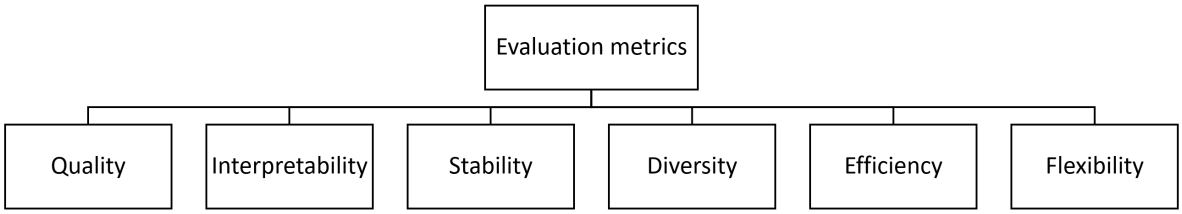


Figure 2: Topic models evaluation criteria.

5.1. Quality and Perplexity

Perplexity measures the model's capability to generate the documents in corpus based on the learned topics. It measures the model's predictive power but not the latent structure, so perplexity only shows how well the model explains the data. A model is perplexed if the information gain of learning the random variable's outcome is small, so the lower the perplexity, the better the model in explaining the observed documents. The posterior probability calculation can be optimized for perplexity.

Note that perplexity must be normalized to the vocabulary size in the corpus since it changes with different corpus and topic sizes. This dependence on vocabulary size makes it harder to compare different models. A method to estimate the probability of held-out documents indicative of the model quality is proposed in [59]. Their method can be used to compare different models quantitatively. Note that models with low perplexity do not necessarily generate human-interpretable topics.

5.2. Interpretability and Topic coherence

A topic is a discrete distribution over words. This set of words is evaluated for being human-interpretable. For this to occur, the generated words should be associated with a single semantic meaning. One way of assessing the interpretability is to evaluate the coherence of the words in the generated topic as in [60], and [61].

In [61], coherence is assessed by observing the lexical similarity between pairs of words. Their approach considered the top 10 words for each topic. Then they calculated an asymmetric- similarity measure between all combinations of word pairs. The authors experimented with different similarity measures and found mutual information to be the most consistent performer. Finally, they aggregate all obtained pairwise scores (45 scores) using an aggregation function such as the mean or median.

The pointwise mutual information, PMI, between word pair (wi, wj) is calculated as below

$$PMI\ (w_i,\ w_j)\ =\ \frac{log\ p\ (w_i,\ w_j)}{p(w_i)p(\ w_j)} \tag{5}$$

For the above equation. PMI, too, must be normalized. Note that the PMI can only be evaluated after a solution is obtained. Alternatively, the interpretability can be optimized from the beginning by considering a different measure such as Word Embedding Topic Comprehension, $WETC$ [56]. It is also important to highlight the trade-off between perplexity and coherence. Optimizing perplexity usually results in decreased coherence [62]. Note that hyperparameters, such as the number of topics, will also affect this metric.

5.3. Topic Diversity (Uniqueness)

This metric describes how semantically diverse the obtained topics are. One way to define diversity is introduced in [8]. It defines topic diversity as the percentage of unique words in the top 25 words of all topics. A topic model should generate diverse topics and score high on this metric. A low score indicates redundant topics, meaning the model could not sufficiently disentangle the corpus's themes. Note that selecting a number of topics in the model affects the topic diversity. A too large number could result in similar topics with overlapping top words. A too small number could result in broad topics with poor interpretability.

5.4. Stability: Topic similarity across runs

One of the salient challenges in topic modeling is topic instability, which means that top topics ranking can change significantly over different runs. Moreover, documents' assignments to different topics can vary throughout the experiments. For example, LDA suffers from instability due to order effects, i.e., different input orderings can lead to different generated topics. This results in user confusion or, in the case of using TM for classification, results in the deterioration of the classifier [31].

To quantify stability, several metrics were proposed. The essence is to measure the similarity between topics across several runs. Notice that topics are represented by their top n words. More model stability is obtained if similar topics are generated across different runs. The previous approach requires no expertise in the domain to which topic models are applied. Human judgment and PMI similarity measures require either direct or indirect expert knowledge, respectively. One way to measure stability is to use the median Jaccard similarity of generated topics -top n words- between runs across all topics [31], and [63]. Jaccard similarity considers the intersection between the top n words in any two generated topics. It does not consider their order. Alternatively, [64] proposed Rank Biased Overlap (RBO) to measure topic stability as in the equation below.

$$RBO(T_1,\ T_2,\ p,\ d)\ =\ \frac{x_d}{d}\ .\ p^d\ +\ \frac{1-\ p}{p}\ \sum_{i=1}^{p}\frac{x_i}{i}\ \ .\ p^i \tag{6}$$

Where $T1$ and $T2$ represent an ordered list of top n words in any two topics (across different runs), d is the evaluation depth typically set to 10 as we compare the top 10 words. x_d is the intersection between $T1$ and $T2$ at depth d. p controls how the metric is affected by the order. Setting p to 1 removes all dependence on order so that the metric considers only the intersection. Setting p to smaller values gives more weight to the order of words. Typically, a value of 0.9 is used as suggested by the authors [64].

A solution to address instability was elaborated in [65] where different ensemble learning strategies were tested in the context of matrix factorization. Among all strategies, the K-Fold ensemble method has achieved the best results. Another solution was proposed in [66]. The authors realize that overcoming instability can be done through careful initialization of the model hyperparameters. They used differential evolution (DE) to optimize the selection of hyperparameters. They reported topic stability -measured with Jaccard similarity- of nearly 88% or more.

5.5. Efficiency; Computation and memory complexity

In latent variable models, the computation of the posterior distribution is always intractable. So, we use approximate solutions such as Markov Chain Monte Carlo (MCMC) or variational inference. We may also try to estimate it by sampling methods (Gibbs sampling) which tend to be faster to develop and run. Variational inference is easier to parallelize. Different algorithms differ in their computation time complexity. Thus, computation complexity is a metric used to assess and compare different models.

Since topic models usually deals with large corpora that may not fit in memory. Radim Řehůřek, [66] proposed a software framework based on the idea of document streaming. Their framework implemented an LDA inference algorithm independent of the corpus size.

5.6. Flexibility

This evaluation aspect is not measurable quantitatively. It is worth noting that different models have different capacities to deal with text corpora exhibiting different characteristics. The required preprocessing steps and the distribution of words in documents, for example, impact the performance. Some models can train jointly with others or can be optimized for certain aspects, such as coherence.

6. Topic models tools and applications

Topic models have applications in several disciplines, from NLP to bioinformatics. In the following subsections, a review is introduced.

6.1. Topic Modeling applications

Topic models represent documents in terms of their topics. This representation can be used as features for other machine learning models, for example, document classification models. These topical features usually augment the engineered features as an input to a model. The topical distribution can also be used to assess the entropy of a document to evaluate its information richness which is used in information retrieval tasks as a feature. Text matching can compare the topical distributions of two documents using any distance metric such as Jensen-Shannon Divergence (JSD).

NLP leverage TM. In [57], the authors applied TM for word sense discrimination. Also, [58] used TM for document summarization, and [67] used it for text segmentation. TM has also been used in other tasks such as discourse segmentation and machine translation.

The representation can also aid humans in making sense of large sets of documents and thus can serve in many disciplines, varying from genetics to education, economics, sociology, and more. Owing to their interpretability, topic models can help scholars from different disciplines explore their data. For example, by leveraging the power of social media, Topic models have been used to understand what people are saying about a company [68] [69]. In the transportation field, researchers used TM to explore the trends in the field [69], [70]. Using LDA, the authors of [70] aggregated topics over time. Hence, they could recognize general trends, like sustainability and travel behavior and some other temporal trends. Accordingly, this paper provides the research community with trends, research areas, and gaps in the transportation field.

TM has also been used in economics. For example, in [71], an investigation of economics's historical evolution over a specific time window and its changing structure was discussed. The idea is to track the most relevant topics discussed by economists over each decade of the twentieth century. Moreover, a web-based interactive tool called "LDAvis" is developed to visualize the topics. In software engineering (SE), topic models have been used in SE tasks such as source code analysis [72], testing [73], requirements

engineering [74], software architecture [75], and others. Coming to the field of bioinformatics, the authors of [76] reviewed the applications of TM in the field. In [77], the bioinformatics field is analyzed whereas [78] analyzed gene sequence data using TM, and [79] used TM to classify gene expression data. In medicine, with the large scale of documents, it is becoming progressively hard to find relevant documents. In [44], a novel fuzzy TM approach was used toward achieving this. Numerous other research papers leveraged TM such as [80], [81] in communication research, and [82] who discovered business intelligence specifically for airlines by applying topic models to documents of customers' reviews.

Due to the long-tailed distribution of languages, topic models in big data sets need to learn a large number of coherent word sets (topics). This large number of topics is too big to comprehend by humans. In [83], industrial applications such as search engines and online advertisement systems use LDA-based TM with at least 105 topics to serve millions of users.

6.2. Software tools

Many available software tools implement various topic models. Most open software tools implement LDA and LSI, whereas many other models lack open-source implementation except for a few exceptions [84] and [85].

Some of the early popular software tools was the Stanford Topic Modeling Toolbox (TMT), which addresses social scientists. It was written using an old version of Scala and is no longer supported. Other toolboxes and software frameworks are presented in table 2 and described briefly. Gensim provides parallelized implementations of several models and can accommodate corpora larger than the memory. It also provides storage for many text datasets accessible through a standard API.

Data cleaning is crucial before feeding the data to a model. For each model, a play with hyperparameters and the preprocessing steps can dramatically affect the quality of the obtained topics. Basic cleaning steps are tokenization, removal of stop words, and stemming. Tokenization decomposes the document into its constructs, usually words. Stop words are these words that do not count as 'constructs' since they have little topical content, such as "the", "and", "if". They are meaningless to the model and are thus removed. The list of stopwords is subjective and usually depends on the corpus and the application. In [86], the authors argue that removing stop words prior to model inference provides a superficial improvement. Alternatively, removing stopwords past model inference is more transparent and produces similar results. Stemming words reduces them to their stems. Stemming is important since words with the same stem have the same topical content and, thus, must be seen as the same by the model. Mallet and Gensim can generate different topics (even with the same set of hyperparameters selected and the same order of feeding documents). This difference is due to the differences in the internals of preprocessing steps in different implementations.

Table 2: Available open-source libraries.

Tool	Interface	Models	Workflow	Pros	Cons
Mallet [1]	Console app.	LDA, PAM, HLDA.	Represent documents as a string, then clean (use std or extra stop word lists or use RegExp) and import. Finally, train (choose hyperparameters (# topics, alpha, and beta, # iterations)	User friendliness.	Limited fine grain control.
Gensim [2]	Python API.	LSI, LDA.	Represent documents as a string, clean, then represent documents by a feature vector, then apply the model.	Highly optimized, memory independent, and allows for fine control to tweak the models.	Harder.
Familia [3]	Python API.	LDA, Sentence LDA, Topical word embedding.	Represent documents as a string, clean, and then call appropriate off-the-shelf trained model using the proper API function.	It offers ready topic models trained in large industrial corpora. It offers two models' implementation beyond LDA	Memory limitations
OCTIS [4]	Python Package.	LSI, NMF, LDA, HDP, ETM, CTM, NeuralLDA, ProdLDA	Use an available dataset, or customize yours. Set the model hyperparameters and train.	One interface to run many topic models. It estimates optimal hyperparameters.	Not all models are available. Sacrifices details for usability.

Familia is more industry oriented. It provides off-the-shelf topic models trained on large-scale industrial corpora. It can be used in the industry for semantic representation or semantic matching tasks, such as document classification, document clustering, and personalized recommendations. Note that Familia, for example, uses several topic models from different categories. For example, it implements LDA from the probabilistic category of models and topical word embeddings from the neural category of models.

[1] http://mallet.cs.umass.edu/index.php
[2] https://radimrehurek.com/gensim
[3] https://github.com/baidu/Familia
[4] https://github.com/MIND-Lab/OCTIS

OCTIS or Optimizing and Comparing Topic Models is Simple! [85] provides a unified interface to train various topic models and optimize the selection of their hyperparameters. Various models from algebraic, probabilistic, and neural categories are available. OCTIS also contains a repository of public preprocessed datasets readily available to load and experiment with.

6.3. Datasets

Tables 3 and 4 present some publicly available datasets for topic modeling. Note that the 20 Newsgroup dataset has been particularly used as a benchmark dataset [87], [88]. Moreover, it is observed that the other mentioned datasets are often used in a limited number of research publications.

Table 3: Available long text datasets.

Dataset	Description	Size	NO. of topics	Language	Date
Scientific journal articles	Articles from 22 leading transportation journals	17,163 articles	50	English	From 1990 to 2015
Economics articles [5]	188 economics articles stored in the JSTOR database	250,846 articles	25,50 and 100	English	From 1845 to 2013
20 Newsgroups [6]	A collection of newsgroup documents, partitioned across 20 different newsgroups	20000 posts	20	English	2008
M10 [7] [89]	Publications from 10 distinct research areas queried from CiteSeerX using the keywords from Microsoft Academic Search.	10,310 publications.	10	English	-
Applications Information	Google Play Arabic Applications names and descriptions	7657 app descriptions	10	Arabic	-
NIPS [8]	A collection of all papers from NIPS Machine learning conference	1566 documents	20 and 50	English	From 1988 to 2003

[5] https://www.jstor.org/
[6] http://qwone.com/ jason/20Newsgroups/
[7] http://citeseerx.ist.psu.edu/
[8] https://cs.nyu.edu/ roweis/data.html

Table 4: Available short text datasets.

Dataset	Description	Size	No. Of inferred topics	Language	Date
The 20- topics dataset	Tweets from 1,200 prominent Twitter accounts.	4,170,382 tweets	20	English	From March 2015 to February 2016
Real-world snippets	Snippets of an English search engine.	12, 340 Web search snippets	8	English	2008
BaiduQA [9]	Questions from Q&A service in Chinese)	648, 514 questions	35	Chinese	-
Weibo [10]	Posts from Weibo commonly referred to as "Chinese Twitter", is a micro-blogging site.	719,570 posts	-	Chinese	From January 1 to June 30, 2020

7. Comparison of models

It is challenging to have a fair comparison between different topic models. One challenge is that different papers proposing different models evaluate them using different datasets. Moreover, many papers report only a few of the evaluation metrics we mentioned in section 5. The choice of hyperparameters and the existence of several model variants of the same topic model add to the list of challenges. In this section, we conduct experiments to compare several topic models from various categories across the proposed evaluation metrics. Seven topic models are considered; LSI and NMF from the algebraic category, LDA and HDP from the probabilistic category, and ETM, CTM, and ProdLDA from the neural category. The results are evaluated across a common evaluation metrics using the exact implementation. The open source OCTIS python package [85] is used to conduct these experiments.

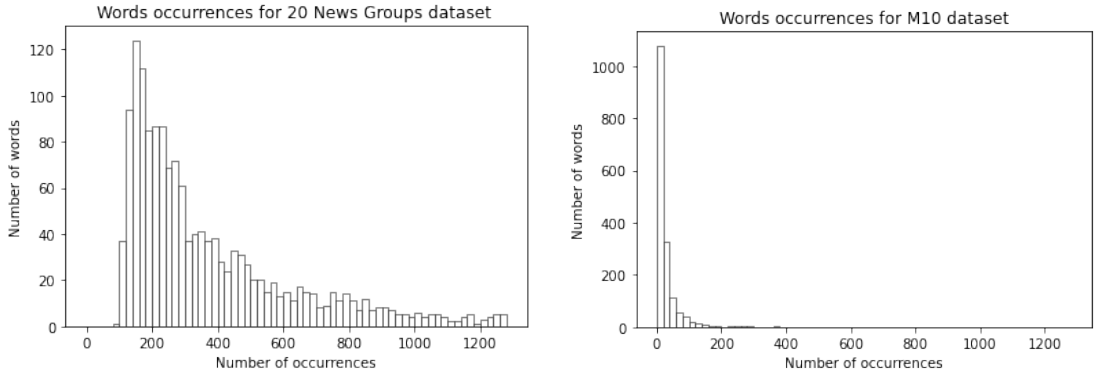
7.1. Experimental setup

Each model is run ten times using the same selection of default hyperparameters. The models are run against two datasets; the OCTIS preprocessed 20 newsgroups dataset containing 16309 documents and the OCTIS preprocessed M10 dataset containing 8355 documents. Both datasets have comparable vocabulary sizes (1612 for 20 newsgroups and 1696 for M10). However, the average word count per document is 48.02 for the 20 newsgroups dataset and 5.91 for the M10 dataset. The M10 dataset thus has shorter documents and a larger number of rare words. Figure 3 shows the number of words in vocabulary for each number of occurrences.

The generated topics by the models are assessed for their stability across every two successive runs. The Rank Based Overlap (RBO) metric discussed in 5.4 is used to measure stability. Normalized PMI is used to measure coherence, and topic diversity is measured considering the unique tokens in the top 10 words. The average score across the runs is reported for coherence, diversity, and computation time.

[9] https://zhidao.baidu.com/

[10] https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DULFFJ



(a) 20 news groups dataset

(b) M10 dataset

Figure 3: Datasets exhibit different word distributions. Note the difference in the y-axis scale between the charts.

The perplexity metric is not evaluated in this experiment since non-generative models, such as LSI and NMF will not have a defined perplexity score. Note that perplexity metric is not correlated with human judgment [62].

Note that all models accept the number of topics as a hyperparameter except for the Hierarchical Topic Model (HDP), which automatically infers the number of topics in the corpus. Note also that the "bert-base-nli-mean-tokens" bert model is used in the zero-shot inference type model variant for CTM.

7.2. Results

The obtained scores are shown in table 5 and table 6.

Table 5: Results of running the topic models against the 20 newsgroups dataset.

Model	Coherence	Diversity	Stability	Time (Seconds)
LSI	0.0128	0.4810	1	3.2521
NMF	0.0821	0.6945	0.8163	4.8802
LDA	0.0575	0.7105	0.7257	8.0566
HDP	-0.1604	0.6264	0.8942	24.4717
ETM	0.0395	0.3775	0.8942	45.9766
CTM	0.0747	0.8900	0.6967	23.981
ProdLDA	0.0552	0.8785	0.6266	27.3077

Table 6: Results of running the topic models against the M10 dataset.

Model	Coherence	Diversity	Stability	Time (Seconds)
LSI	-0.0235	0.546	1	1.6136
NMF	-0.0311	0.713	0.6014	2.3121
LDA	-0.0604	0.627	0.692	2.2156
HDP	-0.5049	0.668	0.3194	7.3764
ETM	-0.0014	0.383	0.8641	32.9827
CTM	0.0197	0.98	0.7739	12.0966
ProdLDA	-0.0017	0.971	0.7881	12.7665

In table 7, we list the best and worst performing models across the evaluated metrics.

Table 7: Best performing models across the evaluated metrics.

Evaluation metric	Best models	Worst models
Coherence	CTM	HDP
Diversity	CTM and ProdLDA	ETM
Stability	LSI and ETM	CTM and ProdLDA. HDP, NMF, and LDA in short documents datasets.
Efficiency	LSI, NMF	ETM

The box plots of the obtained results across the 10 runs are shown in figure 4 and figure 5.

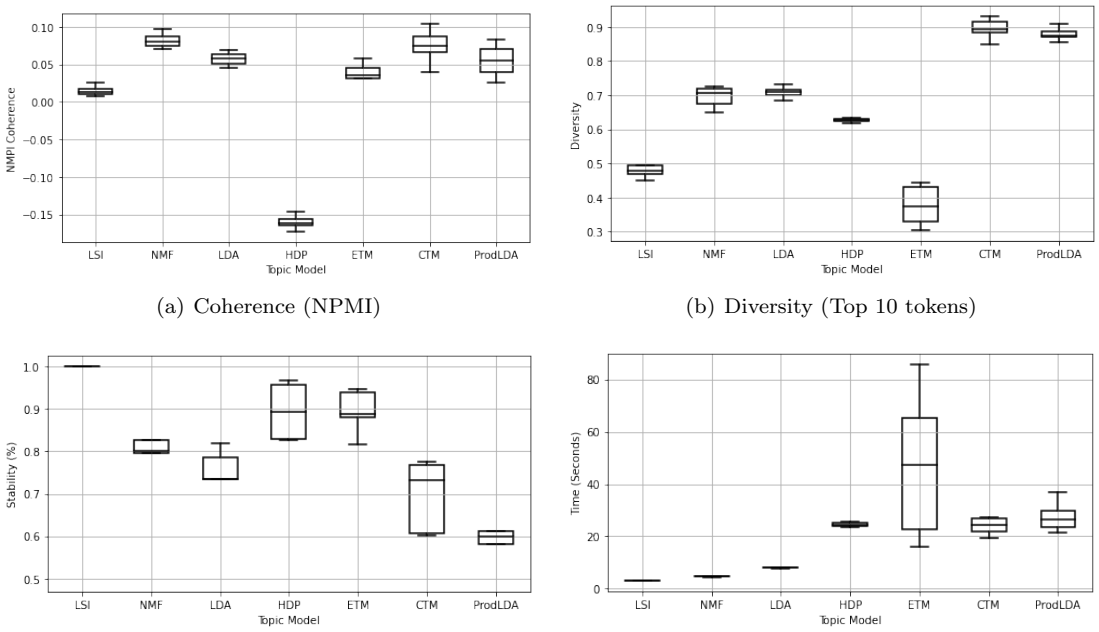
8. Discussion and Analysis

8.1. The models

Examining the results, it is evident that the models' scores vary considerably and that no one model or model family performs best in all aspects. For example, the result of LSI against the 20 newsgroups dataset achieves the second worst scores in diversity and coherence but the best scores in stability and computation time. Moreover, the models' performance varies with the input dataset in some metrics. For example, the ETM coherence score ranks fifth on the 20 newsgroups dataset. However, ETM achieves the second-best coherence score on the M10 dataset. Also notable is the variability between runs, owing to the stochastic nature of many models.

It is observed that a relationship exists between stability and diversity, which is more observable with the 20 newsgroups dataset. As the model stability goes up, its generated topics are less diverse. We hypothesize that it is challenging for models to generate diverse topics without compromising their stability and that diversification is dependent, at least in part, on the stochastic nature of the modeling algorithms. Figure 6 shows a scatter plot of models' diversity versus stability scores with a fitted linear trend line.

In the following sections, we consider evaluation metrics and explore the spectrum of scores achieved by the models.



(a) Coherence (NPMI)

(b) Diversity (Top 10 tokens)

(c) Stability (RBO)

(d) Computation time (Seconds)

Figure 4: Topic models' evaluation metrics scores across ten runs against the 20 newsgroups dataset.

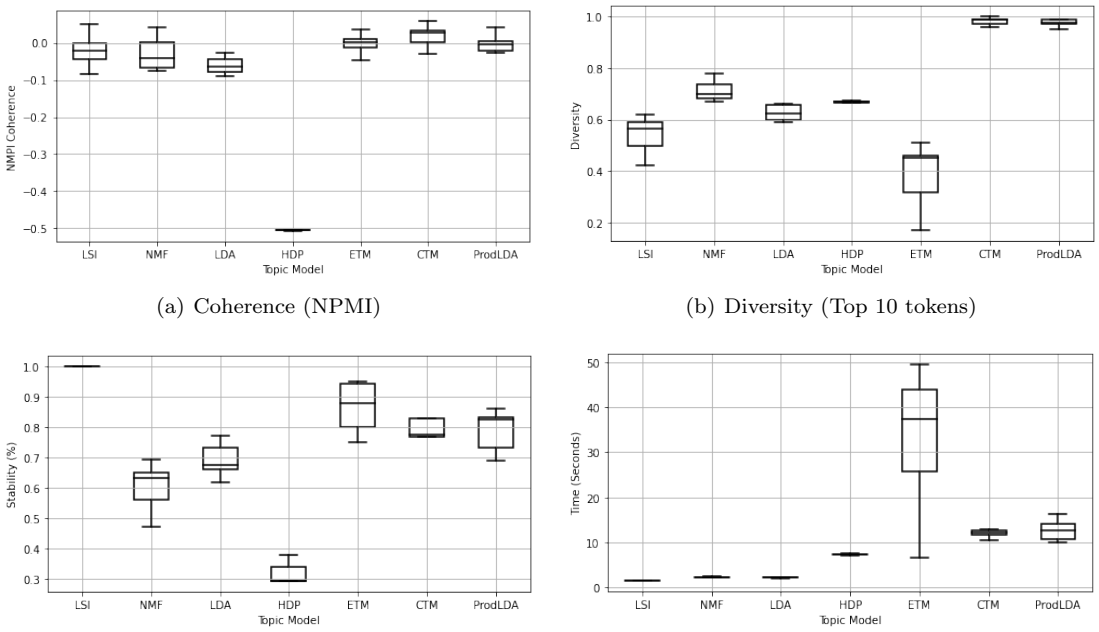
8.1.1. Coherence

The 20 newsgroups dataset results show that NMF achieves the best score, followed by CTM. Next are LDA and ProdLDA, and both models score close to each other. ETM achieves modest results and then comes LSI, which only outperforms HDP. HDP scores the worst coherence. The results of all models on the M10 dataset are generally less than the 20 newsgroups dataset in coherence scores. Moreover, on the M10 dataset, the models are differently ranked, which is attributed to the different dataset characteristics. The difference in performance is noted in the scores of ETM, NMF, and LDA models. Excluding these models from the list of considered models will result in the same ranking of models on both datasets.

The abundance of rare words in the M10 dataset could result in the improvement of the performance of ETM compared to other models. The existence of rare words and shorter documents could also degrade the performance of NMF and LDA due to data sparsity per document. This degradation is because both models uncover topics by observing the word co-occurrence patterns at the document level. NMF maintains its rank above LDA, which is consistent with [65] and [90], who found that NMF is more likely to generate better coherent topics than LDA. Finally, we note that neural models' coherence scores are not always better than algebraic or probabilistic classical models.

8.1.2. Diversity

For diversity, the results of running the models on both datasets are similar, apart from a slight degradation of LDA on the M10 dataset. The best model (CTM) scores close to the second best (ProdLDA). NMF, LDA, and HDP come next. LSI is again the second worst model, and it only outperforms ETM. In [7], the reported diversity scores, obtained from running various models against



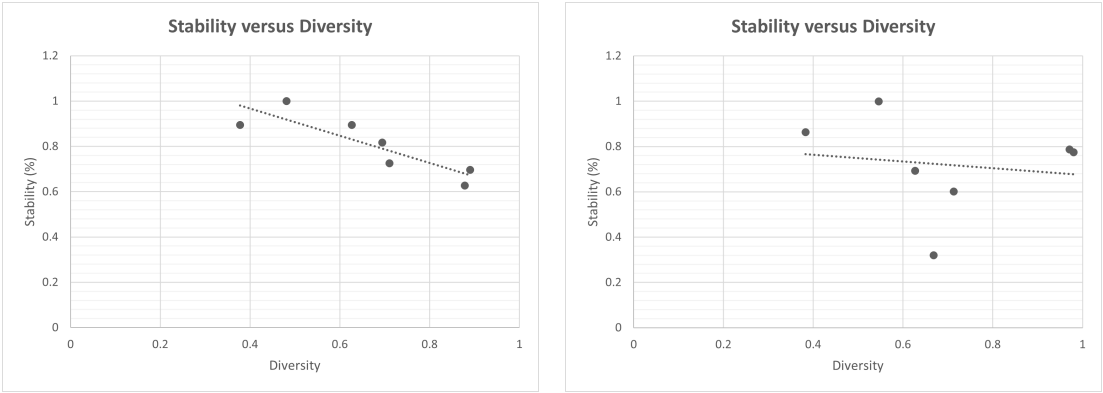
(a) Coherence (NPMI)

(b) Diversity (Top 10 tokens)

(c) Stability (RBO)

(d) Computation time (Seconds)

Figure 5: Topic models' evaluation metrics scores across 10 runs against the M10 dataset.



(a) 20 newsgroups dataset

(b) M10 dataset

Figure 6: Diversity scores versus Stability scores of the examined topic models.

the 20 newsgroups dataset, show that CTM achieves the best diversity, outperforming LDA and NMF, which is consistent with our obtained results.

8.1.3. Stability

On both datasets, LSI achieves maximum stability and is followed by ETM. LDA results are consistent in both datasets. HDP achieves a high stability score on the 20 newsgroups dataset and a low stability score on the M10 dataset. NMF stability score also degraded from 82% to 60%. Finally, CTM and ProdLDA achieve the worst and second worst stability scores on the 20 newsgroups dataset but achieve a high stability score on the M10 dataset.

In general, on the 20 newsgroups dataset, neural models' stability scores are poor compared to classical models. This conclusion agrees with the findings of the authors of [91], who conducted an empirical study to compare neural models to classical models such as LDA and NMF. The authors considered the neural models that are based on encoder-decoder architecture and whose decoders can work with a bag of words representation of documents such as ProdLDA [9]. The authors concluded that, compared to classical models, neural models are generally less stable, and their performance will vary significantly from run to run.

However, neural models' stability scores are improved on the M10 dataset. This observation could suggest that neural topic models outperform classical topic models in stability scores for datasets with short documents and a high percentage of rare words.

Since topic models exhibit instability in their results if they have stochastic elements in any of their steps, deterministic algorithms such as LSI are very stable and score maximum stability. LSI gives the same output every time it is run. In practice, there could be a slight variability with LSI-generated topics due to implementation differences in Singular Value Decomposition (SVD) computation. The exact implementation is used in this experiment, so no variability is observed.

NMF exhibits instability due to the random initialization in the implemented algorithm. Note, however, that apart from the examined NMF variant, there is another NMF variant that uses a non-random initialization approach, namely, Non-negative Double Singular Value Decomposition (NNDSVD) [92], which chooses initial factors based on a sparse SVD approximation of the original data matrix. In this case, the NMF algorithm is deterministic and thus very stable save for SVD implementation details, causing slight variability.

LDA requires randomization in the inference algorithm (e.g. MCMC or variational inference) and accordingly will have inherent instability.

8.1.4. Efficiency

Apart from ETM, the computation time is stable between runs. LSI, NMF, and LDA are the most efficient algorithms and run faster than HDP, CTM, and ProdLDA. On average, ETM takes longer than any other model. These results are consistent in both datasets. This result agrees with [65] and [7] who conducted a study involving several models and found that NMF is fastest, followed by LDA, Top2Vec [93], BERTopic and finally CTM. Analytically, LSI complexity is $O(mn^2)$ to decompose the DTM of m rows (documents) and n columns (terms). In practice, truncated SVD is used since not all singular values are needed, which makes LSI very fast. LDA algorithm computational complexity varies depending on the effective number of topics per document from polynomial-time to NP-hard [94] which is generally faster than neural models (ETM, CTM, and ProdLDA). We note that different models' computation times scale differently with the number of documents in the dataset. For example, LDA computation time increases more rapidly compared to NMF.

8.2. Model selection

There are different aspects to consider when selecting a topic model for an application. We propose to consider four aspects pertaining to the nature of the application as shown in figure 7. These aspects specify the relative importance of the models' evaluation criteria. One aspect checks whether the topic model's output will be consumed by humans or machine learning models. The coherence of the output

is more critical in the case of human consumption, such as the case in social sciences applications. If, however, the topic model output is deployed in another downstream task and is consumed by a machine, interpretability and hence coherence are less critical. Another aspect checks if a comprehensive coverage of input corpus is necessary. A diverse set of generated topics is more likely to cover the available themes in the corpus, which is important for applications such as text summarization and classification.

If the application involves spatial or temporal analysis or if it compares between different datasets, then stability is important. This importance is emphasized more for applications whose output is consumed by humans. Finally, if the application is interactive or the input corpus is vast, the model's efficiency becomes important.

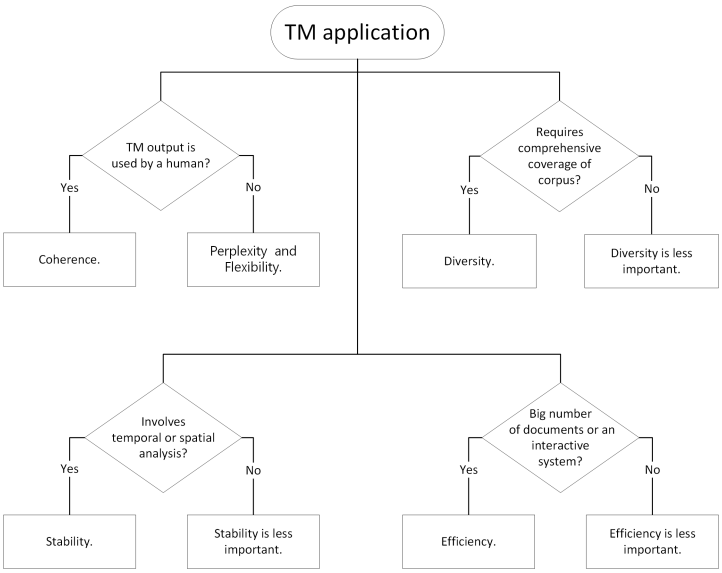


Figure 7: Four aspects to consider in applications to identify the relative importance of evaluation criteria.

Apart from the nature of the application, there are other aspects to consider when selecting a topic model. One aspect pertains to the input dataset, whose characteristics can impact the models' performance as mentioned in 5.4. For example, datasets with short documents are challenging to the classical topic models, such as LSI and LDA. These models reveal topics by capturing the word co-occurrence patterns at the document level; thus, they suffer from data sparsity. Also, in practice, the ease of use, availability of software packages, and need for rapid development play a role in determining which topic model to use. It is usually advisable to deploy a baseline model at first. LSI, NMF, and LDA are amongst the most common choices as a baseline model.

8.3. Research trends

Probabilistic models enjoyed several years of research following the advent of LDA in 2003. It was later extended in several ways to offer better corpora modeling, such as those spanning a large time frame or those involving short documents. The limited interpretability of the models grabbed the attention of researchers to design better models with better coherence scores. Moreover, how the models are evaluated was researched to invent better metrics. Figure 8 shows the trends observed from the surveyed papers. We note that although research in probabilistic topic modeling is still active, the focus in the research has shifted to neural models in the last five years. In particular, their flexibility in jointly training them with other models and their scalability. Currently, research into polylingual

models is gaining momentum. In general, transparency of machine learning models has become a hot topic, and topic modeling research follows the same trend.

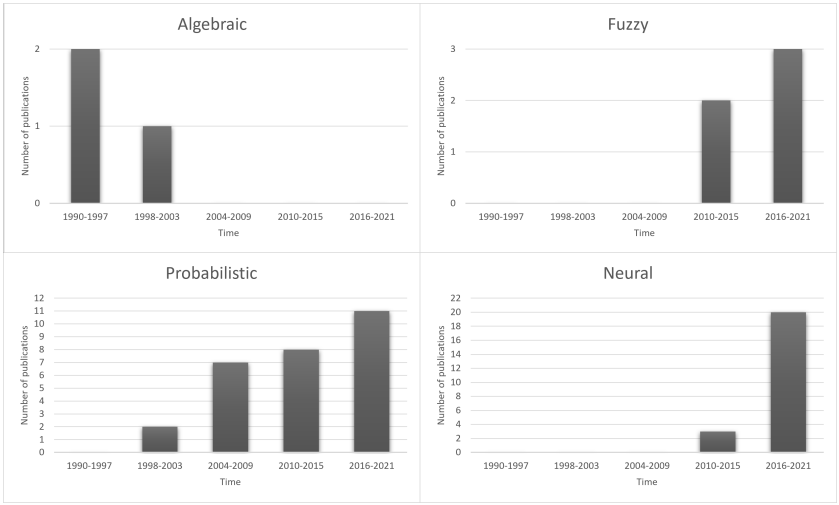


Figure 8: Evolution of topic models according to surveyed papers.

8.4. Research gaps

There are several research challenges and opportunities in TM, which are categorized into three groups. Better models, automatic evaluation, and data discovery. These research areas are illustrated as follows. Some research efforts go into better models that provide improved stability and coherence [95]. Regularization of the models and optimal selection of hyperparameters is also an area of ongoing research. While other research develops models that adapt better to various corpora characteristics, such as short text documents [96]. Computational efficiency is quite essential, which aptly grabs the attention of some researchers too [97]. The use of deep transformer-based language models to encode text documents has proved effective and is increasingly deployed in NTM [98].

Another research area is automatic evaluation [99], [100], [101]. While human evaluation is typically considered the gold standard in NLP, it is usually elaborate, costly, and prone to errors. There exists no unified system of evaluation that researchers adopt. Different papers vary in their evaluation processes and metrics, which makes it harder to compare different models. Therefore, there is a need for benchmarks and a unified evaluation system of topic models [85].

The last area of research is multi-disciplinary applications [82], [102], [103], where researchers from other disciplines, such as political science [104], neuroscience [105], psychology [106], sociology [107], bioinformatics [108], engineering [109] and many others, use topic models to explore their data and form hypotheses about it.

9. Conclusion and Future work

Topic modeling has been an active area of research in the past two decades. In this paper, topic models are categorized into four categories based on their modeling techniques. The modeling techniques are algebraic, fuzzy, probabilistic, and neural. Different models and model categories have different characteristics. They co-exist to serve in different contexts and on different corpora characteristics.

Probabilistic models provide a straightforward interpretation but scaling them up to complex models is challenging. On the other hand, neural models offer scalability and flexibility and hence have been the focus of research efforts in recent years. The models are evaluated against a set of criteria to assess them. They are quality, interpretability, diversity, stability, efficiency, and flexibility. Using two common benchmark datasets, we conducted experiments to evaluate seven topic models from different categories across the proposed criteria. We discussed how the performance changes due to different input dataset characteristics. The evaluation criteria are competing, and scoring high in one criterion could challenge the others, such as the case with stability and diversity. Accordingly, we conclude that no single model achieves the best results in all criteria. We proposed considering four key aspects of the nature of an application to help determine which topic model to use. Evaluating a topic model is challenging. Thus, research efforts are ongoing to provide better metrics to evaluate the models automatically and provide a unified set of benchmarks, including datasets. Other research goes into developing better models to generate more coherent, diverse, and stable topics in a computationally efficient manner. Another research direction is incorporating the rich semantics representation of the pre-trained language models. Due to their interpretability, topic models continue to be deployed in various disciplines ranging from medicine to literature and beyond. To extend our study, future work to compare more models on several datasets exhibiting different characteristics is required. This comparison will help stratify different use cases in the industry and highlight key metrics for evaluation.

References

[1] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (Jan) (2003) 993–1022.

[2] S. P. Crain, K. Zhou, S.-H. Yang, H. Zha, Dimensionality Reduction and Topic Modeling: From Latent Semantic Indexing to Latent Dirichlet Allocation and Beyond, in: C. C. Aggarwal, C. Zhai (Eds.), Mining Text Data, Springer US, Boston, MA, 2012, pp. 129–161. doi:10.1007/978-1-4614-3223-4_5.

[3] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, R. Harshman, Indexing by latent semantic analysis, Journal of the American Society for Information Science 41 (6) (1990) 391–407. doi:https://doi.org/10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9.

[4] J. Lafferty, D. Blei, Correlated topic models, in: Y. Weiss, B. Schölkopf, J. Platt (Eds.), Advances in Neural Information Processing Systems, Vol. 18, MIT Press, 2005, p. 8.

[5] D. M. Blei, J. D. Lafferty, Dynamic topic models, in: Proceedings of the 23rd international conference on Machine learning - ICML '06, ACM Press, Pittsburgh, Pennsylvania, 2006, pp. 113–120. doi:10.1145/1143844.1143859.

[6] Z. Cao, S. Li, Y. Liu, W. Li, H. Ji, A Novel Neural Topic Model and Its Supervised Extension, in: Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.

[7] M. Grootendorst, BERTopic: Neural topic modeling with a class-based TF-IDF procedure, arXiv:2203.05794 [cs] (Mar. 2022). doi:10.48550/arXiv.2203.05794.
URL http://arxiv.org/abs/2203.05794

[8] A. B. Dieng, F. J. R. Ruiz, D. M. Blei, Topic Modeling in Embedding Spaces, Transactions of the Association for Computational Linguistics 8 (2020) 439–453. doi:10.1162/tacl_a_00325.

[9] A. Srivastava, C. Sutton, Autoencoding Variational Inference For Topic Models, arXiv:1703.01488 [stat]ArXiv: 1703.01488 (Mar. 2017).

[10] Y. Miao, E. Grefenstette, P. Blunsom, Discovering Discrete Latent Topics with Neural Variational Inference, arXiv:1706.00359 [cs]ArXiv: 1706.00359 (May 2018).

[11] F. Bianchi, S. Terragni, D. Hovy, Pre-training is a Hot Topic: Contextualized Document Embeddings Improve Topic Coherence, arXiv:2004.03974 [cs]ArXiv: 2004.03974 (Jun. 2021).

[12] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv:1810.04805 [cs]ArXiv: 1810.04805 (May 2019).

[13] R. Alghamdi, K. Alfalqi, A Survey of Topic Modeling in Text Mining, International Journal of Advanced Computer Science and Applications 6 (1) (2015). doi:10.14569/IJACSA.2015.060121.

[14] D. Sharma, A Survey on Journey of Topic Modeling Techniques from SVD to Deep Learning, International Journal of Modern Education and Computer Science Vol. 9 (2017) PP.50–62. doi:10.5815/ijmecs.2017.07.06.

[15] B. V. Barde, A. M. Bainwad, An overview of topic modeling methods and tools, in: 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), IEEE, Madurai, 2017, pp. 745–750. doi:10.1109/ICCONS.2017.8250563.

[16] L. Xia, D. Luo, C. Zhang, Z. Wu, A survey of topic models in text classification, in: 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD), 2019, pp. 244–250. doi:10.1109/ICAIBD.2019.8836970.

[17] H. Jelodar, Y. Wang, C. Yuan, X. Feng, X. Jiang, Y. Li, L. Zhao, Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey, Multimedia Tools and Applications 78 (11) (2019) 15169–15211. doi:10.1007/s11042-018-6894-4.

[18] S. Likhitha, B. S., H. M., A Detailed Survey on Topic Modeling for Document and Short Text Data, International Journal of Computer Applications 178 (39) (2019) 1–9. doi:10.5120/ijca2019919265.

[19] H. Zhao, D. Phung, V. Huynh, Y. Jin, L. Du, W. Buntine, Topic Modelling Meets Deep Neural Networks: A Survey, arXiv:2103.00498 [cs]ArXiv: 2103.00498 (Feb. 2021).

[20] T. Hofmann, Probabilistic latent semantic indexing, in: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '99, Association for Computing Machinery, New York, NY, USA, 1999, p. 50–57. doi:10.1145/312624.312649.
URL https://doi.org/10.1145/312624.312649

[21] H. Rubenstein, J. B. Goodenough, Contextual correlates of synonymy, Communications of the ACM 8 (10) (1965) 627–633. doi:10.1145/365628.365657.

[22] Z. S. Harris, Distributional Structure, WORD 10 (2-3) (1954) 146–162, publisher: Routledge _eprint: https://doi.org/10.1080/00437956.1954.11659520. doi:10.1080/00437956.1954.11659520.

[23] Y. Bengio, A. Courville, P. Vincent, Representation Learning: A Review and New Perspectives, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (8) (2013) 1798–1828, conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. doi:10.1109/TPAMI.2013.50.

[24] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, A. Lerchner, Understanding disentangling in β-VAE, arXiv:1804.03599 [cs, stat]ArXiv: 1804.03599 (Apr. 2018).

[25] D. E. Klein, G. L. Murphy, The representation of polysemous words, Journal of Memory and Language 45 (2001) 259–282.

[26] A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, M. Telgarsky, Tensor Decompositions for Learning Latent Variable Models:, Tech. rep., Defense Technical Information Center, Fort Belvoir, VA (Dec. 2012). doi:10.21236/ADA604494.

[27] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (6755) (1999) 788–791. doi:10.1038/44565.

[28] T. Shi, K. Kang, J. Choo, C. K. Reddy, Short-Text Topic Modeling via Non-negative Matrix Factorization Enriched with Local Word-Context Correlations, in: Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18, ACM Press, Lyon, France, 2018, pp. 1105–1114. doi:10.1145/3178876.3186009.

[29] I. Korshunova, H. Xiong, M. Fedoryszak, L. Theis, Discriminative topic modeling with logistic lda, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc., 2019.

[30] T. L. Griffiths, M. Steyvers, Finding scientific topics, Proceedings of the National Academy of Sciences 101 (suppl 1) (2004) 5228–5235, publisher: National Academy of Sciences Section: Colloquium. doi:10.1073/pnas.0307752101.

[31] A. Agrawal, W. Fu, T. Menzies, What is Wrong with Topic Modeling? (and How to Fix it Using Search-based Software Engineering), Information and Software Technology 98 (2018) 74–88, arXiv: 1608.08176. doi:10.1016/j.infsof.2018.02.005.

[32] H. M. Wallach, Topic modeling: beyond bag-of-words, in: Proceedings of the 23rd international conference on Machine learning - ICML '06, ACM Press, Pittsburgh, Pennsylvania, 2006, pp. 977–984. doi:10.1145/1143844.1143967.

[33] Y. Jo, A. H. Oh, Aspect and sentiment unification model for online review analysis, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 815–824. doi:10.1145/1935826.1935932.
URL https://doi.org/10.1145/1935826.1935932

[34] W. Li, A. McCallum, Pachinko allocation: DAG-structured mixture models of topic correlations, in: Proceedings of the 23rd international conference on Machine learning - ICML '06, ACM Press, Pittsburgh, Pennsylvania, 2006, pp. 577–584. doi:10.1145/1143844.1143917.

[35] M. Rosen-Zvi, T. Griffiths, M. Steyvers, P. Smyth, The author-topic model for authors and documents, in: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, UAI '04, AUAI Press, Arlington, Virginia, USA, 2004, p. 487–494.

[36] J. Chang, D. Blei, Relational Topic Models for Document Networks, in: Artificial Intelligence and Statistics, PMLR, 2009, pp. 81–88, iSSN: 1938-7228.

[37] Y. W. Teh, M. I. Jordan, M. J. Beal, D. M. Blei, Hierarchical Dirichlet Processes, Journal of the American Statistical Association 101 (476) (2006) 1566–1581, publisher: Taylor & Francis. doi:10.1198/016214506000000302.

[38] X. Yan, J. Guo, Y. Lan, X. Cheng, A biterm topic model for short texts, in: Proceedings of the 22nd international conference on World Wide Web - WWW '13, ACM Press, Rio de Janeiro, Brazil, 2013, pp. 1445–1456. doi:10.1145/2488388.2488514.

[39] M.-S. Yang, A survey of fuzzy clustering, Mathematical and Computer Modelling 18 (11) (1993) 1–16. doi:10.1016/0895-7177(93)90202-A.

[40] A. Karami, A. Gangopadhyay, B. Zhou, H. Kharrazi, Fuzzy approach topic modeling for health and medical corpora, International Journal of Fuzzy Systems 20 (08 2017). doi:10.1007/s40815-017-0327-9.

[41] N. Akhtar, M. M. Sufyan Beg, H. Javed, Topic Modelling with Fuzzy Document Representation, in: M. Singh, P. Gupta, V. Tyagi, J. Flusser, T. Ören, R. Kashyap (Eds.), Advances in Computing and Data Sciences, Communications in Computer and Information Science, Springer, Singapore, 2019, pp. 577–587. doi:10.1007/978-981-13-9942-8_54.

[42] J. Rashid, S. M. A. Shah, A. Irtaza, Fuzzy topic modeling approach for text mining over short text, Information Processing & Management 56 (6) (2019) 102060. doi:10.1016/j.ipm.2019.102060.

[43] S. Abri, R. Abri, Providing a Personalization Model Based on Fuzzy Topic Modeling, Arabian Journal for Science and Engineering 46 (4) (2021) 3079–3086. doi:10.1007/s13369-020-05048-7.

[44] A. Karami, A. Gangopadhyay, B. Zhou, H. Kharrazi, FLATM: A Fuzzy Logic Approach Topic Model for Medical Documents, arXiv:1911.10953 [cs]ArXiv: 1911.10953 (Nov. 2019).

[45] S. Basu, Y. Yu, R. Zimmermann, Fuzzy clustering of lecture videos based on topic modeling, in: 2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI), IEEE, Bucharest, Romania, 2016, pp. 1–6. doi:10.1109/CBMI.2016.7500264.

[46] R. Das, M. Zaheer, C. Dyer, Gaussian LDA for Topic Models with Word Embeddings, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Beijing, China, 2015, pp. 795–804. doi:10.3115/v1/P15-1077.

[47] C. E. Moody, Mixing Dirichlet Topic Models and Word Embeddings to Make lda2vec, arXiv:1605.02019 [cs]ArXiv: 1605.02019 (May 2016).

[48] D. Q. Nguyen, R. Billingsley, L. Du, M. Johnson, Improving Topic Models with Latent Feature Word Representations, arXiv:1810.06306 [cs]ArXiv: 1810.06306 (Oct. 2018).

[49] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, arXiv:1301.3781 [cs]ArXiv: 1301.3781 (Sep. 2013).

[50] C. Li, H. Wang, Z. Zhang, A. Sun, Z. Ma, Topic Modeling for Short Texts with Auxiliary Word Embeddings, in: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, ACM, Pisa Italy, 2016, pp. 165–174. doi:10.1145/2911451.2911499.

[51] S. Sia, A. Dalmia, S. J. Mielke, Tired of Topic Models? Clusters of Pretrained Word Embeddings Make for Fast and Good Topics too!, arXiv:2004.14914 [cs] (Oct. 2020). doi:10.48550/arXiv.2004.14914.

[52] L. McInnes, J. Healy, S. Astels, hdbscan: Hierarchical density based clustering, The Journal of Open Source Software 2 (11) (2017) 205. doi:10.21105/joss.00205.

[53] J. He, Z. Hu, T. Berg-Kirkpatrick, Y. Huang, E. P. Xing, Efficient Correlated Topic Modeling with Topic Embedding, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, Halifax NS Canada, 2017, pp. 225–233. doi:10.1145/3097983.3098074.

[54] D. P. Kingma, M. Welling, Auto-Encoding Variational Bayes, arXiv:1312.6114 [cs, stat]ArXiv: 1312.6114 (May 2014).

[55] D. P. Kingma, M. Welling, An Introduction to Variational Autoencoders, Foundations and Trends® in Machine Learning 12 (4) (2019) 307–392, arXiv: 1906.02691. doi:10.1561/2200000056.

[56] R. Ding, R. Nallapati, B. Xiang, Coherence-Aware Neural Topic Modeling, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 830–836. doi:10.18653/v1/D18-1096.

[57] S. Brody, M. Lapata, Bayesian Word Sense Induction, in: Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), Association for Computational Linguistics, Athens, Greece, 2009, pp. 103–111.

[58] A. Haghighi, L. Vanderwende, Exploring Content Models for Multi-Document Summarization, in: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Boulder, Colorado, 2009, pp. 362–370.

[59] H. M. Wallach, I. Murray, R. Salakhutdinov, D. Mimno, Evaluation methods for topic models, in: Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09, Association for Computing Machinery, New York, NY, USA, 2009, pp. 1105–1112. doi:10.1145/1553374.1553515.

[60] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, A. McCallum, Optimizing semantic coherence in topic models, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11, Association for Computational Linguistics, USA, 2011, p. 262–272.

[61] D. Newman, J. H. Lau, K. Grieser, T. Baldwin, Automatic evaluation of topic coherence, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10, Association for Computational Linguistics, USA, 2010, p. 100–108.

[62] J. Chang, S. Gerrish, C. Wang, J. Boyd-graber, D. Blei, Reading tea leaves: How humans interpret topic models, in: Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, A. Culotta (Eds.), Advances in Neural Information Processing Systems, Vol. 22, Curran Associates, Inc., 2009.

[63] M. Mäntylä, M. Claes, U. Farooq, Measuring LDA Topic Stability from Clusters of Replicated Runs, Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (2018) 1–4ArXiv: 1808.08098. doi:10.1145/3239235.3267435.

[64] W. Webber, A. Moffat, J. Zobel, A similarity measure for indefinite rankings, ACM Transactions on Information Systems 28 (4) (2010) 1–38. doi:10.1145/1852102.1852106.

[65] M. Belford, B. Mac Namee, D. Greene, Stability of topic modeling via matrix factorization, Expert Systems with Applications 91 (2018) 159–169. doi:10.1016/j.eswa.2017.08.047.

[66] R. Rehurek, P. Sojka, Software framework for topic modelling with large corpora, in: IN PROCEEDINGS OF THE LREC 2010 WORKSHOP ON NEW CHALLENGES FOR NLP FRAMEWORKS, 2010, pp. 45–50.

[67] Q. Sun, R. Li, D. Luo, X. Wu, Text Segmentation with LDA-Based Fisher Kernel, in: Proceedings of ACL-08: HLT, Short Papers, Association for Computational Linguistics, Columbus, Ohio, 2008, pp. 269–272.

[68] B. Jeong, J. Yoon, J.-M. Lee, Social media mining for product planning: A product opportunity mining approach based on topic modeling and sentiment analysis, International Journal of Information Management 48 (2019) 280–290. doi:10.1016/j.ijinfomgt.2017.09.009.

[69] J. Zhu, S. Chowdhury, Towards the Ontology Development for Smart Transportation Infrastructure Planning via Topic Modeling, ISARC Proceedings (2019) 507–514Publisher: IAARC.

[70] L. Sun, Y. Yin, Discovering themes and trends in transportation research using topic modeling, Transportation Research Part C: Emerging Technologies 77 (2017) 49–66. doi:10.1016/j.trc.2017.01.013.

[71] A. Ambrosino, M. Cedrini, J. B. Davis, S. Fiori, M. Guerzoni, M. Nuccio, What topic modeling could reveal about the evolution of economics, Journal of Economic Methodology 25 (4) (2018) 329–348. doi:10.1080/1350178X.2018.1529215.

[72] B. Dit, M. Revelle, M. Gethers, D. Poshyvanyk, Feature location in source code: a taxonomy and survey, Journal of Software: Evolution and Process 25 (1) (2013) 53–95, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/smr.567. doi:10.1002/smr.567.

[73] H. Hemmati, Z. Fang, M. V. Mäntylä, B. Adams, Prioritizing manual test cases in rapid release environments, Software Testing, Verification and Reliability 27 (6) (2017) e1609, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/stvr.1609. doi:10.1002/stvr.1609.

[74] A. Hindle, C. Bird, T. Zimmermann, N. Nagappan, Relating requirements to implementation via topic analysis: Do topics extracted from requirements make sense to managers and developers?, in: 2012 28th IEEE International Conference on Software Maintenance (ICSM), 2012, pp. 243–252, iSSN: 1063-6773. doi:10.1109/ICSM.2012.6405278.

[75] J. Garcia, D. Popescu, C. Mattmann, N. Medvidovic, Y. Cai, Enhancing architectural recovery using concerns, in: Proceedings of the 2011 26th IEEE/ACM International Conference on Automated Software Engineering, ASE '11, IEEE Computer Society, USA, 2011, pp. 552–555. doi:10.1109/ASE.2011.6100123.

[76] L. Liu, L. Tang, W. Dong, S. Yao, W. Zhou, An overview of topic modeling and its current applications in bioinformatics, SpringerPlus 5 (1) (2016) 1608. doi:10.1186/s40064-016-3252-8.

[77] G. E. Heo, K. Y. Kang, M. Song, J.-H. Lee, Analyzing the field of bioinformatics with the multi-faceted topic modeling technique, BMC Bioinformatics 18 (S7) (2017) 251. doi:10.1186/s12859-017-1640-x.

[78] M. La Rosa, A. Fiannaca, R. Rizzo, A. Urso, Probabilistic topic modeling for the analysis and classification of genomic sequences, BMC Bioinformatics 16 (S6) (2015) S2. doi:10.1186/1471-2105-16-S6-S2.

[79] S. J. Kho, H. B. Yalamanchili, M. L. Raymer, A. P. Sheth, A Novel Approach for Classifying Gene Expression Data using Topic Modeling, in: Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology,and Health Informatics, ACM, Boston Massachusetts USA, 2017, pp. 388–393. doi:10.1145/3107411.3107483.

[80] D. Maier, A. Waldherr, P. Miltner, G. Wiedemann, A. Niekler, A. Keinert, B. Pfetsch, G. Heyer, U. Reber, T. Häussler, H. Schmid-Petri, S. Adam, Applying LDA Topic Modeling in Communication Research: Toward a Valid and Reliable Methodology, Communication Methods and Measures 12 (2-3) (2018) 93–118. doi:10.1080/19312458.2018.1430754.

[81] C. Puschmann, T. Scheffler, Topic Modeling for Media and Communication Research: A Short Primer, SSRN Scholarly Paper ID 2836478, Social Science Research Network, Rochester, NY (Aug. 2016). doi:10.2139/ssrn.2836478.

[82] S. Srinivas, S. Ramachandiran, Discovering Airline-Specific Business Intelligence from Online Passenger Reviews: An Unsupervised Text Analytics Approach, arXiv:2012.08000 [cs]ArXiv: 2012.08000 version: 1 (Dec. 2020).

[83] Y. Wang, X. Zhao, Z. Sun, H. Yan, L. Wang, Z. Jin, L. Wang, Y. Gao, C. Law, J. Zeng, Peacock: Learning Long-Tail Topic Features for Industrial Applications, arXiv:1405.4402 [cs]ArXiv: 1405.4402 (Dec. 2014).

[84] D. Jiang, Z. Chen, R. Lian, S. Bao, C. Li, Familia: An Open-Source Toolkit for Industrial Topic Modeling, arXiv:1707.09823 [cs]ArXiv: 1707.09823 version: 1 (Jul. 2017).

[85] S. Terragni, E. Fersini, B. G. Galuzzi, P. Tropeano, A. Candelieri, OCTIS: Comparing and Optimizing Topic models is Simple!, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Online, 2021, pp. 263–270.

[86] A. Schofield, M. Magnusson, D. Mimno, Pulling out the stops: Rethinking stopword removal for topic models, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, Association for Computational Linguistics, Valencia, Spain, 2017, pp. 432–436.
URL https://aclanthology.org/E17-2069

[87] S. Kesiraju, O. Plchot, L. Burget, S. V. Gangashetty, Learning document embeddings along with their uncertainties, IEEE/ACM Transactions on Audio, Speech, and Language Processing 28 (2020) 2319–2332, arXiv: 1908.07599 version: 3. doi:10.1109/TASLP.2020.3012062.

[88] Y. Miao, L. Yu, P. Blunsom, Neural Variational Inference for Text Processing, arXiv:1511.06038 [cs, stat]ArXiv: 1511.06038 version: 4 (Jun. 2016).

[89] K. W. Lim, W. Buntine, Bibliographic analysis with the citation network topic model, in: D. Phung, H. Li (Eds.), Proceedings of the Sixth Asian Conference on Machine Learning, Vol. 39 of Proceedings of Machine Learning Research, PMLR, Nha Trang City, Vietnam, 2015, pp. 142–158.

[90] D. O'Callaghan, D. Greene, J. Carthy, P. Cunningham, An analysis of the coherence of descriptors in topic modeling, Expert Systems with Applications 42 (13) (2015) 5645–5657. doi:10.1016/j.eswa.2015.02.055.

[91] T.-N. Doan, T.-A. Hoang, Benchmarking Neural Topic Models: An Empirical Study, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, Online, 2021, pp. 4363–4368. doi:10.18653/v1/2021.findings-acl.382.

[92] C. Boutsidis, E. Gallopoulos, SVD based initialization: A head start for nonnegative matrix factorization, Pattern Recognition 41 (4) (2008) 1350–1362. doi:10.1016/j.patcog.2007.09.010.

[93] D. Angelov, Top2Vec: Distributed Representations of Topics, arXiv:2008.09470 [cs, stat] (Aug. 2020). doi:10.48550/arXiv.2008.09470.

[94] D. Sontag, D. Roy, Complexity of Inference in Latent Dirichlet Allocation, in: Advances in Neural Information Processing Systems, Vol. 24, Curran Associates, Inc., 2011.

[95] R. Wang, D. Zhou, Y. Xiong, H. Huang, Variational Gaussian Topic Model with Invertible Neural Projections, arXiv:2105.10095 [cs]ArXiv: 2105.10095 version: 1 (May 2021).

[96] S. Sia, K. Duh, Adaptive Mixed Component LDA for Low Resource Topic Modeling, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Association for Computational Linguistics, Online, 2021, pp. 2451–2469.

[97] H. Zhao, L. Du, W. Buntine, G. Liu, MetaLDA: a Topic Model that Efficiently Incorporates Meta information, arXiv:1709.06365 [cs, stat]ArXiv: 1709.06365 version: 1 (Sep. 2017).

[98] F. Bianchi, S. Terragni, D. Hovy, D. Nozza, E. Fersini, Cross-lingual Contextualized Topic Models with Zero-shot Learning, arXiv:2004.07737 [cs]ArXiv: 2004.07737 (Feb. 2021).

[99] A. Hoyle, P. Goel, D. Peskov, A. Hian-Cheong, J. Boyd-Graber, P. Resnik, Is Automated Topic Model Evaluation Broken?: The Incoherence of Coherence, arXiv:2107.02173 [cs]ArXiv: 2107.02173 version: 1 (Jul. 2021).

[100] C. Doogan, W. Buntine, Topic Model or Topic Twaddle? Re-evaluating Semantic Interpretability Measures, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Online, 2021, pp. 3824–3848. doi:10.18653/v1/2021.naacl-main.300.

[101] S. Koltcov, V. Ignatenko, M. Terpilovskii, P. Rosso, Analysis and tuning of hierarchical topic models based on Renyi entropy approach, arXiv:2101.07598 [cs, stat]ArXiv: 2101.07598 version: 1 (Jan. 2021).

[102] J. Marjanen, E. Zosa, S. Hengchen, L. Pivovarova, M. Tolonen, Topic modelling discourse dynamics in historical newspapers, arXiv:2011.10428 [cs]ArXiv: 2011.10428 version: 1 (Nov. 2020).

[103] C. Schöch, Topic Modeling Genre: An Exploration of French Classical and Enlightenment Drama, arXiv:2103.13019 [cs]ArXiv: 2103.13019 (Mar. 2021).

[104] J. Dehler-Holland, K. Schumacher, W. Fichtner, Topic Modeling Uncovers Shifts in Media Framing of the German Renewable Energy Act, Patterns 2 (1) (2021) 100169. doi:10.1016/j.patter.2020.100169.

[105] T. Renteria-Vazquez, W. S. Brown, C. Kang, M. Graves, F. Castelli, L. K. Paul, Social Inferences in Agenesis of the Corpus Callosum and Autism: Semantic Analysis and Topic Modeling, Journal of Autism and Developmental Disorders (Mar. 2021). doi:10.1007/s10803-021-04957-2.

[106] M. D. Armstrong, D. Maupomé, M.-J. Meurs, Topic Modeling in Embedding Spaces for Depression Assessment, Proceedings of the Canadian Conference on Artificial Intelligence (Jun. 2021). doi:10.21428/594757db.9e67a9f0.

[107] A. Arseniev-Koehler, S. D. Cochran, V. M. Mays, K.-W. Chang, J. G. Foster, Integrating topic modeling and word embedding to characterize violent deaths, arXiv:2106.14365 [cs]ArXiv: 2106.14365 (Aug. 2020). doi:10.31235/osf.io/nkyaq.

[108] A. Pancheva, H. Wheadon, S. Rogers, T. Otto, Using topic modeling to detect cellular crosstalk in scrna-seq, bioRxiv (2021). arXiv:https://www.biorxiv.org/content/early/2021/07/26/2021.07.26.453767.full.pdf, doi:10.1101/2021.07.26.453767.

[109] S. Rani, M. Kumar, Topic modeling and its applications in materials science and engineering, Materials Today: Proceedings 45 (2021) 5591–5596. doi:10.1016/j.matpr.2021.02.313.