PAPER

Automatic label curation from large-scale text corpus

To cite this article: Sandhya Avasthi and Ritu Chauhan 2024 *Eng. Res. Express* **6** 015202

Engineering Research Express

PAPER

Automatic label curation from large-scale text corpus

RECEIVED
23 September 2023

REVISED
9 February 2024

ACCEPTED FOR PUBLICATION
15 February 2024

PUBLISHED
27 February 2024

Sandhya Avasthi[1,*] **and Ritu Chauhan**[2]

[1] Department of CSE, ABES Engineering College, Ghaziabad, India
[2] AI and IoT Lab, Center for Computational Biology and Bioinformatics, Amity University, Noida, UP, India
[*] Author to whom any correspondence should be addressed.

E-mail: sandhya_avasthi@yahoo.com **and** rituchauha@gmail.com

Keywords: automatic labeling, contextual word embedding, latent dirichlet allocation, topic modeling, topic coherence, topic label

Abstract

The topic modeling technique extracts themes based on their probabilistic measurements from any large-scale text collection. Even though topic modeling pulls out the most important phrases that describe latent themes in text collections, a suitable label has yet to be found. Interpreting the topics extracted from a text corpus and identifying a suitable label automatically reduces the cognitive load for the analyst. Extractive methods are used typically to select a label from a given candidate set, based on probability metrics for each candidate set. Some of the existing approaches use phrases, words, and images to generate labels using frequency counts of different words in the text. The paper proposes a method to generate labels automatically to represent each topic based on a labeling strategy to filter candidate labels and then apply sequence-to-sequence labelers. The objective of the method is to get a meaningful label for the result of the Latent Dirichlet Allocation algorithm. The BERTScore metric is used to evaluate the effectiveness of the proposed method. The proposed method generates good interpretative labels as compared to baseline models for topic words or terms automatically. The comparison with the label generated through ChatGPT API shows the quality of the generated label with the experiment performed on Four Datasets NIPS, Kindle, PUBMED, and CORD-19.

1. Introduction

In this competitive world, dependency on digital media has increased tremendously and so is the availability of data in structured and unstructured forms. The data in unstructured forms such as scientific articles, news, reviews, tweets, and blog posts is huge, and organization processes these documents daily to segregate documents or to gain valuable insights from the text. The professionals and analysts face difficulty in interpreting such a large collection of text documents. Any scientific articles, news, and social media text collection require efficient machine learning methods for correct and accurate data interpretability. A lot of time the data are unstructured and without labels as a result, unsupervised machine learning methods are used for their interpretability. A valuable technique is topic modeling which is used to extract terms from the text corpus that best describes the text collections. Some widespread techniques for identifying themes of any collections through topic modeling are Latent Dirichlet Allocation, Dynamic Topic model, Latent semantic analysis, and Correlated Topic model [1, 2]. These methods automatically discover the inherent themes from the document collections in the form of a group of terms or words known as topics where each topic represents word distribution over text collection. Topic modeling is a useful method for tracking the flow of information. As a result, it's a key component of visualization platforms, allowing users to peruse massive amounts of textual data. Based on a prioritized list of likely words, humans are normally allowed to interpret and understand them. Automatic topic labeling [3] was developed as a job of uttermost practical interest to handle this problem which provides an efficient method to offer a text label to describe the overall content in any given text collection and its various topic groups.

Although much study has been done in the domain of labeling text documents, automatic topic classification from texts remains an unsolved challenge. The manual labeling of topics was one of the first

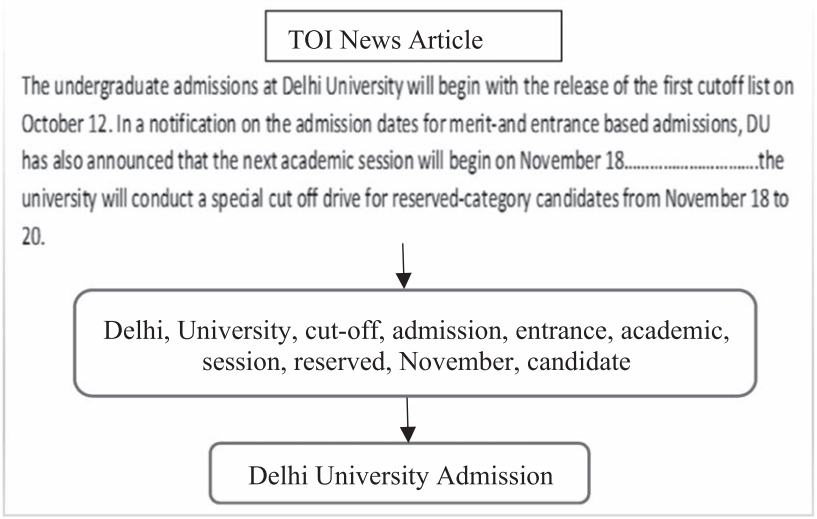

Figure 1. TOI news article example and best label to describe the article.

methods to label text. Some earlier approaches for labeling revolve around the manual labeling of topics. In labeling tasks when done manually, an expert user looks at a group of words that belong to a given topic and then chooses a suitable label that semantically captures the topics. This manual process requires a lot of human involvement and so it is a slow process. To keep up with the rising output and digitalization of texts, it is critical to develop automated ways for improving the search and mining of the large text corpus of literature. By highlighting the most important topics in a paper, key phrases give a succinct depiction. Numerous supervised algorithms utilize local context to predict the label for each token and outperform their unsupervised counterparts significantly. However, this strategy fails in the case of short text documents with a lack of context. In reality, supervised learning addresses the problem provided the data used for learning purposes is labeled already in categories like scientific, political, music, education, or sports. Most of the time the extracted terms provide useful insights and good intuitive meaning of a text but sometimes interpreting the terms becomes a very challenging task. Unsupervised learning [4, 5] is the second category of techniques. Extracting knowledge from data is what it entails. This information identifies common patterns or structures in the data samples based on frequency count or probability of words or their occurrence with other words in the neighborhood. The outcome of unsupervised learning is patterns that can be used to divide data into clusters or groups based on similarity or distance. The learning process is built on a set of metrics for computing observational similarity, correlation, and aggregation distance. Furthermore, the primary purpose of this strategy is to ensure that intra-group similarity is minimized while inter-group dissimilarity is increased.

The interpretation of discovered topics is vital to understanding and retrieving inherent themes from the documents. In addition, not all topics discovered are relevant to the text mining purpose so ranking the topics is another major challenge. For example, some top words extracted for a topic are: {*model neural similarity metric analyst office setting accuracy machine research Japan feature statistics*}. A good label for these groups of words would be' *Scientific Research*', '*Machine learning*', or '*Science*'. Even for a domain expert fully understanding this topic is difficult if he is ignorant about the document collections. When a user is faced with many overlapping topics where the top terms in topics are common, the whole process of choosing a suitable topic becomes even more complicated. Instead of providing the topic's original words, the labeling approach aims to produce a description that is closer to what a human would say about the text data.

Several research publications have presented approaches for labeling topics based on words, concepts, and images obtained from the text corpus [6, 7]. For example, for a news article in the Times of India, the top 10 terms are {Delhi, University, cut-off, admission, entrance, academic, session, reserved, November, candidate} by applying topic modeling. For these groups of words, there may be many suitable candidate labels to describe the article as shown in figure 1. Such a label makes it easier for a user to understand the documents and the topics better. However, in practice, having phrases or concepts as topic labels is insufficient since phrases are short and the labels do not express the theme of documents fully for easy comprehension by the user. Such an approach for curating labels for topics of documents is called the extractive approach where words or phrases can be found from several candidate labels identified through probabilistic measure or score. The extractive approach presents a limitation in selecting suitable interpretative labels for users. In this research paper, an efficient method without this limitation is proposed. The contribution of this paper is described as follows:

- To study different methods for topic labeling and identify their limitations.
- To propose an efficient method for automatic labeling of topical terms.
- The evaluation of the effectiveness of the generated topic labels.

There are five sections in this paper; section 1 introduces the topic labeling problem and its importance. Section 2 examines related research on different topic labeling methods and their limitations are discussed in detail. The proposed topic labeling methods and various steps are discussed in section 4. Section 3 states the problem, symbols used, and basic definitions of various terms used to describe the topic labeling context. Further, section 6 discusses the results obtained and gives an evaluation, and the conclusion and future work are given in section 7.

2. Literature review

The topic modeling technique processes large text corpus and is capable of discovering themes, hot spots, latest trends from the text corpus efficiently. The meaningful labeling of these words in each topic can help users understand the discovered topic easily. Automatically creating semantically meaningful labels for the text segment or group of words is the goal of topic labeling. Good research studies on the problem of finding topics and labels from text collections or corpora are presented here [8, 9] through an extensive study of literature. The top N-words in each topic are traditionally interpreted, or each topic is manually labeled using domain expertise and subjective interpretation [10].

2.1. Topic labeling

The purpose of topic labeling is to come up with a good label or title that will explain what makes a topic homogeneous to others [10]. Labeling the topics is a very important information extraction task when it comes to understanding the themes or content of a large document collection. A suitable label easily describes the theme of the document collection and can be used in the grouping of documents and interpretation of the overall content of the collection. When topic modeling is performed, extracted topics are represented through top n words or terms. These terms from topic modeling are ranked based on the conditional probability for that topic. Automatic topic labeling is a logical extension of Lau *et al* [11] paper, which selects the best topic terms and chooses any one term as a label. The method was based on a reranking approach to determine the top ten subject phrases based on how effectively each term reflects a topic when used in isolation [12] proposed a method to find good interpretable topic labels by exploiting the phrases or n-grams. Further, the problem of labeling topics is considered an optimization problem that involves Kullback Leibler (KL) divergence concepts, and how KL value minimization affects mutual information between a label and a topic model [11], perform labeling by using phrases proposed as a supervised learning method to rank the best candidate labels from extracted topics. In the work, the labels are a few noun chunks from Wikipedia articles and Top-5 terms from topics. In the paper [13], two efficient algorithms were proposed that assign labels automatically to each topic by utilizing relationships such as parent-child and siblings among topics.

Even though the academic community has focused heavily in the last decade on LDA extended versions by incorporating external knowledge of different kinds, it was noticed that LDA findings are still challenging to interpret for humans. The author [2], coined the expression 'reading tea leaves' to emphasize how difficult it is, to decipher results produced by topic models. To deal with such challenges, some natural language processing researchers proposed methods to improve the comprehensibility of LDA results [14, 15], such as topic tagging. One of the first publications to propose topic labeling as an optimization problem [3] considers labeling problem as an optimization problem that requires decreasing the Kullback-Leibler divergence between word distributions while maximizing mutual information between a label and a topic model. Further, the technique [16], advocated labeling subjects using external information. Several methods for associating a term or phrase with a topic based on the top-k terms have been proposed. External resources such as Wikipedia were used to find a title by [16] and [17], as well as some supervision. A few latest solutions utilize letter trigram vectors and word embeddings as explained in [18]. Another alternative is to employ numerous metrics to boost the chances of finding the proper phrase [19]. The phrase-based methods to generate labels are not adequate for interpreting the topics due to their short length. The text summaries as labels can be effective in many situations.

2.1.1. Limitations of various topic labeling methods

The limitations of various topic labeling methods are given in table 1, also the method is described briefly.

Some of the recent research proposes text annotation or label generation using GPTs, (Generative Pre-trained Transformers (GPT). The methods based on GPT have made breakthrough changes in automatic

Table 1. Research papers summary on topic labeling method and their limitations.

Paper	Proposed method	limitation
[3]	Uses Kullback-Leibler divergence and Mutual information for semantic relevance and generates candidate labels.	Inferior quality of labels, extractive approach
[20]	Based on finding most representative labels using graph entailment over phrases.	The ranking of candidate phrases can be improved by ranking strategies.
[21]	Proposed a method to identify junk topics from legitimate topics. Uses a 4-phase weighted approach.	The evolution of topics on text data is not considered.
[22]	Proposed a labeling framework based on the consideration that labeling is a k-nearest neighbor problem. Uses sensitive hashing technology.	The method utilizes labels from the topic-label database, which are used as standard labels.
[23]	The method is based on the structured data provided by DBpedia and uses graph centrality measures to evaluate.	It is still not a fully automated way to generate labels.
[24]	Uses a dynamic topic model, and proposed a semi-automatic transfer learning to find labels.	The data of UK-house speeches is used, limited only to the political domain.
[16]	Finds candidate label set from top-ranking terms and phrases from Wikipedia data.	Based only on phrases from Wikipedia articles.
[25]	Introduced a two-phase neural embedding approach utilizing graph-based ranking	The method does not provide redundancy control and semantic understanding
[26]	Proposes a filtering method to enhance topic labels, uses Hellinger distance to aggregate redundant topics	The method is good for large-scale data, also the filtering process can be improved.
[27]	Proposes automatic labeling methods-based BERT and word2-vec methods.	Limited only to customer complaint data and faces issues of data availability.
[28]	GPT −3-based approaches have been utilized for sequence and token level NLP tasks, and evaluation was done.	Limited to only generating labels for training data in sentiment analysis task not for generating interpretative labels for topic/word clusters.
[29]	This research shows the use of LLMs for text annotation tasks through zero-shot ChatGPT classification.	Suitable for supervised tasks, where some predefined labels have been chosen, hence cannot be used for unsupervised tasks.

labeling tasks for data in supervised machine learning tasks [28–30]. With the recent launch of ChatGPT in year 2022, GPT became popular for various natural language processing (NLP) tasks. GPT is based on deep learning models pre-trained on large text corpora and can solve problems such as sentiment analysis, text generation, language modeling, etc. However, the research shows significant bias in labeling results due to the bias present in the training data such as cultural bias and context-specific sensitivity [30].

2.2. Strategies based on n-grams

The paper [31], proposes a multi-strategy approach to generate labels for topics by applying strategic measures in different ways. A sequence of p words, also called candidate term t is given a score, this is also known as n-grams. These candidates are considered possible labels to represent a given topic $t = (w_1, w_2 \ldots w_p)$. The posterior probability $P\left(\frac{w}{z}\right)$ of word w for a topic z and the probability $P\left(\frac{z}{d}\right)$ of topic z for given document d can be found using the topic modeling algorithm, LDA. Three different topic labelers are proposed that are M-order [31], T-order, and Document-based labelers. The relevance score in the M-order strategy is calculated by equation (1), where Z is the set of extracted topics. The T-order labelers utilize the notion of 'term-hood' [32], where, a substring of a long term t' is called a short-term (t). In the computer science corpus, 'Neural' is a short term, commonly nested within the phrase 'Neural Network'. Here, in this case, ignoring the term 't' will improve the result. This T-order measure is given by equation (2).

$$M - order(t, z) = \sum_{i=1}^{p} \log \frac{p(w_i|z)}{\frac{1}{|Z|-1}\sum_{z' \neq z} p(w_i|z')} \tag{1}$$

$$T - order(t, z) = \begin{cases} 0 \; if \; t \; is \; short \\ \frac{1}{len(t)}.M - order(t, z) \quad otherwise \end{cases} \tag{2}$$

2.3. Labeling based on sequence-to-sequence method

Two recurrent neural networks (RNN) are used to implement the sequence-to-sequence labeling process, the first RNN becomes an encoder and the second one is used as a decoder. The job of the encoder is to take a given sequence of values $X = (x_1, x_2, \ldots x_T)$ as input and give a hidden representation $H = (h_1, h_2, \ldots h_T)$ that is passed on to the decoder for further processing. An RNN generates a sequence of outputs $y = (y_1, y_2, \ldots y_T)$ through the

Table 2. Variable and symbols used.

Symbol	Description
D	Number of documents
N	Total words
w	Words in corpus
w_i	ith word (as in the corpus)
z	Topic assignment
z_i	ith topic in documents
α	Dirichlet hyperparameter
β	Dirichlet hyperparameter
θ	Topic probability
ϕ	Probability of words given topic
σ^2, σ	Variance and standard deviation

equations (3) and (4):

$$h_t = \text{sigm}(W^{hx}x_t + W^{hh}h_{t-1}) \tag{3}$$

$$y_t = W^{hx}h_t \tag{4}$$

The output symbol can be predicted as $P(y_t \mid \{y_1, y_2,\ldots,y_{t-1}\}, X)$.

2.4. Topic embedding

The various topics extracted through the LDA method are represented by 10, 20, or top N words or terms. The topic labeling task can be improved by utilizing two types of term embedding methods to represent discovered topics from the text corpus [18]. The following equation (equation (5)) describes the term embeddings.

$$E_{Mean}(T) = \sum_{w \epsilon T} E_{w2v}(w) P_T(w) \tag{5}$$

The discovered topic from the text corpus is T; E_{Mean} is the average of word embeddings of all top N-words in respective topic T; E_{w2v} value is Word2Vec embedding of term w. In each topic 'T', the marginal probability of a specific word 'w' is $P_T(w)$.

3. Problem statement

Automatic topic labeling has become a useful way to give end users different ways to see topics, which makes it easier for them to understand the topic that the LDA model has found. The main types of topic labels made by topic labeling methods are sentences, summaries of the text, and pictures. The topic labeling methods use two common processes sentence scoring and sentence selection. In sentence scoring, the relevance score between a candidate sentence and a given topic is calculated, and the score is utilized to choose the best sentence. The second process 'sentence selection' usually is the process of selecting the sentences with high relevance from the ranked sentences by relevance score.

A collection of text documents is represented by a set of latent themes that have been retrieved through topic modeling. The topics are a multinomial distribution over words; the purpose is to generate phrases and sentences to use as labels for the latent topics' words. In this section, we explain useful definitions for use in the following section. The symbols are described in table 2, these symbols are frequently used in representing topic modeling and labeling concepts.

Text corpus: A text corpus is a vast, unstructured collection of text that is processed electronically and its main use is in statistical analysis, hypothesis testing, as well as checking occurrences and validating linguistic rules within a language.

Topic: Each topic θ is a probability distribution of words $\{p_\theta(w)\}$, where V is the vocabulary set, and we have $\Sigma_{w \epsilon V} p_\theta(w)=1$. To get latent topics from the text data Latent Dirichlet Allocation method is used.

Topic label: A suitable label 'l' to represent a topic θ, is a sequence of words that describes the latent meaning of the topic and is semantically meaningful.

High relevance: The text and summary should represent an entire document, and it should be semantically relevant to the topic. The quality of the summary is measured through relevance metrics.

High coverage: The summary should provide a complete idea about the whole document i.e.; it should give as much semantic information as possible. This measure is the same as the diversity requirement of multi-document summarization.

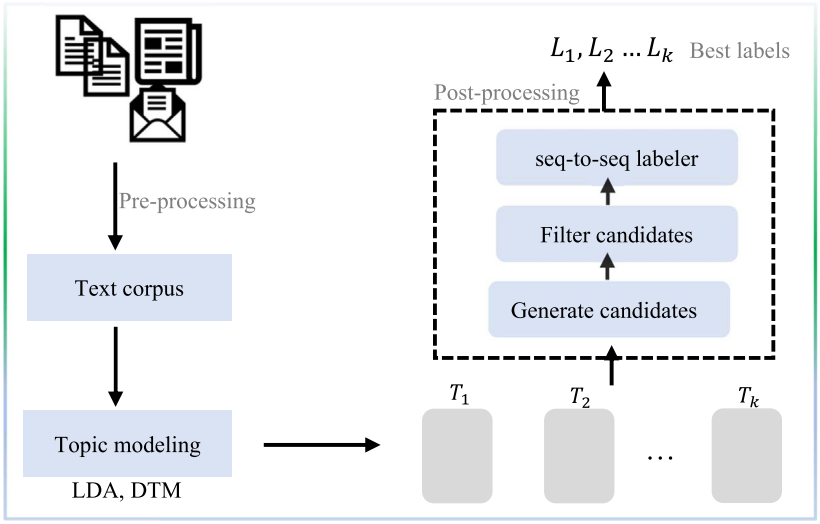

Figure 2. Auto-labeling framework for text documents.

For a text corpus, if we extract a set of topics θ, the labeling problem identifies the candidate label set, $L = \{l_1, l_2,\ldots,l_m\}$ according to their relevance score. The labeling task is a very significant step of text mining over large-scale text data. To solve the topic labeling problem the sequence of operation would be extraction of candidate labels, finding a relevance score, ranking or filtering candidate labels, and then generating labels from candidates having top relevance scores. A good topic label should be easy to understand, relevant, provide good coverage of topics, and be discriminative enough. In section 4, an automatic labeling approach is proposed based on filtering and sequence-to-sequence labeling concepts.

4. The proposed topic labeling method

The automatic labeling of a topic can naturally be accomplished by choosing the best word from all the terms/words. This approach of choosing the best word as a label does not express the topic fully which a multi-word label or a term that is not even there in the top 10 terms can do. For example, a group of terms {design, apparel, fabric, trend, Paris, …} can be labeled as 'Fashion week' but the term itself is not there in the top 10 words. After due consideration of the lack of expression of previous methods, the proposed method considers labeling as a three-step process: generating candidate labels, filtering using ranking strategies, and then applying the sequence-to-sequence labelers. The labels generated in this way have high relevance, coverage, and discrimination for all topics. The framework for the proposed method consists of three phases: finding candidate labels, using strategy to filter candidates, and applying sequence to sequence topic labelers to find suitable labels. Figure 2 illustrates the working of the proposed topic labeling method. The overall process starts with text data that goes through preprocessing steps such as stop word removal, URL removal, punctuation marks removal, and lemmatization. This text corpus is trained using LDA or DTM (topic modeling technique) to produce k number of topics and groups of words in each topic. After topic modeling topic labeling known as a postprocessing step is performed to give a suitable label describing each group of words in a topic 'k'.

4.1. Generate and filter candidate labels

After performing topic modeling on various text corpus, the topic is extracted which is a multinomial distribution over the entire text document collection. Here each topic is distributed over words in the text. To find candidates for the next step, we find out the top terms based on the probabilities from the topic modeling method used in the process. In general top-10 topics, and terms can be used to query restricted Google searches. This process returns the set of titles through the Google search engine to be used as the initial set of primary candidate labels.

The next step is chunking, which is done using OpenNLP chunker to find out all noun chunks. Further n-grams components are generated from the set of noun chunks and then the n-grams are removed based on T-order and M-order measures.

4.2. Sequence-to-sequence labelers

The filtered candidate from previous steps becomes the input to the encoder in sequence-to-sequence labeler and, is used in embedding layer mapping with more than 300-dimensional embeddings. In the next step maps with 200 units of bidirectional GRU (Gated Recurrent Unit). The choice of Bi-directional GRU is based on the fact that GRUs are computationally efficient and have less complicated structures as compared to LSTM. While processing large-scale text corpus which generates a long sequence of inputs, GRU is preferred over LSTM [33]. The GRU scans input terms in original order in the forward direction. In the backward phase, GRU takes the terms in reverse order. The output by GRU for both forward and backward directions is given by equations (6) and (7).

$$hf_t = GRU(x_t, h_{t-1}) \tag{6}$$

$$hb_t = GRU(x_t, h_{t-1}) \tag{7}$$

$$h_t = [hf_t; hb_t]$$

The labels are predicted word after word in the decoding phase, and at the 't 'timestamp, the decoder calculates the value in the hidden state s_t.

$$s_t = GRU(y_{t-1}, s_{t-1}, c_t) \tag{8}$$

$$c_t = \sum_{j=1}^{T_x} \alpha_{tj} h_j \tag{9}$$

In equation (8), y_{t-1} is the predicted word from the previous step, and feedback to predict the next word, whereas s_{t-1} is the hidden state. The c_t is the context vector at time t (equation (9)), which is used by the decoder. The context vector (c_t) is equal to a weighted sum of weights α_t over encoder, hidden states utilize an attention mechanism [34].

4.3. Applications of topic labeling

Topic labeling can be used to understand scientific documents because these documents have specialized use of words and the writing style is formal. Many common words can have different meanings when used in technical and scientific contexts. Efficient automatic labeling will help the analyst in better organization of documents and interpretation of development and trends in research. A good analysis of scientific documents is useful in identifying innovation and novelty in research. Applying topic modeling and labeling to abstracts of scientific collections of various domains might provide useful insights and discoveries.

Topic labels can also be used to analyze document sentiments and then the organization of documents based on the sentiments of text data. The text data comes from social media and blog posts. Interpreting historical news articles and how a specific theme has changed over time is one very important use of topic modeling and labeling. Some other applications include business analysis, product analysis, social media monitoring, search engine optimization, and brand monitoring. In many businesses, monitoring what people discussing about a specific brand gives insights using topic identification.

5. Text preprocessing and experiments

In this research, experiments were performed using chosen datasets on Python IDE. Python library pandas, matplotlib, and Gensim are utilized to perform topic modeling tasks. The LDA and DTM method from the Gensim package is primarily used in performing topic modeling and extracting words from each topic from the text corpus. The parameters considered for it were taken from previous studies on this topic, the topic number K was fixed at 20 and 30.

5.1. Baseline models

To evaluate the effectiveness of the proposed method two baseline methods for topic labeling tasks have been used. Baseline 1(Top-2 Label or Top-3 Label) generates labels utilizing either top-2 words or top-3 words based on their marginal probabilities order in a specific topic. After that chosen words are concatenated to produce a label for that topic. This method is an extractive approach and only assigns labels from a restricted set.

Baseline 2 generates labels from the Wikipedia text data pool mainly consisting of article titles. The model works in two steps; in the first step candidate labels are selected and then selected labels are re-ranked to their semantic similarity to the topic terms/words. Experimental result shows proposed labeling method generates good descriptive and interpretative labels for topics (Refer to section 6 for results samples).

5.2. Dataset and preprocessing

To assess our method's domain/genre independence, we conducted topic labeling tests utilizing document sets from four separate domains. For all four datasets used here, the preprocessing goes through steps like tokenization, lemmatization, and stop word removal. The terms having a frequency of less than five were removed from the vocabulary.

Kindle reviews: the dataset is of Amazon product reviews available at Kaggle [21] with more than 60000 reviews of books. These reviews are posted by customers who are buying and reading books.

PUBMED: A collection of 10000 PubMed biomedical abstracts published between 2010 to 2018. The size of each abstract is 150–250 words [9].

NIPS: This dataset is a collection of Neural Information Processing Systems conference papers year 1987 to 2016 [35]. This is a collection of scientific papers presented during the conference.

CORD19: CORD-19 dataset of scientific papers metadata, is prepared in response to the COVID-19 pandemic during the year 2020 [36, 37]. For the experiment purpose, the abstract and title from the table are considered.

For the topic modeling process, a bag-of-words representation for each text document is created. The modeling process was performed for two different values for k (k=30 and k=50), where k is the number of topics. In the filtering process, the average PMI score for each topic was calculated and filtered for all topics with a PMI score of less than 0.421. After this process, a few topics having less than 5 terms from the top 10 terms are default values and are also filtered to have more coherent topics [38]. Note that for evaluation purposes the traditional method of calculating precision and recall is not appropriate due to the real value rating of labels.

5.3. Hyperparameter setting

For the preprocessing of text corpus, implementation purpose, and evaluation task we used Python and Google Collaboratory. The Genism module in Python was used to discover latent topics using the LDA and DTM methods. All the parameters were. Set as per previous studies on the topic. The main parameters for setting are α and β, learning rate, and the number of topics k. The hyperparameter tuning was done by taking random samples of a size of 50 or more.

Hyperparameter setting and fine-tuning during the modeling process are crucial in improving the efficacy of models, by selecting the optimal hyperparameter's overall result, the generated topical words can significantly be improved. For instance, for a large value of k such as 60 or 80, the topic coherence value reduces.

5.4. Training and test data

To generate a label using the proposed model, text data is required to train the model. The LDA and DTM methods provide the required set of topics (consisting of terms according to their probabilistic occurrence in documents) and associated labels. All four datasets described here were used in training. The training and test set was created in an 80:20 ratio. The pair of topics and labels become input to the sequence-to-sequence neural network model. The technique used here is the BERT method. In dataset1 [21], as it is a review text, there is an absence of the Title of each review/document, and labels were generated with the help of human annotators. For the training of the proposed method for the labeling task, the top 20 words from each topic ranked by LDA or DTM were taken as our topic terms to form terms and label pairs. For the other three datasets of the abstract of articles, article titles were used as labels, and the top 20 words from each article as returned by LDA modeling were used as terms. In the training phase of sequence-to-sequence labeling using RNN, the sigmoid function was used as an activation function and Glorot-Uniform distribution to get samples. This process utilized pre-trained contextual embedding Word2Vec.

5.5. Metric to find similarities between texts

Cosine Similarity is a widely used metric for measuring the textual similarity between two papers of any size. A word is a vector, and text documents are stored in n-dimensional vector space. Mathematically, the cosine similarity metric is defined as the cosine of the angle created by two n-dimensional vectors projected in a multidimensional space. The mathematical expression (equation (9)) for the cosine similarity of two non-zero vectors is as follows:

$$S = \cos\theta = \frac{\boldsymbol{A.B}}{\|A\|\,\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} {A_i}^2}\sqrt{\sum_{i=1}^{n} {B_i}^2}} \tag{9a}$$

To measure similarity, find out the word embedding and then directly measure any two-word similarity. In table 3, the neighbors of the words 'Late' and 'Algorithm' are shown based on the Skip-Ngram model and similarity score value.

Table 3. Cosine similarity values of different words.

Neighbors of the word 'Late'	Cosine similarity (Skip-Ngram)	Neighbors of 'Algorithm'	Cosine similarity (Skip-Ngram)
later	0.748	online	0.924
wait	0.741	scheme	0.906
Postponed	0.719	complexity	0.833
overloaded	0.683	base	0.813
annoyed	0.661	apply	0.804
early	0.637	distribute	0.803
apologies	0.622	theoretical	0.800

Table 4. Statistics of standard dataset used in research.

Dataset/domain	Size (Document size)	Unique values	Description
Kindle /Amazon	982619	982267	Review text posted by customers
PUBMED	9719	9682	Abstracts of articles on PUBMED
NIPS	7241	7237	Abstracts of conference papers
CORD-19	57633	46410	Abstracts of the article of COVID-19 related papers

5.6. Evaluating labels

The quality of the generated labels from sequence-to-sequence labeling was assessed using BERTScore, a metric for calculating the similarity between predictions and references based on contextual embeddings that have been found to correspond with human assessments [39–41]. Because BERTScore does not require exact matches between predicted and gold-standard labels, it can identify relevant label terms that are missing from the gold labels. The usage of contextual embeddings improves BERTScore performance. Between generated label 'l' and reference labels $\{l_{r1}, l_{r2}, \ldots, l_{rn}\}$, the BERTScore can be computed using equation (10). In equation (10), l_t refers to the generated label for topic 't'. Symbol l_{ri} represents ith reference label taken from standard gold labels. 'T' is the total number of tokens in the sequence.

$$score_t = \max_{i=[1,\ldots,n]} BERTScore(l_t, l_{ri}) \tag{10}$$

The mean score overall of the topics is the final score (equation (11)).

$$final_{score} = \frac{1}{T}\sum_{t=1}^{T} score_t \tag{11}$$

The BERTScore is easy to use and resolves several limitations of commonly used metrics [42]. The Precision score is calculated by matching each token in the candidate set to each token in the reference set.

5.7. Using ChatGPT 3.5 for comparison

Amazon MTurk gets label generation through crowd-workers efficiently but comes with significant cost, especially in the case of large-scale text data, it is even more expensive [29, 30]. Therefore, the Open AI application ChatGPT 3.5 API is utilized to show the comparison of results from the proposed labeling method. ChatGPT API is fast and efficient for problems like text annotation and language understanding and can be used in NLP applications. The labels produced by the proposed labeling method are compared with the same set of groups of words from various topics in the case of all four experimental datasets. For many samples, the relevance score is comparable in both results from the proposed method and ChatGPT.

6. Results and analysis

First topic modeling is performed, in the next step candidate label generation takes place, then the sequence-to-sequence method returns the best label describing the groups of terms. Note that, for evaluation purposes, the traditional method of calculating precision and recall is not appropriate due to the real value rating of labels. The summary of all four datasets used for the experiments is given in table 4. Choosing k = 30, the number of topics for all datasets except CORD19 gives distribution over words for k = 30 topics. Table 5 provides a sample of groups of words in topic numbers 17,14, 22, and 15 for CORD-19, NIPS, and Amazon review datasets

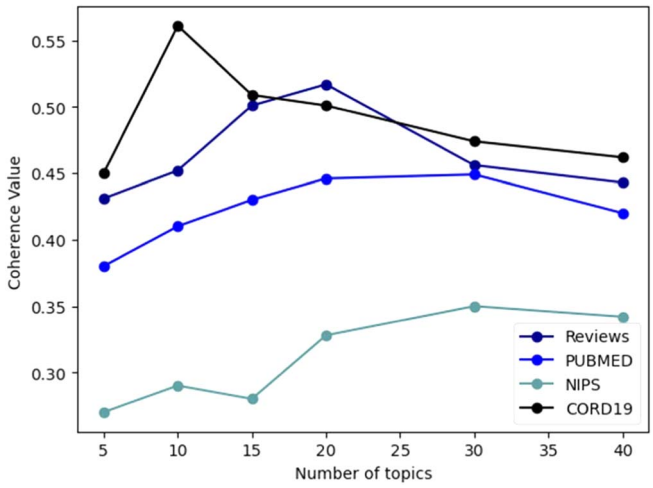

Figure 3. Coherence score for different values of 'k'.

Table 5. Sample topic for CORD-19 and NIPS.

Dataset	Topic #	Top 10 words
CORD-19	T-17	Initial, unknown, discovery, modeling, filter, modular, resolution, gradually, shadow, cardiac
NIPS	T-14	Image, feature, object, use, classification, scene, representation, learn, training, dataset
Kindle/Amazon	T-22	Movie, kid, child, adult, wizard, fog, wood, young, church, gavin
PUBMED	T-15	Brain, disorder, induce, activity, effect, memory, increase, expression, may, genetic

respectively. The words in groups have a high-frequency count as compared to other words in collections. Figure 3 shows coherence values for different values of 'k', the diagram shows k=30 as the optimum value for k (number of topics) except for the CORD19 dataset. For the CORD19 dataset the optimal value for 'k' is 10 because the coherence metric is the highest for it.

Overall computational complexity depends on the Topic Modeling step for a large text corpus, and then applying the automatic labeling step (Proposed method). For the topic modeling primarily LDA technique is used in the experiment. The memory requirement is proportional to the size of text corpus, the requirement grows as corpus size grows. The time complexity of LDA is O(NK) where N is the total number of words in the text corpus and K is the number of topics. The sequence-to-sequence labeling step depends on the size of word embeddings. In general, for sequence size 'n' and contextual embedding size 'd', the time complexity is estimated to be $O(nd^2)$.

Table 6 shows samples from all four datasets used for the automatic topic labeling and their average score calculated for various label candidates using the proposed method for labeling as given per equation (11). Table 6, shows the labels generated through ChatGPT API, and comparison shows the labels are significantly describe the theme of each topic using top 10 words. All the samples taken is a group of 10 words from a specific topic. Among the various label candidates, the label 'Voice-activation' is considered good based on the score for specific sample from Kindle dataset, but its ChatGPT label 'InterActZone' is not so descriptive of the words. Similarly, from sample terms extracted from the PUBMED dataset 'Blood-cancer' is returned as the most suitable label based on the calculated score. For samples from the NIPS dataset, 'Convolution Neural Network' is the most suitable label to describe the group of words. For the last sample word group from PUBMED data both the labels 'Clinical-Study' and 'Clinical Trial' generated through proposed method and ChatGPT API is equally good.

The model generates labels by utilizing the top 10 terms and 10 additional terms. The training performed on various datasets gives good precision, recall, and F-measure scores. Table 7 illustrates the comparison of the proposed method with the other two Baseline1 and Baseline2 methods. The proposed method gives the best score on all three metrics Precision, Recall, and F-measure for BERTScore. The illustration of the proposed method as compared to the other two baseline methods based on Precision, Recall, and F-measure is shown in figure 4 for the NIPS dataset. The experiment demonstrated the goodness of generated labels through Precision, Recall, and F-measure higher values for all four datasets used here. Figure 5 gives a comparison of the precision (ratio of good labels from the total results of the returned labels through the chosen method) value of Baseline 1, and Baseline 2 with the proposed method on three datasets used in modeling.

Table 6. Datasets, Topic samples, labels, and the average relevance score for each label candidate.

Dataset/domain	Group of words from Topics	Label candidate	Average score	Labels from ChatGPT API
Kindle Reviews Amazon	Tap, echo,button, voice, add, activate,push, command, anywhere, well	Voice-activation	2.61	InterActZone
Abstracts/PUBMED	CNN, patient, MRI, leukemia, nanotube, machine, muscle, treatment rendering blood	Blood-cancer	2.45	MRI-Based Leukemia Detection
Abstract/NIPS	Convolutional, generative, model, variational, inference, recurrent, gaussian, hidden deep layer	Convolution-neural network	2.56	Dynamic Fusion
Abstracts/CORD-19	sars-cov, sars, coronavirus, ace2, coronaviruses, ncov, covid-19, Wuhan, spike, sars-cov-2	COVID-19	2.38	Coronavirus
Kindle Reviews Amazon	world, tale, dragon, war, novel, magic, adventure, earth, ship, fairy_tale	Fiction-Fantasy	2.58	Fantasy Literature
Abstract/NIPS	learn, grammar, set, node, model, use, algorithm, kernel, approach, figure	Linguistics	2.51	Computational Linguistics
Abstracts/PUBMED	treatment, patient, study, intervention, group, pain, outcome, effect, follow, week	Clinical-Study	2.43	Clinical Trial

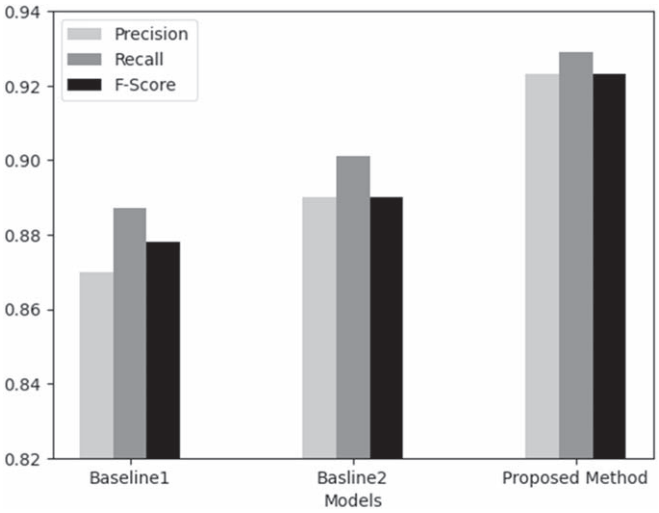

Figure 4. Precision, Recall, and F-measure for the proposed method for the NIPS dataset.

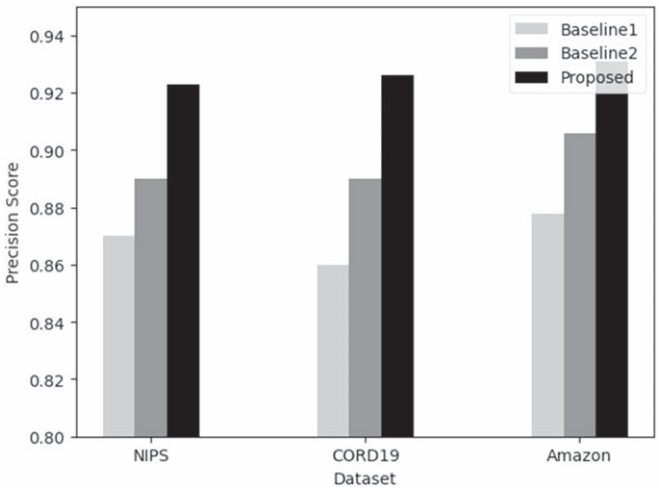

Figure 5. Precision for Baseline1, Baseline2, and Proposed method for three datasets NIPS, CORD19, and Amazon reviews.

Table 7. Labels using the Proposed method and BERTScore value.

Method	BERTScore(→)	P	R	F
Baseline1 (Top-2 or Top-3)	Abstracts/NIPS	0.87	0.887	0.878
	Abstracts/CORD-19	0.86	0.89	0.87
	Review/Amazon	0.878	0.90	0.88
Baseline2(Wikipedia)	Abstracts/NIPS	0.89	0.901	0.89
	Abstracts/CORD-19	0.89	0.913	0.902
	Review/Amazon	0.906	0.92	0.896
Proposed Method	Abstracts/NIPS	**0.923**	**0.929**	**0.923**
	Abstracts/CORD-19	**0.926**	**0.931**	**0.926**
	Review/Amazon	**0.931**	**0.942**	**0.928**

7. Conclusion

Topic modeling over time turned out to be a very interesting field of study. It has many applications in both machine learning and text mining, and it can be used for both. Even though there have been a lot of studies about

how to use these models, no method could provide a quick and stable automatic label generation process for extracting useful labels for representing the content of the subject. Topic modeling techniques discover a diverse range of topic terms automatically from a large text corpus but the clear and precise interpretation is difficult for a user. The applicability of these models to real-world applications is reduced if they are not labeled appropriately. This problem can be solved by giving a suitable label to each topic. Because of recent and upcoming data growth forecasts, rapid and scalable solutions would be favored above those requiring a significant number of resources.

The research paper describes the problem of topic labeling and provides a solution to find the topic label for generated topics after topic modeling of a text corpus. The proposed method is based on the filtering process and applies the sequence-to-sequence approach to generate phrase and sentence labels. The phrase and sentence label produced by the proposed method is highly relevant and has a good interpretation of the topics from the text. The BERTScore and cosine similarity measures are used to evaluate the goodness of generated labels. The results is compared with generated labels using ChatGPT 3.5 that shows proposed method efficiency and accuracy in case of large text corpus. In the future, the method can be extended to a multilingual text corpus and can be applied to automatically generate the labels for the topic terms.

Data availability statement

The data cannot be made publicly available upon publication because they are not available in a format that is sufficiently accessible or reusable by other researchers. The data that support the findings of this study are available upon reasonable request from the authors.

ORCID iDs

Sandhya Avasthi https://orcid.org/0000-0003-3828-0813
Ritu Chauhan https://orcid.org/0000-0001-9905-2270

References

[1] Blei D M 2012 Probabilistic topic models *Commun. ACM* **55** 77–84
[2] Chang J, Gerrish S, Wang C, Boyd-Graber J L and Blei D M 2009 Reading tea leaves: How humans interpret topic models *Proceedings of the 22nd International Conference on Neural Information Processing Systems* 288–96
[3] Mei Q, Shen X and Zhai C 2007 Automatic labeling of multinomial topic models *In Proc. of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 490–9
[4] Shi C, Chen Q, Sha L, Li S, Sun X, Wang H and Zhang L 2018 Auto-dialabel: Labeling dialogue data with unsupervised learning *In Proc. of the 2018 Conference on Empirical Methods in Natural Language Processing* 684–9
[5] Avasthi S, Chauhan R and Acharjya D P 2021 Techniques, applications, and issues in mining large-scale text databases *In Advances in Information Communication Technology and Computing* (Springer) pp 385–96
[6] Aletras N and Stevenson M 2013 Representing topics using images *HLT-NAACL In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 158–167
[7] Hulpus I, Hayes C, Karnstedt M and Greene D 2013 Unsupervised graph-based topic labelling using dbpedia *In Proceedings of the Sixth ACM International Conference on Web Search and Data Mining* 465–474
[8] Blei D M, Ng A Y and Jordan M I 2003 Latent dirichlet allocation *Journal of Machine Learning Research* **3** 993–1022
[9] Avasthi S, Chauhan R and Acharjya D P 2021 Processing large text corpus using N-gram language modeling and smoothing *In Proceedings of the Second International Conference on Information Management and Machine Intelligence* (Springer) 21–32
[10] Wang X and McCallum A 2006 Topics over time: a non-markov continuous-time model of topical trends *In Proc. of the 12th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD 2006)* 424–33 Philadelphia, USA
[11] Lau J H, Grieser K, Newman D and Baldwin T 2011 Automatic labeling of topic models *Association for Computational Linguistics. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* vol 1
[12] Mei Q, Shen X and Zhai C X 2007 Automatic labeling of multinomial topic models *Proc. of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 1536–1545 ACM
[13] Mao X-L, Ming Z-Y, Zha Z-J, Chua T S, Yan H and Li X 2012 Automatic labeling hierarchical topics *In Proc. of the 21St ACM International Conference on Information and Knowledge Management* 2383–6 ACM
[14] Lau J H, Newman D and Balwin T 2014 Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. *In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics* 530–539
[15] Blei D and Lafferty J 2009 Topic Models *Text Mining: Classification, Clustering, and Applications* (New York: Taylor and Francis) (https://doi.org/10.1201/9781420059458)
[16] Lau J H, Grieser K, Newman D and Baldwin T 2011 Automatic labelling of topic models *In Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* 1536–45
[17] Bhatia S, Lau J H and Baldwin T 2016 Automatic labelling of topics with neural embeddings arXiv: 1612.05340
[18] He D, Ren Y, Khattak A M, Liu X, Tao S and Gao W 2021 Automatic topic labeling using graph-based pre-trained neural embedding *Neurocomputing* **463** 596–608
[19] Gourru A, Velcin J, Roche M, Gravier C and Poncelet P 2018 United we stand: Using multiple strategies for topic labeling *In Int. Conf. on Applications of Natural Language to Information Systems* (Springer) 352–63

[20] Mehdad Y, Carenini G, Ng R and Joty S 2013 Towards topic labeling with phrase entailment and aggregation *In Proc. of the 2013 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 179–89
[21] AlSumait L, Barbará D, Gentle J and Domeniconi C 2009 Topic significance ranking of LDA generative models *In Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (Springer) 67–82
[22] Mao X L, Zhao Y J, Zhou Q, Yuan W Q, Yang L and Huang H Y 2016 A novel fast framework for topic labeling based on similarity-preserved hashing *In Proc. of COLING 2016, the 26th Int. Conf. on Computational Linguistics: Technical Papers* 3339–48
[23] Hulpus I, Hayes C, Karnstedt M and Greene D 2013 Unsupervised graph-based topic labelling using dbpedia *In Proc. of the Sixth ACM International Conference on Web Search and Data Mining* 465–74
[24] Herzog A, John P and Mikhaylov S J 2018 Transfer topic labeling with domain-specific knowledge base: an analysis of UK House of commons speeches 1935-2014 arXiv:1806.00793
[25] He D, Ren Y, Mateen Khattak A, Liu X, Tao S and Gao W 2021 Automatic topic labeling using graph-based pre-trained neural embedding *Neurocomputing* **463** 596–608
[26] Tarifa A, Hedhili A and Chaari W L 2020 A Filtering process to enhance topic detection and labelling *Procedia Computer Science* **176** 695–705
[27] Tang X, Mou H, Liu J and Du X 2021 Research on automatic labeling of imbalanced texts of customer complaints based on text enhancement and layer-by-layer semantic matching *Sci. Rep.* **11** 1–11
[28] Ding B, Qin C, Liu L, Chia Y K, Joty S, Li B and Bing L 2022 Is gpt-3 a good data annotator? arXiv: 2212.10450
[29] Gilardi F, Alizadeh M and Kubli M 2023 Chatgpt outperforms crowd-workers for text-annotation tasks arXiv: 2303.15056
[30] Yenduri G, Ramalingam M, Chemmalar Selvi G, Supriya Y, Srivastava G, Maddikunta P K R and Athanasios V 2023 Generative pre-trained transformer: A comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions arXiv:2305.10435
[31] Gourru A, Velcin J, Roche M, Gravier C and Poncelet P 2018 United we stand: Using multiple strategies for topic labeling *In Int. Conf. on Applications of Natural Language to Information Systems* (Springer) 352–63
[32] Lauriola I, Lavelli A and Aiolli F 2022 An introduction to deep learning in natural language processing: Models, techniques, and tools *Neurocomputing* **470** 443–56
[33] Kumar D and Aziz S 2023 *Performance Evaluation of Recurrent Neural Networks-LSTM and GRU for Automatic Speech Recognition. In 2023 Int. Conf. on Computer, Electronics & Electrical Engineering & their Applications (IC2E3)* (IEEE) 1–6
[34] Soydaner D 2022 Attention mechanism in neural networks: where it comes and where it goes *Neural Computing and Applications* **34** 13371–85
[35] Amazon Reviews: Kindle https://kaggle.com/bharadwaj6/kindle-reviews
[36] Wang L L, Lo K, Chandrasekhar Y, Reas R, Yang J, Eide D and Kohlmeier S 2020 *Cord-19: The covid-19 open research dataset*
[37] Perrone V, Jenkins P A, Spano D and Teh Y W 2017 Poisson random fields for dynamic feature models *Journal of Machine Learning Research* **18** 1–45 arXiv:1611.07460
[38] Avasthi S, Chauhan R and Acharjya D P 2022 Topic modeling techniques for text mining over a large-scale scientific and biomedical text corpus *International Journal of Ambient Computing and Intelligence (IJACI)* **13** 1–18
[39] Avasthi S, Prakash A, Sanwal T, Tyagi M and Yadav S 2023 Tourist reviews summarization and sentiment analysis based on aspects *In 2023 13th Int. Conf. on Cloud Computing, Data Science & Engineering (Confluence) (Piscataway, NJ)* (IEEE) 452–6
[40] Zhang T, Kishore V, Wu F, Weinberger K Q and Artzi Y 2019 Bartscore: Evaluating generated text as text generation *Advances in Neural Information Processing Systems* **34** 27263–77
[41] Mao J and Liu W 2019 A BERT-based approach for automatic humor detection and scoring *In IberLEF@ SEPLN.* **2421** 197–202
[42] Chan C R, Pethe C and Skiena S 2021 Natural language processing versus rule-based text analysis: Comparing BERT score and readability indices to predict crowdfunding outcomes *Journal of Business Venturing Insights* **16** e00276