AI

MDPI

Article

Emerging Research Topic Detection Using Filtered-LDA

Fuad Alattar * and Khaled Shaalan

Faculty of Engineering and IT, The British University in Dubai, Dubai 345015, United Arab Emirates; Khaled.shaalan@buid.ac.ae

* Correspondence: fuad.alattar@hotmail.com

Abstract: Comparing two sets of documents to identify new topics is useful in many applications, like discovering trending topics from sets of scientific papers, emerging topic detection in microblogs, and interpreting sentiment variations in Twitter. In this paper, the main topic-modeling-based approaches to address this task are examined to identify limitations and necessary enhancements. To overcome these limitations, we introduce two separate frameworks to discover emerging topics through a filtered latent Dirichlet allocation (filtered-LDA) model. The model acts as a filter that identifies old topics from a timestamped set of documents, removes all documents that focus on old topics, and keeps documents that discuss new topics. Filtered-LDA also genuinely reduces the chance of using keywords from old topics to represent emerging topics. The final stage of the filter uses multiple topic visualization formats to improve human interpretability of the filtered topics, and it presents the most-representative document for each topic.

Keywords: emerging topic detection; research trend detection; topic discovery; topic modeling; hot topics; trending topics; FB-LDA; Filtered-LDA

Citation: Alattar, F.; Shaalan, K. Emerging Research Topic Detection Using Filtered-LDA. *AI* **2021**, *2*, 578–599. https://doi.org/10.3390/ai2040035

Academic Editor: Amir Mosavi

Received: 12 June 2021
Accepted: 21 October 2021
Published: 31 October 2021

1. Introduction

Finding the right hot scientific topic is a common challenging task for many students and researchers. To illustrate, a PhD student should read many scientific papers on a specific field to identify candidate emerging topics before proposing a dissertation. This time-consuming exercise usually covers multiple years of publications to spot evolution of new topics. During the last two decades, multiple techniques were introduced to handle similar tasks, wherein large sets of timestamped documents are processed by a text mining program to automatically detect emerging topics. Some of these techniques shall be briefly described here in both Sections 2 and 3. However, we focus in this paper only on those techniques that employ topic models, which use statistical algorithms to detect topics that appear in texts. Topic models treat each part of data as a word document. A collection of word documents forms a corpus. Topics can usually be predicted based on some similar words that appear inside each document. Therefore, each document may consist of multiple topics, whereas its dominant topic is the one which is discussed more inside that document.

Some topic models are nonprobabilistic, like the latent semantic analysis (LSA) [1] and the non-negative matrix factorization (NMF) [2], whereas other topic models are probabilistic, like probabilistic latent semantic analysis (PLSA) [3] and latent Dirichlet Allocation (LDA) [4].

LDA is one of the most used topic models because of its good performance and ability to produce coherent outputs for many applications [5]. LDA represents each document by a distribution of fixed number of topics. Each one of these topics is represented by a distribution of words.

Figure 1 shows a graphical LDA model, which is based on [6]. It includes three levels of representations. Corpus-level representation uses hyperparameters α and β, which are sampled once when a corpus is generated. The document-level representation's variables

are $\vec{\theta m}$, which are sampled once for each document. The word-level representation's variables are $z_{m,n}$ and w_{mn}, which are sampled once for each word.

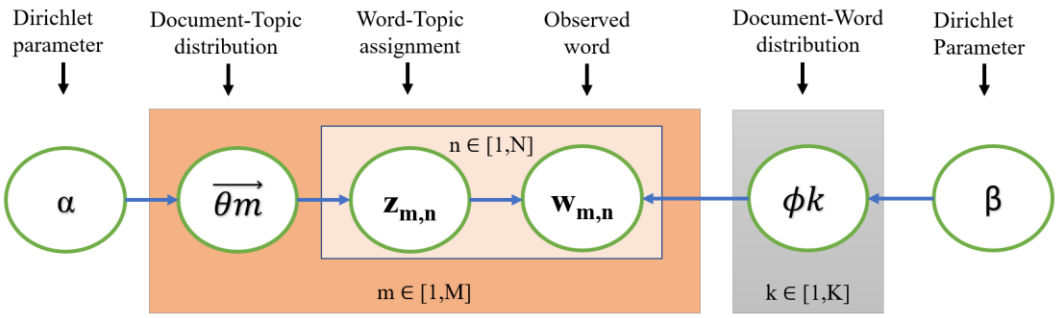

Figure 1. Latent Dirichlet allocation (LDA) model.

Given that M is the number of documents in a corpus, and n is the number of words in a document, then D_m is the document number m in the corpus, whereas $\vec{\theta m}$ *is* the multinomial distribution which includes D_m in the latent topic $Z_{m,n}$.

The hyperparameter α for this distribution follows the Dirichlet distribution. "K" represents the number of topics which can be either statistically calculated or selected manually by the user. $\vec{\phi k}$ is the words' multinomial distribution for K topics. This distribution has the hyperparameter β which follows the Dirichlet distribution. Probability of the word $w_{m,n}$ is decided by $p(w_{m,n} \mid z_{m,n}, \beta)$.

Tuning LDA hyperparameters is important for obtaining accurate results. In general, alpha (α) decides mixture of topics inside a document, whereas beta (β) decides mixture of words for each topic. For instance, increasing Alpha would increase mixture of topics [7]. Figure 2—which is obtained from [7]—shows an example of Wikipedia article on Mitt Romney, which demonstrates the smoothing impact of hyperparameter alpha. When Alpha is low, weight is assigned to one topic, whereas weight gets distributed among topics when alpha is high.

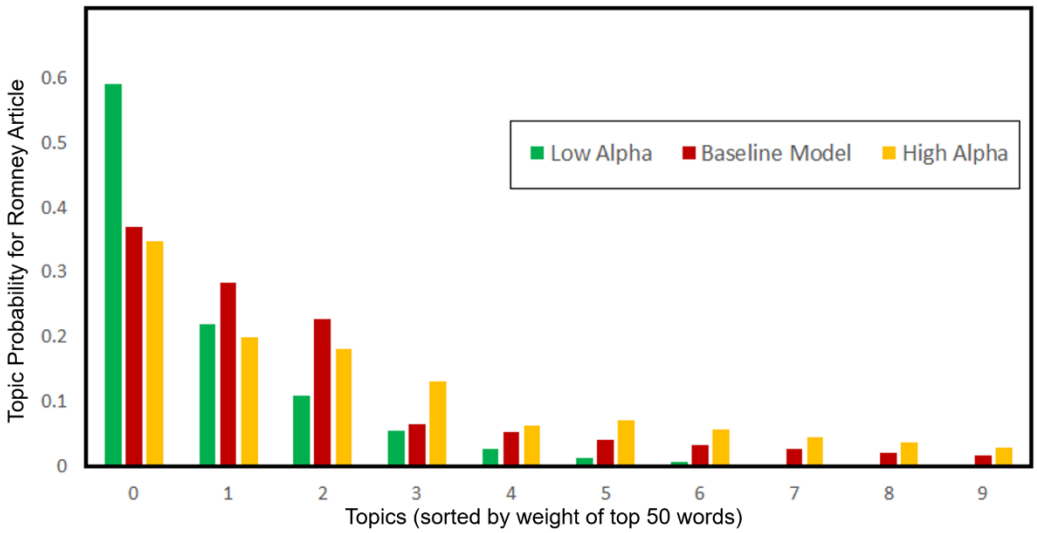

Figure 2. Smoothing impact of LDA hyperparameter Alpha.

Regular topic models like LDA use unsupervised learning techniques that are designed to discover discussed topics within text, however they do not have mechanisms to distinguish between old and new topics. As a result, some approaches were developed by various researchers to address the emerging topic detection problem.

In the next sections, we review three main base methods for emerging topic detection, then we introduce our new model that overperforms base methods. In Section 2, we summarize main research work related to emerging topic detection. Section 3 briefly describes the three base methods, which are the dynamic topic model, the partially labeled Dirichlet allocation model, and the foreground-background LDA model. In Section 4, we describe the two datasets which are used in Section 5 to test the performance of base methods. The main contribution of our paper is developing mechanisms to enhance performance of topic models for tracking topics over time. Multiple frameworks are proposed to implement these mechanisms. Section 6 introduces our new filtered-LDA model and its two options of implementation through two separate frameworks, then it tests these frameworks to analyze their performance. Finally we summarize our results in Section 7.

2. Related Work

Our research work addresses capturing emerging topics when two sets of documents are compared, therefore research work related to event detection and tracking is not relevant to our paper as the task of our frameworks is not detecting bursts of events only, but to detect emerging topics even when they are not among the top popular topics.

Erten et al. [8] visualized temporal patterns to detect emerging topics using citations and relations between documents from conference proceedings of the Association for Computing Machinery (ACM). Their proposed visualization method identifies gradually fading topics and fast-growing topics. It detects the highest five frequently words that are used in the papers' titles. However, it is not easy to apply this technique on other unlabeled datasets as it utilizes the ACM special classification of labeled papers to discover the research emerging topics in each year.

Le et al. [9] introduced a model to discover scientific trends from academic articles. The model employs both hidden Markov models (HMMs) [10] and maximum-entropy Markov models (MEMMs) [11], and it could achieve a maximum accuracy of 58% when tested on 100 papers.

Tan et al. [12] applied the foreground and background latent Dirichlet allocation (FB-LDA) model on the application of scientific trend detection. Their model could discover emerging topics when applied on SIGIR dataset. However, many old topics were also detected among the emerging topics of FB-LDA. Furthermore, the authors [12] did not provide clear guidelines for selecting the variable "K".

Shen and Wang [13] used the VOSviewer software, the look-up tool in MS Excel, and charts of MS PowerPoint to track topics related to the Perovskite solar cell (PSC). However, their approach requires manual tuning of the threshold related to minimum term's occurrences. Furthermore, emerging topics are visually detected by the user in this approach as it did not propose any mechanism to automatically articulate detected topics. A similar topic visualization approach was proposed by [14–18].

Some other emerging topic detection methods are domain-specific, hence their performance on other domains is unknown. For example, Bolelli et al. [19] proposed an LDA-based model using a segmented author topic by inserting the temporal ordering inside CiteSeer research papers. Like partially labeled Dirichlet allocation (PLDA) [20], their model divides the duration of the publications into multiple time slots, it applies LDA on the first time slot, then it tries to correlate the topics of other time slots with the LDA topics of the first time slot. Morinaga and Yamanishi [21] introduced a model using a standard clustering process to examine the variations in time of the detected components to monitor emerging topics. However, their method relies on a collection of email documents, which makes it difficult to evaluate their model's performance on other domain of documents, like research papers. Behpour et al. [22] could avoid such domain-dependency limitation by introducing a temporal bias to the clustering process and they could improve the sharpness of topic trends, which means that they could enhance the model's ability to discover emerging topics.

Marrone [23] used an entity linking approach to examine topic popularity in a set of papers of the Information Science Journal. This approach uses a knowledge base to link word strings to entities, then it automatically identifies topics based on entity mentions. Multiple indicators are used to identify which topics are active. Although this approach could solve the problem of identifying the number of topics for standard topic modeling methods, it has a limitation of relying on rapidly growing topics. This makes this approach closer to the event detection methods which cannot identify emerging topics before they form a surge in the topic's pipeline.

3. Topic Modeling Base Methods

Given that our research work focuses only on identifying emerging topics when two sets of documents are compared, we shall not address here other emerging topic detection problems, like event detection for identifying burst topics. In our research problem, an emerging topic may exist inside few documents only, and it may not cause a surge that can be discovered through event detection techniques. Furthermore, we shall not address the citation-based emerging topic methods which create clusters from datasets then analyzes topics that appear in each cluster. Manual labeling of citation information for each document would require additional human efforts, and our research focuses on detecting all emerging topics at an early stage even before they receive high number of citations.

In the following subsections, we briefly address three main topic models that are still being used to handle emerging topic detection. These models are the dynamic topic model (DTM) [24], the partially labeled Dirichlet allocation (PLDA) [20], and the foreground-background latent Dirichlet allocation (FB-LDA) [12]. Later, we test these models on a real-life dataset to examine their performance, then we introduce our proposed filtered-LDA frameworks to overcome limitations of existing models.

3.1. Dynamic Topic Model

DTM was developed to examine evolution of topics inside a large set of documents. It is a probabilistic time series model that utilizes multinomial distribution to represent topics. Unlike static models, posterior inference for a dynamic model is intractable, therefore DTM utilizes wavelet regression and Kalman filters to approximate inference [24].

Blei and Lafferty [24] applied DTM on a large set of science journal documents from 1880 to 2000. They presented the results of both wavelet regression and Kalman filters separately. Both approximation methods could track the variation of target topic frequencies over time, with differences related to smoothing some topic curves and spikes due to dissimilarities in these approximation methods.

In practice, a DTM program divides the complete time span into multiple time points, and the specified number of documents are detected for each time point. During the training process, the program keeps adding a new document to analyze, and finally, the output produces distribution of each topic over the series of time points. Figure 3 shows a DTM as used by the tomotopy [25] topic modeling toolkit, where documents are split into two time slots: the first time slot includes background documents which represent old documents, whereas the second time slot includes foreground documents which represent newer documents. Documents can be divided into multiple number of time slots.

Given that DTM limits its topic discovery to the prespecified number of topics, it keeps tracking these topics until the end of the time series. Therefore, in case a new topic emerges after reaching this prespecified number, it is expected that DTM will not be able to detect this new topic. Increasing the DTM number of topics does not help its ability to detect emerging topics because the algorithm tries to detect same number of topics from each timeslot. We shall demonstrate this limitation later in Section 5 through an experiment.

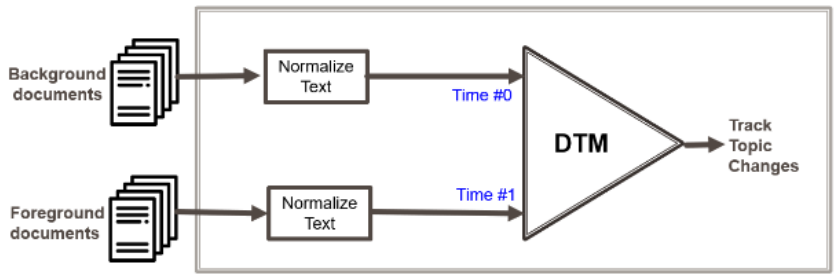

Figure 3. Dynamic topic model (DTM).

3.2. Partially Labeled Dirichlet Allocation

PLDA was introduced to enhance human interpretability of topic models' outputs. PLDA is a semisupervised model, which utilizes available manual topic labels for part of a set of documents. The model scans documents and links discovered topics with their associated label. It also identifies topics that are not represented by any label. Ramage et al. [20] applied PLDA on a large dataset of PhD dissertations, and it could group discovered topics according to the prespecified labels. Moreover, it could also identify those topics which are not represented by any label.

Figure 4 shows a PLDA Model as used by the tomotopy topic modeling toolkit, where labels are provided for the background documents, which represent old documents, then the model would be trained on the unlabeled foreground documents, which represent a newer document set.

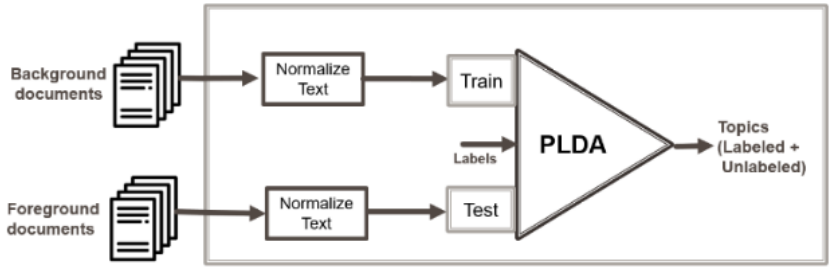

Figure 4. PLDA for background and foreground document.

By using PLDA for solving the emerging topic detection problem, new topics should be detected once the user knows the labels of old topics. This technique looks promising, but when you analyze its mechanism, you realize that discovered unlabeled topics are mainly the global topics which exist in most of the documents.

As a result, it is not expected from PLDA to efficiently detect emerging topics when these are only discussed locally in some documents. This expectation shall be validated later in Section 5 through an experiment using a labeled dataset.

3.3. FB-LDA Topic Model

FB-LDA Model [12] aims to discover emerging topics when two sets of documents are compared. The first set of documents is called the background, whereas the second set is called the foreground. The foreground documents appear during the foreground period in which we are interested in identifying emerging topics that did not appear in the past. The background period is the period which ends just before the start of the foreground period, and its duration is double that of the foreground period. The FB-LDA model detects new topics from the foreground after removing all topics that are discussed in the background.

FB-LDA utilizes a regular LDA model with Gibbs sampling to detect topics from both background and foreground documents. However, after identifying the topics of the background documents, it starts classifying the foreground topics based on their similarity with the background topics. If a foreground topic is not similar to a background topic, it is

identified as an emerging topic. Figure 5 is obtained from [12] and it shows a representation of a FB-LDA topic model.

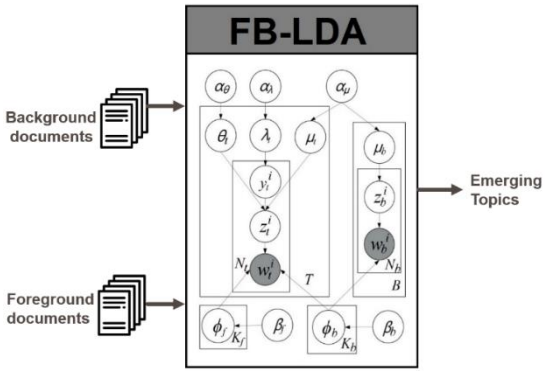

Figure 5. FB-LDA model.

This technique can also be applied to timestamped document sets by slicing the time span into either equal or unequal time slots, then detecting the new topics from any given foreground timeslot when compared to background time slots.

Tan et al. [12] applied FB-LDA to a set of timestamped scientific paper abstracts by splitting them into two groups. The first group represents the recent abstracts which were written during the last 3 years in the dataset, whereas the second group includes the rest of abstracts which were written during the prior 10 years. Then FB-LDA was used to discover the hot scientific topics discussed in the foreground set but not in the background set.

In this paper, we shall repeat the same experiment of Tan et al. [12] to analyze its outcomes. We shall also apply FB-LDA on an additional dataset to verify our findings.

4. Datasets

For our experiments, we shall use two different datasets. The first is the ACM SIGIR dataset [26] which was used by Tan et al. [12]. It consists of 924 scientific paper abstracts about information retrieval separated into two groups as illustrated in Table 1.

Table 1. (a) SIGIR and (b) 10 Newsgroups datasets.

(a)	
ACM SIGIR	**No. of Abstracts**
Background (Years 2000–2009)	**630**
Foreground (Years 2010–2012)	294

(b)				
10 Newsgroups				
Business (100 docs)	**Entertainment** (100 docs)	**Food** (100 docs)	**Graphics** (100 docs)	**Historical** (100 docs)
Technology (100 docs)	**Sport** (100 docs)	**Space** (100 docs)	**Politics** (100 docs)	**Medical** (100 docs)

The first group represents background period from year 2000 to 2009, whereas the second group includes foreground documents from year 2010 to 2012. We manually labeled the timestamp of each abstract by recording its publication year using a simple Google Scholar search for each paper's title. Figure 6 shows the distribution of abstracts from year 2000 to 2012, where background documents are from year 2000 to 2009, and foreground documents are from year 2010 to 2012.

To categorize the topics which are discussed in SIGIR dataset, we read all the 924 abstracts, and we manually labeled the dominant topic of each abstract. As explained earlier, each document consists of multiple topics, and each topic has its own probability

inside that document. The dominant topic for each document is that topic which has the highest probability. We ignored topics that did not appear in more than three abstracts because these do not represent a scientific trend.

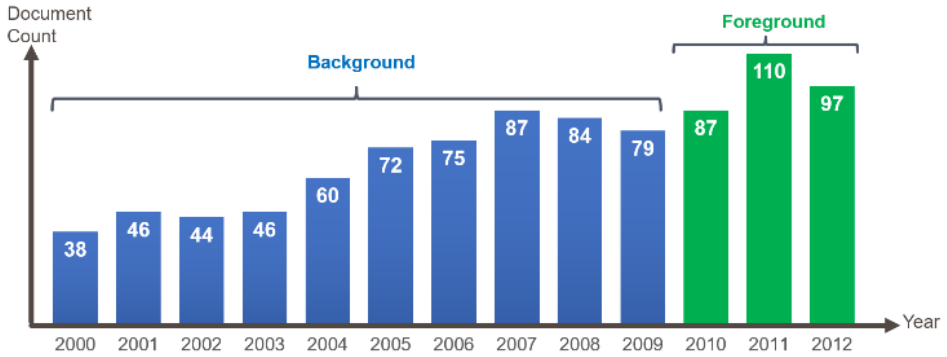

Figure 6. Distribution of ACM SIGIR 2000–2012 documents.

Finally, we grouped all SIGIR abstracts based on their common labels as follows:

(1) Background topics which also appear in foreground:
- User behavior;
- Question answering;
- Search engines queries;
- Relevance feedback;
- Recommendation items;
- Average precision evaluation metrics;
- Learning to rank;
- Image retrieval;
- Web search pages.

(2) Background topics which do not appear in foreground:
- Topic detection and tracking event;
- Document clustering;
- Language translation models.

(3) Emerging topics which mainly appear in foreground:
- Search result diversification;
- Feature space hashing;
- Social network twitter;
- Search result cache invalidation;
- Temporal indexing.

Given that—in practice—an emerging topic could appear in one or two papers during background period, we kept such topics in the emerging topics list if they appear at least in three documents during the foreground period. For example, we consider temporal indexing as an emerging topic because it appears in three abstracts during the foreground period, although it also appears once in the background period.

The second dataset is the 10 Newsgroups dataset [27] which includes 1000 classified news messages on 10 different topics as shown in Table 1.

The second dataset is used for examining the ability of base methods to detect emerging topics by manually splitting the 10 Newsgroups documents into two sets. The first set represents background documents, and the second set represents foreground documents. We randomly selected documents from two topics that represent emerging topics and inserted these documents in the foreground group only. Multiple combinations of emerging topics are used to verify our findings. For example, Table 2 shows the split of background and foreground documents, wherein we included historical and medical documents only in the foreground.

Table 2. Background and foreground of 10 Newsgroups dataset.

10 Newsgroups' Background				
Business (50 docs)	**Entertainment** (50 docs)	**Food** (50 docs)	**Graphics** (50 docs)	**Historical** (0 docs)
Technology (50 docs)	**Sport** (50 docs)	**Space** (50 docs)	**Politics** (50 docs)	**Medical** (0 docs)
10 Newsgroups' Foreground				
Business (50 docs)	**Entertainment** (50 docs)	**Food** (50 docs)	**Graphics** (50 docs)	**Historical** (50 docs)
Technology (50 docs)	**Sport** (50 docs)	**Space** (50 docs)	**Politics** (50 docs)	**Medical** (50 docs)

5. Experiment Using Base Methods

We used the datasets of Section 4 to perform experiments using Python for analyzing outputs of the three base methods: DTM, PLDA, and FB-LDA.

We used a Dell Inspiron 7370 laptop with Intel$^{(R)}$ Core$^{(TM)}$ i7-8550U CPU @1.99 GHz Processor, Installed RAM is 16.0 GB, Windows 10 Home version 20H2 64-bit operating system. Python version 3.8.3 is used with Jupyter Notebook server version 6.1.4.

We applied DTM on SIGIR dataset by splitting it into two timepoints: Time #0, which includes all scientific abstracts from year 2000 to 2009, and Time #1, which includes all abstracts from year 2010.

Before applying the topic modeling algorithms, we used Genism library to carry out simple text normalization and data preprocessing steps which include (1) stripping html tags, (2) removing accent characters, (3) expanding contractions, (4) removing special characters, (5) removing digits, (6) removing extra new lines, (7) removing extra white space, (8) removing stop words, (9) lowercasing each word (10) tokenization, wherein sentences and words are extracted from each document, (11) lemmatizing words by converting all verbs to present tense and grouping different forms of a noun to a common form, and (12) stemming which reduces each word to its root.

To apply DTM and PLDA, we used Python wrapper for tomotopy, which is a topic modeling tool written in C++ based on Gibbs sampling. For FB-LDA, we used the Python Gibbs sampling implementation program [28], which is provided by the authors of [12].

5.1. Dynamic Topic Model

We applied DTM on SIGIR dataset by splitting it into two timepoints: Time #0, which includes all scientific abstracts from year 2000 to 2009, and Time #1, which includes all abstracts from year 2010 to 2012.

We also applied DTM to the 10 Newsgroups dataset by splitting it into two timepoints: Time #0, which includes all background documents of Table 2, and Time #1, which includes all foreground documents of same table. We selected (K = 10) as we are aware that the number of main topics in the dataset is 10. However, as shown in Table 3, DTM also failed to detect any of the two emerging topic, the historical and medical topics. A similar result was obtained when we doubled number of topics to become (K = 20). These two experiments on DTM confirm our expectations that DTM cannot detect emerging topics when these do not appear in the background time slot.

The number (K = 17) is selected as we are aware that the number of main topics in the dataset is 17 based on our manual labeling process. Our program ignored topics which do not appear in more than three documents. However, as shown in Table 4, DTM failed to detect any of the new topics which are listed above in the five emerging topics which mainly appear in foreground. A similar result was obtained when number of topics was doubled to become (K = 34).

Table 3. DTM results for 10 Newsgroups dataset.

Topic No.	Time Slot #0	Time Slot #1
Topic # 0	t = 0: company service look want start	t = 1: back user way company day
Topic # 1	t = 0: minute need company European want	t = 1: launch party day back army
Topic # 2	t = 0: next call help give firm	t = 1: help day number com need
Topic # 3	t = 0: much part face firm look	t = 1: force run second want great
Topic # 4	t = 0: cup bit way call part	t = 1: power give second low three
Topic # 5	t = 0: cup call mobile need write	t = 1: day service battle party three
Topic # 6	t = 0: minute big business day program	t = 1: next base food launch list
Topic # 7	t = 0: minute three user country problem	t = 1: way general party plan since
Topic # 8	t = 0: uk need place tool number	t = 1: great want information need
Topic # 9	t = 0: uk next day market need	t = 1: food way give help ally
Topic # 10	t = 0: give tell run minute much	t = 1: party back call early army
Topic # 11	t = 0: large uk service help business	t = 1: graphic com army general next
Topic # 12	t = 0: second report help minister give	t = 1: com food nasa want base
Topic # 13	t = 0: uk information service program report	t = 1: great write number food information
Topic # 14	t = 0: expect call issue tell end	t = 1: look way help force party
Topic # 15	t = 0: three cup cut look part	t = 1: give information party cost look
Topic # 16	t = 0: report test cup need end	t = 1: food army give general run
Topic # 17	t = 0: give week call need great	t = 1: still day plan write force
Topic # 18	t = 0: give country look back tell	t = 1: write give much second day
Topic # 19	t = 0: next service company user run	t = 1: plan information force great company

Table 4. DTM results for the SIGIR 2000–2012 dataset.

Topic No.	Time Slot #0	Time Slot #1
Topic # 0	t = 0: new system text algorithm such	t = 1: two users framework proposed This
Topic # 1	t = 0: new This these learning used	t = 1: such learning different propose users
Topic # 2	t = 0: new not it between language	t = 1: proposed propose more large present
Topic # 3	t = 0: more This system each these	t = 1: different This Web problem more
Topic # 4	t = 0: This Web new propose has	t = 1: users propose Web it two
Topic # 5	t = 0: Web Our new learning different	t = 1: different propose evaluation not task
Topic # 6	t = 0: text such has users more	t = 1: users different two evaluation other
Topic # 7	t = 0: two propose such text users	t = 1: propose users such different it
Topic # 8	t = 0: not text this system more	t = 1: users Web more it over
Topic # 9	t = 0: use text systems new has	t = 1: has proposed two text problem
Topic # 10	t = 0: This not different these it	t = 1: users different propose proposed effectiveness
Topic # 11	t = 0: problem terms propose has use	t = 1: problem different their has it
Topic # 12	t = 0: propose systems This proposed evaluation	t = 1: has propose proposed users it
Topic # 13	t = 0: This more such system has	t = 1: users more these This between
Topic # 14	t = 0: different new system Web evaluation	t = 1: users proposed propose different these
Topic # 15	t = 0: two This system different such	t = 1: propose more different such proposed
Topic # 16	t = 0: Web text has new different	t = 1: users such different into proposed

5.2. Partially Labeled Dirichlet Allocation

The methodology of this experiment is to provide the labels of background dataset, hoping that PLDA will be able to detect unlabeled topics in the foreground dataset. We provided the labels of background documents to PLDA, then we applied it on the foreground dataset, which includes emerging topics in addition to background topics.

For the 10 Newsgroups dataset, we selected (K = 10) for the number of PLDA topics as we are aware that the number of main topics in the dataset is 10. However, as shown in Table 5, PLDA failed to detect any emerging topic as none of the new unlabeled topics is related to historical or medical topics.

Table 5. PLDA results for 10 Newsgroups dataset.

Labeled Topic	Unlabeled New Topic
Label business: turn economy put must foreign	New Topic 0: hope decision charge leave open
Label entertainment: article champion athens money together	New Topic 1: trial green possible product even
Label food: list salt turn let process	New Topic 2: cross concern begin combine lot
Label graphics: salt fail satellite hour ask	New Topic 3: hour list cover general official
Label politics: athens hit blair global ask	New Topic 4: share google event little future
Label space: astronaut leave police indoor recently	New Topic 5: google four risk almost return
Label sport: sport meet thank major cook	New Topic 6: athens hope live serve third
Label technology: speed economy google comedy ban	New Topic 7: chicken far google hand support
	New Topic 8: move beat comment video salt
	New Topic 9: continue miss list something concern

Although PLDA is good for tracking all topics which are detected during the training process, its detection of unlabeled topics failed to discover emerging topics as it could only detect new subtopics that are related to labeled topics.

5.3. FB-LDA Topic Model

We applied FB-LDA on the 10 Newsgroups dataset and selected (K = 10) as the number of topics. As clear from Table 6, the model could detect emerging topics successfully, however the keywords of 60% of detected topics do not belong to emerging historical and medical topics.

Table 6. FB-LDA (K = 10) results for 10 Newsgroups dataset.

No.	FB-LDA Topic Keywords
1	power just city ac iphone point green view john announced
2	medical article medicine writes know disease health effect dont case
3	data available ftp image graphics email contact file anonymous package
4	image van polygon het editing line een xv pat vote
5	european time good best mark place jump long took form
6	soviet hitler japan ii japanese moscow union general iphone japans
7	war germany august world battle france great soviet russian north
8	food foods people study blood chocolate risk like help levels
9	united russia states july austriahungary 1939 june land 28 verdun
10	film best festival ancient awards oscar films award box wheat

For example, the keywords of the first topic belong to the technology topic, whereas the keywords of the third topic belong to the graphics topic. Furthermore, we noticed that the word “phone” from the background technology topic appeared in the emerging historical topic. When the number of topics was reduced to 3, the model could detect only one emerging topic as shown in Table 7.

Table 7. FB-LDA (K = 3) results for 10 Newsgroups dataset.

No.	FB-LDA Topic Keywords
1	war german british germans germany army august french battle military
2	food like dont image people just writes article help study
3	data available ftp graphics email image film line program information

In our experiment, FB-LDA is also applied on SIGIR dataset by selecting (k = 10) as the number of topics, and we could obtain similar results to the ones published by Tan et al. in [12] as shown in Table 8. However, it is noticed that 50% of detected topics are not emerging topics. As a result, we conclude that FB-LDA can detect some of the emerging topics when number of topics is selected correctly, though the authors of [12] did not provide clear guidelines for selecting the setpoint of number of topics. However, the output of FB-LDA showed some background topics most of the time, and in a few cases, it also showed a background word along with the top keywords of an emerging topic.

Table 8. FB-LDA results for SIGIR dataset.

Topic No.	Research Topic	FB-LDA Top Words
1	Exploiting users' behavior data	behavior search model user click log session data
2	Probabilistic factor models for recommendation	user recommandation person interest facet factor latent
3	Search result diversification	result search vertical diverse diversify subtopic show
4	Query suggestions	query search suggest engine log reformulation predictor
5	Quality of user-generated content	label quality book crowdsource select flaw impact sample
6	Twitter stream mining	stream twitter context tweet entity toponym context-aware
7	Image search and annotation	image visual attribute estimate face privacy flickr facial
8	Search result cache invalidation	time result temporal cache evaluate update invalidate
9	Temporal indexing	collect index time web structure temporal archive time-travel
10	Hashing for scalable image retrieval	retrieval hash example code method propose application

6. Filtered-LDA Model

Our experiments of Section 5 revealed that both DTM and PLDA could not detect emerging topics from our two datasets efficiently, whereas FB-LDA could do the job with some limitations. In this section, we introduce our novel model to overcome limitations of base methods.

We called the new model "Filtered-LDA" as it directs its topic modeling components to filter out old topics and keep new topics only. Figure 7 presents the proposed Filtered-LDA model, which attempts to emulate human approach for discovering new topics from a large set of documents. A human would first skim through all documents to isolate the ones which he/she thinks they contain new topics. Then he/she would only go through the isolated set of documents to identify high-frequency emerging topics. With this simple technique, the impact of isolation/clustering errors would be insignificant because the wrongly classified documents would not be among the high-frequency emerging topics. Nevertheless, this novel model introduces additional measures to ensure high performance of emerging topic detection by genuinely reducing the chance of detecting old topics.

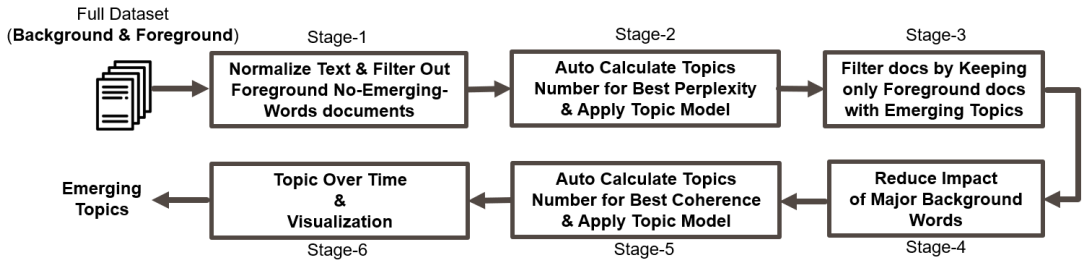

Figure 7. Filtered-LDA Model.

The first stage of the model aims to reduce number of background topics from the foreground documents to minimum by ignoring all foreground documents that do not contain new words when compared to background documents. In practice, emerging topics could appear once or twice in the background document set, but they do not appear as a trend in the background duration. The threshold of the number of documents that represent a trend is a variable that shall be defined by the user. For instance, if a user is looking for a completely new topic that never appeared in the background, then the threshold setpoint is 0. However, if a user accepts emerging topics even if they appear once or twice in the background then the threshold setpoint is 2. In our experiments, we selected multiple options of hot trend threshold between 0 to 10, and we could achieve consistent results with accuracy above 90%.

The first stage of the model also ignores words that have low frequency in the background documents. This aims to reduce possibility of mixing foreground emerging topics with background topics due to presence of some low frequency words in the background dataset. The second stage uses topic modeling for clustering topics in the complete dataset. To select optimum number of topics for the topic model, usually number of topics that gives highest coherence score is selected. As explained in [29], there are multiple methods for measuring the topic's coherence score, and the C_V measurement showed the best performance. Therefore, in our experiments we select the C_V measurement which uses a sliding window, normalized pointwise mutual information (NPMI), and the cosine similarity. However, we noticed that highest coherence score does not guarantee best representation of dataset. Figure 8 shows an example in which topic models are used to analyze our SIGIR dataset. We know from our manual review of the dataset that the minimum number of topics is 17 as explained earlier in Section 4. However, regular topic models that use bag-of-words (BoW) text representation showed highest coherence scores for topic numbers that are much lower than 17. For instance, Gensim LDA showed the highest coherence score when (K = 8), which means that remaining nine topics would not be captured. Hence, at this stage, we shall select the number of topics "K" that makes our model more fit for the test dataset or the held-out data without caring much about the human interpretability of the model because this output is not the one which will be finally presented to the user. Therefore, this stage focuses on the perplexity score of the topic model, rather than focusing on the coherence score. There are multiple ways for carrying out Stage-2 task efficiently. We shall propose two different options of frameworks that can ensure good clustering performance for our model.

The third stage identifies all topics that do not have documents in the background dataset and topics that form a trend in foreground dataset but do not form a trend in background dataset. As explained earlier, the number of documents that represents a trend is a variable that the user shall select in advance. For our experiment, a trend or a hot topic shall be discussed at least in three documents. As a result, the output of stage 3 is all foreground documents discussing topics that do not represent major background topics. To ensure that keywords of major background topics do not appear in the emerging topics keywords, the fourth stage filters out high frequency background words from the obtained foreground documents of third stage.

The fifth stage automatically selects the optimum number of topics "K" for the filtered foreground documents by calculating the coherence scores for a wide range of "K" values so that the output of the topic model at the sixth stage can be easily interpreted by the user.

The final stage presents detected emerging topics in multiple forms including topic-over-time curves, topic keyword list, topic keyword Wordcloud, visual representation of topic probabilities, visual representation of topic similarities and overlap, and finding the most representative document for each topic. Automatic topic labeling [30] is also used at this stage to select best few keywords that represent the topic. This ensures better interpretation of discovered topics and that the user can monitor evolution of topics over time.

6.1. First Framework

Low perplexity score for a model ensures that it represents the held-out data well [31], hence "K" that gives a low LDA perplexity score shall be selected for Stage 2 of the model. However, to reduce the computational power requirements for this step, an alternative method is chosen by using the hierarchical Dirichlet processes (HDP) as shown in Figure 9 which presents the first framework. HDP has low perplexity scores regardless of the number of topics [32] when compared to LDA as shown in Figure 10. Moreover, HDP also provides good Coherence Scores when compared to other models as shown in the example of Figure 11 in which we compared various types of topic models. It is noticed that HDP scores are higher than models that use BoW text representation.

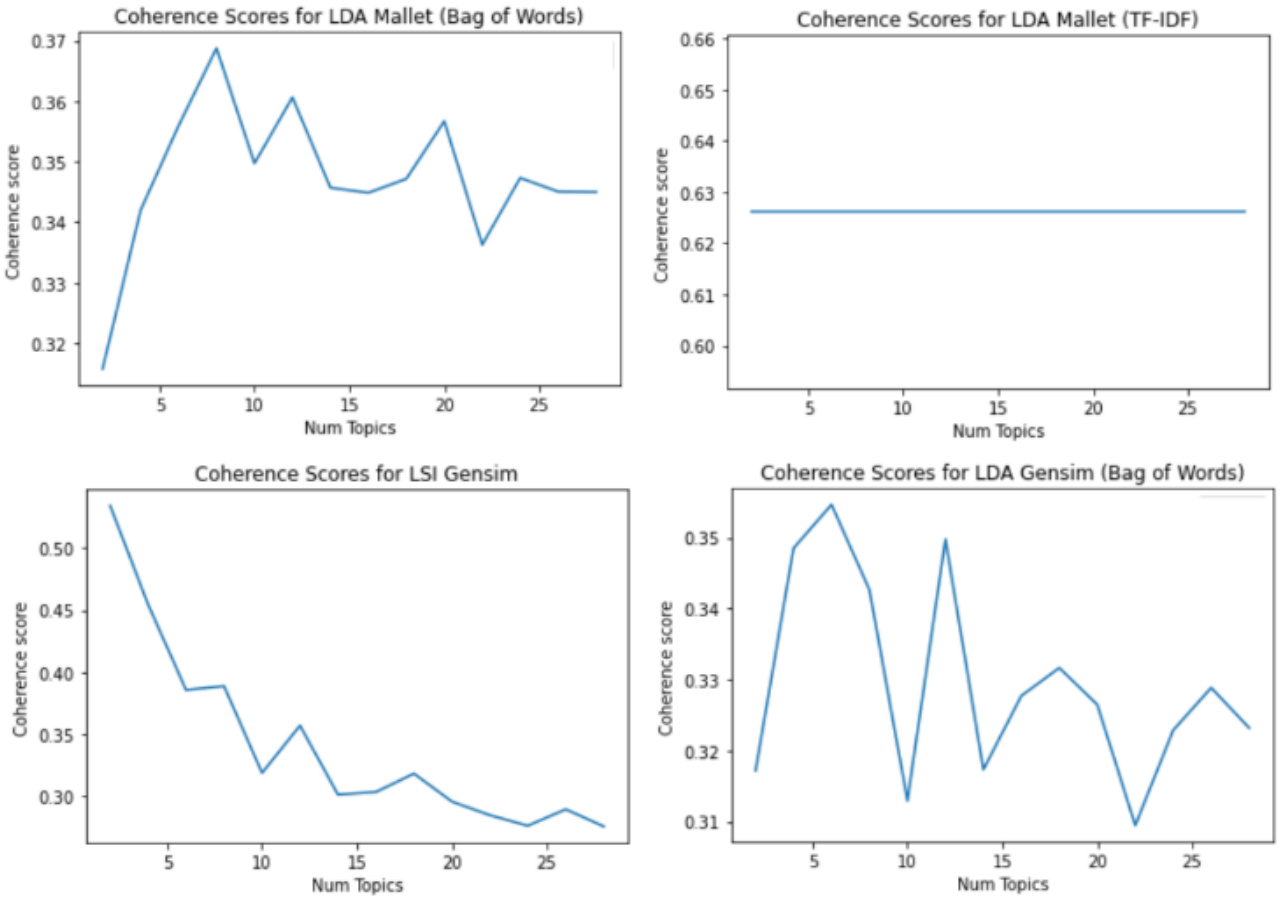

Figure 8. Topic models' coherence scores curves for the SIGIR dataset.

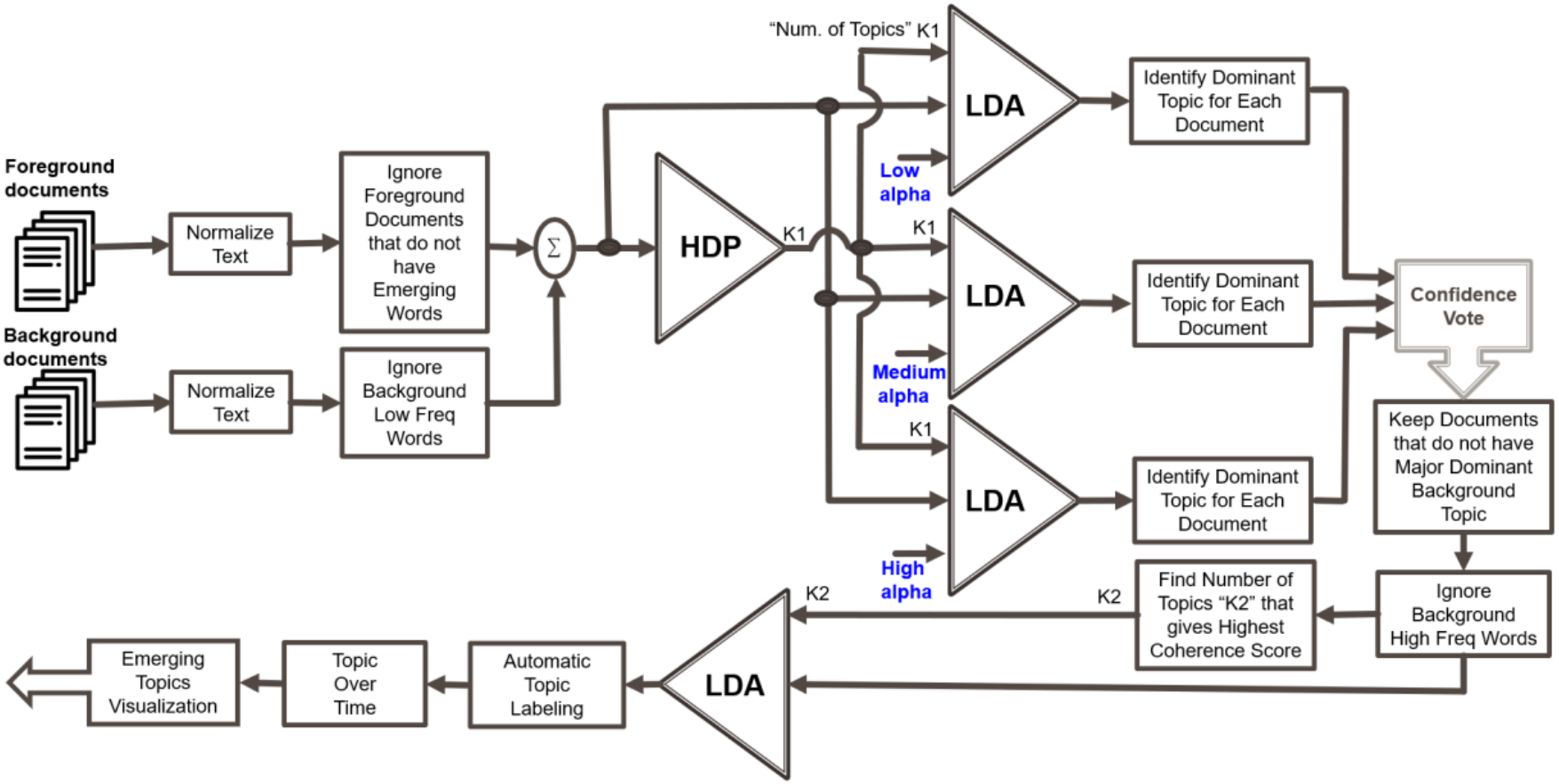

Figure 9. Emerging topic detection filtered-LDA framework #1.

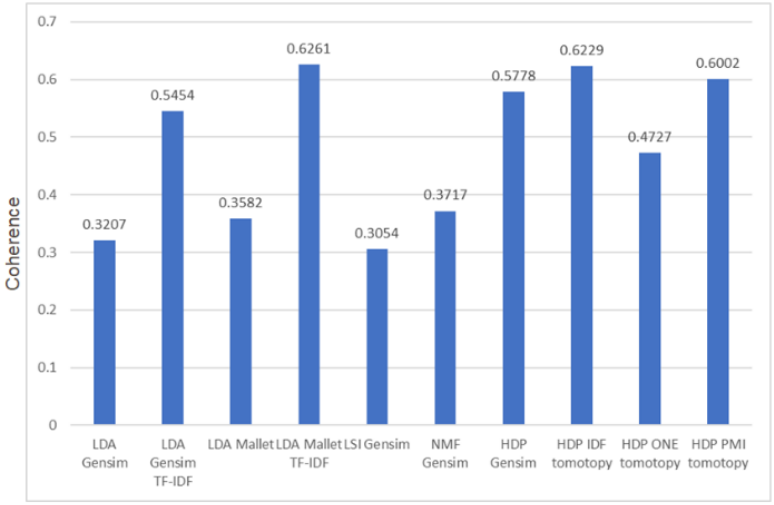

Figure 10. Comparison of LDA and HDP perplexity [32].

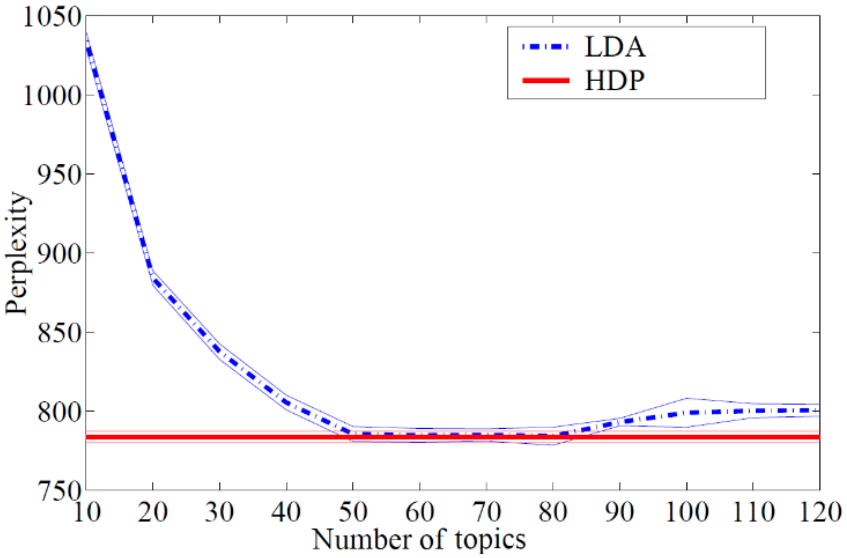

Figure 11. Topic models' coherence scores for SIGIR dataset at K = 20.

Employing HDP for the second stage also solves the problem of manual selection of the number of topics "K" as HDP automatically calculates optimum number of topics for the document set. However, the precision of the HDP model depends on the selected term weighting scheme (TM) [33], therefore a comparison between these schemes shall be carried out for the dataset, and the TM that provides highest coherence score shall be automatically selected.

We selected the LDA Model for the actual clustering work because we could achieve slightly higher accuracy using LDA when compared to HDP results. Similar findings were confirmed earlier by Limsettho et al. [34].

Cascaded LDA models are used to look at the dataset from multiple distances by choosing different value of Alpha hyperparameter for each LDA block so that emerging topics would not be merged by mistake with other old topics due to wrong smoothing factor. We used in Figure 9 three cascaded LDA models, one with low alpha ($\alpha = 1$), a second with medium alpha ($\alpha = 50$), and a third with high alpha ($\alpha = 100$). These settings are used with Gensim wrapper for Mallet toolkit. Of course, cascaded LDA block can have additional LDA models to cover wider range of alpha, including very high settings. In case program speed is not important for an application, we recommend using high number of

cascaded LDA that covers wide range of alpha settings to enhance the zooming effect of the model.

A simple confidence voting process follows the output of cascaded LDA to select those documents that appeared multiple times. To illustrate, if a document is identified as an emerging topic's document at the output of LDA Models of both low alpha and medium alpha, it shall be considered as an emerging topic's document.

Afterwards, high frequency background words are removed from all selected documents to ensure that final emerging topics do not use common background keywords.

The next stage includes a standard LDA model with optimum coherence score as described earlier in Stage 6 of the model.

Finally, topic over time is drawn using LDA calculated probabilities for topics during each time slot. Each document is automatically labeled with a single topic based on its dominant topic which has the highest probability in that document. Then each topic is tracked over time based on the number of documents where it has highest probability.

To implement the framework, Python is used along with multiple packages including Gensim [35] for topic modeling, and pyLDAvis [36] for topic visualization. Given that the current Gensim version supports only variational Bayes sampling for LDA, we used Python wrapper for Mallet [37] toolkit to implement LDA with Gibbs sampling because it showed better results as illustrated in Figure 10. We also used Python wrapper for tomotopy toolkit to implement HDP and automatic topic labeling.

Figure 12 shows some visualization formats that can be obtained through Python functions and codes, like the pyLDAvis which was introduced for the first time in year 2015. To test the framework, experiments are carried out on both the SIGIR and 10 Newsgroups datasets. For the HDP model in stage 2, the term weighting scheme that provides highest coherence score for the model is automatically selected to ensure good precision. For example, in Figure 11, the inverse document frequency term weighting TM IDF showed highest coherence score for the SIGIR dataset when compared to the pointwise mutual information term weighting TM PMI and the TM ONE which considers every term equal.

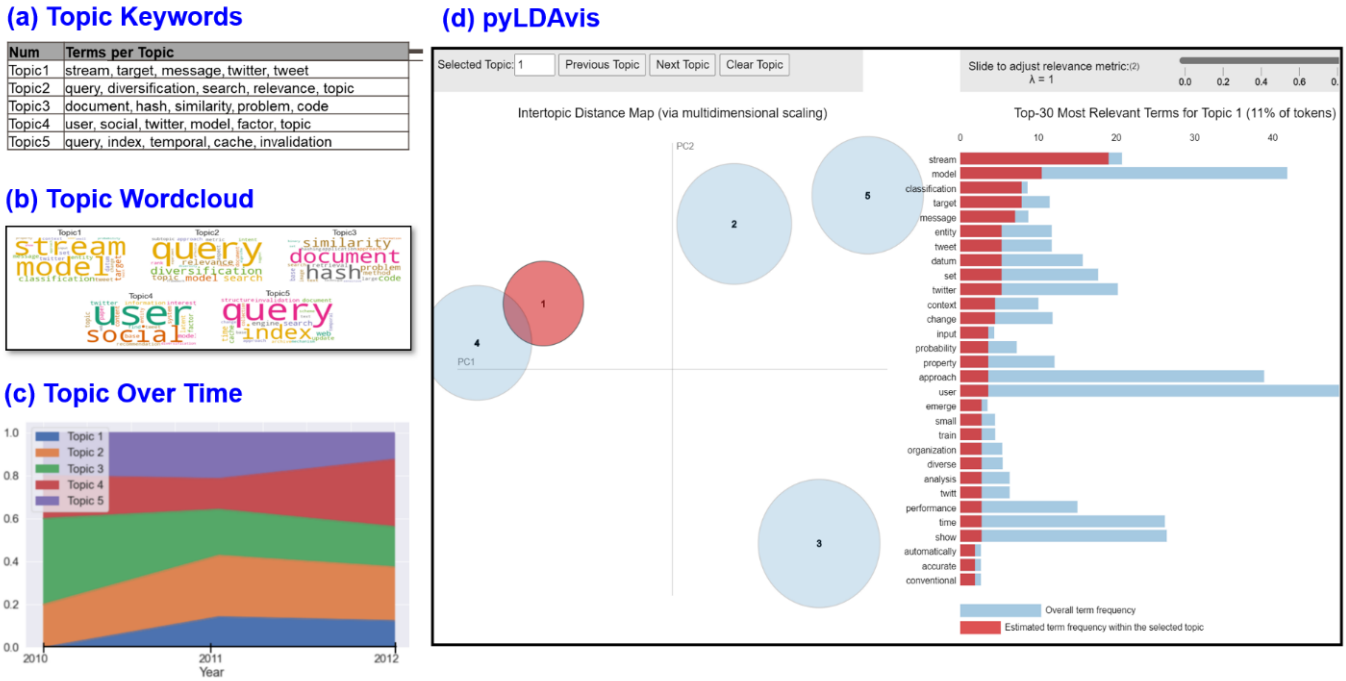

Figure 12. Sample topic visualization for SIGIR dataset at (K = 5).

The automatic number of topics "K" for HDP at stage 2 for the SIGIR and 10 Newsgroups datasets were 100 and 92 respectively, whereas the number of LDA topics "K" at Stage 5 were 5 and 2 respectively. Table 9 shows the emerging topics of both datasets.

Table 9. Framework #1 emerging topics for (a) SIGIR and (b) 10 Newsgroups datasets.

(**a**)			
Topic No.	**Topic Keywords**	**Title of Most Representative Abstract**	**Year of Most Representative Abstract**
1	diversification, topic, improve, search, feedback	Combining implicit and explicit topic representations for result diversification	2012
2	large, feature, image, similarity, hashing	Integrating hierarchical feature selection and classifier training for multi-label image annotation	2011
3	engine, index, cache, framework, invalidation	Online result cache invalidation for real-time web search	2012
4	social, twitter, stream, entity, temporal	TwiNER: named entity recognition in targeted twitter stream	2012
5	Index, temporal, search, queries, collections	Temporal index sharding for space-time efficiency in archive search	2011
(**b**)			
Topic No.	**Topic Keywords**	**Most Representative Document begins with:**	
1	Disease, medicine, alternative, head, drug	New England Medical Journal in 1984 ran the heading: "Ninety Percent of Diseases are not Treatable by Drugs...	
2	austria, war, german, ally, front	World War I, also called First World War or Great War, an international conflict that in 1914–1918 embroiled most...	

All five detected emerging topics for SIGIR dataset did not appear in the background documents. Furthermore, the detected two emerging topics for the 10 Newsgroups dataset represent both historical and medical emerging topic trends.

To verify our results, we repeated our experiment using different selections of emerging topics for 10 Newsgroups dataset with different number of documents varies between 25 to 100 documents, and we could achieve similar results as all detected topics were always emerging topics, however the number of emerging topics varied each time because of inclusion of subtopics that belong to emerging topics.

By repeating our experiment for SIGIR dataset, we noticed that the temporal indexing topic is sometimes replaced by another emerging topic that do not form a trend in the foreground dataset. Given that there are only three documents in the foreground dataset that discuss this topic, and that some of its keywords are similar to other indexing topics in the background, the model was merging it with background topics during some of LDA runs. We could avoid this problem by reducing the threshold of a trending topic from three to two documents, therefore the user should consider reducing the desired threshold by one or two documents when setting the hot trend's variable.

6.2. Second Framework

As shown in Figure 13, instead of representing input documents by bag of words, the second framework uses term frequency inverse document frequency (TF-IDF) representation of texts as an input for the topic model in the second stage. Blei and Lafferty [38] indicated that using TF-IDF for LDA may enhance the speed of the model because it reflects in advance how significant a word is to a document in the dataset. Figure 11 shows that coherence scores for topic models with TF-IDF are generally higher than other models that use BoW representation.

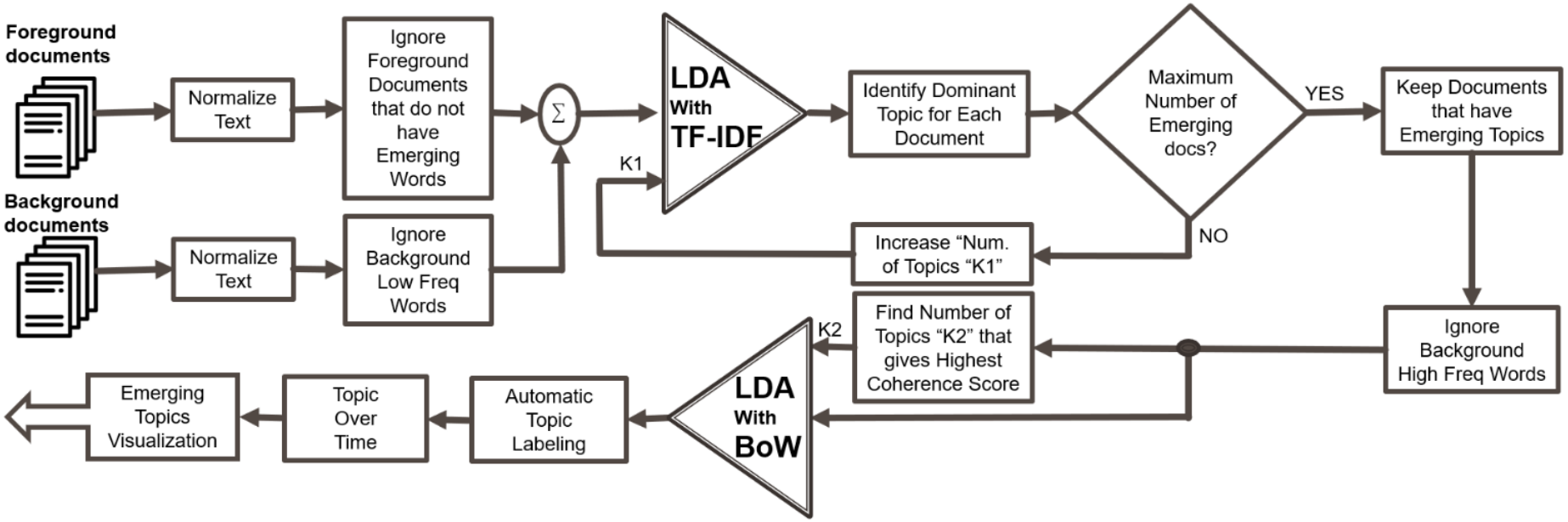

Figure 13. Emerging topic detection filtered-LDA Framework #2.

To select the number of documents "K" for our TF-IDF LDA model, perplexity and coherence scores are monitored while automatically increasing the model's number of topics. At each number of topics, the number of emerging topics that only appear in foreground documents is checked. Finally, the system automatically selects the number of topics that produces maximum number of documents that contain emerging topics because this represents the state in which the model performs best clustering. In other words, it is the state when the model represents the test data well. It is also noticed that the model has a good perplexity score at this setpoint.

For the 10 Newsgroups dataset, Figure 14 shows the curves of Perplexity scores, Coherence Scores, and the number of emerging topics against the setpoint of the number of topics "K". The optimum number of topics that provides maximum number of documents with emerging topics is 99, whereas the number for SIGIR dataset increased to 100 topics.

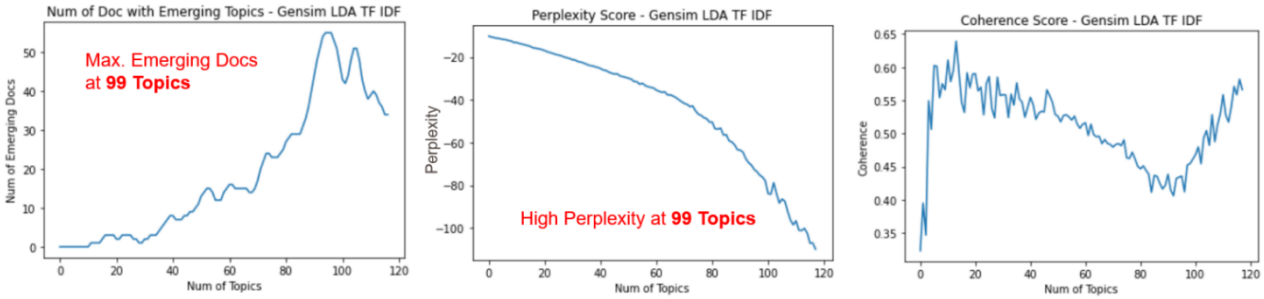

Figure 14. 10 Newsgroups perplexity and coherence scores for LDA with TF-IDF.

As shown in Table 10, the output of second Framework for SIGIR dataset include five emerging topics. However the temporal indexing topic was not detected as it was merged with a background topic. Instead of the temporal indexing topic, a fifth emerging topic was detected although it did not represent a trend. For the 10 Newsgroups dataset, the framework could detect both emerging topics successfully without adding any background topic, however the historical topic appears twice in the output.

Table 10. Framework #2 emerging topics for (a) SIGIR and (b) 10 Newsgroups datasets.

(**a**)			
Topic No.	**Topic Keywords**	**Title of Most Representative Abstract**	**Year of Most Representative Abstract**
1	code, hashing, binary, bit, teach	Self-taught hashing for fast similarity search	2010
2	clickthrough, attribute, position, conceptual, photo	Where is who: large-scale photo retrieval by facial attributes and canvas layout	2012
3	stream, tweet, twitter, replication, allocation	TwiNER: named entity recognition in targeted twitter stream	2012
4	diversification, redundancy, explicit, formal, diversify	Explicit relevance models in intent-oriented information retrieval diversification	2012
5	cache, invalidation, update, stale, cached	Caching search engine results over incremental indices	2010
(**b**)			
Topic No.	**Topic Keywords**	**Representative Abstract begins with:**	
1	german, army, force, ally, division	The final offensive on the Western Front It was eventually agreed among the Allied commanders . . .	
2	war, hitler, soviet, world, say	World War I, also called First World War or Great War, an international conflict that in 1914–1918 . . .	
3	disease, alternative, medical, edu, compartment	poster for being treated by a licensed physician for a disease that did not exist. Calling this physician . . .	

7. Conclusions

Our two frameworks could achieve better results when compared to base methods. They did not only detect most of emerging topics successfully from both the SIGIR and 10 Newsgroups datasets, but they could also block old trends and major background topics.

The first framework can zoom into the text by varying the value of alpha hyperparameter to detect trending topics, even when they appear only in few documents. Such low-frequency topics could be merged with other background topics when a fixed value of alpha is selected, or a low number of topics is specified.

The second framework uses LDA with TF-IDF text representation, which showed higher coherence and lower perplexity scores when compared to standard LDA models with BoW text representation. As a result, good topic clustering could be achieved without the need of applying cascaded LDA blocks and varying the value of alpha hyperparameter. However, the computational cost of the second framework is still higher than the first one because of the automatic search for maximum number of documents that contain a dominant emerging topic. Furthermore, unlike the first framework, the second one failed to capture the temporal indexing topic for the SIGIR dataset. We believe this is the result of using a single value of alpha hyperparameter in the second framework, which make it possible to merge a new low-frequency topic with background topics when they share some of the topic's keywords. This gives an advantage to the first framework, which uses multiple values of alpha hyperparameter. Table 11 summarizes the results of the three base models and our two new frameworks.

Table 11. Summary of experimental results.

MODEL	Emerging Topic Detection for SIGIR Dataset	Emerging Topic Detection for 10 Newsgroups Dataset
DTM	Not detected	Not detected
PLDA	Not applicable (labels are not available for this dataset)	Not detected
FB-LDA	50% of detected topics are not emerging topics	66% of detected topics are not emerging topics
Filtered-LDA (1st Framework)	100% accuracy, all emerging topics are detected; no old topic is presented.	100% accuracy, all emerging topics are detected; no old topic is presented.
Filtered-LDA (2nd Framework)	80% accuracy, 4 out of 5 emerging topics are detected; no old topic is presented.	100% accuracy, all emerging topics are detected; no old topic is presented.

The proposed emerging topic detection model is general and can be applied to other applications where two sets of documents are compared to discover new topics. In our future work, we shall use the new model to interpret sentiment variations in Twitter by comparing foreground and background tweets.

Author Contributions: Supervision, K.S.; Writing—original draft, F.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Deerwester, S.; Dumais, S.; Furnas, G.; Landauer, T.; Harshman, R. Indexing by latent semantic analysis. *J. Am. Soc. Inform. Sci.* **1990**, *41*, 391–407. [CrossRef]
2. Lee, D.; Seung, H. Learning the parts of objects by non-negative matrix factorization. *Nature* **1999**, *401*, 788–791. [CrossRef] [PubMed]

3. Hofmann, T. Probabilistic latent semantic analysis. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Berkeley, CA, USA, 15–19 August 1999.
4. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
5. Alattar, F.; Shaalan, K. A Survey on Opinion Reason Mining and Interpreting Sentiment Variations. *IEEE Access* **2021**, *9*, 39636–39655. [CrossRef]
6. Lee, J.; Kang, J.; Jun, S.; Lim, H.; Jang, D.; Park, S. Ensemble Modeling for Sustainable Technology Transfer. *Sustain. J.* **2018**, *10*, 2278. [CrossRef]
7. Hansen, J. Inside Latent Dirichlet Allocation: An Empirical Exploration. *Knowl. Inf. Syst.* **2016**, *3*, 1–21.
8. Erten, C.; Harding, P.; Kobourov, S.; Wampler, K.; Yee, G. Exploring the computing literature using temporal graph visualization. *SPIE Int. Soc. Opt. Eng.* **2003**, *5295*, 525945–525956.
9. Le, M.; Ho, T.; Nakamori, Y. Detecting Emerging Trends from Scientific Corpora. *Int. J. Knowl. Syst. Sci.* **2005**, *2*, 53–59.
10. Rabiner, L. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. IEEE* **1989**, *77*, 257–286. [CrossRef]
11. McCallum, D.F.; Pereira, F. Maximum entropy Markov models for information extraction. In Proceedings of the 17th International Conference on Machine Learning, San Francisco, CA, USA, 29 June–2 July 2000.
12. Tan, S.; Li, Y.; Sun, H.; Guan, Z.; Yan, X.; Bu, J.; Chen, C.; He, X. Interpreting the Public Sentiment Variations on Twitter. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 1158–1170.
13. Shen, X.; Wang, L. Topic Evolution and Emerging Topic Analysis Based on Open Source Software. *J. Data Inf. Sci.* **2020**, *5*, 126–136. [CrossRef]
14. Swan, R.; Jensen, D. *TimeMines: Constructing Timelines with Statistical Models of Word Usage*; University of Massachusetts: Amherst, MA, USA, 2000; pp. 73–80.
15. Havre, S.; Hetzler, E.; Whitney, P.; Nowell, L. Theme River: Visualizing thematic changes in large document collections. *IEEE Trans. Vis. Comput. Graph.* **2002**, *8*, 9–20. [CrossRef]
16. Eck, N.J.; Waltman, L. Software survey: VOSviewer, a computer program for bibliometric mapping. *Scientometrics* **2010**, *84*, 523–538. [PubMed]
17. Vorontsov, K. *BigARTM Documentation Release 1. 0*; Moscow Institute of Physics and Technology: Moscow, Russia, 2020.
18. Fedoriaka, D.S. *Hierarchic Topic Models Visualization*; MIPT: Moscow, Russia, 2016.
19. Bolelli, L.; Ertekin, S.; Giles, C. Topic and Trend Detection in Text Collections Using Latent Dirichlet Allocation. In Proceedings of the ECIR2009: 31st European Conference on Information Retrieva, Oulouse, France, 6–9 April 2009.
20. Ramage, D.; Manning, C.; Dumais, S. Partially labeled topic models for interpretable text mining. In Proceedings of the International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD), Singapore, 14–18 August 2011.
21. Morinaga, S.; Yamanishi, K. Tracking dynamics of topic trends using a finite mixture model. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, 22–25 August 2004.
22. Behpour, S.; Mohammadi, M.m.; Albert, M.V.; Alam, Z.S.; Wangc, L.; Xiao, T. Automatic trend detection: Time-biased document clustering. *Knowl.-Based Syst.* **2021**, *220*, 106907. [CrossRef]
23. Marrone, M. Application of entity linking to identify research fronts and trends. *Scientometrics* **2020**, *122*, 357–379. [CrossRef]
24. Blei, D.; Lafferty, J. Dynamic Topic Models. In Proceedings of the 23rd International Conference on Machine Learning (ICML 2006), Pittsburgh, PA, USA, 25–29 June 2006.
25. Tomotopy Documentation, Ver. 0.10.2. Available online: https://pypi.org/project/tomotopy (accessed on 28 February 2021).
26. Tan, S. SIGIR Dataset from ACM Digital Library, 2000–2012. Available online: https://github.com/laos1984/FB-LDA/blob/master/sigir_data.zi (accessed on 20 February 2021).
27. Lang, K. 10Newsgroups Dataset. 2008. Available online: https://www.kaggle.com/jensenbaxter/10dataset-text-document-classification (accessed on 21 February 2021).
28. Tan, S. FB-LDA Python Implementation of the Collapsed Gibbs Sampler for Foreground and Background Latent Dirichlet Allocation. Available online: https://github.com/laos1984/FB-LDA (accessed on 13 June 2018).
29. Röder, M.; Both, A.; Hinneburg, A. Exploring the Space of Topic Coherence Measures. In Proceedings of the WSDM 15: The Eighth ACM International Conference on Web Search and Data Mining, Shanghai, China, 2–6 February 2015.
30. Mei, Q.; Shen, X.; Zhai, C. Automatic labeling of multinomial topic models. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, CA, USA, 12–15 August 2007.
31. Yarnguy, T.; Kanarkard, W. Tuning Latent Dirichlet Allocation Parameters using Ant Colony Optimization. *J. Telecommun. Electron. Comput. Eng.* **2018**, *10*, 21–24.
32. Teh, Y.W.; Jordan, M.I.; Beal, M.J.; Blei, D.M. Sharing clusters among related groups: Hierarchical Dirichlet Processes. *Adv. Neural. Inf. Process. Syst.* **2004**, *17*, 1385–1392.
33. Wilson, C.P. Term weighting schemes for latent dirichlet allocation. In Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, CA, USA, 2–4 June 2010.
34. Limsettho, N.; Hata, H.; Matsumoto, K. Comparing Hierarchical Dirichlet Process with Latent Dirichlet Allocation in Bug Report Multiclass Classification. In Proceedings of the 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Las Vegas, NV, USA, 30 June–2 July 2014.
35. Gensim Project Description, Ver. 3.8.3. Available online: https://pypi.org/project/gensim (accessed on 3 March 2021).

36. pyLDAvis History, Dates of Each pyLDAvis Version. Available online: https://pyldavis.readthedocs.io (accessed on 3 March 2021).
37. MALLET, MAchine Learning for LanguagE Toolkit Website. Available online: http://mallet.cs.umass.edu (accessed on 3 March 2021).
38. Blei, D.; Lafferty, J. Topic models. *Text Min. Classif. Clust. Appl. J.* **2009**, *10*, 101–124.