Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Computers in Biology and Medicine 138 (2021) 104920

Contents lists available at ScienceDirect

Computers in Biology and Medicine

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

PAN-LDA: A latent Dirichlet allocation based novel feature extraction model for COVID-19 data using machine learning

Aakansha Gupta, Rahul Katarya *

Big Data Analytics and Web Intelligence Laboratory, Department of Computer Science & Engineering, Delhi Technological University, New Delhi, India

ARTICLE INFO

Keywords:
COVID-19
Latent dirichlet allocation
Collapsed gibbs sampling
Data mining
Feature extraction
Backpropagation

ABSTRACT

The recent outbreak of novel Coronavirus disease or COVID-19 is declared a pandemic by the World Health Organization (WHO). The availability of social media platforms has played a vital role in providing and obtaining information about any ongoing event. However, consuming a vast amount of online textual data to predict an event's trends can be troublesome. To our knowledge, no study analyzes the online news articles and the disease data about coronavirus disease. Therefore, we propose an LDA-based topic model, called PAN-LDA (Pandemic-Latent Dirichlet allocation), that incorporates the COVID-19 cases data and news articles into common LDA to obtain a new set of features. The generated features are introduced as additional features to Machine learning (ML) algorithms to improve the forecasting of time series data. Furthermore, we are employing collapsed Gibbs sampling (CGS) as the underlying technique for parameter inference. The results from experiments suggest that the obtained features from PAN-LDA generate more identifiable topics and empirically add value to the outcome.

1. Introduction

The Coronavirus Disease 2019 (COVID-19) outbreak was originated in Wuhan, China, and has rapidly spread worldwide. On March 11, 2020, the WHO has declared a pandemic. There is a consequent increase in confirmed cases and deaths worldwide due to coronavirus, which has instantly led to more information on social media platforms. Moreover, the availability of a massive amount of everyday data on online platforms has built a relationship between ongoing events and online data. Therefore, reducing the online textual data into topic distribution increase the value of this relationship. The numerical data can also be extended to use for technical analysis to extract more valuable information about the event with time.

In recent years, the rapid growth in powerful text mining techniques entails a significant change in the research of extracting information and prediction. These techniques enhance the ongoing research efforts; improve the efficiency and speed of existing approaches. After the emergence of text mining approaches, the research for extracting information from unstructured textual data has been taken into account more often.

While addressing unstructured textual data, one approach is to develop specialized search engines. For example, several researchers developed search engines in order to find the data related to a particular interest from the COVID-19 related publications across scientific disciplines [1–3]. However, such engines are limited to use numerical data, keeping the textual data aside. Some studies focused on the textual information leaving behind the numerical data. For example, A. Khadjeh Nassirtoussi, S. Aghabozorgi, T. Ying Wah et al. [4] used only the news article headlines for text mining. Another study mined only the new articles and forecasted Argentine and Brazilian currency markets' movement, which employed topic clustering, sentiment analysis, and regression analysis [5].

Another typical way for text mining is using the statistical model, such as topic models, to comprise enormous textual information based on the topic distribution of documents [6,7]. With the emergence of text mining methods, the research on topic modeling usually focused on textual data to obtain the results. While working with the unstructured textual data, there are numerous instances where the topic models, especially, LDA model, were utilized or improved to discovers topics from the online text [8–10]. All these studies incorporate textual data for the evolution of topics in different ways. Although the traditional latent Dirichlet allocation (LDA) model has been studied well, these methods do not consider numerical data. Recent studies have employed topic modeling methodologies to anticipate prices using unstructured data such as broadcast news and social media data [11]. In addition to online text mining for time-series predictions, researchers have widely applied

* Corresponding author.
E-mail addresses: aakanshagupta.74@gmail.com (A. Gupta), rahuldtu@gmail.com (R. Katarya).

https://doi.org/10.1016/j.compbiomed.2021.104920
Received 23 July 2021; Received in revised form 5 October 2021; Accepted 5 October 2021
Available online 12 October 2021

sentiment analysis for processing unstructured text. For example, X. Li, W.Shang, and S.Wang have considered news sentiment text features and grouped them according to their topics using topic modeling to increase the prediction accuracy [12,13]. However, none have defined a topic model based upon the combination of statistical data and textual data for prediction. Therefore, depending on which aspect is used, there is a scope for improvement. Our approach is to incorporate the time series numerical data along with online textual data in the standard LDA to extract better topics. Accordingly, we introduced PAN-LDA, a modified LDA, to take COVID-19 cases numerical data into account to improve the feature extraction. The topic features are then presented in machine learning algorithms to have an advantage from our PAN-LDA model.

Various statistical methods have been developed, including ML techniques and time series methods, to track and predict the events' evolution with time. Regardless of the ongoing re-rise and prevalence of conventional ML algorithms, boosting strategies are still increasingly valuable for a medium dataset as the preparation time is generally extremely quick, and they do not require quite a while to tune its parameters. In recent years, gradient boosting based ML methods such as Extreme Gradient Boosting (XGBoost) [14] and Light Gradient Boosting Machines (LightGBM) [15] are applied by some researchers for a robust prediction of future events in different research fields [16–19].

Generally, in the data mining task, the main components in achieving the outcome are data collection, data preparation, modeling, and evaluation. In the data preparation process, feature extraction models like LDA provide new lower dimension features. Accordingly, PAN-LDA is a topic model for extracting the features during the data preparation stage. Therefore, our focus will be on describing the flow of this model in various phases of data mining and performing experiments to understand the future benefits of the extracted features. In the modeling phase, evaluating the PAN-LDA model's performance is done by using ML algorithms. We intend to follow advanced ML algorithms, such as XGBoost and LightGBM, as they are fast, and their complexity will improve the prediction performance. As a summary, the significant contributions of our study are as follow:

- We proposed a latent Dirichlet allocation (LDA) based model, PAN-LDA, to create a new set of features from integrating texts from news articles and COVID-19 case data.
- The features from our model served as an additional feature to Machine learning algorithms for outbreak case prediction.
- The developed model, PAN-LDA, employed collapsed Gibbs sampling (CGS) as the underlying algorithm for inference in topic modeling.
- We provided framework details for applying our model, PAN-LDA, in text mining, even though our model focused on extracting features in the data preparation phase.
- We used ML algorithms to justify the benefits of the features extracted from the proposed model.
- Our proposed model delivered superior results for all ML algorithms and generated more identifiable topics than other baseline methods.

To the best of our knowledge, this is the first effort to define a topic model based on the integration of news articles and the data of daily new cases of coronavirus. We collect the COVID-19 dataset of global news articles archived by 'Aylien' [20] and the data of corona cases published by 'Our World in Data' [21]. Next, we develop a new topic model to generate structured information and produce better latent topics from the collected data. We show that our model 1) uncovers a significant number of more identifiable topics than LDA, 2) the features obtained from PAN-LDA empirically add advantage for prediction 3) and performed significantly well other baseline approaches.

The remaining paper is arranged as follows: Section 2 covers the related work. Section 3 introduces the PAN-LDA model and describes the procedure for topic inference for new documents, along with the framework to apply our model in text mining. Section 4 presents the experiments and results. Section 5 is the discussion section, where we have discussed and analyzed the outcomes of the results. Section 6 ends the paper with a conclusion remark and suggestions for future work.

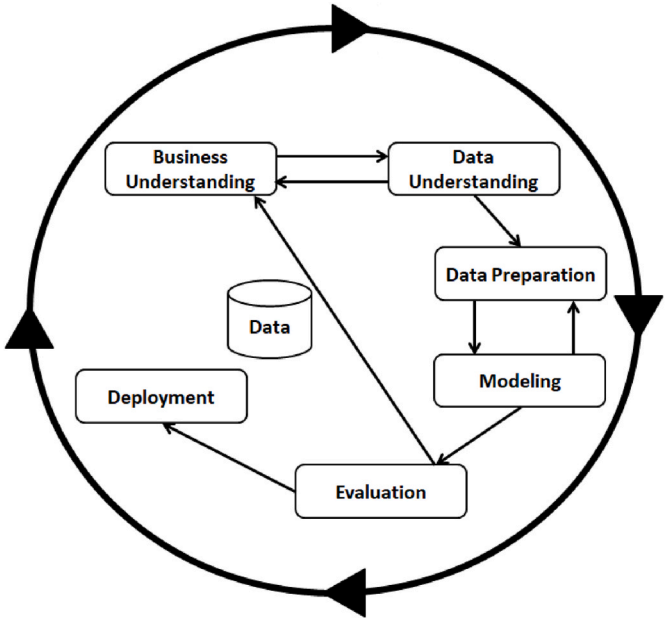

Fig. 1. Essential steps to the data mining process based on CRoss-Industry Standard Process for Data Mining [23].

2. Related work

2.1. Data mining

Usually, there are six phases in any data mining project [22], as shown in Fig. 1. These six phases can be implemented in either manner; however, it involves backtracking towards previous steps. Our research aims to improve the result in the data preparation phase by using our PAN-LDA model.

Though the particular concern of this paper is on refining the data preparation task by using our PAN-LDA yet, in the modeling phase, we aim to provide the performance comparison of different features obtained from our model and conventional approaches.

Recently, ML algorithms, including XGBoost and LightGBM, have been used in many studies and were proved useful in predicting time series. For example, H. Qiu, L. Luo, Z. Su et al. applied six machine learning algorithms, including XGBoost and LightGBM, for building predictive models having a unique feature set [17]. They showed that the LightGBM model outperforms logistic regression (LR), Support Vector Machine (SVM), and Artificial Neural Network(ANN) with the highest AUC (0.940, 95% CI: 0.900–0.980), but its performance did not vary much from that of Random Forest (RF) and XGBoost. Another paper [24] extracted time-dependent characteristics from time series and inputted them into three models: RF, XGBoost, and LightGBM, to predict Sepsis. LightGBM has proved great potential in predicting the market price movement in finance and economics [18]. Y. Tounsi, L. Hassouni, and H. Anoun have introduced a new model CSMAS to predict problems in data mining of credit scoring domain using state-of-the-art gradient boosting methods (XGBoost, CatBoost, and LightGBM) [25]. Sunghyeon Choi forecasted solar energy output by employing RF, XGBoost, and LightGBM models [26]. Moreover, J.Cordeiro, O. Postolache, and J. Ferreira used the XGBoost model and the LightGBM model to predict the height of children [27]. Accordingly, in the modeling phase, we adopted the XGBoost and LightGBM to have the benefits of the features of our PAN-LDA model. Table 1 summarizes the various machine learning techniques used to solve the prediction tasks in multiple domains.

Table 1
A summary of ML models used for various time series prediction.

Reference	Year	Base model	Time series prediction	Data set
[28]	2021	SVM, LR, Multi-layer perceptron, RF	Covid-19 pandemic cumulative case forecasting	Data of COVID-19 between January 20, 2020, and September 18, 2020, for the USA, Germany, and global was obtained from the World Health Organization website.
[17]	2020	LR, SVM, ANN, RF, XGBoost, LightGBM	Prediction of peak demand days of cardiovascular disease(CVD) admissions	Health Information Center of Sichuan Province, China: the daily number of admissions of CVD patients in hospital Chengdu Meteorological Monitoring Database: Meteorological data China National Environmental Monitoring Cente: air pollutants data
[25]	2020	XGBoost, CatBoost, LightGBM	Prediction of problems in data mining of credit scoring domain	Home Credit Default Risk from Kaggle Challenge
[26]	2020	RF, XGBoost, LightGBM	Photovoltaic Forecasting	Data of a Photovoltaic plant in South Korea
[18]	2020	LightGBM	Cryptocurrency price trend	Daily trading data from https://www.investing.com/
[29]	2020	XGBoost, ARIMA	Hemorrhagic fever with renal syndrome	Monthly hemorrhagic fever with renal syndrome incidence data from 2004 to 2018 from the official website of the National Health Commission of the People's Republic of China
[27]	2019	XGBoost, LightGBM	Child's Target Height Prediction	Dataset based on the famous study 1885 of Francis Galton, which included the trait observation of 928 children and their parents (205 pairs)
[30]	2018	Recurrent Neural Network, XGBoost	Taxi Demand Prediction	The real-world dataset generated by taxis
[31]	2018	Recurrent Neural Network, Gated Recurrent Unit	Stock price prediction	S&P 500 historical time series data

*SARIMA: Seasonal Autoregressive Integrated Moving Average.

XGBoost is an ensemble learning algorithm based on gradient boosting [14]. It is a widely applicable ML technique for both classification and regression. This has been justified in various real applications, such as malware classification, text classification, sales prediction, customer behavior prediction, risk prediction [14]. XGBoost is a helpful approach to optimize the gradient boosting algorithm by combining a linear model with a boosting tree model.

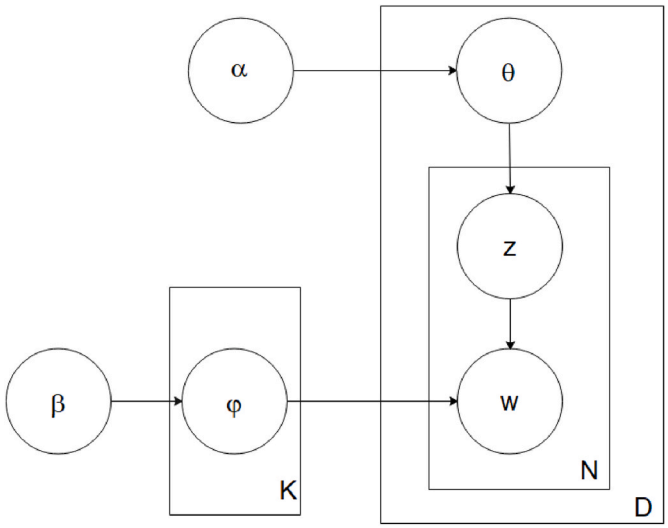

Fig. 2. Graphical Model representation of LDA.

Suppose a dataset D consists of n samples and m features, $D = \{(x_i, y_i)\ (|D| = n)\}$, where x_i are the independent variables; each of these variables has m features such that $\{x_i \in R_m\}$. And y_i is the dependent variable corresponding to x_i, $\{y_i \in R\}$. For a given (x_i, y_i), the objective function in XGBoost is given by:

$$obj(\theta) = \sum_{i}^{n} L(y_i, \widehat{y_i}) + \sum_{t=1}^{T} W(f_t) \tag{1}$$

where L is a loss function and f_t is the t^{th} tree.

LightGBM, proposed by G. Ke et al., is an improved framework based on the Gradient Boosting Decision Tree algorithm [15]. This algorithm is widely used in classification as well as regression. Moreover, we used LightGBM for regression in our model. This algorithm is mainly featured by two novel techniques: the Gradient-based One-Side Sampling (GOSS) alongside the Exclusive Feature Bundling (EFB) [32].

For a given training set $D = \{(x_i,\ y_i)\ (|D| = n)\})\}$, the objective function in LightGBM is defined as:

$$\widehat{f} = \min_{f} \sum_{1}^{n} L(y_i, f\ (x_i)) \tag{2}$$

where

L is the loss function
x_i are the independent variables and y_i is the dependent variable corresponding to x_i.

LightGBM integrates several regression trees to approximate the final model:

$$f_T(X) = \sum_{t=1}^{T} f_t(X) \tag{3}$$

While XGBoost enforces level-wise loss, LightGBM is grown leaf-wise. Leaf-wise growth strategy has several advantages compared to a level-wise growth strategy, such as reducing large errors, handling extensive data, higher accuracy, faster training, etc.

As mentioned in this article, the data preparation task is done by PAN-LDA, an LDA-based model; we first describe the baseline LDA model.

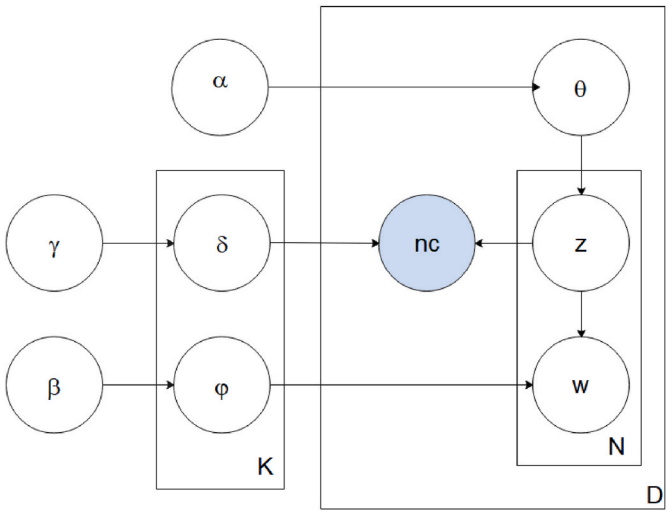

Fig. 3. Graphical model representation of PAN-LDA.

2.2. Latent Dirichlet allocation (LDA)

LDA is a generative probabilistic model proposed by Blei et al. [33] to compute the latent topics from various text documents. It is an unsupervised model that takes the primary units of data, i.e., the words in text documents. Fig. 2 represents the graphical model of smoothed LDA. The $w_{d,n}$, the index of word w in document d, represents the input of the model. The output from the model is the K, predefined number of latent topics. Each topic k, $k \in \{1, \ldots..., K\}$ is represented by a discrete probability distribution φ_k over the vocabulary V and generated from a Dirichlet distribution $\varphi_k \sim Dir(\beta)$. Additionally, every document d, $d \in \{1, \ldots..., D\}$ comes from a Dirichlet distribution $\theta_d \sim Dir(\alpha)$, which is the topic distribution for each document d. From θ_d we calculate, $z_{d,n}$, per word topic assignment in document d, where β and α are the Dirichlet parameters.

For a set of given observed words $w = \{w_{d,n}\}$, inference methods aim to determine the posterior distribution over the unknown parameters. There are several approximate inference algorithms, such as variational Bayes, Markov random fields, expectation propagation, Markov Chain Monte Carlo, etc. However, collapsed Gibbs sampling is an efficient inference procedure to learn the model from data [34], where only the latent variable z is sampled, and the random variables such as φ and θ are marginalized out. Once the latent variables are sampled out, the random variables φ and θ can be estimated.

3. Proposed model

This section describes our model PAN-LDA, followed by parameter inference. This model incorporates the changes in the number of daily new COVID-19 confirmed cases on a one-day interval along with the globally published news articles. Our model can discover hidden topics discussed in news articles related to COVID-19 in various countries. We also discuss the place of our model in text mining tasks for outbreak activity prediction.

3.1. Model description

It is noted that with the increase in the severity of pandemic cases, there is an association between the topic generation and trend prediction. Inspired by this, we developed PAN-LDA, a modification of LDA, which incorporates the statistics of daily new coronavirus cases along with news articles for feature extraction. The overall framework of the PAN-LDA approach is depicted in Fig. 3.

The graphical model of PAN-LDA is represented in Fig. 3. Our model incorporates the change in the number of reported coronavirus cases after a news article d is published, nc_d . Furthermore, the distribution of changes in reported cases per topic δ_k, is added in the figure to connect with the other distribution via nc_d . The past data of the infected corona cases are processed to find the change in the number of new reported

ALGORITHM 1: PAN-LDA

$\forall\, k \in \{1, \ldots\ldots, K\}$
  Simulate $\varphi_k \; Dir(\beta)$
  Simulate $\delta_k \sim Dir(\gamma)$
$\forall\, d \in \{1, \ldots\ldots, D\}$
  Simulate $\theta_d \sim Dir(\alpha)$
  $\forall\, w \in \{1, \ldots\ldots, N_d\}$
    Simulate $z_{d,n} \sim Multi\,(\theta_d)$
    Simulate $w_{d,n} \sim Multi\,(\varphi_{z_{d,n}})$
  Simulate $nc_d \sim Multi\,(\delta_{z_d})$

Parameters and variables:

- K: the total number of latent topics
- D: number of documents
- N_d: number of work tokens in document d
- α, β, γ: Dirichlet parameters
- φ_k: per topic word distribution
- δ_k: change in the number of daily COVID-19 cases distribution for topic k
- θ_d: per document topic distribution
- $z_{d,n}$: topic index of the word
- $w_{d,n}$: index of the word w in document d
- $Multi$: Multinomial distribution

ALGORITHM 2: collapsed Gibbs sampling

Input: $w \in d, nc$

initialize z and increment counters
for each iteration do
  for each document do
    for $n = 1$ to N_d do
      topic = $z[n]$
      decrease $N_{d,topic}$, $N_{topic,w}$, $N_{topic,nc}$ and N_{topic} by one
      for each topic k do

$$p(z_{d,n} = k | z_{-d,n}, w, nc, \alpha, \beta, \gamma) \propto (N_{d,k} + \alpha) \frac{\left(N^{-d,n}_{k,w_{d,n}} + \beta\right)}{\left(N^{-d,n}_{k} + V\beta\right)} \frac{\left(N^{-d,n}_{k,nc_d} + \gamma\right)}{\left(N^{-d,n}_{k} + C\gamma\right)}$$

      topic $\leftarrow \sim p(z| \cdot)$
      $z[n] \leftarrow$ topic
      Increase $N_{d,topic}$, $N_{topic,w}$, $N_{topic,nc}$ and N_{topic} by one
return z, $N_{d,k}$, $N_{k,w}$, $N_{k,nc}$ and N_k

cases, nc_d, after publishing the document d. The time lag that we considered for the collection of data is of one day. In the training process, data for daily new corona-infected cases were available. The presence of corona case data in PAN-LDA affects the distribution of changes in reported cases, affects the per document topic distribution, θ_d and also the per topic word distribution, φ_k. After the parameter estimation, the latent topics of a document can be obtained. For a new document, the latent topic distribution is obtained using the estimated word distribution from parameter estimation in the inference methods on the document. The received topic features can then be introduced as input features in any ML method. In summary, the incorporation of news articles and changes in the daily new corona infected cases is used by the proposed model to refine the parameter estimation and topic distribution inference on previously unseen documents. The obtained topic distribution can serve as input features for ML models to predict the time series.

For a given collection of D documents having fixed vocabulary V, with N_d word tokens, $(w_{d,1}, \ldots\ldots, w_{d,N_d})$ in document d, each having an index in the vocabulary, $w_{d,n} \in \{1, \ldots\ldots, V\}$. It is assumed that the number of latent topics, K, is predetermined.

In LDA, per word topic assignment, $z_{d,n}$, is drawn from the probability list of K topics and depends on the previously drawn topic proportion in document d, $\theta_{d,k}$. And, a word instance, $w_{d,n}$, is presumed to be deduced from a probability list of V words and depends on the word distribution, $\varphi_{d,k}$ and the topic index of the word, $z_{d,n}$. Based on LDA, PAN-LDA is also a probabilistic generative model. Accordingly, in PAN-LDA, the change in the number of new coronavirus disease infected confirmed cases, nc_d, is drawn from a list of probabilities of C categories and depends on the changes in the number of cases distribution δ_{z_d}.

ALGORITHM 1 describes the generative process of the PAN-LDA model:

ALGORITHM 1. PAN-LDA

When processing a single document, the proposed PAN-LDA algorithm distributes random accesses across an O(DK) document-topic count matrix or an O(KV) topic-word count matrix, where K, V, and D represents the total number of latent topics, the vocabulary size, and the number of documents respectively.

The variables are distributed via probability distribution:

$$p\left(z_{d,n} = k | \theta_d\right) = (\theta_d)_k \tag{4}$$

$$p\left(w_{d,n} = v | z_{d,n}, \varphi_1, \ldots\ldots, \varphi_K\right) = \left(\varphi z_{d,n}\right)_v \tag{5}$$

$$p(nc_d = c | z_d, \delta_1, \ldots\ldots, \delta_K) = \left(\delta_{z_d}\right)_c \tag{6}$$

The joint distribution of latent variables and observed data is then:

$$p(w, z, nc, \theta, \varphi, \delta | \alpha, \beta, \gamma) = \prod_k p(\varphi_k | \beta) \prod_k p(\delta_k | \gamma) \prod_d \left[p(\theta_d | \alpha) \left[\prod_n p\left(z_{d,n} | \theta_d\right) p\left(w_{d,n} | z_{d,n}, \varphi\right) \right] p(nc_d | z_d, \delta) \right] \tag{7}$$

Inferences about the posterior parameters typically yield topics where the probability mass of each topic is assigned to frequently co-occurred words that are semantically strongly related.

3.2. Topic inference

The central issue in topic modeling is posterior inference, which includes learning the posterior probabilities of the observed data, i.e., words in documents, w, and the change in the number of reported corona infected cases after the documents were published, nc, and the latent variables, i.e., θ, φ, δ and z. In PAN-LDA, the posterior probabilities of latent variables can be calculated as:

$$p(\varphi, \theta, \delta, z | w, nc, \alpha, \beta, \gamma) = \frac{p(\varphi, \theta, \delta, z, w, nc, \alpha, \beta, \gamma)}{p(w, nc, \alpha, \beta, \gamma)} \tag{8}$$

Unfortunately, the computation of this posterior distribution is intractable. The computation of the normalization factor, particularly $p(w, nc, \alpha, \beta, \gamma)$, cannot be done accurately. Among the various inference methods, CGS proposed by T. Griffiths and M. Steyvers [34] is known for model estimation with high accuracy.

Using the CGS with LDA, we are interested in computing the posterior distribution of a topic z is allocated to a word w, given the remaining words are assigned to other topics, as follows:

$$p(z_i | z_{-i}, w, \alpha, \beta, \gamma) \tag{9}$$

where z_{-i} means all topic allocations, excluding z_i.

Also, we must appeal to approximated inference, where some of the parameters are marginalized out. Therefore, we applied collapsed Gibbs Sampling for finding inference, which marginalizes out parameters φ, θ, and δ and on each iteration recovers topic, $z_{d,n}$ of a word token w, from a distribution conditioned on the present values of remaining variables. In a space containing all the variables, by sampling in a collapsed space, collapsed Gibbs Sampling usually converges much faster than a common

ALGORITHM 3: Topic Inference from a Previously Unseen Document

Input: w_{pu} and φ

initialize α, β
initialize $z_{pu,n}$
for each iteration do
  for $i = 0$ to N_{pu-1} do
    topic ← $\sim p(z_{pu,n} = k | z_{-pu,n}, w_{pu}, \alpha, \beta, \varphi) \propto \frac{(N_{pu,k}^{-pu,n}+\alpha)(N_{k,w_{new,n}}^{-pu,n}+\beta)}{(N_{k}^{-pu,n}+V\beta)}$
    update $\theta_{pu,k}$
  end
end

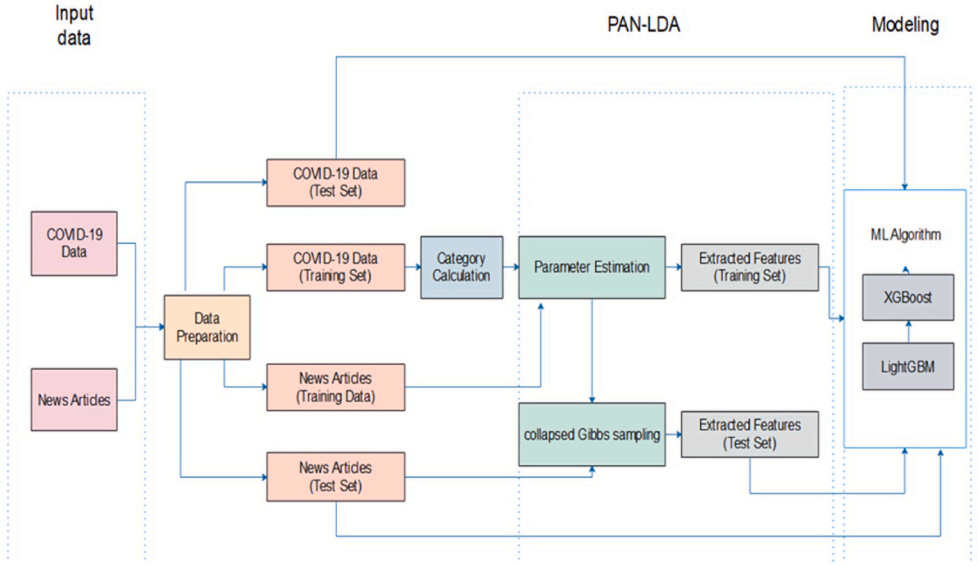

Fig. 4. Flowchart of the proposed model, PAN-LDA.

Gibbs sampler. By simply computing all the K topic assignments, a naive implementation of Eq. (10) has a complexity O(K) per token. The posterior distribution for sampling the latent variable z, after the random variables are marginalized out, for PAN-LDA, is:

$$p(z_{d,n} = k | z_{-d,n}, w, nc, \alpha, \beta, \gamma) \propto (N_{d,k} + \alpha) \frac{\left(N_{k,w_{d,n}}^{-d,n} + \beta\right)}{\left(N_{k}^{-d,n} + V\beta\right)} \frac{\left(N_{k,nc_d}^{-d,n} + \gamma\right)}{\left(N_{k}^{-d,n} + C\gamma\right)} \quad (10)$$

where,

$N_{d,k}$ denotes the total words allocated to topic k in document d.
$N_{k,w_{d,n}}$ is the number of words of type $w_{d,n}$ allocated to topic k.
N_k is the total words allocated to topic k.s
N_{k,nc_d} is the number of words belonging to category nc_d and assigned to topic k.

For topic modeling, we estimated the topic distribution per document θ_d, the word distribution per topic, φ_k, and the topic assignments per word $z_{d,n}$. Using topic allocations, φ, θ, and δ can be computed as:

$$\theta_{d,k} = \frac{N_{d,k} + \alpha}{N_d + K\alpha} \quad (11)$$

$$\varphi_{k,w_{d,n}} = \frac{N_{k,w_{d,n}} + \beta}{N_k + V\beta} \quad (12)$$

$$\delta_{k,nc_d} = \frac{N_{k,nc_d} + \gamma}{N_k + C\gamma} \quad (13)$$

Equations 11–13 estimate the quantity probabilities that word w belongs to topic k, $\varphi_{k,w_{d,n}}$, topic k is generated in document d, $\theta_{d,k}$ and the document d with category nc_d assigned to topic k, δ_{k,nc_d}, respectively. The CGS procedure is outlined in ALGORITHM 2.

ALGORITHM 2. collapsed Gibbs sampling

where count matrices are the following notations: $N_{d,k}$, $N_{k,w}$, $N_{k,nc}$ and N_k .

As shown in Fig. 3, other parameters of the model will not get affected by the δ and γ, if there is an unavailability of the corona case data. Therefore the inference from a previously unseen document, without having corona case data, will be the same as the inference from LDA.

The following pseudo-code, i.e., ALGORITHM 3, shows the inference from a new document, without θ, φ, and δ.

ALGORITHM 3. Topic Inference from a Previously Unseen Document

3.3. PAN-LDA in text and data mining

Here, we discuss how the calculations flow when our model is used for COVID-19 case prediction (see Fig. 4). The figure shows the flowchart representation of our proposed approach.

This paper deals with the application of our PAN-LDA model for forecasting coronavirus cases over time. Firstly, a usual requirement in the data preparation phase is the preprocessing of the raw data. In our experiments, we preprocessed the collected text by noise removal, case folding, tokenization, stemming, lemmatization, and stopword removal. Next, in the data preparation phase, we focused on feature extraction while working on topic modeling with PAN-LDA.

In the data preparation phase, our model needs the word vectors and the statistics of the COVID-19 cases. The vector of words in PAN-LDA is provided by the 'bag-of-words 'model. However, the case statistics need to be calculated and categorized for our model PAN-LDA-see Fig. 4. Our model generates three probability lists after training from the training set, i.e., topic distribution per document, word distribution per topic, and distribution of change in corona infected cases per topic. The topic distribution obtained serves as an additional feature for training an ML algorithm. The word distribution becomes the input parameter for inferring the topic from a previously unseen document, which in result, generates the new topic distribution for the new document, which serves as an additional feature for the ML algorithm for predicting outcomes in the next phase of time series data.

In the modeling phase, we trained machine learning algorithms to compute regressions. In this paper, we choose two machine learning algorithms, i.e., XGBoost and LightGBM, to perform our experiment. The topic distributions from the training data, along with other features, were used to train the selected ML algorithms. Ultimately, the model can be utilized to forecast the selected index.

4. Experiments and results

This section discusses the experimental setup and the comparative results. As discussed, we experimented with four different models and explored four different feature sets, i.e.:

- COVID-19 cases statistics as a base features, FS1
- COVID-19 cases statistics with topic distributions from LDA, FS2
- COVID-19 cases statistics with topic distributions from LDA and sentiment scores to the latent topics, FS3
- COVID-19 cases statistics with topic distributions from PAN-LDA, FS4

In FS1, only historical COVID-19 daily case data are used as base features in the prediction models. For FS2, in addition to the historical data, the topic distribution from the LDA is integrated into the prediction model. In FS3, we use the COVID-19 cases data along with the computed topic distribution of the news articles and sentiment scores specific to the extracted latent topics. Due to the fact that the topics extracted from news articles do not have any associated sentiments, sentiment analysis of reviews is also done by using VADER to compute the relative sentiment scores with respect to the topics. The historical data, extracted topics, and their sentiments are used as input features to a machine learning prediction model. Finally, FS4 denotes the feature set obtained from the topic distributions from PAN-LDA and the COVID-19 historical data.

Once the models are trained, we used the backtesting approach for time series forecasting. For backtesting, we used the walk-forward testing [35] routine, which accounts for model performance at different time windows.

4.1. Data selection and gathering

- **COVID-19 data:** The data for the number of confirmed coronavirus infected cases used to experiment are the official data published by 'Our World in Data' [21]. They have provided global and reliable data to study statistics on the COVID-19 pandemic. The dataset is updated daily from the World Health Organization (WHO) situation reports [36]. We use available data on the daily new cases of coronavirus-infected people from January 2020 to May 2020.
- **News Articles:** The news articles dataset used in this paper was gathered from the Aylien [20]. Aylien has aggregated and published the COVID-19 dataset that can be used to analyze global news during the outbreak. Aylien has transformed the COVID-19 dataset into structured and actionable data using NLP and ML. The data analyzed in this study correspond to the period that stretches between January 2020 to May 2020. As a result, a total of 1147454 articles were considered for the experiment.

4.2. Data preparation

Consequently, we trained our model on a collection of more than 1 million news articles, which contributes to text documents for this experiment. We preprocessed the data by tokenization, stemming, and removal of stop words. The text in new articles is represented by a vector using the 'bag-of-words' model in Gensim [37].

We collected one-day level new corona infected data and classified the changes in the number of infected cases into three different categories. The threshold was considered based on an average of one-day change for collected data. There is an average change of 0.1285**%** in both directions. In this paper, we experimented with a **0.10%-0.15%** change as the threshold. As a result, the data is divided into three categories, i.e.,

- category 1, if the change in the number of new coronavirus cases lies above the threshold,
- category −1 if the change in the number of coronavirus cases is below the threshold, and
- category 0 if it lies within the threshold

So, the value for nc_d in PAN-LDA is explained in three levels in Equation (14).

We found that 767345 articles were falling in the category $-1(nc_d =$ -1), 145773 articles in category $0(nc_d = 0)$ and category $1(nc_d = 1)$ has 234336 articles.

After all of the data has been preprocessed, we then divide it into

$$nc_d = \begin{cases} 1, \ if \ \dfrac{number\ of\ the\ new\ coronavirus\ case_{t+1\ day} - number\ of\ the\ new\ coronavirus\ case_t}{number\ of\ the\ infected\ person_t} * 100 > 0.15 \\ -1, \ if \ \dfrac{number\ of\ the\ new\ coronavirus\ case_{t+1\ day} - number\ of\ the\ new\ coronavirus\ case_t}{number\ of\ the\ infected\ person_t} * 100 < 0.10 \\ 0, \ otherwise \end{cases} \tag{14}$$

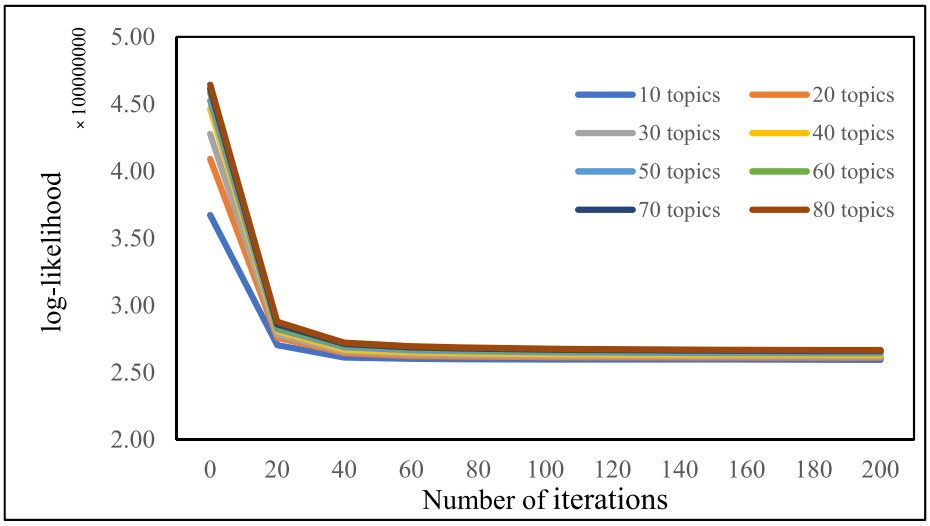

Fig. 5. The log-likelihood for PAN-LDA with collapsed Gibbs sampling.

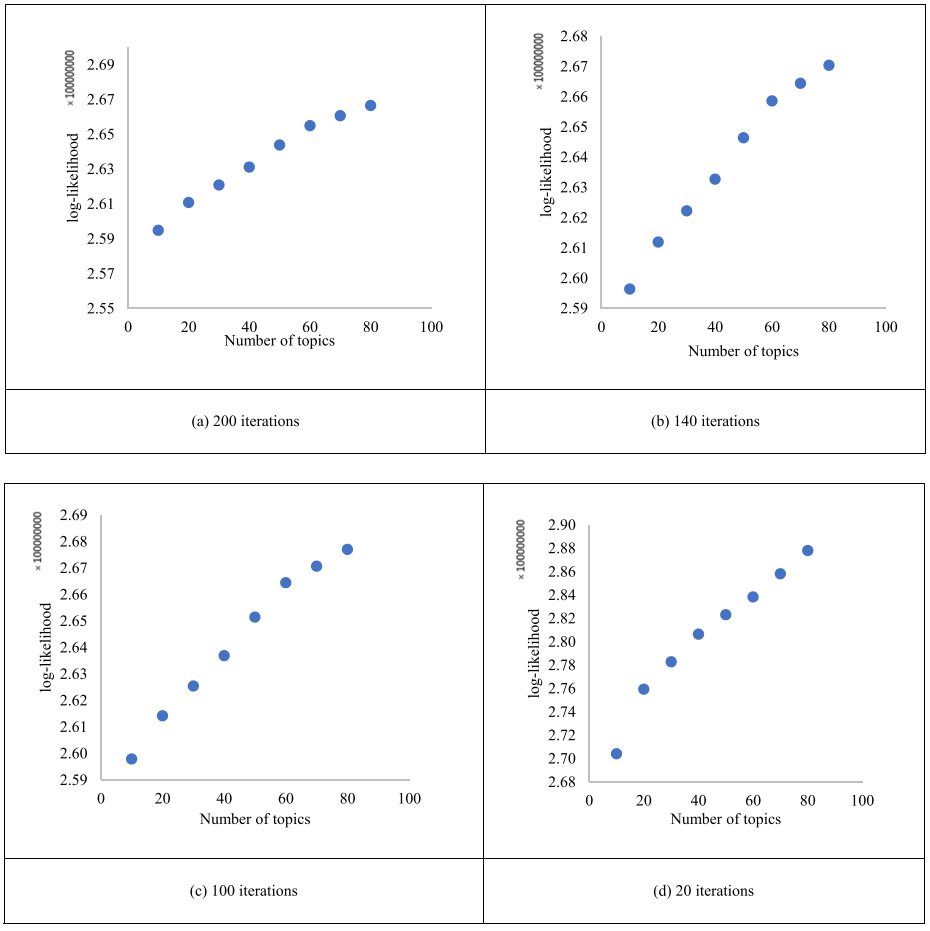

Fig. 6. The log-likelihood against the number of topics.

training and test sets using walk-forward validation [35], a variant of cross-validation. This method includes splitting the data into a series of overlapping training-testing sets, and each set is moved forward through the time series. In this paper, the whole dataset was split into 11 overlapping datasets with a 19-day window. Each of these datasets was divided into training and test sets as 25% testing and 75% training.

The input dataset for PAN-LDA consists of all the word tokens in D documents $w_{d,n}$, and category values, nc_d. Next, we set the hyperparameters of the Dirichlet distributions, i.e., α, γ, and β. Without any particular basis, different research papers have chosen different values of these hyperparameters, e.g. Refs. [38,39], used $\beta = 0.1$, $\alpha = 50/K$ [40,41], used $\beta = 0.1$, $\alpha = 0.1$ and [42], used $\beta = 1/K$, $\alpha = 1/K$. In order to have few words with a high probability per topic and numerous latent topics with a high probability per document, the values of Dirichlet

Table 2a
Examples of topics generated by LDA.

Sports	Finance	Business	Entertainment	Country	Politics
League	Market	business	Time	australia	Trump
Football	Economy	company	People	Reuters	president
Season	Global	ceo	Lockdown	new_zealand	donald_trump
Players	China	officer	social_media	government	white_house
Club	Markets	financial	Instagram	australian	president_trump
premier_league	Oil	year	Family	bank	house
Team	Economic	bank	Star	editing	virus
Sports	Year	industry	Quarantine	reporting	washington
England	Energy	companies	Year	reuters_reuters	bill
Training	Reuters	airline	Video	sydney	congress
United	Virus	insurance	Food	european	year
Sport	Stock	group	life	germany	trump_administration
Games	Bank	chief_executive	social	france	fox_news
Clubs	Prices	cash	facebook	prime_minister	senate
Game	Demand	businesses	times	french	federal

Table 2b
Examples of topics generated by PAN-LDA

Sports	Finance	Business	Entertainment	Country	Health
League	Bank	company	instagram	new_york	virus
Football	Economy	business	time	city	hospital
Season	Market	year	star	county	patients
Team	Economic	ceo	video	california	infection
Players	Global	officer	social_media	governor	disease
Sports	Markets	industry	live	york	health
Game	Financial	companies	twitter	florida	vaccine
Games	Oil	sales	family	texas	testing
Club	Reuters	market	music	los_angeles	symptoms
Events	Stimulus	stock	film	virus	test
Time	Energy	group	story	department	tests
Year	Prices	production	years	order	people
premier_league	unemployment	supply_chain	life	mayor	fever
United	Money	quarter	series	chicago	medical
Event	Rate	nasdaq	netflix	people	medicine

distributions were set as $1/K$,i.e., $\alpha = \beta = \gamma = 1/K$ for all the topic models.

Selecting the optimum number of topics(K) in topic modeling is also a significant problem. In order to estimate the optimum values of K and the number of iterations (iter), we ran 200 iterations, iter = 200, and noted the computed value at every 20 iterations. We evaluated the effect of iteration count with different numbers of topics on log-likehoods at these savings points. The results are presented in Fig. 5, which depicts the stability in results when iter >140.

Also, for a better understanding of the parameters' values, we computed the log-likehoods for PAN-LDA in Fig. 6. The graphs in Fig. 6 suggested that the optimum value is achieved at $K = 10$. And as the graph showed stable results when iter >140, therefore, we set the iter = 160 for our experiment. Accordingly, we set the same values for the LDA model.

After setting the parameter values, we extracted topics from the LDA, news-text-sentiment feature grouping using LDA and PAN-LDA models. Tables 2a and b show 6 of the 10 topics discovered by LDA and PAN-LDA, respectively, with their top 15 words. The remaining topics are shown in Table S1 and Table S2 in the supplementary information.

Tables 2a and b suggest that some topics from both models have the same words or words with similar implicit meanings. Though, their ranking order, suggesting their importance, is different. Moreover, some topics from the two models are entirely different.

Based on the extracted words, we interpreted the meaning of the topic and assigned labels to each, i.e., 'Sports', 'Finance', 'Business', 'Entertainment', 'Country', 'Health' and 'Politics'. We assign the same color to the words belonging to the same topic. Topic 1, 'Sports', colored in orange, has similar sets of words for both the models. Though the words in the models have a different order, indicating their importance.

Table 3
Performance metrics and their calculations.

Metrics	Calculation
R^2	$1 - \frac{\sum (c_i - \hat{c}_i)^2}{(c_i - \bar{c})^2}$
RMSE	$\sqrt{\frac{1}{N}\sum_1^N (c_i - \hat{c}_i)^2}$
MAE	$\frac{1}{N}\sum_1^N \lvert c_i - \hat{c}_i \rvert$
MAD	$\frac{1}{N}\sum_1^N \lvert c_i - \bar{c} \rvert$

Topic 2, 'Finance', also contains similar words for both the models, yet, the absence of an important word, i.e., "stock", can be noted in the vocabulary words of PAN-LDA.

Similarly, the word 'unemployment' is absent from the top vocabulary words of LDA. In topic 3, 'Business', the words generated by both the models are quite different but appear to be similar in their implicit meanings. In the next topic, i.e., 'Entertainment', PAN-LDA has generated more meaningful words related to the topic in comparison to LDA. In LDA, the words are vaguely present and do not contribute much to extract a single topic. The words from LDA in topic 5 seems to be a combination of two topics. PAN-LDA isolates more coherent topics, such as health, social anxiety, as compared to LDA. In LDA, the remaining words talk about politics, lockdown, etc. We noted that the rest of the generated words in LDA do not contribute much to form new identifiable topics. Also, the remaining topics generated by PAN-LDA are vaguely present in LDA. Some topics in LDA seemed to be a combination of topics

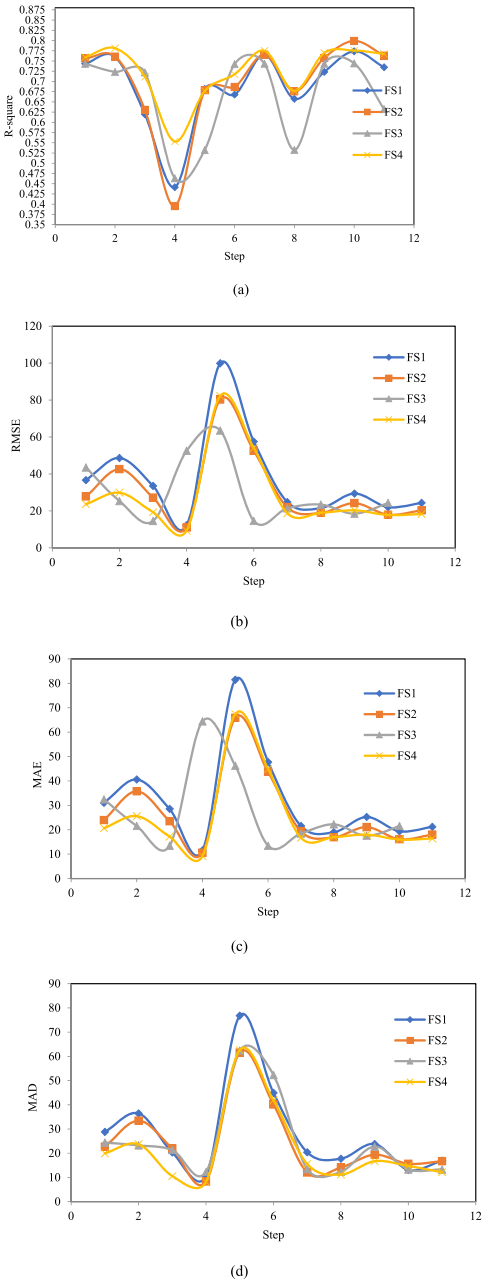

Fig. 7. The presentation of (a) R^2 (b) RMSE, (c) MAE, and (d) MAD between the actual and the predicted number of new confirmed cases for FS1, FS2, FS3, and FS4 by XGBoost.

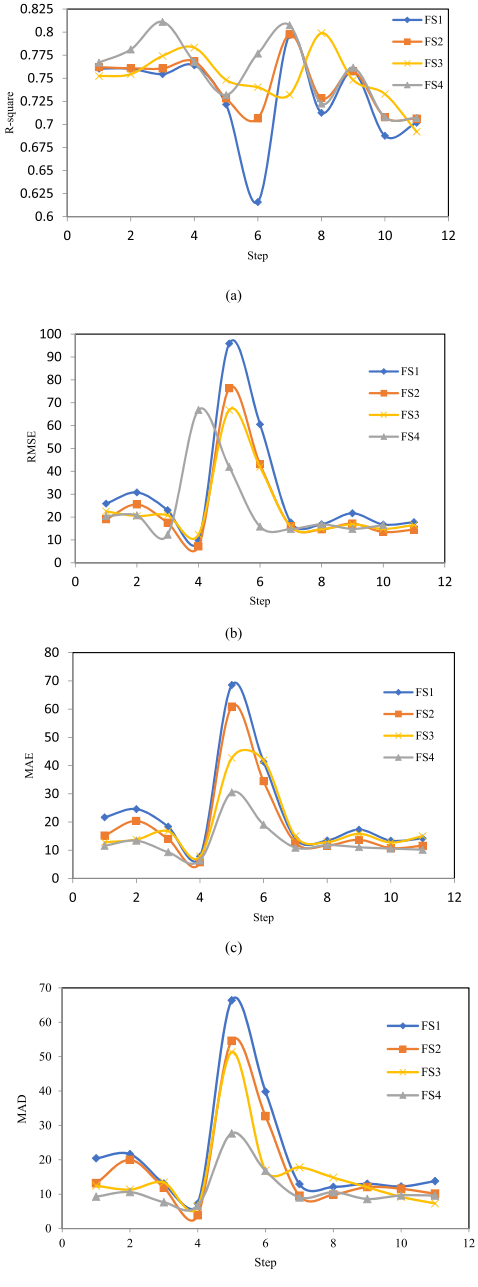

Fig. 8. The presentation of (a) R^2 (b) RMSE, (c) MAE, and (d) MAD between the actual and the predicted number of new confirmed cases for FS1, FS2, FS3, and FS4 by LightGBM.

from PAN-LDA. Additionally, PAN-LDA produced more identifiable topics.

As the topics obtained from the models are mainly used for inferring the topic distributions from the text, the interpretation and meaning of topics are not of much concern in this experiment. From the parameter estimation process, the topic distributions for all training set documents along with estimated word distributions for inference were obtained. The obtained topic distribution then served as an input feature for ML algorithms without any need to interpret their meaning. It can be noted that two different models generated different topics. This suggests that adding a new feature, i.e., changes in data of new corona cases, successfully influenced per topic word distribution in the parameter estimation, which is evaluated in the next step.

Following that, using the estimated φ values and all words in the test set, $w_{d,n}$, the topic distributions were inferred from documents in the test set. The resulted topic distributions were fed into machine learning models, i.e., XGBoost and LightGBM, for testing in the next phase.

The prepared data is then arranged into four feature sets, FS1, FS2, FS3, and FS4, for both training and test sets.

4.3. Evaluation indicators

We used four widely accepted statistical indicators, i.e., the determination coefficient, R^2 (R-Square), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Deviation (MAD) for performance comparison of ML algorithms trained with the above-said feature sets. Each metric is computed, as mentioned in Table 3.where N corresponds to the total in the test data, c and $\hat{c}$ are the i^{th} value of the

Table 4
Comparison of results of XGBoost and LightGBM with FS4.

Step	R^2		RMSE		MAE		MAD	
	XGBoost	LightGBM	XGBoost	LightGBM	XGBoost	LightGBM	XGBoost	LightGBM
1	0.7592	**0.7670**	23.6395	**15.7655**	20.6263	**11.5822**	19.9906	**9.2664**
2	**0.7812**	**0.7812**	29.8801	**16.7653**	25.6069	**13.3801**	23.7570	**10.6228**
3	0.7112	**0.8112**	19.3234	**11.6423**	17.1817	**9.29156**	10.5774	**7.6562**
4	0.5531	**0.7676**	9.34649	**07.9876**	9.21931	**6.37479**	8.18504	**6.6132**
5	0.6766	**0.7317**	82.2345	**78.2345**	67.3902	**30.4953**	62.5700	**27.6917**
6	0.7175	**0.7768**	54.0757	**28.8765**	44.9171	**19.0459**	41.9074	**16.7886**
7	0.7752	**0.8075**	18.7480	**13.6653**	16.7225	**10.9061**	15.5794	**9.0570**
8	0.6766	**0.7226**	19.0169	**14.8265**	16.9371	**11.8328**	11.2149	**10.5660**
9	**0.7697**	0.7615	20.3450	**13.8643**	17.9971	**11.0649**	16.7289	**8.6037**
10	**0.7755**	0.7079	17.8170	**13.2345**	15.9795	**10.5622**	14.7570	**9.6441**
11	**0.7673**	0.7070	18.3331	**12.7654**	16.3914	**10.1879**	12.1495	**9.6112**
avg	0.7239	**0.7583**	28.4327	**20.6934**	24.4517	**13.1567**	21.5834	**11.4655**

observed and forecasted number of new COVID-19 cases in the testing period and $\bar{c}$ denotes the mean value of c.

It should also be noted that lower values of the RMSE, MAE and MAD indicate a better fit.

4.4. Results

Though the focus of our study is on extracting the better feature in the data preparation stage, yet we use two ML algorithms, i.e., XGBoost and LightGBM, to validate the performance of PAN-LDA. Next, we sought to evaluate the models using four statistical metrics, i.e., R^2, RMSE, and MAE provided by scikit-learn [43–45] and MAD provided by mad function in class Series in pandas [46].

4.4.1. Results of XGBoost

Initially, as we were interested in a fair comparison among the results of different feature sets, we trained the XGBoost with the default parameter values [47] in the modeling phase. The results obtained for the different feature sets are given in the Supplementary information, i.e., Tables S3–S6.

To better understand the XGBoost results, Fig. 7(a)–(d) show the distribution of the four evaluation metrics mentioned in this paper. In each figure, we compare the outputs (vertical axis) of evaluation indicators for all four feature sets against each step of walk-forward testing.

4.4.2. Results of LightGBM

The results of all the feature sets from LightGBM are presented in the supplementary information (Tables S7–S10), using 11 overlapping training–test sets from the walk-forward validation. Fig. 8 (a)-(d) illustrate that the performances of LightGBM with different sets of features, i.e., FS1-FS4.

5. Discussion

After text mining became popular and viable for extracting information from text, public health research often incorporated unstructured textual data. This paper presented a feature extraction model based on changes in daily COVID-19 cases data and news articles. To derive the inputs for the ML prediction models, we used the PAN-LDA model to extract relevant features. To take advantage of our PAN-LDA, the topic distributions from PAN-LDA are then used in a machine learning model. We compared the performance of our approach in predicting COVID-19 cases one day after news articles were released. By comparing the final results while employing the four different feature sets, FS1, FS2, FS3, and FS4, an experiment was conducted to demonstrate the benefits of integrating the features generated using PAN-LDA. Furthermore, we compare the results from XGBoost and LightGBM for all four feature sets, using the 11 overlapping training-test sets. The proposed model's prediction errors were lower than those of the other techniques. When the features from the proposed model are employed in both XGBoost and LightGBM, the results reveal that they empirically add value to the prediction.

The results for R^2, RMSE, MAE, and MAD for XGBoost are provided using graphs in Fig. 7 (a)–(d). We observed that the best results are obtained with FS4 for 7 out of 11 overlapping datasets with all statistical measures. Comparing the results with FS1, FS2, and FS3 as shown in Fig. 7 (a)–(d) shows that FS1, FS2, and FS3 have larger values for average RMSE, average MAE, and average MAD but smaller average R^2 than the FS4. Compared to the baseline methods, the proposed model improves average RMSE by 24-3% and MAE by 22-7%. The MAD in Fig. 7(d) reveals that results from XGBoost when using FS2 were better in some datasets, but the best result for average MAD is achieved for the FS4, followed by FS2, FS3, and FS1. We concluded that FS4 provided better input features than FS1, FS2, and FS3. The result is consistent in all of the evaluation indicators. Also, the XGBoost gave the best average performance when used with the feature set FS4.

The results of LightGBM are shown in Fig. 8 (a)–(d). We can see these figures show the FS4 has smaller RMSE, MAE, and MAD than FS3, FS2, and FS1 but larger R^2 in 8 out of 11 steps of backtesting. It implies that the performance of LightGBM when using features from PAN-LDA was better than that when using other sets of features. The average performance from 11 overlapping datasets indicates that LightGBM with FS1 has the worst performance, followed by FS2 and FS3 and the best with FS4 features. The R^2 for FS4 was improved by 3.84% than when using FS1. The average RMSE from PAN-LDA, 20.6934, is significantly better than FS1, 30.5913, FS2, 24.0235, and FS3 27.1294. The average MAE and MAD show the same results. Fig. 8(d) shows that the LightGBM, when used with the features from our model, outperformed FS1 by 45.78% and FS2 by 33.39%, and FS3 by 27.06%. Additionally, in Fig. 8 (a)–(d), the performance of all the feature sets is compared by taking all the evaluation metrics, showing the benefit of PAN-LDA clearly. These figures also suggest that the performance with FS4 was much better than FS1, FS2, and FS3 for all the evaluation metrics, namely R^2, RMSE, MAE, and MAD.

As for both XGBoost and LightGBM, on average, the results from FS4 are better than the results from FS3, FS2, and FS1. Fig. 7(a)–(d) shows that the PAN-LDA model outperforms the baseline models in terms of MAE but not so much in terms of RMSE as RMSE penalizes larger prediction errors, while MAE stands for the absolute difference between observed and predicted values. Therefore, in Table 4, a comparison of these two machine learning algorithms using FS4 has been demonstrated. It can be noted from Table 4 that the highest correlations were achieved for LightGBM with an average R^2 of 0.7583. Also, the LightGBM has a smaller average value of RMSE, MAE, and MAD than XGBoost.

The experimental results and comparison of the proposed PAN-LDA model's performance with baseline models clearly show that

supplementary/side information, such as new article content, is a valuable and expressive source of information for improving ML algorithms' predictions. Because historical data only captures the general perception of the target item, it cannot be used to generate precise forecasts. The features extracted from the LDA model do not seem to give much advantage for data forecasting over time. Moreover, incorporating sentiment scores as an additional feature in the prediction model has improved performance with less prediction error, such as MAE and RMSE. Besides, the results from FS1 are the worst for both XGBoost and LightGBM with the walk-forward testing. It can be concluded that incorporating the infectious disease data along with news articles in PAN-LDA gave better performance than LDA, which incorporates news articles only. This suggests the benefit of additional features in PAN-LDA. However, it seems that adding XGBoost resulted in only little changes with the PAN-LDA model. Overall, it can be implied that LightGBM can forecast more closely to the actual values of COVID-19 cases than the XGBoost method. Also, including changes in the number of COVID-19 cases into account in PAN-LDA for prediction with time series, esp. with LightGBM.

In this study, the overall time-period of the research is short because of limited availability of the reliable new articles data [20]. We will improve our model with more data in the future.

6. Conclusions and future directions

In this work, we proposed an LDA-based mathematical model, PAN-LDA, which integrates news articles and data of confirmed COVID-19 cases for better feature extraction. The resultant features can be input as additional features to any ML algorithm to forecast trends with time series. In our paper, we introduced the extracted features from the PAN-LDA model to two gradient boosting-based ML algorithms, i.e., XGBoost and LightGBM, to validate the feasibility of applying PAN-LDA compared to baseline methods. The features from PAN-LDA significantly added value to the goal output when used in ML algorithms. Moreover, LightGBM gave a considerably better performance than XGBoost.

In summary, the features from PAN-LDA generated more identifiable topics and empirically added value to the prediction when they were used in LightGBM.

In the future, we will focus on incorporating the other sophisticated features, e.g., daily death cases, the number of recovered cases, etc., as well as on hyperparameters tuning. In the present paper, we have used default values of hyperparameters for both ML algorithms, i.e., XGBoost and LightGBM. So it cannot be guaranteed that the used hyperparameters' values rates are the best. Choosing the optimal values of the hyperparameters of ML algorithms will be investigated in future work.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compbiomed.2021.104920.

References

[1] A. Esteva, A. Kale, R. Paulus, K. Hashimoto, W. Yin, D. Radev, R. Socher, COVID-19 information retrieval with deep-learning based semantic search, question answering, and abstractive summarization, Npj Digit. Med. 4 (2021), https://doi.org/10.1038/s41746-021-00437-0.

[2] E. Zhang, N. Gupta, R. Tang, X. Han, R. Pradeep, K. Lu, Y. Zhang, R. Nogueira, K. Cho, H. Fang, J. Lin, in: Covidex: Neural Ranking Models and Keyword Search Infrastructure for the COVID-19 Open Research Dataset, 2020, pp. 31–41, https://doi.org/10.18653/v1/2020.sdp-1.5.

[3] A. Köksal, H. Dönmez, R. Özçelik, E. Ozkirimli, A. Özgür, Vapur: a search engine to find related protein - compound pairs in COVID-19 literature, in: Proc. 1st Work. NLP COVID-19 (Part 2) EMNLP 2020, Association for Computational Linguistics, Stroudsburg, PA, USA, 2020, https://doi.org/10.18653/v1/2020.nlpcovid19-2.21.

[4] A. Khadjeh Nassirtoussi, S. Aghabozorgi, T. Ying Wah, D.C.L. Ngo, Text mining of news-headlines for FOREX market prediction: a Multi-layer Dimension Reduction Algorithm with semantics and sentiment, Expert Syst. Appl. 42 (2015) 306–324, https://doi.org/10.1016/j.eswa.2014.08.004.

[5] F. Jin, N. Self, P. Saraf, P. Butler, W. Wang, N. Ramakrishnan, Forex-foreteller: currency trend modeling using news articles, in: Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., 2013, pp. 1470–1473, https://doi.org/10.1145/2487575.2487710.

[6] A. Tissaoui, S. Sassi, R. Chbeir, Probabilistic topic models for enriching ontology from texts, SN comput. Sci. 1 (2020), https://doi.org/10.1007/s42979-020-00349-y.

[7] X. Li, L. Lei, A bibliometric analysis of topic modelling studies (2000–2017), J. Inf. Sci. 47 (2021) 161–175, https://doi.org/10.1177/0165551519877049.

[8] B. Zhu, X. Zheng, H. Liu, J. Li, P. Wang, Analysis of spatiotemporal characteristics of big data on social media sentiment with COVID-19 epidemic topics, Chaos, Solit. Fractals 140 (2020), 110123, https://doi.org/10.1016/j.chaos.2020.110123.

[9] C. Ordun, S. Purushotham, E. Raff, Exploratory analysis of covid-19 tweets using topic modeling, UMAP, and DiGraphs, ArXiv. https://radimrehurek.com/gensim/models/ldamulticore.html, 2020. (Accessed 27 November 2020).

[10] A. Rortais, F. Barrucci, V. Ercolano, J. Linge, A. Christodoulidou, J.P. Cravedi, R. Garcia-Matas, C. Saegerman, L. Svečnjak, A topic model approach to identify and track emerging risks from beeswax adulteration in the media, Food Control 119 (2021), 107435, https://doi.org/10.1016/j.foodcont.2020.107435.

[11] T. Chuluunsaikhan, G.A. Ryu, K.H. Yoo, H. Rah, A. Nasridinov, Incorporating deep learning and news topic modeling for forecasting pork prices: the case of South Korea, Agric. For. 10 (2020) 1–22, https://doi.org/10.3390/agriculture10110513.

[12] X. Li, W. Shang, S. Wang, Text-based crude oil price forecasting: a deep learning approach, Int. J. Forecast. 35 (2019) 1548–1560, https://doi.org/10.1016/j.ijforecast.2018.07.006.

[13] A. Mahadevan, M. Arock, Integrated topic modeling and sentiment analysis: a review rating prediction approach for recommender systems, Turk. J. Electr. Eng. Comput. Sci. 28 (2020) 107–123, https://doi.org/10.3906/elk-1905-114.

[14] T. Chen, C. Guestrin, XGBoost: a scalable tree boosting system, in: Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., 2016, pp. 785–794, https://doi.org/10.1145/2939672.2939785.

[15] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.Y. Liu, LightGBM: a highly efficient gradient boosting decision tree, in: Adv. Neural Inf. Process. Syst., 2017, pp. 3147–3155. https://github.com/Microsoft/LightGBM. (Accessed 16 August 2020).

[16] J.C. Wang, T. Hastie, Boosted varying-coefficient regression models for product demand prediction, J. Comput. Graph Stat. 23 (2014) 361–382, https://doi.org/10.1080/10618600.2013.778777.

[17] H. Qiu, L. Luo, Z. Su, L. Zhou, L. Wang, Y. Chen, Machine learning approaches to predict peak demand days of cardiovascular admissions considering environmental exposure, (n.d.). https://doi.org/10.1186/s12911-020-1101-8.

[18] X. Sun, M. Liu, Z. Sima, A novel cryptocurrency price trend forecasting model based on LightGBM, Finance Res. Lett. 32 (2020), 101084, https://doi.org/10.1016/j.frl.2018.12.032.

[19] Y. Liang, J. Wu, W. Wang, Y. Cao, B. Zhong, Z. Chen, Z. Li, Product marketing prediction based on XGboost and LightGBM algorithm, in: ACM Int. Conf. Proceeding Ser., 2019, pp. 150–153, https://doi.org/10.1145/3357254.3357290.

[20] Free coronavirus news dataset - updated - AYLIEN, n.d. https://blog.aylien.com/free-coronavirus-news-dataset/. (Accessed 16 August 2020).

[21] Coronavirus Pandemic (COVID-19), Statistics and research - our World in data, n.d. https://ourworldindata.org/coronavirus. (Accessed 16 August 2020).

[22] V. Sharma, A. Stranieri, J. Ugon, P. Vamplew, L. Martin, An agile group aware process beyond CRISP-DM: a hospital data mining case study, in: ACM Int. Conf. Proceeding Ser., Association for Computing Machinery, 2017, pp. 109–113, https://doi.org/10.1145/3093241.3093273.

[23] P.C. Ncr, J.C. Spss, R.K. Ncr, T.K. Spss, T.R. Daimlerchrysler, C.S. Spss, R. W. Daimlerchrysler, Step-by-step data mining guide, SPSS Inc. 78 (2000) 1–78. http://www.crisp-dm.org/CRISPWP-0800.pdf. (Accessed 20 September 2020).

[24] Q. Yu, X. Huang, W. Li, C. Wang, Y. Chen, Y. Ge, Using features extracted from vital time series for early prediction of Sepsis, in: 2019 Comput. Cardiol. Conf., 2019, https://doi.org/10.22489/cinc.2019.067.

[25] Y. Tounsi, H. Anoun, L. Hassouni, CSMAS: improving multi-agent credit scoring system by integrating big data and the new generation of gradient boosting algorithms, in: ACM Int. Conf. Proceeding Ser., 2020, https://doi.org/10.1145/3386723.3387851.

[26] S. Choi, J. Hur, An ensemble learner-based bagging model using past output data for photovoltaic forecasting, Energies 13 (2020), https://doi.org/10.3390/en13061438.

[27] J.R. Cordeiro, O. Postolache, J.C. Ferreira, Child's target height prediction evolution, Appl. Sci. 9 (2019) 5447, https://doi.org/10.3390/app9245447.

[28] S. Ballı, Data analysis of Covid-19 pandemic and short-term cumulative case forecasting using machine learning time series methods, Chaos, Solit. Fractals 142 (2021), 110512, https://doi.org/10.1016/j.chaos.2020.110512.

[29] C.-X. Lv, S.-Y. An, B.-J. Qiao, W. Wu, Time Series Analysis of Hemorrhagic Fever with Renal Syndrome in Mainland China by Using XGBoost Forecasting Model, n. d..

[30] U. Vanichrujee, T. Horanont, T. Theeramunkong, W. Pattara-Atikom, T. Shinozaki, Taxi demand prediction using ensemble model based on RNNs and XGBOOST, in: 2018 Int. Conf. Embed. Syst. Intell. Technol. Int. Conf. Inf. Commun. Technol. Embed. Syst. ICESIT-ICICTES 2018, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 1–6, https://doi.org/10.1109/ICESIT-ICICTES.2018.8442063.

[31] M.A. Hossain, R. Karim, R. Thulasiram, N.D.B. Bruce, Y. Wang, Hybrid deep learning model for stock price prediction, in: Proc. 2018 IEEE Symp. Ser. Comput.

Intell. SSCI 2018, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1837–1844, https://doi.org/10.1109/SSCI.2018.8628641.
[32] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.Y. Liu, LightGBM: a highly efficient gradient boosting decision tree, in: Adv. Neural Inf. Process. Syst., 2017, pp. 3147–3155. https://github.com/Microsoft/LightGBM. (Accessed 17 September 2020).
[33] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022, https://doi.org/10.1016/b978-0-12-411519-4.00006-9.
[34] T.L. Griffiths, M. Steyvers, Finding scientific topics, Proc. Natl. Acad. Sci. U.S.A. 101 (2004) 5228–5235, https://doi.org/10.1073/pnas.0307752101.
[35] L.J. Cao, F.E.H. Tay, Support vector machine with adaptive parameters in financial time series forecasting, IEEE Trans. Neural Network. 14 (2003) 1506–1518, https://doi.org/10.1109/TNN.2003.820556.
[36] WHO, Coronavirus disease (COVID-19) situation reports in Bangladesh, World Heal. Org. (2020) 1. https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports. (Accessed 16 August 2020).
[37] R. Rehurek, gensim: topic modelling for humans. https://radimrehurek.com/gensim/index.html, 2014. (Accessed 16 August 2020).
[38] D. Surian, S. Chawla, Mining outlier participants: insights using directional distributions in latent models, in: Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2013, pp. 337–352, https://doi.org/10.1007/978-3-642-40994-3_22.
[39] L. Wang, Y. Zhang, Y. Zhang, X. Xu, S. Cao, Prescription function prediction using topic model and multilabel classifiers, evidence-based complement, Altern. Med. (2017), https://doi.org/10.1155/2017/8279109, 2017.
[40] A. Panichella, A Systematic Comparison of search-Based approaches for LDA hyperparameter tuning, Inf. Software Technol. 130 (2021), 106411, https://doi.org/10.1016/j.infsof.2020.106411.
[41] T. Yoshida, R. Hisano, T. Ohnishi, Gaussian hierarchical latent dirichlet allocation: bringing polysemy back. http://arxiv.org/abs/2002.10855, 2020. (Accessed 7 April 2021).
[42] J. Vosecky, D. Jiang, K.W.T. Leung, W. Ng, Dynamic multi-faceted topic discovery in twitter, in: Int. Conf. Inf. Knowl. Manag. Proc., 2013, pp. 879–884, https://doi.org/10.1145/2505515.2505593.
[43] sklearn.metrics.r2_score — scikit-learn 0.23.2 documentation, n.d. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html. (Accessed 16 August 2020).
[44] sklearn.metrics.mean_absolute_error — scikit-learn 0.23.2 documentation, n.d. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_error.html. (Accessed 16 August 2020).
[45] sklearn.metrics.mean_squared_error — scikit-learn 0.23.2 documentation, n.d. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_error.html. (Accessed 16 August 2020).
[46] pandas.Series.mad — pandas 1.2.4 documentation, n.d. https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Series.mad.html. (Accessed 18 April 2021).
[47] XGBoost Python Package, Python Package Introduction — xgboost 1.4.0-SNAPSHOT documentation. https://xgboost.readthedocs.io/en/latest/python/python_intro.html, 2020. (Accessed 16 August 2020).