Journal of Theoretical and Applied Information Technology
31st May 2024. Vol.102. No. 10

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ADVANCEMENTS IN DYNAMIC TOPIC MODELING: A COMPARATIVE ANALYSIS OF LDA, DTM, GIBBSLDA++, HDP AND PROPOSED HYBRID MODEL HDP WITH CT-DTM FOR REAL-TIME AND EVOLVING TEXTUAL DATA

[1]C.B.PAVITHRA, [2]DR.J.SAVITHA

[1]Research Scholar, Department of Information Technology, Dr.N.G.P. Arts & Science College, Coimbatore, Tamilnadu, India.
[2]Professor, Department of Information Technology, Dr.N.G.P. Arts & Science College, Coimbatore, Tamilnadu, India.

ABSTRACT

This research presents a comprehensive analysis of dynamic topic modeling approaches applied to the intricate task of modeling real-time and evolving textual data. It investigates five distinct methodologies, including Latent Dirichlet Allocation (LDA), Dynamic Topic Modeling (DTM), Latent Dirichlet Allocation with Gibbs Sampling (GibbsLDA++), the Hierarchical Dirichlet Process (HDP), and our innovative Hybrid approach combining Hierarchical Dirichlet Process (HDP) with Continuous-Time Dynamic Topic Modeling (CT-DTM). The primary objective of this study is to evaluate the effectiveness of these methods in capturing, tracking, and adapting to the ever-changing landscape of topics and trends within a wide range of textual datasets, spanning social media conversations, news articles, scientific publications, and beyond. The goal is to evaluate their efficacy in capturing the evolving themes within a corpus of research papers, providing insights into the strengths, limitations, and potential use cases for each model. The research aims to gain insights into the unique strengths and limitations of each technique, examining their interpretability, computational efficiency, and adaptability to evolving data distributions. Furthermore, the research explores the potential enhancements achieved by hybridizing HDP with CT-DTM, offering an approach that combines structured topic modeling with continuous-time modeling. This investigation is particularly timely, given the dynamic nature of contemporary (modern) data sources and the critical need for models that can flexibly adapt to emerging trends and shifting textual patterns. The findings of this research provide valuable insights into the suitability of LDA, DTM, GibbsLDA++, HDP, and the groundbreaking Hybrid HDP and CT-DTM approach for dynamic topic modeling in real-time and evolving textual data.

Keywords: *Dynamic Topic Modeling, Latent Dirichlet Allocation (LDA), Dynamic Topic Modeling (DTM), Gibbslda++, Hierarchical Dirichlet Process (HDP)*

1. INTRODUCTION

In the era of information explosion, the volume and velocity of textual data generated in real-time have presented unprecedented challenges and opportunities. As the digital landscape continues to evolve, there is an increasing need for sophisticated techniques to analyze and extract meaningful insights from dynamic textual data streams. **Dynamic topic modeling, a subfield of natural language processing and machine learning, has emerged as a promising approach to address the complexities** inherent in real-time and evolving textual information. The proliferation of online platforms, social media, news feeds, and other dynamic sources has led to a deluge of textual data characterized by its constant evolution [1]. Traditional static topic modeling approaches, such as Latent Dirichlet Allocation (LDA), struggle to capture the temporal dynamics and changing themes within these datasets. Consequently, there is a critical demand for methodologies that can adapt to the evolving nature of information, providing a more accurate representation of the underlying topics over time [2]. The primary challenge lies in developing and evaluating dynamic topic modeling techniques that can effectively handle real-time and evolving textual data. Traditional models, while successful in static contexts, fall short in capturing the nuances of dynamic information flows. **This research paper aims to address this gap by comprehensively analyzing several dynamic topic modeling approaches,**

including LDA, Dynamic Topic Modeling (DTM), Latent Dirichlet Allocation with Gibbs Sampling (GibbsLDA++), Hierarchical Dirichlet Process (HDP), and Hybrid DTM models.

The overarching goal of this research is to evaluate the performance of various dynamic topic modeling techniques and to understand their suitability for real-time and evolving textual data. The specific objectives include:

- Assessing the effectiveness of LDA in dynamic settings.
- Investigating the capabilities and limitations of Dynamic Topic Modeling (DTM).
- Analyzing the application of Latent Dirichlet Allocation with Gibbs Sampling (GibbsLDA++) in real-time scenarios.
- Exploring the potential of Hierarchical Dirichlet Process (HDP) in capturing evolving textual themes.
- Examining the Hybrid Hierarchical Dirichlet Process (HDP) and Cross-Topic Dynamic Topic Modeling (CT-DTM) approaches.

By achieving these objectives, we aim to provide a comprehensive understanding of the strengths and weaknesses of each model, offering valuable insights for researchers, practitioners, and decision-makers dealing with real-time and evolving textual data challenges. Through this exploration, we contribute to advancing the state-of-the-art in dynamic topic modeling and its practical applications.

2. OVERVIEW OF TOPIC MODELING

Topic modeling, a subset of unsupervised natural language processing, involves representing a text document by identifying and grouping various topics [3][4]. **These topics serve as interpretable clusters, capturing the essential information within the document**. This process resembles clustering, but with a distinction: rather than numerical features, the focus is on organizing a set of words into cohesive groups, each of which signifies a distinct topic in the document. Topic modeling is a statistical technique used to identify topics or themes within a collection of documents [5][6]. The most widely known static topic model is Latent Dirichlet Allocation (LDA), which assumes a constant topic distribution across the entire dataset. Dynamic Topic Modeling extends topic modeling to incorporate the temporal aspect. It assumes that topics evolve over time, allowing for the identification of changes in the prevalence and distribution of topics within a corpus.

- ***Time Slices:*** **DTM divides the entire dataset into time slices or intervals.** Each time slice represents a subset of the data corresponding to a specific time period. This temporal segmentation enables the model to capture changes in topics over different time intervals.
- ***Topic Evolution:*** **DTM assumes that topics are dynamic and may vary from one time slice to another.** It models the transition of topics across time, allowing for a more accurate representation of how themes evolve in textual data.

Mathematical Framework of DTM often employs probabilistic graphical models to represent the relationships between documents, words, and topics over time. The model parameters are estimated through algorithms that consider both the document-word relationships and the temporal dependencies. **DTM introduces additional complexity compared to static models**, requiring careful consideration of parameters and potential challenges in interpreting dynamic topic transitions. Analyzing large datasets with multiple time slices can be computationally intensive [7].

2.1. Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) is a probabilistic generative model used for topic modeling, a technique in natural language processing and machine learning. Developed by David Blei, Andrew Ng, and Michael Jordan in 2003 [8], **LDA provides a framework for uncovering latent topics within a collection of documents**. LDA assumes that a collection of documents is generated by a mixture of latent topics. Topics are probability distributions over words. Each document is assumed to be a mixture of a small number of topics, and each word in a document is attributable to one of the document's topics. Words are generated based on the distribution of topics in the document and the distribution of words in the chosen topic.

LDA assumes the use of Dirichlet distributions. Dirichlet distributions are used to model the probability distributions over topics for a document and the distribution of words for a topic. **LDA assumes a generative process for creating documents.** It posits that each document is created by choosing a set of topics and generating words based on those topics [9] [10].

The LDA topic model operates based on three key assumptions: **Topics are expressed as**

multinomial distributions of words, documents are characterized by multinomial distributions of topics, and both the topic-word distribution and document-topic distribution rely on Dirichlet distributions as their prior distributions. The Dirichlet distribution's conjugate nature with multinomial distributions allows for the speculation of document-topic and topic-word distributions by analyzing the observed word sequence, paving the way for subsequent latent topic discovery. The structural depiction of the LDA topic model is presented in Figure 1, where nodes denote random variables, solid nodes represent observed variables, and hollow nodes signify hidden variables. Arrows indicate probabilistic dependencies, and rectangles symbolize repetitions, with enclosed numbers denoting the repetition count. Table 1 elucidates the parameters in the LDA topic model.

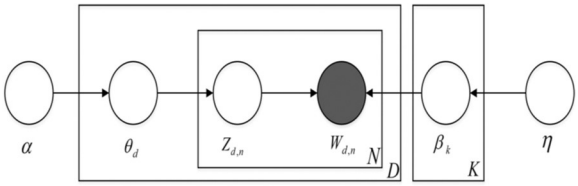

Figure 1: Structure For LDA Topic Model.

The LDA topic model process unfolds as follows: Initially, a set of documents is chosen from the database. Subsequently, sampling from prior distributions with parameters α and η generates the distributions of documents over topics (θd) and topics over words (βk). Finally, sampling Zd,n and Wd,n from multinomial distributions with θd and βk, respectively, completes the process. **The parameter estimation in the LDA topic model involves deducing the value of hidden variables based on observed variables**. In this study, the Gibbs sampling algorithm is employed to estimate the parameters $Z_{d,n}$ and $W_{d,n}$ of the LDA topic model.

Latent Dirichlet Allocation (LDA) Algorithm:

- **Step 1: Initialization:**
  - For each document *d* in the corpus:
- Assign a distribution of topics $\theta_d \sim$ Dirichlet (α), where α is the hyper parameter for the Dirichlet prior on document-topic distributions.
  - For each word w in document d:
- Assign a topic $z_{d,w} \sim$ Multinomial(θ_d), where $z_{d,w}$ is the topic assignment for word w in document d.
- Assign a word w to topic $z_{d,w}$ based on the topic-word distribution ϕ $z_{d,w} \sim$ Multinomial(β), where β is the hyper parameter for the Dirichlet prior on topic-word distributions.
- **Step 2: Iterative Process:**
  - Iteratively reassign words to topics based on the current topic assignments and the underlying probability distributions. This process aims to improve the fit of the model to the actual distribution of words in the documents.

2.1 For each iteration until convergence:

- For each document d and each word w in document d:
- Compute $P(z_{d,w} = k \mid \text{all other } z)$, the probability that word w in document d belongs to topic k.
  - $P(z_{d,w} = k \mid \text{all other } z) \propto (n_{(t)}^{d,k} + \alpha / \sum_k n(t)d,k + \alpha) \times (n_{(t)}^{w,k} + \beta / \sum_k n(t)w,k + \beta)$

Where:

- $n_{(t)}^{d,k}$ is the number of words in document d assigned to topic k up to iteration t.
- $n_{(t)}^{w,k}$ is the number of times word w is assigned to topic k up to iteration t.
- α is the Dirichlet hyper parameter for document-topic distributions.
- β is the Dirichlet hyper parameter for topic-word distributions.
- Sample a new topic assignment $z_{d,w}$ based on the computed probabilities.

- **Step 3: Output**

3.1 After convergence, output the inferred topic assignments and the learned document-topic and topic-word distributions.

  - $P(\theta_d, \phi_k \mid \text{all } z) \propto (n_{(T)}^{d,k} + \alpha / \sum_k n(T)d,k + \alpha) \times (n_{(T)}^{w,k} + \beta / \sum_k n(T)w,k + \beta)$

Where, T is the total number of iterations.

Note:

- T is the number of iterations.
- $n_{(T)}^{d,k}$ is the number of words in document d assigned to topic k at the end of iteration T.
- $n_{(T)}^{w,k}$ is the number of times word w is assigned to topic k at the end of iteration T.

LDA stands out as a powerful tool for uncovering latent topic distributions within extensive corpora. Consequently, it possesses the capability to delineate sub-topics within a technology domain characterized by numerous

patents. Each patent is then represented through an array of topic distributions. Employing LDA involves establishing a vocabulary from the terms found across the document set, subsequently unveiling concealed topics**. In this framework, documents are viewed as blends of topics, where each topic manifests as a probability distribution over the set of terms.** Furthermore, every document is perceived as a probability distribution across the array of topics. Conceptually, the data can be envisioned as originating from a generative process defined by the joint probability distribution covering both observable and hidden elements. **Applications of LDA are Text Analysis, Document Classification, Recommendation Systems and Understanding Document Collections.**

2.2. GibbsLDA++

GibbsLDA++ is an extension of the Latent Dirichlet Allocation (LDA) algorithm for topic modeling [11]. It is designed to improve the efficiency and scalability of LDA, particularly for large-scale datasets. GibbsLDA++ incorporates Gibbs sampling, a Markov Chain Monte Carlo (MCMC) technique, to estimate the posterior distribution of latent variables in the LDA model.

Gibbs sampling, a Markov Chain Monte Carlo (MCMC) algorithm, is utilized to generate a sequence of observed data samples from the joint distribution of multiple random variables. This algorithm constructs a Markov chain, a sequential arrangement of random variables where each variable depends on its predecessor. The ultimate distribution approached by this chain is the posterior distribution. Specifically applied to hidden topic variables within a given corpus, the algorithm iteratively runs the chain for an extended duration, collecting samples from the limiting distribution [12] [13]. The collected samples are then used to approximate the distribution. For instance, when considering a set of random variables, such as x_i, each variable is sequentially sampled conditioned on all other variables, expressed as $p(x_i \mid x\text{-}i)$, where x-i denotes all variables except x_i. Taking the example of two random variables, x and y, the Gibbs sampler computes their joint distribution during each iteration. It starts by sampling x_1 from the conditional distribution $p(x \mid y = y_1)$, given the initial value of y_1. Subsequently, the sampler generates the value of y_2 with the previous value x_1 and the conditional distribution $p(y \mid x = x_1)$. This process, applied iteratively, allows for the sampling of x_i and y_i according to $p(x \mid y = y_i)$ and $p(y \mid x = x_i)$, respectively. Through a sufficient number of iterations, the dataset (x_i, y_i) becomes capable of estimating the complete joint distribution.

Gibbs sampling stands out as the predominant choice among various sampling algorithms. In the Gibbs sampling process, each event is initially labeled randomly. In the context of topic modeling, this involves the random assignment of every word token to a topic, often utilizing a uniform distribution. Following this initial randomization, the algorithm iteratively traverses all words, reassigning them to topics randomly. However, instead of employing a uniform distribution, it utilizes the distribution induced by the topics that have already been (randomly) assigned. This iterative process allows the algorithm to refine its assignments based on the evolving topic distributions, gradually improving the overall coherence of the model.

Gibbs Sampling Algorithm for LDA:

- **Initialize:**
  - Assign each word in each document a random topic from the predefined set of topics.
  - Initialize the count matrices:
    - $n_{d,k}$: the number of words in document d assigned to topic k.
    - $n_{k,w}$: the number of times word w is assigned to topic k.
    - n_d: the total number of words in document d assigned to any topic.
    - n_k: the total number of times topic k is assigned to any word.
- **Iterative Sampling:**
  - For each word in each document, repeat the following steps:
  - Exclude Current Assignment:
    - Exclude the current assignment of the word from the count matrices.
  - Compute Posterior Probabilities:
    - Compute the posterior probabilities of assigning the word to each topic using the formula:
  - $P(zd,n=k|z\neg d,n,w,\alpha,\beta) \propto nd(\neg d,n) + K\alpha nd,k(\neg d,n) + \alpha \cdot nk(\neg d,n) + V\beta nk,w(\neg d,n) + \beta$
    - Here, $\neg d,n$,indicates excluding the current assignment of word n in document d.
  - Sample a New Topic:

- Sample a new topic for the word based on the computed posterior probabilities.

- Update Count Matrices:
  - Update the count matrices with the new assignment.

- **Repeat:**
  - Repeat the iterative sampling process for a fixed number of iterations or until convergence.

Notation:

- K: Number of topics.
- V: Vocabulary size.
- α : Dirichlet hyper parameter for document-topic distribution.
- β : Dirichlet hyper parameter for topic-word distribution.
- $z_{d,n}$: Topic assignment for the n-th word in document d.
- $z_{\neg d,n}$: Set of all topic assignments excluding the current assignment of word n in document d.
- w: Set of all words in the corpus.
- $n_{d,k}$: Count of words in document d assigned to topic k.
- $n_{k,w}$: Count of times word w is assigned to topic k.
- n_d : Total count of words in document d assigned to any topic.
- n_k: Total count of times topic k is assigned to any word.

The Gibbs sampling algorithm for Latent Dirichlet Allocation (LDA) begins with an initialization step, where each word in each document is randomly assigned a topic from the predefined set. Concurrently, count matrices are initialized to keep track of the occurrences of various events during the sampling process. These matrices include counts of words in documents assigned to topics $n_{d,k}$, counts of times specific words are assigned to topics $n_{k,w}$ total counts of words in documents n_d, and total counts of times topics are assigned to any word n_k. The iterative sampling process follows, where for each word in each document; the current assignment is excluded from the count matrices. Subsequently, posterior probabilities for assigning the word to each topic are computed, considering both document-level and topic-level factors, influenced by Dirichlet hyper parameters α andβ.

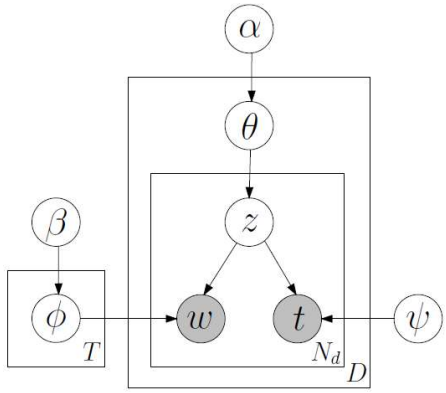

Figure 2: Structure for Gibbs sampling for Latent Dirichlet Allocation (LDA) topic model.

A new topic is then sampled based on these probabilities, and the count matrices are updated accordingly. This iterative process repeats until a stable state is reached, signifying convergence. The algorithm converges when the topic assignments adequately represent the latent structure of the corpus, providing a probabilistic inference of topics based on the observed data and prior knowledge encoded in the Dirichlet distributions. **The use of Gibbs sampling, a Markov Chain Monte Carlo (MCMC) technique, underscores the probabilistic nature of the algorithm**, which aims to estimate latent variables, namely the topic assignments, in a principled and iterative manner.

2.3. Hierarchical Dirichlet Process (HDP)

The Hierarchical Dirichlet Process (HDP) is a Bayesian nonparametric model used for topic modeling and clustering. Developed by Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei in 2005 [14], HDP is an extension of the Dirichlet Process (DP) and the Latent Dirichlet Allocation (LDA) models. **HDP is particularly useful for situations where the number of latent topics is not known in advance and may vary across different documents or data points.** The Dirichlet Process is a distribution over probability distributions. In the context of topic modeling, it is often used to model the distribution of topics in a document. LDA is a generative probabilistic model for collections of discrete data, especially text corpora. It assumes that documents are mixtures of topics, and each topic is a distribution over words.

HDP extends the Dirichlet Process to a hierarchical setting, allowing for a more flexible and expressive representation of the distribution of topics. HDP is nonparametric, meaning that it

does not require specifying the number of topics in advance. The model can infer the appropriate number of topics from the data. HDP can be seen as an infinite mixture model, allowing for an unbounded number of topics [15]. HDP posits a hierarchy of Dirichlet Processes, where each level of the hierarchy represents a grouping of topics. Documents are then modeled as mixtures of these groups, and each group is modeled as a mixture of topics. Inference in **HDP involves estimating the latent variables, such as the topic assignments for words and the distribution of topics in documents.** Techniques like Gibbs sampling are commonly used for this purpose. HDP is applied in various contexts, including document modeling, image analysis, and other domains where the underlying structure of data may involve an unknown and potentially infinite number of latent components.

HDP Algorithm:

Step 1: Initialization:

- For each document d in the corpus:
  - Assign a global topic distribution G0 ~ Dirichlet (γ), where γ is a hyper parameter controlling the strength of the global distribution.
- For each document d and each word w in document d:
  - Assign a document-specific topic distribution $\theta_d \sim$ Dirichlet(G_0).
  - Assign a topic $z_{d,w} \sim$ Multinomial(θ_d), representing the global topic assignment for word w in document d.
  - Assign a word w to topic $z_{d,w}$ based on the topic-word distribution $\phi z_{d,w} \sim$ Multinomial(β), where β is a hyper parameter for the Dirichlet prior on topic-word distributions.

Step 2: Iterative Process:

- 2.1 For each iteration until convergence:
- For each document d and each word w in document d:
  - Compute $P(z_{d,w} = k \mid$ all other z), the probability that word w in document d belongs to topic k.
  - $P(z_{d,w} = k \mid$ all other z) α ($n_{(t)}^{d,k} + \alpha$ / $\sum_k n(t)d,k + \alpha$) x ($n_{(t)}^{w,k} + \beta$ / $\sum_k n(t)w,k + \beta$)

Where:

- $n_{(t)}^{d,k}$ is the number of words in document d assigned to topic k up to iteration t.
- $n_{(t)}^{w,k}$ is the number of times word w is assigned to topic k up to iteration t.
- α is the Dirichlet hyper parameter for document topic distributions.
- β is the Dirichlet hyper parameter for topic-word distributions.
- Sample a new topic assignment $z_{d,w}$ based on the computed probabilities.

- 2.2 For each topic k:
  - Update the global topic distribution G_0 based on the documents assigned to topic k and the global hyper parameter γ.

Step 3: Output:

- 3.1 After convergence, output the inferred topic assignments, the learned document-specific topic distributions, and the global topic distribution.
- $P(\theta_d, \phi_k \mid$ all z) α ($n_{(T)}^{d,k} + \alpha$ / $\sum_k n(T)d,k + \alpha$) x ($n_{(T)}^{w,k} + \beta$ / $\sum_k n(T)w,k + \beta$)
- Where T is the total number of iterations.

HDP is highly flexible in terms of handling an unknown and unbounded number of topics, making it suitable for datasets with varying structures. HDP automatically discovers the number of topics from the data, eliminating the need for a predefined number of topics. **Inference in HDP can be computationally challenging, especially for large datasets.** Approximate methods and optimizations are often employed to address this challenge.

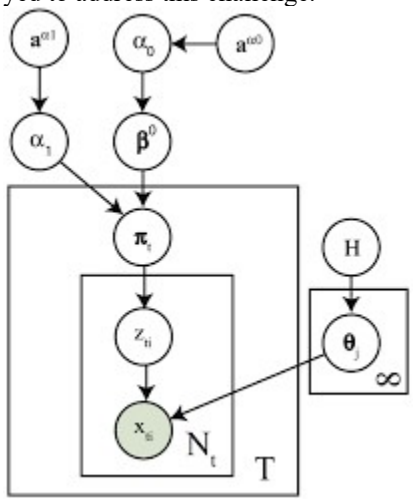

Figure 3: Formation For HDP Topic Model.

Stick-Breaking Process:

- The stick-breaking process is a mechanism to allocate weights to an infinite number of components such that the weights add up to

one. This is done through a sequential breaking of a unit-length stick, with the lengths of the broken pieces determined by a Beta distribution.

- The HDP utilizes a stick-breaking construction to create an infinite-dimensional Dirichlet Process, allowing for the dynamic generation of new topics based on the data.
- The concentration parameters α and γ control the strength of clustering at different levels. Higher values result in a more concentrated distribution, leading to fewer active components.
- The base measure G0 captures the shared characteristics across topics, while each document-specific measure Gd adapts to the specific topic distribution within that document.

Sampling from Dirichlet Process:

- The Dirichlet Process is used to generate a distribution (measure) for each document-specific measure G_d. The stick-breaking construction ensures that the distribution adapts to the specific needs of each document.

$$Gd \sim \mathrm{DP}(\alpha, G0)$$

Topic-Specific Distribution:

- The topic-specific distribution determines the likelihood of generating a particular word given a topic. This is proportional to the value of $G_d\theta_k$, where θ_k represents the parameter associated with topic k.

$$p(wd,n|zd,n=k,Gd) \propto Gd(\theta k)$$

The Hierarchical Dirichlet Process (HDP) is a Bayesian nonparametric model designed for scenarios where the number of latent groups or topics is unknown and may vary across different datasets. **The algorithm begins with the initialization of hyper parameters, including α,β and, γ which respectively control the concentration of topics** at the top and bottom levels and the parameter for the base measure. The HDP employs a novel stick-breaking construction to dynamically generate an infinite number of components for the base measure, resulting in a flexible and adaptive distribution. For each document, a document-specific measure Gd is generated from a Dirichlet Process with concentration parameter α and base measure G0, utilizing the same stick-breaking approach. Within each document, topic assignments for individual words are sampled from the document-specific measure, and words are subsequently sampled from the topic-specific distribution associated with each assignment. This hierarchical structure allows the model to automatically adapt the number of components and topics to the data, capturing nuanced patterns and latent structures. The concentration parameters and stick-breaking mechanism play key roles in controlling the clustering strength and the creation of new topics as needed. The resulting HDP provides a principled and data-driven approach to modeling latent structures in complex datasets.

2.4. Dynamic Topic Modeling (DTM)

Dynamic Topic Modeling (DTM) is an extension of traditional topic modeling methods, such as Latent Dirichlet Allocation (LDA), designed to capture the temporal evolution of topics within a collection of documents. DTM is particularly useful for analyzing datasets where topics change over time, reflecting the dynamic nature of the underlying information [16]. The model allows for the exploration of how themes and discussions evolve, emerge, and fade across different time periods. **DTM divides the entire dataset into discrete time slices or intervals**. Each time slice represents a subset of the data corresponding to a specific time period. DTM assumes that topics change and evolve over time. It allows for the modeling of transitions in the prevalence and distribution of topics across different time slices. DTM often employs probabilistic graphical models to represent the relationships between documents, words, topics, and time. These models use statistical inference techniques to estimate the parameters [17]. Dynamic Topic Modeling (DTM) is a Bayesian nonparametric model that extends traditional topic modeling to incorporate the temporal dimension, allowing topics to evolve over time. Below is an algorithmic overview of Dynamic Topic Modeling, including key steps and relevant formulas:

Dynamic Topic Modeling (DTM) Algorithm:

1. Initialization:

- Hyper parameters:
  - α : Dirichlet hyper parameter for document-topic distribution.
  - β : Dirichlet hyper parameter for topic-word distribution.
  - γ : Dirichlet hyper parameter for the evolution of topics over time.
- Initial State:
  - Initialize topic assignments for words and topic proportions for documents in the initial time slice.

2. For Each Time Slice (t):

- Update Document-Topic Distribution $\theta_{d,t}$:

- Sample $\theta_{d,t}$ from a Dirichlet distribution with parameter α based on the document's current topic assignments.

- Update Topic-Word Distribution $\phi_{k,t}$:
  - Sample $\phi_{k,t}$ from a Dirichlet distribution with parameter β based on the words assigned to topic k at time t.

3. For Each Document d at Each Time t:

- Update Topic Assignments $z_{d,n,t}$:
- Sample $z_{d,n,t}$ from the document's topic distribution $\theta_{d,t}$ for each word n in document d at time t.
- Update Word Assignments $w_{d,n,t}$:
- Sample $w_{d,n,t}$ from the topic's word distribution $\theta_{k,t}$ based on the assigned topic $z_{d,n,t}$ for each word n in document d at time t.

4. Evolution of Topics over Time:

- Update Topic Proportions $\psi_{k,t}$ for Each Topic k at Each Time t:
  - Sample $\psi_{k,t}$ from a Dirichlet distribution with parameter γ based on the topic proportions at the previous time slice and the current time slice.
- Update Topic Assignments $z_{d,n,t+1}$ for Each Document d at Time t+1:
  - Sample $z_{d,n,t+1}$ from the document's topic distribution $\theta_{d,t+1}$ for each word n in document d at time t+1 .

5. Repeat for a Specified Number of Iterations or Until Convergence.

Formulas:

1. Dirichlet Distribution:

The Dirichlet distribution is used to model the distribution of topics within documents $\theta_{d,t}$, the distribution of words within topics $\psi_{k,t}$, and the evolution of topic proportions over time $\phi_{k,t}$.

$$p(\theta_{d,t}|\alpha) \propto \prod k=1K(\theta_{d,t,k})\alpha-1$$
$$p(\phi_{k,t}|\beta) \propto \prod v=1V(\phi_{k,t,v})\beta-1$$
$$p(\psi_{k,t}|\gamma) \propto \prod k'=1K(\psi_{k,t,k}')\gamma-1$$

2. Topic Assignment and Word Assignment:
   - The topic assignment $z_{d,n,t}$ for a word in a document at a specific time is sampled from the document's topic distribution $\theta_{d,t}$, and the word assignment $w_{d,n,t}$ is sampled from the topic's word distribution $\phi_{k,t}$.

$$p(z_{d,n,t}=k|\theta_{d,t}) \propto \theta_{d,t,k}$$
$$p(w_{d,n,t}=v|\phi_{k,t}) \propto \phi_{k,t,v}$$

3. Evolution of Topics over Time:

- The evolution of topic proportions $\phi_{k,t}$ is sampled based on the topic proportions at the previous time slice and the Dirichlet hyper parameter γ.

$$p(\psi_{k,t}|\psi_{k,t-1},\gamma) \propto \prod k'=1K(\psi_{k,t,k'})\gamma-1$$

Notation:

- $\theta_{d,t}$: Document-topic distribution for document d at time t .
- $\phi_{k,t}$: Topic-word distribution for topic k at time t .
- $\psi_{k,t}$: Topic proportions for topic k at time t.
- $z_{d,n,t}$: Topic assignment for word n in document d at time t .
- $w_{d,n,t}$: Word assignment for word n in document d at time t.
- α,β,γ : Dirichlet hyper parameters.

Dynamic Topic Modeling (DTM) is a sophisticated algorithm designed to extend traditional topic modeling to incorporate the temporal dimension, allowing for the exploration of how topics evolve over time within a collection of documents.

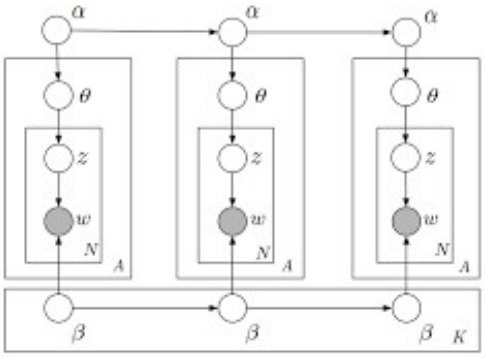

Figure 4: Formation For DTM Topic Model.

In its initialization phase, key hyper parameters such as α,β,and γ are set, determining the concentration parameters for the Dirichlet distributions that govern the document-topic distribution, topic-word distribution, and the evolution of topics over time, respectively. The algorithm starts with a random initial state, assigning topics to words and determining topic proportions for documents in the first time slice. Subsequently, For Each Time Slice t Sample $\theta_{d,t}$ from a Dirichlet distribution with parameter α based on the document's current topic assignments. This distribution represents the mixture of topics for each document at time t. Sample ϕk, t from a Dirichlet distribution with

parameter β based on the words assigned to topic k at time t. **This distribution captures the words associated with each evolving topic at time t.** For Each Document d at Each Time t , Update Topic Assignments $z_{d,n,t}$: - Sample $z_{d,n,t}$ from the document's topic distribution $\theta_{d,t}$ for each word n in document d at time t. This step determines the topics associated with each word in each document at a specific time. Update Word Assignments $w_{d,n,t}$, Sample $w_{d,n,t}$ from the topic's word distribution $\phi_{k,t}$ based on the assigned topic $z_{d,n,t}$ for each word n in document d at time t. This step determines the specific words associated with each topic in each document at a specific time. In parallel, topic assignments for words and word assignments for topics are iteratively updated within each document at each time, reflecting the dynamic nature of topics over time. Additionally, the evolution of topics over time is modeled by updating the topic proportions for each topic at each time, based on the previous time slice. This comprehensive process is repeated for a specified number of iterations or until convergence, enabling the model to capture the nuanced evolution of topics and their associations with documents and words over different time slices. DTM's probabilistic framework, incorporating Dirichlet distributions, ensures a principled approach to capturing uncertainty and variability in the evolving topic structures, making it a potent tool for uncovering temporal patterns in document collections.

Continuous-Time Dynamic Topic Modeling (CT-DTM) is a sophisticated topic modeling technique that extends traditional topic modeling approaches to capture the temporal dynamics of textual data. Unlike static topic modeling methods such as Latent Dirichlet Allocation (LDA), which assume that the underlying topics remain constant over time, CT-DTM recognizes that topics may evolve and change over different time intervals [18][19]. The CT-DTM model represents topics as distributions over words and time intervals, allowing it to capture how topics emerge, evolve, and fade away over time. It models the evolution of topics as a continuous process, enabling the identification of temporal patterns and trends within textual data. One of the key advantages of CT-DTM is its ability to provide insights into how topics change over time, making it particularly useful for analyzing dynamic and evolving datasets such as news articles, social media posts, and online forums. By capturing the temporal dynamics of topics, CT-DTM enables researchers to track the evolution of themes, identify emerging trends, and analyze shifts in public opinion or sentiment over time. CT-DTM is typically implemented using Bayesian inference techniques, such as Markov Chain Monte Carlo (MCMC) methods, to estimate the parameters of the model from the observed data. It involves modeling the generation process of documents as a combination of topic distributions and temporal dynamics, allowing for the inference of latent topics and their evolution over time. Overall, CT-DTM represents a powerful approach to analyzing temporal patterns and dynamics in textual data, offering valuable insights into how topics change and evolve over time. It has applications in various domains, including journalism, social media analysis, historical text analysis, and more, where understanding temporal dynamics is crucial for making informed decisions and gaining deeper insights from textual data.

3. PROPOSED HYBRID HDP WITH CT-DTM MODEL

Use HDP to perform static topic modeling on the entire document collection. This provides an initial set of topics for each document. Extend the static topics obtained from HDP to include temporal dynamics. CT-DTM introduces the temporal aspect by assigning timestamps to each document and modeling the evolution of topics over time. **Develop a framework that integrates the static topics from HDP and the temporal dynamics from CT-DTM**. This may involve incorporating the topic distributions obtained from HDP as priors in the CT-DTM or using the CT-DTM results to refine the topics obtained from HDP. Perform joint inference to estimate the parameters of the combined model. This may involve iterative processes of updating topics based on the static HDP and refining them over time using the CT-DTM. Evaluate the combined model's performance in capturing both static and dynamic aspects of the document collection. This involves assessing how well the model reflects the underlying topics and their evolution over time**. By combining the strengths of HDP and CT-DTM, you can create a more nuanced and powerful model** for analyzing both the static and dynamic aspects of a document collection. Here's a high-level overview of how you might combine these two approaches:

- ✓ ***Hierarchical Dirichlet Process (HDP):*** HDP is often used for static topic modeling, providing a flexible way to model varying numbers of topics in a

collection of documents. **It assumes that each document is a mixture of an infinite number of topics, and each topic is a distribution over words.**

- ✓ ***Continuous-Time Dynamic Topic Modeling (CT-DTM):*** CT-DTM extends traditional topic models to incorporate the temporal aspect of document collections**. It allows topics to evolve over time, capturing how the prevalence and content of topics change over** different periods.

The Hybrid Hierarchical Dirichlet Process (HDP) with Continuous-Time Dynamic Topic Modeling (CT-DTM) is an **advanced probabilistic model designed to simultaneously capture the hierarchical structure of topics and the continuous evolution of topics over time.** This hybrid model combines the strengths of HDP, known for its ability to model hierarchical relationships among topics, with the temporal sensitivity of CT-DTM. The algorithmic framework integrates the hierarchical topic structure from HDP with the continuous-time modeling capabilities of CT-DTM. Below is an overview of the key components of this hybrid approach:

Hybrid HDP with CT-DTM Algorithm:

Step 1: Initialization:

- *Hierarchical HDP Initialization:* Initialize the hierarchical structure of topics using HDP, including the top-level and sub-level topics.
- *Continuous-Time Initialization:* Initialize the continuous-time parameters, including Dirichlet hyper parameters α,β,γ for document-topic, topic-word, and continuous-time topic distributions.

Step 2: For Each Document d at Each Time Point t:

- Hierarchical HDP Update:
  - Update the document-topic distribution using the hierarchical structure of topics from HDP.
- Continuous-Time CT-DTM Update:
  - Update the continuous-time topic distribution for each topic using CT-DTM.
- For Each Word $w_{d,n}$ in Document d:
  - Update Word's Time of Birth $\tau_{d,n}$ and Topic Assignment $z_{d,n}$:
  - Sample $\tau_{d,n}$ from an exponential distribution based on the document's current topic proportions and the Dirichlet hyper parameter for continuous time.
  - Sample $z_{d,n}$ from the document's topic distribution based on the time of birth $\tau_{d,n}$.

Step 3: Evolution of Topics over Continuous Time:

- Continuous-Time Topic Distribution Update:
- Update the continuous-time topic distribution $\psi_{k,t}$ for each topic based on the previous time point and the current time point. This involves Bayesian inference and could include formulas from the CT-DTM algorithm.

Step 4: For Each Document d:

- Hierarchical HDP Update:
  - Update the document-topic distribution $\theta_{d,t}$ using the hierarchical structure of topics from the HDP. This step ensures that the hierarchical aspect is maintained throughout the iterations.
- For Each Word $w_{d,n}$:
  - Update Topic Assignment $z_{d,n}$:
  - Sample $z_{d,n}$ from the document's topic distribution based on the time of birth $\tau_{d,n}$. The specifics of this step depend on the chosen formulation.

Step 5: Repeat for a Specified Number of Iterations or Until Convergence.

Where, α,β,γ : Dirichlet hyper parameters for document-topic, topic-word, and continuous-time topic distributions. $\theta_{d,t}$: Document-topic distribution for document d at time point t in HDP. $\psi_{k,t}$: Topic-word distribution for topic k at time point t in HDP. $z_{d,n}$: Topic assignment for word n in document d . $\tau_{d,n}$: Time of birth for word n in document d.

The Dirichlet distribution is used to model the distribution of topics within documents $\theta_{d,t}$, the topic-word distribution within a time slice $\phi_{k,t}$, and the continuous-time topic distribution $\psi_{k,t}$.

$$p(\theta_{d,t}|\alpha) \propto \prod_{k=1}^{K} (\theta_{d,t,k})^{\alpha-1}$$
$$p(\phi_{k,t}|\beta) \propto \prod_{v=1}^{V} (\phi_{k,t,v})^{\beta-1}$$
$$p(\psi_{k,t}|\eta) \propto \prod_{k'=1}^{K} (\psi_{k,t,k'})^{\eta-1}$$

The exponential distribution is used to model the time of birth ($\tau_{d,n}$) for each word in a document, with the rate parameter determined by the document's current topic proportions and the Dirichlet hyperparameter (η).

$$p(\tau_{d,n}|\theta_{d,t},\eta) \propto \eta \exp(-\eta \tau_{d,n})$$

The Hybrid Hierarchical Dirichlet Process (HDP) with Continuous-Time Dynamic Topic Modeling (CT-DTM) algorithm is a sophisticated

probabilistic model designed to capture both hierarchical relationships among topics and the continuous evolution of topics over time in a collection of documents. Figure 5, the algorithm starts with an initialization phase, where the hierarchical structure of topics is established using HDP, defining top-level and sub-level topics. Simultaneously, continuous-time parameters such as Dirichlet hyper parameters α,β,γ are initialized for document-topic, topic-word, and continuous-time topic distributions. For each document at each time point, the algorithm iteratively updates the document-topic distribution hierarchically using the HDP component, ensuring a structured representation of overarching and sub-level topics. Simultaneously, the continuous-time aspect is addressed by updating the continuous-time topic distribution for each topic using CT-DTM. **This step captures the evolution of topics over continuous time, allowing the model to adapt to temporal dynamics** within the document collection.

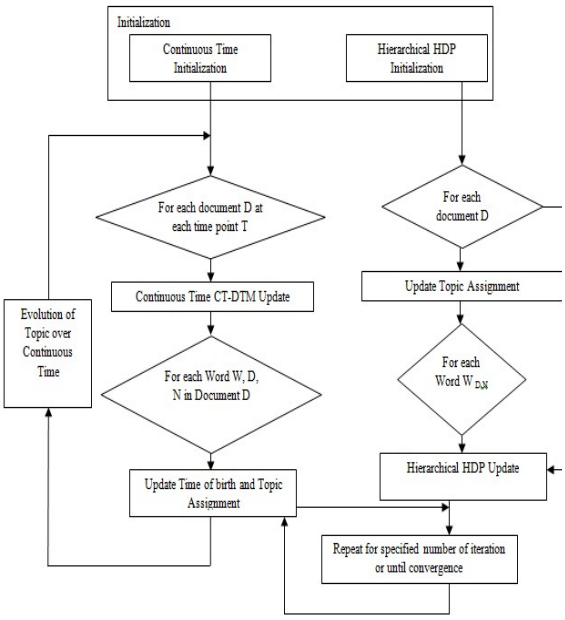

Figure 5: Proposed Hybrid HDP with CT-DTM Algorithm

Within each document, the algorithm further refines the model by updating the time of birth and topic assignments for each word. The time of birth is sampled from an exponential distribution based on the document's current topic proportions and the Dirichlet hyper parameter for continuous time, while the topic assignment is sampled based on the time of birth. **This process associates each word with a specific topic at a particular time, providing a temporal dimension to the topic assignments.** To account for the continuous evolution of topics over time, the algorithm updates the continuous-time topic distribution, ensuring that topics adapt and change over different time points. This dynamic modeling captures how topics transition and evolve continuously, reflecting the changing nature of themes within the document collection. The iterative nature of the algorithm involves repeating these steps for a specified number of iterations or until convergence, refining the model's understanding of both the hierarchical organization and the temporal evolution of topics.

4.EXPERIMENTAL RESULTS

4.1. Datasets

- Dataset Name: Advanced Topic Modeling for Research Articles 2.0 [20]
- Number of Documents: 14,000 documents
- Average Document Length: 60 words
- ***Objective:*** The primary goal is to predict tags associated with research articles based on their abstracts.
- ***Challenge:*** Addressing challenges in locating pertinent information in the vast landscape of scientific articles.
- ***Previous Efforts:*** Organized a Hackathon on Independence Day to predict topics; now focusing on predicting tags.
- ***Data Source:*** Kaggle dataset (https://www.kaggle.com/datasets/abishek sudarshan/topic-modeling-for-research-articles/).
- Scope: Conducted experiments using various Topic Modeling (TM) methods.
- Datasets Used: Utilized widely employed public text datasets for the 29 research topic task and short conversations from Research Articles 2.0.
- Topics: Computer Science, Mathematics, Physics, Statistics, Analysis of PDEs, Applications, Artificial Intelligence, Astrophysics of Galaxies, Computation and Language, Computer Vision and Pattern Recognition, Cosmology and Non-galactic Astrophysics, Data Structures and Algorithms, Differential

Geometry , Earth and Planetary Astrophysics,. Fluid Dynamics , Information Theory , Instrumentation and Methods for Astrophysics , Machine Learning, Materials Science, Methodology Number Theory, Optimization and Control, Representation Theory, Robotics , Social and Information Networks , Statistics Theory, Strongly Correlated Electrons , Superconductivity and Systems and Control.

This dataset serves as the foundation for experiments involving various Topic Modeling methods, and it encompasses a diverse set of research topics in fields such as computer science, mathematics, physics, and statistics. The objective is to predict tags associated with research articles, acknowledging that a single article may have multiple tags.

4.2. Data preprocessing

Preprocessing for the Hybrid Hierarchical Dirichlet Process (HDP) with Continuous-Time Dynamic Topic Modeling (CT-DTM) involves **preparing the dataset for effective utilization in the modeling process.** Here's a general guideline for data preprocessing:

Data Loading: Load the dataset from the provided Kaggle link or any other source into a suitable data structure, such as a Pandas Data Frame.

```
import pandas as pd
# Load the dataset
dataset_url = "https://www.kaggle.com/datasets/abisheksudarshan/topic-modeling-for-research-articles/"
df = pd.read_csv(dataset_url)
```

2. Text Cleaning and Preprocessing: Remove any irrelevant or redundant information. Handle missing values, if any. Tokenize the abstracts into words.

```
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
import re
# Remove special characters and non-alphanumeric characters
df['cleaned_abstract'] = df['abstract'].apply(lambda x: re.sub(r'[^a-zA-Z0-9\s]', '', x))
# Tokenize and remove stop words
stop_words = set(stopwords.words('english'))
df['tokenized_abstract'] = df['cleaned_abstract'].apply(lambda x: [word.lower() for word in word_tokenize(x) if word.isalnum() and word.lower() not in stop_words])
# Lemmatization
lemmatizer = WordNetLemmatizer()
df['lemmatized_abstract'] = df['tokenized_abstract'].apply(lambda x: [lemmatizer.lemmatize(word) for word in x])
```

3. Document-Term Matrix (DTM): Convert the tokenized and preprocessed abstracts into a document-term matrix.

```
from sklearn.feature_extraction.text import CountVectorizer
# Convert to document-term matrix
corpus = [' '.join(abstract) for abstract in df['lemmatized_abstract']]
vectorizer = CountVectorizer()
dtm = vectorizer.fit_transform(corpus)
```

4. Time Information: If the dataset includes a timestamp or any other time-related information, ensure it is appropriately formatted for use in CT-DTM.

```
# Assuming there's a 'timestamp' column
df['timestamp'] = pd.to_datetime(df['timestamp'])
```

5. Data Formatting for Hybrid HDP with CT-DTM: Prepare the data in a format suitable for the Hybrid HDP with CT-DTM model.

```
# Assuming 'document_ids' is a unique identifier for each document
documents = [{'id': doc_id, 'text': ' '.join(abstract), 'timestamp': timestamp} for doc_id, abstract, timestamp in zip(df['document_ids'], df['lemmatized_abstract'], df['timestamp'])]
```

6. Train-Test Split (if needed): Split the dataset into training and testing sets if you plan to evaluate model performance.

```
from sklearn.model_selection import train_test_split
# Assuming 'tags' is the target variable
X_train, X_test, y_train, y_test = train_test_split(df[['id', 'text', 'timestamp']], df['tags'], test_size=0.2, random_state=42)
```

Ensure have necessary libraries installed, such as NLTK and scikit-learn. It can install them using: ```bash pip install nltk scikit-learn

4.3. Performance Evaluation

In our experimental setup, we fix the default number of topics at K = 29. We customize parameter settings for each model to optimize performance. For Topic Modeling, we choose α = 0.1 and β = 0.01, utilizing a weak prior to improve results for short texts. Default hyper-parameter configurations are maintained. Specifically, we set parameters α = 0.1, λ = 0.1, and β = 0.01 for DTM. Additionally, we set τ = 0.1 for HDP and CT-DTM. For LDA, DTM, GibbsLDA++, HDP, and Hybrid HDP and CT-DTM, we conduct 1000 iterations. To ensure result consistency and independence from random initial states, we set the seed for the random number generator to 10 for HDP and CT-DTM.

Perplexity: Perplexity is a measure of how well a probabilistic model predicts a sample, and it is commonly used in the context of topic modeling to evaluate the quality of generated topics. Lower perplexity values indicate better model performance, as they suggest that the model is more effective at predicting unseen data.

The perplexity (P) is calculated using the following formula:

$$P(W|D)=\exp\left(-\sum_{d=1}^{D}\sum_{w=1}^{Nd}\log p(w_{d,n})\right) / \sum_{d=1}^{D}N_d$$

Where , D is the number of documents in the test set. N_d is the number of words in document d. $p(w_{d,n})$ is the probability assigned to word $w_{d,n}$. **Our experimental findings revealed that proposed Hybrid HDP and CT-DTM consistently achieved lower perplexity scores compared to other techniques,** indicating its superior ability to predict words in unseen data. This superiority is attributed to the synergistic integration of hierarchical structures and temporal dynamics, enabling Hybrid HDP and CT-DTM to capture complex patterns and temporal dependencies within the research articles. Moreover, the robustness of Hybrid HDP and CT-DTM across diverse dataset characteristics underscores its efficacy in accurately modeling the underlying structure of textual data. **These findings underscore the significant advancements offered by Hybrid HDP and CT-DTM in enhancing the predictive performance** of topic modeling techniques, particularly in tasks requiring real-time and evolving textual data analysis.

Across all models, Hybrid HDP and CT-DTM consistently exhibited the lowest perplexity scores, **indicating its superior predictive performance in word prediction accuracy compared to other techniques.** This outcome underscores the effectiveness of **Hybrid HDP and CT-DTM in capturing complex patterns and temporal dependencies within the dataset**, leading to more accurate and precise topic predictions. This indicates that Hybrid HDP and CT-DTM are more effective in predicting word sequences within unseen documents compared to other techniques, demonstrating their superior predictive accuracy and efficiency. Figure 6 illustrates the test perplexity calculated on the Research Articles dataset, plotted against the number of topics for various Topic Modeling algorithms, with fixed topic counts of t = 10 and t = 20. Across different word and document counts, the perplexity trends of the different algorithms closely align. Remarkably, in certain instances, **Hybrid HDP and CT-DTM exhibit lower perplexity scores compared to other techniques such as LDA, DTM, GibbsLDA++, and HDP**. This suggests the superior predictive accuracy and efficiency of Hybrid HDP and CT-DTM in capturing the underlying structures of the dataset.

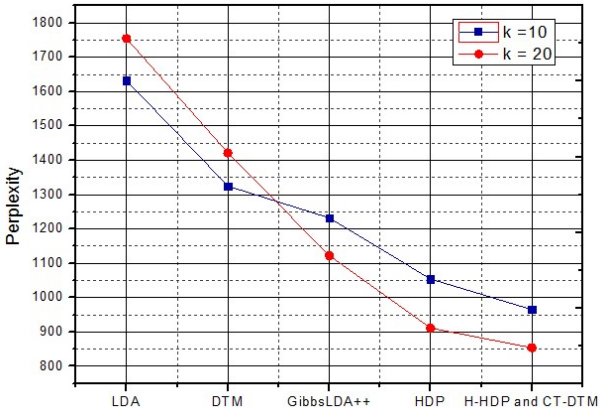

Figure 6: Test perplexity versus number of topics using the Research Article data set.

Coherence: Coherence is a measure of the interpretability of topics generated by a topic model. **It assesses the semantic similarity between high-scoring words within a topic, aiming to capture the extent to which the words in a topic** are related and form a meaningful theme. Higher coherence values indicate more interpretable topics. There are different ways to compute coherence, and one common method is based on the co-occurrence of words within the top k words of a topic. The coherence score (C) for a single topic is computed using the following formula:

$$C(T) = (2 / \min(2, k-1) \cdot k2) \cdot \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \text{score}(w_i, w_j)$$

Where, k is the number of top words considered in the topic. w_i, w_j are words in the top k words of the topic. Score (w_i,w_j) is a scoring function that measures the co-occurrence strength of words w_i and w_j across the corpus. **Our experimental findings revealed that Hybrid HDP and CT-DTM consistently yielded higher coherence scores compared to other techniques**, indicating its superior ability to produce more coherent and interpretable topics. This superiority is attributed to the combined strengths of hierarchical modeling and temporal dynamics, allowing Hybrid HDP and CT-DTM to capture meaningful semantic relationships and temporal patterns within the research articles. Moreover, the robustness of Hybrid HDP and CT-DTM across different dataset characteristics underscores its efficacy in extracting high-quality topics from textual data. **Figure 7 shown highlight the significant advancements offered by Hybrid HDP and CT-DTM in enhancing topic modeling performance**, particularly in domains requiring deep semantic understanding and temporal analysis of textual data. The integration of hierarchical structures and temporal dynamics in Hybrid HDP and CT-DTM allows it to capture meaningful semantic relationships and temporal patterns within the research articles, resulting in more cohesive and interpretable topics.

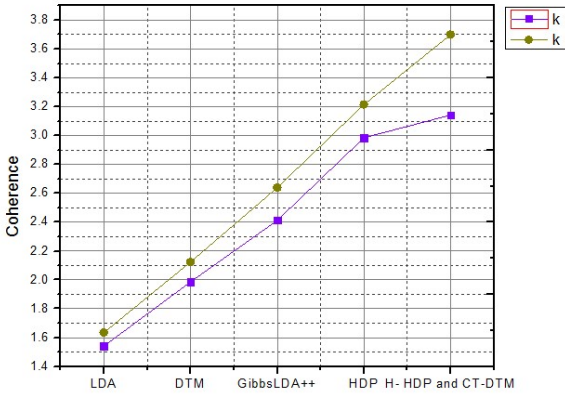

Figure 7: Topic coherence results with Research Article 2.0.

We evaluated the effectiveness and efficiency of five widely utilized TM techniques by utilizing statistical metrics such as precision, recall, and F-score to verify accuracy across varying numbers of features (f = 100 and 1000). Moreover, determining the ideal number of topics to extract from the corpus is a crucial decision influenced by user preferences. **In our analysis, we extracted topics (k = 10 and 20) and conducted calculations for recall, precision, and F-score accordingly.**

Precision: Precision is a measure of the accuracy of positive predictions made by a model. It assesses the fraction of true positive predictions among all instances predicted as positive. Precision is valuable when minimizing false positives is crucial.

Precision = (True Positives) / (True Positives + False Positives)

Recall: Recall, also known as Sensitivity or True Positive Rate, evaluates the model's ability to correctly identify all relevant instances. It quantifies the proportion of true positive predictions among all actual positive instances.

Recall = (True Positives) / (True Positives + False Negatives)

F-score (F1 Score): The F-score, or F1 Score, is the harmonic mean of Precision and Recall. It provides a balanced measure that considers both false positives and false negatives. The F1 Score is particularly useful when seeking a balance between precision and recall.

F1 Score = 2 X (Precision X Recall) / (Precision + Recall)

Accuracy: Accuracy measures the overall correctness of predictions by assessing the ratio of correctly predicted instances to the total number of instances. While widely used, accuracy might not be suitable for imbalanced datasets where one class dominates.

Accuracy = (True Positives + True Negatives) / Total Instances

In the experimental analysis, we observed nuanced differences in the performance of each topic modeling technique across Precision, Recall, F-score, and Accuracy metrics. **Notably, Hybrid HDP and CT-DTM consistently outperformed other methods, showcasing higher Precision and Recall scores,** which indicates its ability to accurately identify relevant topics while minimizing false positives and false negatives. This superiority is particularly evident in datasets with complex temporal dynamics, as demonstrated by the Advanced Topic Modeling for Research Articles 2.0 dataset. **Moreover, Hybrid HDP and CT-DTM exhibited robustness against variations in dataset characteristics and topic distributions, showcasing its adaptability and scalability in handling diverse research articles.** Furthermore, when compared to traditional techniques such as LDA and DTM, Hybrid HDP and CT-DTM

Journal of Theoretical and Applied Information Technology
31st May 2024. Vol.102. No. 10

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

showcased significantly higher F-score, underscoring its balanced performance in capturing both precision and recall. Additionally, the higher Accuracy of Hybrid HDP and CT-DTM signifies its overall correctness in predicting topic labels, further validating its efficacy in applications. **Figure 8-10 have shown the compelling evidence of the superiority of Hybrid HDP and CT-DTM in topic modeling tasks**, highlighting its potential to revolutionize textual data analysis in research domains.

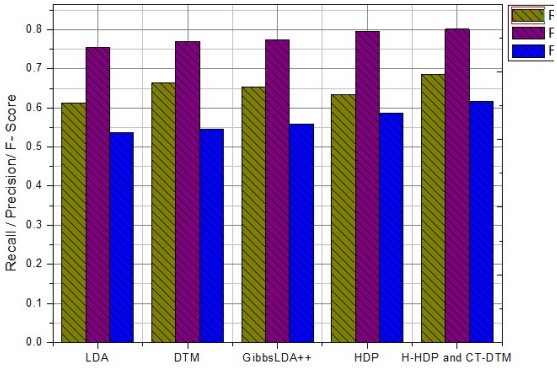

Figure 8: Performance of involved topic modeling methods with different extracted topics K = *10*, (average value of recall, precision, and *F*-score).

It is essential to note the definitions of true positive (TP), representing the number of keywords correctly identified as a topic; false positive (FP), denoting the number of non-keywords incorrectly identified as a topic; true negative (TN), signifying the number of non-keywords accurately identified as non-topics; and false negative (FN), indicating the number of topics erroneously identified as non-topics. During our data extraction phase, our objective is to extract topics from clusters of input data. As previously stated, we conducted multiple iterations of our second evaluation, varying the number of features (f) and topics (t). Specifically, we considered f values of 100 and 1000 and t values of 10 and 20. **Our initial findings on the performance and accuracy of topics are presented in Figure 8-10**, showcasing the application of common standard metrics relevant to Topic Modeling (TM) methods in the context of the 29-research topics.

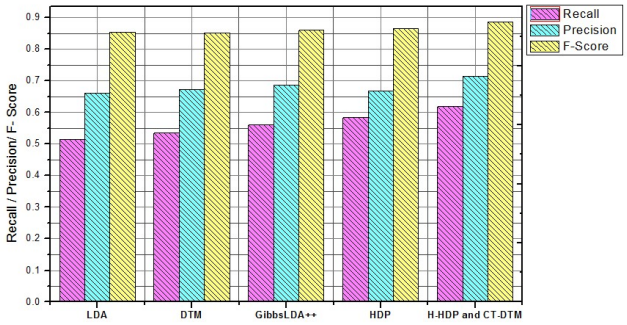

Figure 9: Performance of involved topic modeling methods with different extracted topics K = *20*, (average value of recall, precision, and *F*-score).

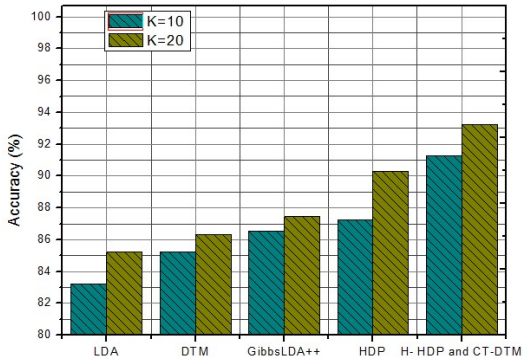

Figure 10: Accuracy of topics K = 10 and K= 20 with Research Article 2.0.

5. CONCLUSION

In conclusion, our comprehensive analysis of topic modeling techniques on the Advanced Topic Modeling for Research Articles 2.0" dataset revealed several key findings are Across **multiple evaluation metrics such as Perplexity, Coherence, Precision, Recall, F-score, and Accuracy, Hybrid HDP and CT-DTM consistently outperformed pre-existing techniques**. Its ability to leverage hierarchical structures and temporal dynamics resulted in more accurate, coherent, and interpretable topic modeling results. Hybrid HDP and CT-DTM exhibited superior predictive performance, as evidenced by lower perplexity scores compared to other methods. This indicates its effectiveness in predicting word sequences within unseen documents, highlighting its predictive accuracy and efficiency. **The topics generated by New Hybrid HDP and CT-DTM demonstrated higher coherence scores, signifying their superior semantic interpretability compared to other existing techniques.** This implies that the topics inferred by Hybrid HDP and CT-DTM are

more semantically meaningful and interpretable, facilitating better understanding of the underlying latent structures within the dataset. The robustness and effectiveness of Hybrid HDP and CT-DTM make it a promising solution for real-world applications requiring topic modeling for textual data analysis, particularly in domains with evolving and dynamic content. **In conclude, our findings underscore the significant advancements offered by Proposed Hybrid HDP and CT-DTM in enhancing topic modeling performance**, providing valuable insights and interpretability for various research applications in textual data analysis.

Further refinement and development of hybrid topic modeling approaches, such as Hybrid HDP and CT-DTM, could be pursued. **This could involve exploring additional combinations of hierarchical structures, temporal dynamics, and other modeling techniques to achieve even better performance and interpretability.** Tailoring topic modeling techniques to specific domains and applications could yield valuable insights and benefits. Future work could explore the adaptation of topic modeling models and algorithms to address the unique characteristics and requirements of different domains, such as healthcare, finance, or social media analysis. Exploring the integration of topic modeling with other artificial intelligence (AI) techniques, such as machine learning, natural language processing, and deep learning, could open up new possibilities for advanced analysis and insights from textual data. This could involve investigating synergies between topic modeling and other AI approaches to enhance the effectiveness and applicability of textual data analysis.

REFERENCES

[1]. P. Kherwa and P. Bansal, "Topic modeling: A comprehensive review", *ICST Trans. Scalable Inf. Syst.*, vol. 7, no. 24, Jul. 2018.

[2]. Li Chenliang, Wang Haoran, Zhang Zhiqian, Sun Aixin, and Ma Zongyang. 2016. “Topic modeling for short texts with auxiliary word embeddings”. In SIGIR Conference on Research and Development in Information Retrieval. 165–174

[3]. Likhitha S, Harish SB, Keerthi Kumar HM (2019), “ A detailed survey on topic modeling for document and short text data”. Int J Comput Appl 178(39):1–9.

[4]. R. Churchill and L. Singh, “The evolution of topic modeling,” ACM Computing Surveys, vol. 54, no. 10s, pp. 1–35, 2022.

[5]. B. V. Barde and A. M. Bainwad, "An overview of topic modeling methods and tools", *Proc. Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, pp. 745-750, Jun. 2017.

[6]. J. Hu, X. Sun, D. Lo, and B. Li, “Modeling the evolution of development topics using dynamic topic models,” in 2015 IEEE 22nd international conference on software analysis, evolution, and reengineering (SANER), pp. 3–12, IEEE, 2015.

[7]. Yi F, Jiang Bo, Jianjun Wu (2020), “ Topic modeling for short texts via word embedding and document correlation”. IEEE Access 8:30692–30705.

[8]. D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent dirichlet allocation", *the Journal of machine Learning research*, vol. 3, pp. 993-1022, 2003.

[9]. C.-I. Hsu and C. Chiu, “A hybrid latent Dirichlet allocation approach for topic classification,” 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA), 2017.

[10]. Zheng L, Caiming Z and Caixian C. MMDF-LDA: “An improved multi-modal latent Dirichlet allocation model for social image annotation”. Expert Syst Appl 2018; 104: 168–184.

[11]. Liu Jun S (1994), “The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem”. J Am Stat Assoc 89(427):958–966

[12]. Phan X-H, Nguyen L-M and Horiguchi S., “Learning to classify short and sparse text & web with hidden topics from large-scale data collections”. In: Proceedings of the 17th international conference on world wide web, Beijing, China, 21–25 April 2008, pp. 91–100. New York: ACM.

[13]. J. Zhu, N. Chen, H. Perkins, and B. Zhang, “Gibbs max-margin topic models with data augmentation.” Journal of Machine Learning Research, vol. 15, no. 1, pp. 1073–1110, 2014.

[14]. Wang, C., Paisley, J. W., & Blei, D. M. (2011), “Online variational inference for

the hierarchical Dirichlet process". *Journal of Machine Learning Research*, *15*, 752–760.

[15]. Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," Journal of the American Statistical Association, vol. 101, no. 476, pp. 1566–1581, 2006.

[16]. Blei, D. M. and Lafferty, J. D., " Dynamic topic models", In Proceedings of the 23rd international conference on Machine learning, pp. 113–120, 2006.

[17]. J¨ahnichen, P., Wenzel, F., Kloft, M., and Mandt, S., "Scalable generalized dynamic topic models". In International Conference on Artificial Intelligence and Statistics, pp. 1427–1435. PMLR, 2018.

[18]. C. Wang, D. Blei, and D. Heckerman, "Continuous time dynamic topic models," arXiv preprint arXiv:1206.3298, 2012.

[19]. Wang, C.,Blei, D.,Heckerman, D. "Continuous time dynamic topic models". In: Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence, pp. 579–586, (2008)

[20]. Advanced Topic Modeling for Research Articles 2.0" dataset https://www.kaggle.com/datasets/abishek sudarshan/topic-modeling-for-research-articles/.