IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 September 2022, accepted 25 September 2022, date of publication 3 October 2022, date of current version 10 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3211396

RESEARCH ARTICLE

Enhancing Big Social Media Data Quality for Use in Short-Text Topic Modeling

BELAL ABDULLAH HEZAM MURSHED[1,2], JEMAL ABAWAJY[3], (Senior Member, IEEE), SURESHA MALLAPPA[1], MUFEED AHMED NAJI SAIF[4], SUMAIA MOHAMMED AL-GHURIBI[5,6], AND FAHD A. GHANEM[7,8]

[1]Department of Studies in Computer Science, Mysore University, Mysore-570006, Karnataka, India
[2]Department of Computer Science, College of Engineering and IT, Amran University, Amran, Yemen
[3]School of Information Technology, Faculty of Science, Engineering and Built Environment, Deakin University, Geelong, VIC 3220, Australia
[4]Department of Computer Applications, Sri Jayachamarajendra College of Engineering (Affiliated to VTU), Mysore 570006, Karnataka, India
[5]Department of Computer Science, Faculty of Applied Sciences, Taiz University, Taiz, Yemen
[6]Center for Artificial Intelligent Technology, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia
[7]Department of Computer Science & Engineering, PES College of Engineering (Affiliated to University of Mysore), Mandya 571401, India
[8]Department of Computer Science, College of Education—Zabid, Hodeidah University, Hodeidah, Yemen

Corresponding author: Belal Abdullah Hezam Murshed (belal.a.hezam@gmail.com)

ABSTRACT With the emergence of microblogging platforms and social media applications, large amounts of user-generated data in the form of comments, reviews, and brief text messages are produced every day. Microblog data is typically of poor quality; hence improving the quality of the data is a significant scientific and practical challenge. In spite of the relevance of the problem, there has been not much work so far, especially in regard to microblog data quality for Short-Text Topic Modelling (STTM) purposes. This paper addresses this problem and proposes an approach called the Social Media Data Cleansing Model (SMDCM) to improve data quality for STTM. We evaluate SMDCM using six topic modelling methods, namely the Latent Dirichlet Allocation (LDA), Word-Network Topic Model (WNTM), Pseudo-document-based Topic Modelling (PTM), Biterm Topic Model (BTM), Global and Local word embedding-based Topic Modeling (GLTM), and Fuzzy Topic modelling (FTM). We used the Real-world Cyberbullying Twitter (RW-CB-Twitter) and the Cyberbullying Mendeley (CB-MNDLY) datasets in the evaluation. The results proved the efficiency of the GLTM and WNTM over the other STTM models when applying the SMDCM techniques, which achieved optimum topic coherence and high accuracy values on RW-CB-Twitter and CB-MNDLY datasets.

INDEX TERMS Social media, big data, microblogging platforms, topic modeling, data cleansing, data quality, topic coherence, purity.

I. INTRODUCTION

Microblogging platforms such as Twitter have emerged as the primary sources of big data [1], [2], and [3], giving organizations access to previously unattainable opportunities to obtain vital intelligence that will guide their decisions and drive insights. Data quality in the big data context is a specific and critical problem, particularly with Twitter data [4]. In this regard, data cleansing is the most critical process in maintaining the quality of data. It is the most time-consuming step in any text mining process and negatively affects the accuracy of the data if done improperly. Despite the growing literature on the use of Twitter data for various applications [1], [5], [6], [7], [8], [9], [10], [11], [12], [13], and [14], most of the extant work have mainly concentrated on mining and classification of Twitter data (i.e., tweets) while the quality of the data is mostly overlooked [1].

High-quality data is a prerequisite for data-driven applications such as predicting flu trends from twitter data [6], sentiment analysis to assess Airport Service Quality (ASQ) [7], multimodal sentiment analysis [9], and analyzing and capturing tourist activities [15] to guarantee the quality of the

The associate editor coordinating the review of this manuscript and approving it for publication was Arianna Dulizia.

analysis outcome. However, issues with Twitter data quality continue to be a serious and challenging research concern [4], [16]. The conventional data cleansing techniques focus on stop words removal, plural words, and frequent words. The traditional cleansing methods are not fit for microblogging datasets and it may increase the odds of negative data quality. This is because tweets have plenty of anomalies, data sparsity problems, and noises like slang, typos, repeated characters in a word (elongated), complex spelling errors, poorly structured, concatenated words, unconventional usage of acronyms, diversified forms of abbreviations of the same word, short document lengths, varying grammatical structures, and clothed in informal language compared to the long text and normal documents. Therefore, ensuring the data quality collected from microblogging sources is an important research and practical issue that has not yet been adequately addressed [1], [4], [17].

Research in data cleaning has been undertaken in applications such as RFID (Radio Frequency Identification) [18], [19] and online movie reviews [4], [20]. In the context of RFID, the aim is to make the tag read rate as close to the real one as possible by applying data deduplication techniques. In the Topic Modeling (TM) aspect, which is our interest in this paper, converting the slang and acronyms to the normal text, removing the repeating character in a word, splitting the concatenated words of single words, and stemming are the most significant techniques of data cleansing of social media data that are utilized to reduce the feature space, make the task less dependent, and improve the short text topic discovery performance. Also, balanced small-sized datasets are the main focus of the majority of the existing research that examines the impact of preprocessing approaches. Twitter data cleansing in most of the sentiment analysis is largely ignored as the extraction of new sentiment features is mainly the focus [21].

The work presented in this article addresses the problem of data quality issues in the Twitter dataset for use in short-text topic modeling methods. High-quality data is necessary for topic modeling. Topic modeling methods perform poorly when no or little data cleansing is performed [22]. The noisy short text, data sparsity, and scarcity of word co-occurrences nature of Twitter data pose considerable challenges to topic modelling methods [23]. To address this problem, we propose a Social Media Data Cleansing Model (SMDCM) to increase the quality and accuracy of social media data for use in conjunction with modeling methods. The proposed SMDCM can rectify a wide range of abnormalities like typos, slang, complex spelling errors, contraction, emoji, emoticons, repeated characters in a word (elongated), concatenated words, unconventional usage of acronyms, and diversified forms of abbreviations of the same word. The overall contributions of this article can be stated as follows:

- Specific and detailed literature review and comparisons of other short text topic modelling (STTM) using six models: LDA, WNTM, BTM, PTM, GLTM, and FTM.
- A new framework for a data cleansing model specifically tailored to social media dataset for use in topic modelling is proposed.
- We investigate the effects of the Twitter data cleansing method on short-text topic modelling methods' performance.
- We validated the framework through extensive experiments using two real-world social media datasets on six short text topic modelling algorithms in terms of purity, NMI, accuracy, and topic coherence.
- Comparison and choices of different and extensive experiments conducted with various scenarios for evaluating the quality of data, as well as the quality of the topic, utilizing different short text topic modelling algorithms in terms of purity, NMI, accuracy, and topic coherence.
- We present the overall performance improvement rate of the short text topic modelling models with the proposed data cleansing approach as compared to baseline techniques.

The rest of this paper is structured as follows: Section II introduces the problem formulation and related work regarding preprocessing. Section III presents a review of short text topic modeling models. The proposed methodology and mathematical models are described in detail in Section IV. The details of the experimental analysis and results are discussed in Section V. A conclusion with key findings is given in Section VI.

II. PROBLEM FORMULATION AND RELATED WORKS

This section formulates the problem and presents the related works. The problem formulation is presented in the upcoming sub-section.

A. PROBLEM FORMULATION

Given a collection of m Social Media (SM) posts, $\mathcal{D} = \{D_1, D_2, \cdots, D_m\}$, for each social media post D_i in $\mathcal{D}$, $D_i \in \mathcal{D}$, $1 \leq i \leq m$ is an unstructured social media post. The size of each social media post $D_i \in \mathcal{D}$ is defined as given in Eq. (1).

$$Size(D_i) = \sum_{j=1}^{n} \sum_{l=1}^{z} c_{j,l} \tag{1}$$

Each social media post $D_i \in \mathcal{D}$ is tokenized by whitespace, comma, and semicolon and represented as a series of n tokens, $D_i = \left\{Tk_{i,1}, Tk_{i,2}, \cdots, Tk_{i,n}\right\}$. These tokens may be words, numbers, web addresses, URLs, Hashtags, user mentions, punctuation, emoji, emoticon, symbol, elongated words, concatenated words, slang, or acronym. Such that each token $Tk_{i,j} \in D_i$, $1 \leq j \leq n$ consists of z characters $Tk_{i,j} = \left\{c_{j1}, c_{j2}, c_{j3}, \cdots, c_{jz}\right\}$. The SMDCM rules are the operations that remove, transform, or change a token. A set of operations are applied to a collection of tokens in each D_i to generate a new D'_i after applying these operations of SMDCM. For instance, punctuation, URL, and symbol removal rules should return a clean social media post by removing all these noises from every token. When the operations are applied to each social media post in $\mathcal{D}$, then the result is a set of clean and modified social media posts known as $\mathcal{D}'$. The short text topic modelling $\mathcal{T}_{model}$ is to represent a dataset $\mathcal{D}'$ as a set of k topics $T = \{t_j | 1 \leq j \leq k\}$ present in $\mathcal{D}'$. Each social media post, D'_i, will be utilized by $\mathcal{T}_{model}$ model to generate T topics. Subject to the following constraints:

$$S_{min} \leq |Size\ (D_i)| \leq S_{max} \tag{2}$$

$$|\mathcal{D}|_{\mathcal{W}} \approx SMDCM_{\mathcal{W}}, \quad \mathcal{W} \in \{\mathcal{Q}_1\} \tag{3}$$

$$|D|_{\mathcal{W}} \approx \mathcal{T}_{model,\mathcal{W}}, \quad \mathcal{W} \in \{\mathcal{Q}_2, \mathcal{R}, \mathcal{P}, A\} \tag{4}$$

In our study, we will utilize data cleansing techniques to depict their significance and effects for short text topic modelling over social media data.

Where the SMDCM indicates the social media data cleansing techniques, $\mathcal{T}_{\text{model}}$ denotes a short text topic modelling algorithm. Let $\mathcal{Q}_1$ denotes the optimal data quality of the social media posts generated by the SMDCM model, and $\mathcal{Q}_2$ indicates the optimal quality of topics discoverable by $\mathcal{T}_{\text{model}}$. Also, assume that A denotes the optimal accuracy of topics. Let $\mathcal{R}$ indicates the optimal recall of topics discoverable $\mathcal{T}_{\text{model}}$. $\mathcal{P}$ denotes the optimal precision of topics discovered by $\mathcal{T}_{\text{model}}$. Constraint (2) stipulates that the size of every social media post must not be less than (S_{min}) and should not be more than (S_{max}) in social media tweets. This constraint is formulated specifically for social media tweets. In our case, the minimum size of a short text (S_{min}) can be set depending on the quality of the social media post received and S_{max} consists of 280 characters, including blank space. Constraint (3) deals with the optimality of the quality of social media data generated by SMDCM, which enhances the quality of the extracted topics. Constraint (4) deals with the optimality of the four measures: the quality of the extracted topics utilizing topic coherence, recall, precision, and accuracy.

B. RELATED WORK

STTM process mainly includes two phases: preprocessing (social media data cleansing) and topic modelling. This section briefly reviews the existing STTM models based on these phases.

1) PREPROCESSING

Many researchers have investigated the effects of data cleansing and preprocessing on text classification [24]. AL-Ghuribi *et al.* [20] investigated the impacts of different preprocessing methods, such as negation words, stopwords, and the number of occurrence words, in constructing a domain-based lexicon for unbalanced reviews and computing the total review sentiment score. Zin et al. [25] investigated the effectiveness of three preprocessing techniques in Sentiment Analysis (SA): stopwords removal, eliminating (stopwords with meaningless words), and finally eliminating (words less than three characters, numbers, meaningless words, and stopwords).

Sentiment analysis faces major challenges related to data quality [26], [27]. Twitter data cleansing in most of the sentiment analysis is largely ignored as the extraction of new sentiment features is mainly the focus [21]. Murshed et al. [28] investigated the effects of data cleansing on the sentiment analysis performance. Krouska et al. [29] suggested five preprocessing methods to investigate their impacts on the sentiment analysis performance. Sun et al. [30] proposed preprocessing techniques which can conduct (URL, punctuation, numbers, stop word) removal, tokenization, contractions extensions, and lemmatization. Duwairi and El-Orfali [31] studied the impact of various pre-processing techniques like n-gram models, feature correlation on Arabic text sentiment analysis. Some other works studied the impacts of stemming on the Arabic text classification performance, such as [32], [33], [34], and [35].

Topic identification and topic discovery also face major challenges related to data quality [26]. Three prominent and significant heuristic mechanisms have been used to mitigate the data sparsity issue. The first mechanism is to aggregate short texts into pseudo-documents. This mechanism is vastly utilized in social media text data, but it is extremely data-reliant as well. To this extent, Mehrotra et al. [36] aggregated tweets into macro-documents in preprocessing phase based on pooling schemes (Author, hashtags, and burst-score). Hong et al. [37] aggregated all the posts or short texts together which contain the exact term or word. Weng et al. [38] aggregated all the tweets generated by the same twitterer (user). Then, once the pseudo-documents have been generated, the traditional TM approaches, such as LDA, etc., are applied to learn and discover more eminent prevalent significant topics from the richer contexts of the aggregated tweets or short texts. Nonetheless, additional information like hashtags or authorship is not available constantly in real-world applications. The second mechanism is to extend TM by adding robust assumptions of STs documents. Some works, like Lakkaraju et al. [39] and Zhao et al. [40], suppose that each post or ST is a blend of unigrams drawn from a single topic. While other models attempt to leverage the wealthy global word co-occurrence patterns to infer hidden topics such as BTM [41] and PTM [42]. The BTM [41] model was utilized to discover the latent topics from the Short-Texts (STs) by generating word co-occurrence patterns (biterm). Pseudo-document-based Topic Modelling (PTM) and Self-Aggregation-based Topic Modelling (SATM) are the most common models in Self-Aggregation models. SATM [43] assumes that each ST as a sample from a hidden long pseudo-document and merges them automatically to use as a Gibbs sampling [44] for topic extraction; however, it suffers from the over-fitting problem and is computationally expensive. PTM is another model suggested by Zuo et al. [42] for short texts. Here, the pseudo document's concept is to implicitly combine short texts to address data sparsity and the over-fitting issue. Most of these models were developed to mitigate the sparsity issue.

This study takes into account several social media data cleansing/preprocessing techniques and concentrates on an unsupervised STTM instead of the supervised SA and text classification task. Denny and Spirling [45] investigated the effectiveness of the preprocessing techniques on various text classification and topic modelling over political text datasets. However, the investigation with respect to topic modelling was only on Latent Dirichlet Allocation (LDA). Besides, the utilized dataset is smaller than the ones we used for our study, and it is just about 2000 documents. The key aim of the author's research is to study and analyze the differences between supervised and unsupervised learning on text political datasets. Other papers studied the impacts of preprocessing on the performance of topic modelling over speech and newspaper long text. Schofield et al. [46] analyzed and investigated the effectiveness of one preprocessing technique, such as stopwords removal from the corpus, before conducting topic modelling. This method is informative; however, the authors evaluated only one preprocessing method just over newspaper text, and the social media data was not investigated in their study. Churchill and Singh [47] suggested a standardized pre-processing approach for utilizing on-topic modelling over social media data. They showed the influence and usefulness of the proposed approach on topic modelling with various social media data.

Compared to the existing works, this research provides an in-depth analysis of various social media data cleansing models. It investigates their effectiveness and usefulness over short text topic modeling algorithms. We conduct extensive experiments over two real-world social media cyberbullying datasets: the RW-CB-Twitter dataset and CB-MNDLY dataset, and evaluate the topic quality and data quality on short, noisy, and sparse cyberbullying datasets for each scenario utilizing six short text topic modelling algorithms: LDA, BTM, WNTM, PTM, GLTM, and FTM in terms of topic coherence evaluation and two other external evaluations such as short text clustering evaluation, including purity and NMI, and finally, short text classification evaluation such as accuracy.

III. SHORT TEXT TOPIC MODELING

This section presents a review of short text topic modeling techniques. There are numerous models in the literature that address the topic modelling problems based on our previous taxonomy and survey [48]. Some of them concentrated on long text datasets called traditional long text topic modelling models. These models, such as LDA [49], Non-negative Matrix Factorization (NMF), Latent Semantic Analysis (LSA), and Probabilistic LSA (PLSA) are well-known in the unsupervised generative for extracting the hidden topics from the long texts datasets. LDA has been the inspiration for the enormous bulk of other generative TM approaches, such as an extension of the LDA called Twitter-LDA [40], Authorless Topic Models (ATM) [50], and Dynamic topic models (DTM) [51]. As the restrictions of the LDA seem to impact its' performance over short texts, then the research community was focused on the shifting toward conventional LDA modification. To this extent, Chen and Kao [52] suggested an approach to enhance the performance of topic modelling utilizing Re-Organized LDA (RO-LDA), which solves the scarcity of local word co-occurrence of LDA. The limitation of this model is in treating the redundant data. A new model named Time-Sensitive Variational Bayesian inference LDA (TSVB-LDA) was suggested by Fang et al. [53] to discover the latent trending topics with high accuracy. However, TSVB-LDA has demerits in terms of inference of news tweets. The Corpus-based Topic Derivation (CTD) was developed by Sharath et al. [54], which integrates Timestamp-based Popular Hashtag Prediction (TPHP) and Latent Feature-LDA (LF-LDA) utilizing an asymmetric topic model to extract Twitter hidden topics based on corpus semantics. Ni *et al.* [55] introduced the hot event detection approach utilizing Background Removal LDA (BR-LDA), which eliminates the background words from short text tweets. All these models suffer from data sparsity problems with short text datasets due to the scarcity of word co-occurrences in STs.

As a result of the scarcity of word co-occurrences in short texts, many other STTM models have been suggested to extract and reveal the latent topics from the short text datasets. In this research, we focus on the most widely used models for short text datasets, known as STTM. Most of these models were suggested to alleviate the data sparsity problem. Three prominent and significant heuristic mechanisms have been used to mitigate the sparsity of data problems. The first mechanism is to aggregate STs into pseudo-documents. This mechanism is vastly utilized in SM text data, but it is extremely data-reliant as well. To this extent, Mehrotra et al. [36] aggregated tweets into macro-documents in preprocessing phase based on pooling schemes (Author, hashtags, and burst-score), Hong et al. [37] aggregated all the short text posts which contain the same term or word, and Weng *et al.* [37], [38] aggregated all the tweets which generate by the same twitterer (user). Then, once the pseudo-documents have been generated, the traditional TM approaches, such as LDA, etc., are applied in order to learn and discover more eminent prevalent significant topics from the richer contexts of the aggregated tweets or STs. Nonetheless, in real-world applications, additional information like authorship or hashtags are not always available. The second mechanism is to expand TM by adding robust assumptions of STs documents. Some works, like Lakkaraju et al. [39] and Zhao et al. [40], suppose that each post or ST is a blend of unigrams drawn from a single topic.

In contrast, other models attempt to leverage the wealthy global word co-occurrence patterns to infer hidden topics such as BTM [41], PTM [42]. The BTM [41] is one of the well-known Global Word Co-occurrences based models. BTM learns the hidden themes on STs by modelling the generation of biterms directly in the dataset. A biterm is a pair of unordered terms/words that appear together (co-occurrence) in a post/short text. The main concept is that if two terms or words co-occur more repeatedly, they are more probably to pertain to the same topic. Zuo et al. [56] suggested a new method named WNTM utilized for clustering the topic from imbalanced and short texts. WNTM is a novel model that simultaneously addresses the data sparsity problem and imbalance. However, WNTM is unable to express the underlying meaning between words, due to a lack of semantic distance metrics. In addition, WNTM contains a huge data that is not relevant in word-word space. An extension to WNTM, Wang et al. [57] introduced a novel model called Robust WNTM (R-WNTM), which filters the unrelated data during the sampling process is presented as the irrelevant data in the word-word space building procedure of WNTM is high, Jiang *et al.* [58] suggested WNTM with Word2Vector (WNTM-W2V) to discover deep meaning among words to increase the accuracy of relationship among words as well as to improve topic coherence. Wu et al. [59] introduced a clustering method for short texts based on the (BG & SLF–Kmeans) method. In addition, a novel approach named Noise BTM Word Embedding (NBTMWE) was suggested by [60] to resolve the data sparsity problems. This approach integrates the noise BTM and WE from external datasets to ameliorate the coherence of the topic.

Another short text topic modelling is called the Self-Aggregation models. The PTM and SATM are the most common models in Self-Aggregation models. SATM [43] assumes that each ST as a sample from a hidden long pseudo-document and merges them automatically to use as a Gibbs sampling [44] for topic extraction; however, it suffers from the over-fitting problem and is computationally expensive. The PTM is another model proposed by Zuo et al. [42] for short texts. The pseudo document concept implicitly combines short texts to address data sparsity and over-fitting problems. Besides, the authors proposed another model named Sparsity-enhanced PTM (SPTM) by employing Spike and Slab prior method for eliminating the unwanted correlations among the pseudo documents. An extension of the PTM model called Word Embedding-enhanced PTM (WE-PTM) was developed by Zuo *et al.* [61] to leverage pre-trained WEs, which alleviates the data sparsity problem. Feng et al. [62] proposed a User group-based Topic-Emotion model (UGTE) for topic extraction and Emotion detection, which mitigates the data sparsity problems by aggregating the ST of the group into long pseudo-documents. Most of the previous work considered the data sparsity problem; however, they did not consider the sensitivity of word order in short texts.

Moreover, Dirichlet Multinomial Mixture (DMM) based models were proposed to extract and detect the hidden topics from STs. Hence, many studies incorporating the DMM models for STTM followed. Yin and Wang [63] suggested a Gibbs Sampling algorithm for DMM (GSDMM), which used DMM for short text topic clustering and achieved higher efficiency. Besides, they developed a Fast GSDMM (FGSDMM) [64], which acclimatized an online clustering method for initialization. An improved DMM model called Poisson DMM (PDMM) was proposed by Li et al. [65], which is based on modelling the topic number as the Poisson distribution with auxiliary word embedding. An efficient topic modelling named GPU-DMM model was proposed by Li et al. [66] for short text. GPU-DMM enhances the semantic relatedness of words through the sampling process of DMM under the same topic by utilizing the Generalized Polya Urn (GPU) method. These models seem to outperform both the DMM and individual PDMM methods but also involve high computation costs. A new model called a Collaboratively Modeling and Embedding DMM (CME-DMM) was proposed by Liu [67] for capturing coherent hidden topics from STs. All these models were suggested for topic modeling over short text.

In this research, we evaluate the influence of the Social Media Data Cleansing Model (SMDCM) utilizing the most prominent learning short text topic models such as LDA [49], WNTM [56], BTM [41], PTM [42], GLTM [68], and FTM [69] in terms of topic coherence, purity, NMI, accuracy where LDA is the conventional and ubiquitous topic model. The PTM is the prominent model of Self-Aggregation models. BTM, WNTM and GLTM are chosen to represent the Global Word Co-occurrences based Methods. Whereas the FTM is a clustering-based topic modeling, this is based on the fuzzy concept perspective of extracting and discovering the latent topics from the short texts dataset.

IV. PROPOSED METHODOLOGY

The proposed methodology of this research includes the following stages: (I) Social Media data Cleansing Models (SMDCM), (II) Feature extraction using different techniques such as Term Frequency-Inverse Document Frequency (TF-IDF), Global Vectors (GloVe), and Bag of Word (BoW). Then, (III) Applying various short text topic modeling algorithms: LDA [49], BTM [41], WNTM [56], PTM [42], GLTM [68], and FTM [69]. These algorithms are performed and adopted to extract and discover latent topics from social media short text datasets. The workflow of the suggested framework is depicted in Figure 1. As shown in the figure, the input of the framework is the Social Media Short Text which goes through the four stages mentioned above to extract and discover the topics using the short text topic modelling (STTM) algorithms. Then, the results are evaluated using different performance metrics. The findings of the current research are used to better understand how various data cleaning techniques impact the performance improvement rate (PIR) of STTM algorithms.

A. SOCIAL MEDIA DATA CLEANSING MODEL (SMDCM)

The SMDCM stage comprises four sub-stages (as depicted in Figure (1-B), starting with (1) Filtering short texts, (2) Noise elimination, (3) Out of Vocabulary (OOV) cleaning, and (4) Posts Transformation. The aim of these data cleansing and pre-processing stages is to reduce the dimension of social media posts. Thus, the informal posts are converted as possible into formal posts using the proposed SMDCM. Subsequently, the reduced posts are fed into the representation techniques. The following sub-sections present each of these sub-stages in detail.

1) FILTERING AND EXTRACTION POSTS

Filtering is the first stage of the proposed SMDCM model, which focuses only on English posts and ignores the other languages of social media posts from the dataset for further analysis. We utilized the "Tweepy" package for extracting the tweets from Twitter social media platforms using Twitter API streaming. In this stage, the re-tweets are also filtered, and the duplication of tweets is removed utilizing Regular Expression (RegEx) methods to seek hyperlinks in the post and remove the duplication.

2) NOISE ELIMINATION

A noise is a factor that negatively impacts the analysis and the quality of classification results. This subsection presents the pre-processing methods for eliminating the noise of social media data: URL elimination, hashtags/mentions elimination, emoji and emoticon transformation, and punctuation & symbol Elimination.

- *URL ELIMINATION*: it is the process of removing the Uniform Resource Locator (URLs) contained in the posts or tweets. Although these URLs provide a detailed description of the posts, they are deemed unneeded in our study and should be removed from the posts since we just concentrate on meaningful words in the posts. The regular expression based on NLP is used to remove these URLs in the SMDCM model.
- *HASHTAGS/MENTIONS ELIMINATION:* It is the process of removing unnecessary words starting with symbols such as '@' or '#'. The @ symbol is typically used in tweets and posts to mention people's names. While the '#' symbol is used to describe the topic being discussed. For example, "I love when white people talk to black people @Belal #black_people". In our model, the symbols are eliminated using regular expression and the terms, phrases, or the tag which contains the meaningful words or phrases are kept.
- *EMOJI AND EMOTICON TRANSFORMATION:* It is the process of converting the emojis and emoticons to their appropriate word representation to enhance the feature extraction process. Two dictionaries created by NeelShah[1] are used to transform emojis and emoticons into word format. The emojis dictionary has 4,853 emojis, whereas the emoticon dictionary consists of 222 emoticons.
- *CONCATENATED WORDS SPLITTING:* Since the maximum of eliminating special characters such as (%, &, $, etc.), punctuation marks, extra white-space characters, and numbers is to obtain only informative data. We use NLTK and regular expressions for this process. It helps reduce the storage of the dataset and holds just the effective data to be used for other processing, such as classification and topic modelling.

3) OUT OF VOCABULARY (OOV) CLEANSING

This stage is the most crucial one, which identifies and eliminates words/terms that are not in the English dictionary. This stage includes several issues such as concatenated words ('BlackPeopleRacism'), slang (e.g., 'Luv', 'ppl'), elongated words (e.g., 'happppppy'), and contraction. The techniques used to address these issues and enhance data quality are detailed in the following subsections.

- *CONCATENATED WORDS SPLITTING:* Since the maximum capacity for tweets or postings is limited, some Twitter users concatenate their words to create longer tweets. Concatenated words should be broken down into their individual parts, such as the concatenated word "BlackPeopleRacism" should be split into three words ('Black', 'People', 'Racism') using the regular expression technique.
- *ELONGATED WORDS TRANSFORMATION:* This process is responsible for transforming an elongated word into its original word by eliminating repeated letters. On social media, users use elongated words to express their feelings or emotions, such as "I am so happpppppppy to meet you" and "looooooooove you". Using this process, 'happpppppppy' transformed to 'happy' and 'looooooooove' transformed to 'love'. The regular expression technique, the backreferences module, is used to conduct this process. It is a popular technique that permits the text captured by one group in a pattern to be matched to exactly the same text again. It matches and excludes repeated characters from the words of posts.
- *CONTRACTION REPLACEMENT:* This is the most important process in the SMDCM model; it plays a significant role in identifying the tweet or post's sentiments. This process transforms the contractions in social media data such as tweets into a regular lexicon which consists of all the contractions utilized for the transformation. The initial task in this process is to find the contraction pattern and then replace it with the respective pattern from the lexicon. For example, the following contractions: "can't", "didn't", "hasn't", and "won't" should be transformed into "can not", "did not", "has not", and "will not", respectively.
- *SLANGS MODIFICATIONS:* Slang is the vocabulary of an informal language that is frequently used in user-to-user communication, particularly on social media platforms like YouTube, Reddit, Facebook, Instagram, Snapchat, Twitter, and others. The process of converting informal words into their formal (original) words is known as slang modification. Users frequently employ slang words in chat to lower the number of characters in each post or tweet due to the character limit on tweets. For instance, slang terms like "luv," "ppl", and "plz" are not listed in the English dictionary. These

[1] https://github.com/NeelShah18/emot/blob/master/emot/emo_unicode.py

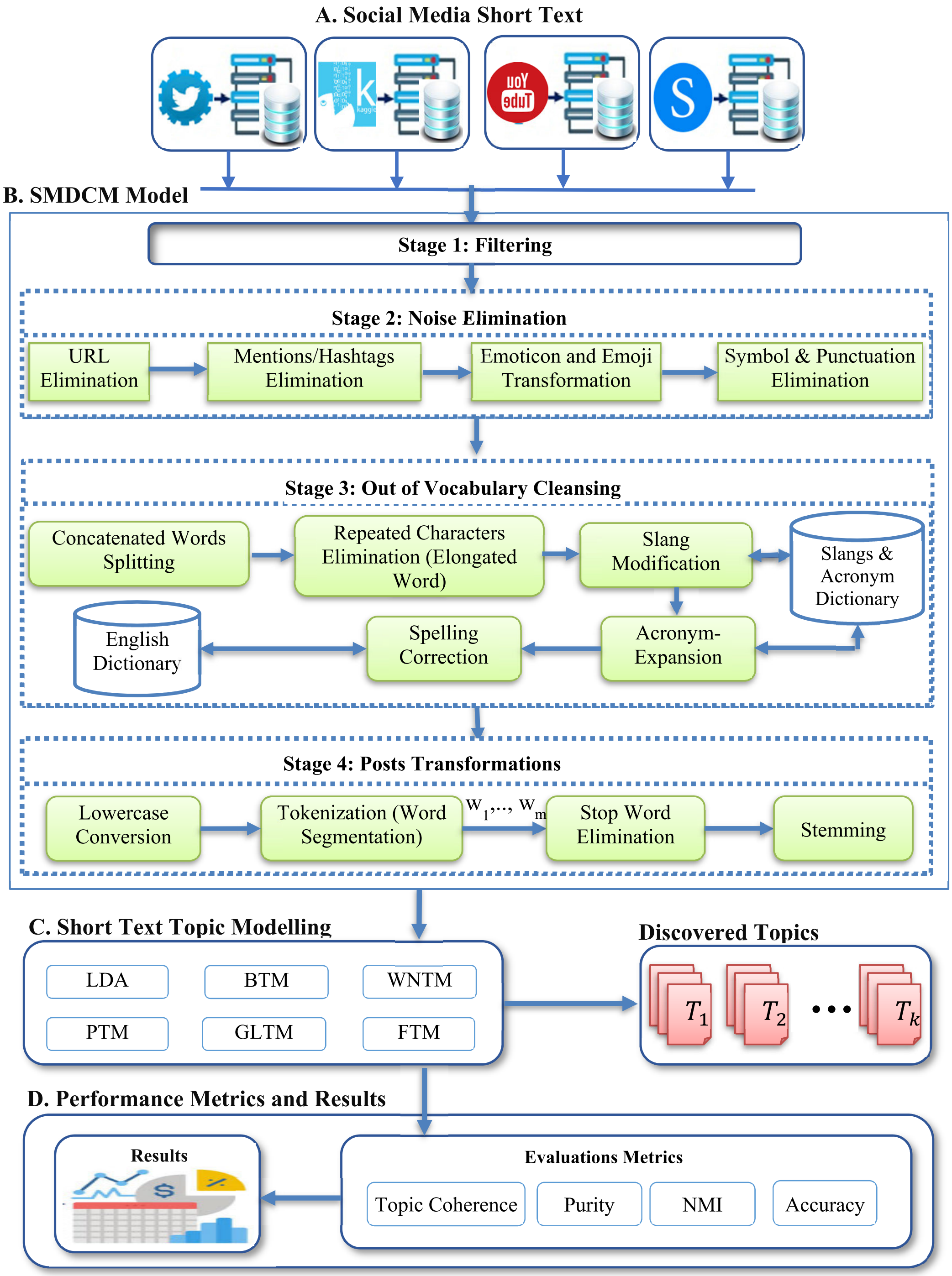
A. Social Media Short Text
B. SMDCM Model
Stage 1: Filtering
Stage 2: Noise Elimination
URL Elimination
Mentions/Hashtags Elimination
Emoticon and Emoji Transformation
Symbol & Punctuation Elimination
Stage 3: Out of Vocabulary Cleansing
Concatenated Words Splitting
Repeated Characters Elimination (Elongated Word)
Slang Modification
Slangs & Acronym Dictionary
English Dictionary
Spelling Correction
Acronym-Expansion
Stage 4: Posts Transformations
Lowercase Conversion
Tokenization (Word Segmentation)
$W_1,.., W_m$
Stop Word Elimination
Stemming
C. Short Text Topic Modelling
LDA
BTM
WNTM
PTM
GLTM
FTM
Discovered Topics
T_1
T_2
T_k
D. Performance Metrics and Results
Results
Evaluations Metrics
Topic Coherence
Purity
NMI
Accuracy

FIGURE 1. Framework of the proposed social media data cleansing and topic modelling.

terms should be changed into official English terms like ''love,'' ''people'', and ''please'', respectively. We constructed a dictionary containing 2864 slang with their formal perspective words to handle this issue. The binary search algorithm is utilized to find the right words in the constructed dictionary for the slang words.

- *SPELLING CORRECTION:* This process involves correcting typos and mistakes that have arisen in the data. The Pyspellchecker package, which can fix a variety of mistakes, is used for this correction.

4) POST TRANSFORMATIONS

This section presents the common pre-processing methods to clean up social media data. These methods are all described in the following subsections.

- *LOWERCASE CONVERSION:* It is the process of lowercasing all letters in all words in a post or tweet in order to give a uniform and consistent format.
- *TOKENIZATION (WORD SEGMENTATION):* It is a fundamental task in most text processing applications. It divides the post or tweet into lexical units (features or words) named tokens. The words in the sentence are often separated by breaks like commas, semicolons, periods, and white space. The NLTK library [70] is utilized for this process.
- *STOP WORDS REMOVAL:* This process eliminates the stop words in the post or tweet. Stop words refer to the words that provide no meaning regarding the content, and most of these words, whether prepositions, pronouns and conjunctions such as 'the', 'she', 'he', 'is', 'a', 'an', etc. The common way of eliminating stopwords is based on pre-compiled lists. Since there are numerous potential stop-word lists, we restrict our attention to choosing whether to eliminate words or terms from posts or tweets using the default list provided using the NLTK library.
- *STEMMING:* This process transforms the word into its base form by removing suffixes and prefixes from the words to get the word roots. It is an important process in NLP because it helps concentrate on the base form of the words in analysis rather than discriminating among different variations of words, which might bring ambiguity during data mining and analysis. As an illustration, the words "eliminate", "eliminated", "eliminating", and "elimination" all have the same root form or stem: "eliminate". The literature contains a wide variety of stemming algorithms. In our model, we used the Porter Stemmer algorithm [71], which is regarded as the most popular technique with English datasets [72].

B. FEATURE EXTRACTION

The preprocessed social media data, such as posts, are represented as a vector of features. Feature extraction is the process of extracting the words from the text or post and converting them into a set of numerical features usable for ML. In this section, three well-known feature extraction methods are used: BoW, TF-IDF, and Glove [73], [74]. The Feature extraction methods are selected for experimental based on original papers of the STTM models. The following subsections describe the mathematical modelling of each feature extraction method.

1) BAG OF WORD (BoW)

The BoW is the most flexible, popular, and simpler technique for extracting features from documents (posts). It is completely based on the occurrence of a term/word in the post. The procedure of tokenization and counting the token occurrences are accomplished in this method. There are numerous parameters in the BoW [75] method that can be used to refine the feature type. The features can be constructed by utilizing these three parameters: the unigram, bigram, and trigram. In our experiment, we utilized unigram. In this case, each term in the post indicates a specific feature name, and the occurrence of each feature is represented using a matrix to make it simpler to comprehend. Hence consider the set of two tweets, T_1 = "I hate when black people talk to black people", and T_2 = "if Black people are old enough to experience racism then white people are old enough to learn about it". The BoW method counts the number of terms that occur most frequently in each post, which may obscure the importance of words that appear less frequently but have more important and relevant features in the post. These are the demerits of BoW, which can be solved by utilizing the *TF-IDF* method, as explained in subsection IV-B-2. After removing the stop words from T_1 and T_2, the first tweet becomes T_1 ="hate black people talk black people" and the second tweet becomes as T_2 =" black people old enough experience racism white people old enough learn". Construct a BoW that consists of all the terms available in both tweets T_1 *and* T_2 without repetition. The Bag can be represented as B = ['hate', 'black', 'people', 'talk', 'old', 'enough', 'experience', 'racism', 'white', 'learn']. After the previous steps are done, the output will be as demonstrated in table 1. Thus, the columns can be represented as features, while the rows can be represented as documents (tweets).

2) TF-IDF

The TF-IDF technique [75] is a weighting matrix mainly utilized as a weighting factor in Information Retrieval (IR). It is utilized to evaluate the significance of a term/word (weight + count) in each post (document) in a given social media dataset. It is made up of two measures: the first measure is Term Frequency (*TF*), and the second measure is Inverse Document Frequency (*IDF*). Mathematicaly, It can be expressed as in a given Eq. (5).

$$TF - IDF = TF\,(w, d) * IDF\,(t, d) \tag{5}$$

$$TF\,(w, d) = \frac{Total\ times\ a\ word\ w\ appear\ in\ document\ d}{Total\ words\ in\ document\ d} \tag{6}$$

$$IDF\,(t, d) = 1 + log\frac{T}{(1 + DF\,(t))} \tag{7}$$

where the term frequency is denoted by $TF\,(w, d)$. It is computed by the total times a word w occurs in the post d by the total words in post d, T is denoted to the total posts presented in the dataset, the number of posts counts (where the term$t$ appears) is denoted by $DF(t)$. Table 2 shows the matrix of TF-IDF for the previous example.

3) GloVe

The GloVe stands for Global Vectors, a word embedding framework that signifies the numerical text representations and offers a semantic similarity measure between the words [73], [74]. Word Embedding (WE) is the words or terms representation in their context and the words or terms around them [76]. WE is generally utilized in various deep learning tasks like semantic analysis, entity recognition, syntactic parsing, etc. The word representations are learned using

TABLE 1. Sparse matrix representation utilizing a bag of word technique.

Tweets/Word	hate	black	people	talk	old	enough	experience	racism	white	learn
T_1	1	2	2	1	0	0	0	0	0	0
T_2	0	1	2	0	2	2	1	1	1	1

TABLE 2. Sparse matrix of TF-ID features.

Tweets/Word	hate	black	people	talk	old	enough	experience	racism	white	learn
T_1	0.407	0.579	0.579	0.407	0	0	0	0	0	0
T_2	0	0.187	0.373	0	0.525	0.525	0.262	0.262	0.262	0.262

the GloVe method by factorizing the w2w (word-word) co-occurrence matrix. The key aim of GloVe is to reduce the reconstruction error (only for positive entries of z), and it is computed as given in Eq. (8).

$$J=\sum_{i=1}^{m}\sum_{j=1}^{m}f(z_{ij})(u_iv_j+b_i^A+b_j^B-\log(z_{ij}))^2 \tag{8}$$

where the vocabulary size is indicated as m, the scalar bias terms associated with words j and i are denoted as b_i^A and $b_j^{\boldsymbol{B}}$, respectively. The $f\left(z_{ij}\right)$ indicates the weighting function, which filters the zero-entries and minimizes the unusual co-occurrences and can be defined as given in Eq. (9).

$$f\left(x_{ij}\right)=\begin{cases}\left(z/z_{max}\right)^{\frac{3}{4}}, & z<z_{max}\\ 1, & Otherwise\end{cases} \tag{9}$$

Hence, GloVe is considered a distributed word depiction framework used to gain vector representation from the words. The GloVe can be applied to discover relations between words like synonyms. The most common drawback of the Glove model is that it requires a lot of memory for storage when trained on the co-occurrence matrix of words. Moreover, if the parameters are changed related to the co-occurrence matrix, then matrix reconstruction is required again, which is very time-consuming.

C. SHORT TEXT TOPIC MODELING ALGORITHMS

The final phase of the STTM framework is discovering and extracting the latent topics from the short text media posts based on the content of the discussion using STTM Models. Topic modelling is automatically discovering the latent topics from the short text dataset. We evaluates the influence of the social media data cleansing model (STDCM) utilizing the most prominent learning short text topic models: LDA [49], BTM [41], WNTM [56], PTM [42], GLTM [68], and FTM [69]. The descriptions of the considered STTM models are discussed in the upcoming subsections.

1) LATENT DIRICHLET ALLOCATION (LDA)

LDA is a prevalent form of an unsupervised and probabilistic topic modelling method used for discovering and extracting the hidden structure topics in social media short text data [49]. The main concept behind LDA is that each document is essentially represented as a probability distribution or a mixture of topics, whereas each topic is represented as a probability distribution over a bunch of words. Those topics are stayed within hidden in the latent layer. The LDA model is based on the assumption of BoW, which neglects the order of words. The generative process of the LDA method for each short text/post $D_i \in \mathcal{D}$ in a corpus $\mathcal{D}$ can be formulated as in the following steps of algorithm 1.

where both parameters β and α represent the dataset level parameters, which are sampled just once in the procedure of generating the dataset. The word-level variables that are taken just once for every word in every document are represented by $z_{d,n}$ and $w_{d,n}$. The number of topics is indicated by K. The parameter φ_k denotes the word probability distribution for the topic k. Finally, the parameter θ_D indicates the document-level variable that is sampled just once per document short text. The posterior called conditional probability is formulated as given in the following Eq. (10).

$$P\left(\beta_k,\theta_D,z_D\,|\,w_D\right)=\frac{P\left(\beta_k,\theta_D,z_D,w_D\right)}{P\left(w_D\right)} \tag{10}$$

where $P\left(w_D\right)$ denotes the marginal probability, which computes the sum of the joint distribution on all instantiations of the latent structure. The variables β_k, θ_D, and z_D are the hidden variables and not observed, which denote the topics, document topic distribution, and word topic assignment, respectively. There are three main kinds of inference methods: expectation propagation [77], a variational method [49], and Gibbs sampling [44], and this article has utilized Gibbs sampling [44].

2) BITERM TOPIC MODEL (BTM)

BTM [41] is one of the well-known Global Word Co-occurrences based models. BTM learns the hidden topics on STs by modelling the generation of biterms directly in the dataset $\mathcal{D}$. A biterm is a pair of unordered words that appear together (co-occurring) in a ST or post. The main concept is that if two words co-occur more repeatedly, they are more probably to pertain to the same topic.

Formally, let us suppose that $\mathcal{D}$ is a dataset consisting of $N_{\mathcal{D}}$ short texts assume it includes n_B biterms $B=\{b_i\}_{i=1}^{n_B}$, where $b_i=\left(w_{i,1},w_{i,2}\right)$. Suppose $z\in[1,K]$ is a topic indicator variable, where the K topics represent over W words in a vocabulary V. The word distribution over topics (for

Algorithm 1 LDA Generative Process

```
1. Sample a topic-word distribution $\beta_{\mathrm{k}} \sim$ Dirichlet ($\varphi$), for each topic k $\in \{1, 2, \ldots, \mathrm{K}\}$
2. For each document d:
3.   Select $\theta_d \sim$ Dirichlet($\alpha$), where $d \in \{1, \ldots, M\}$, Dirichlet ($\alpha$) is the topic distribution with parameter $\alpha$
4.   For each n words $w_n$ in document $d$:
5.       ∘ Select a word-topic assignment $z_{d,n} \sim$ Multinomial($\theta_d$), where $z_{d,n} \in \{1, 2, \ldots, k\}$
6.       ∘ Select a word $w_{d,n} \sim$ Multinomial $\left(\beta_{z_{d,n}}\right)$ where $w_{d,n} \in \{1, 2, \ldots, V\}$
```

example, $P(w \mid z))$ can be defined by $K \times W$ matrix φ. Where the k^{th} row φ_{k} is a w-dimensional multinomial distribution with $\varphi_{k,w} = P(w \mid z = k)$ where $\sum_{w=1}^{W} \varphi_{k,w} = 1$. The propagation of topics in $\mathcal{D}$ dataset (for example, $P(z)$) can be represented by using the K-dimensional multinomial distribution $\theta = \{\theta_k\}_{k=1}^{K}$ with $\theta_k = P(z = k)$ where $\sum_{k=1}^{K} \theta_k = 1$. The generative process of the Biterm Topic Model (BTM) is defined as in algorithm 2.

Algorithm 2 The Generative Process of the BTM

```
1. Draw $\theta \sim$ Dirichlet($\alpha$)
2. For each topic $k \in \{1 \ldots K\}$
3.     Draw $\varphi_k \sim$ Dirichlet($\beta$)
4. For each biterm $b_i \in B$:
5.     Select a topic $z_i \sim$ Multinomial($\theta$)
6.     Select a word $w_{i,1}, w_{i,2} \sim$ Multinomial $\left(\varphi_{z,i}\right)$
```

This model samples the topic $\mathrm{z_i}$ for the biterm $\mathrm{b_i}$ utilizing the collapsed Gibbs sampling technique according to the subsequent conditional distribution as in Eq. (11).

$$P(z_i = k \mid Z_{\neg i}, B) \propto \left(n_{k,\neg i} + \alpha\right) \times \frac{\left(n_{k,\neg i}^{w_{i,1}} + \beta\right)\left(n_{k,\neg i}^{w_{i,2}} + \beta\right)}{\left(n_{k,\neg i} + V\beta + 1\right)\left(n_{k,\neg i} + V\beta\right)} \tag{11}$$

The number of biterms appropriated to topic k, except b_i is represented by $n_{k,\neg i}$, the $z_{\neg i}$ represents the topic's assignments for the entire biterms, excluding the current b_i. The number of times words $w_{i,1}$ and $w_{i,2}$ appropriated to the topic k except b_i are denoted $n_{k,\neg i}^{w_{i,1}}$ and $n_{k,\neg i}^{w_{i,2}}$, respectively. We remove the biterm from its current Topic Feature (TF) vector for every biterm. Thus, by using Eq. (11), we reallocate biterm to the topic. The new topic feature vector is updated using Eq. (12). After completing the iterations, the BTM model estimates φ and θ using Eq. (13) and Eq. (14).

$$n_k^{w_{i,1}} = n_k^{w_{i,1}} + 1, \quad n_k^{w_{i,2}} = n_k^{w_{i,2}} + 1, \quad n_k = n_k + 1 \tag{12}$$

$$\varphi_k^w = \frac{n_k^w + \beta}{n_k + V\beta} \tag{13}$$

$$\theta_k = \frac{n_k + \alpha}{N_B + K\alpha} \tag{14}$$

3) WORD-NETWORK TOPIC MODEL (WNTM)

WNTM utilizes Global Word Co-occurrences (WC) to build a WC Network (WCN). It is a novel framework that simultaneously addresses the data sparsity problem and imbalance texts. WNTM learns the distribution over topics for every word rather than topics for short texts, rendering the WNTM less sensitive to the social media short text length and the topic-distribution heterogeneity. On the other hand, WNTM learns to construct every word's adjacent word-list in the network utilizing hidden word groups and words corresponding to those groups. The following steps of algorithm 3 show the entire pseudo-document generative process.

Algorithm 3 The Entire Pseudo-Document Generative Process in WNTM

```
1. For every hidden word group $z$
2.     Drawn $\varphi_z \sim$ Dirichlet($\beta$), multinomial-distribution
        over words for $z$
3. Draw $\theta_i \sim$ Dirichlet($\alpha$), a latent word group distribution
    for the adjacent word-list $L_i$ of the word $w_i$
4. For each word $w_j \in L_i$:
5.     Choose a hidden word group $z_j \sim \theta_i$
6.     Choose the adjacent word $w_j \sim \varphi_{z,j}$
```

As the WNTM model scans window word by word, two different words in the same window are considered as a co-occurrence. Then, the undirected WCN is constructed by WNTM, where each node of WCN represents one word, and every edge weight denotes the number of the two co-occurrence words. The number of nodes in the networks denotes the number of features in vocabulary V. After that, the Word-Network topic model produces the pseudo-document (PD) l for every vertex v in a word network, which composes of its neighbouring vertices. Following the acquisition of pseudo-documents P, the WNTM model uses Gibbs sampling for LDA to discover hidden topics or themes from generating the PD. The WNTM model infers its hidden topic utilizing the subsequent conditional distribution as given in Eq. (15).

$$P\left(z_{l,w} = k \mid Z_{\neg(l,w)}, P, \alpha, \beta\right) \propto \left(n_{l,\neg(l,w)}^{w} + \alpha\right) \times \frac{n_{k,\neg(l,w)}^{w} + \beta}{n_{k,\neg(l,w)} + V\beta} \tag{15}$$

Assuming a pseudo-document in l is produced from word l, we can compute the topic-word distribution using the given Eq. (16).

$$\varphi_k^w = \frac{n_l^k + \alpha}{n_l + K\alpha} \tag{16}$$

where the number of words or terms in l indicated by n_l. The document-word distribution θ_D is based on the topic-word distribution $\varphi_k^{w_{D,i}}$, it can be computed as given in Eq. (17).

$$\theta_D^w = \sum_{i=1}^{n_d} \varphi_k^{w_{D,i}} p\left(w_{D,i} \mid D\right) \tag{17}$$

$$p\left(w_{D,i} \mid D\right) = \frac{n_D^{w_{D,i}}}{n_D} \tag{18}$$

The number of word $w_{D,i}$ in the post is denoted $n_D^{w_{D,i}}$.

4) PSEUDO-DOCUMENT-BASED TOPIC MODEL (PTM)

PTM is one of the popular self-aggregation based methods proposed specifically for the ST dataset. PTM provided the Pseudo Document's idea to aggregate STs implicitly against the data sparsity problem. PTM supposes an extreme volume of social media short texts are produced from one long pseudo-document pl. Subsequently, learns the hidden topics from long pseudo-documents P instead of STs.

Formally, let us assume that K is a set of topics $\{\varphi_z\}_{z=1}^{K}$, which each represents a multinomial distribution on a vocabulary of size V. Suppose P is the Pseudo-document $\left\{d_l'\right\}_{l=1}^{P}$ and $\mathcal{D}$ is a dataset consisting of d_s short texts $\{d_s\}_{s=1}^{\mathcal{D}}$. Here, the Pseudo-document are the hidden ones, whereas the short texts are the observed ones. A multinomial distribution ψ is utilized for modelling the distribution of STs on pseudo documents. Let us suppose every ST only belongs to one Pseudo-document. Every word or term in ST is produced by first drawing a hidden topic z from the topic distribution θ of the Pseudo-document and subsequently drawing a word $w \sim \varphi_z$. The generative process of the PTM is presented in algorithm 4. In according to the inference, the Sampling pseudo document assignments l for ST ds using collapsed Gibbs sampling. It can be defined as given in Eq. (19).

$$\begin{aligned} p\left(l_{d_s} = 1 \mid rest\right) &\propto \frac{m_l \neg d_s}{D - 1 + P\lambda} \\ &\quad \times \frac{\prod_{z \in d_s} \prod_{j=1}^{n_{d_s}^k} \left(n_{l,\neg d_s}^k + b_{l,k}\alpha + \alpha + j - 1\right)}{\prod_{j=1}^{n_{d_s}} \left(n_{l,\neg d_s} + |A_l|\,\alpha + K\bar{\alpha} + i - 1\right)} \end{aligned} \tag{19}$$

where the length of s^{th} short text d_s is denoted by n_{d_s}, the number of tokens assigned to topic k in d_s is indicated to $n_{d_s}^k$, the short texts number associated with the pseudo-document is denoted by m_l. The total number of tokens in d_l' is represented by n_l, The topic selector of pseudo-document d of topic k is indicated by $b_{l,k}$. The size of $|A_l|$ is denoted by $|A_l|$, $A_l = \{K : b_{l,k} = 1, k \in \{1, \cdots, K\}\}$.

The method to sample the topic assignments k is the same LDA. After getting the pseudo-document, the PTM sample the topic assignments k for each word w in d_s. The θ is drawn from Spike and Slab prior.

$$p\left(z_{d_s,w} = k \mid rest\right) \propto \left(n_{l,d_s}^k + b_{l,d_s}\alpha + \bar{\alpha}\right) \frac{n_k^{w_{d_s}} + \beta}{n_k + V\beta} \tag{20}$$

where $n_k = \sum_{w=0}^{V} n_k^w$, the times number w being assigned to topic k is represented by n_k^w. The document-word distribution is computed using the given Eq. (21).

$$\theta_{,d_s}^k = \frac{n_{d_s}^k + \alpha}{n_{d_s} + K\alpha} \tag{21}$$

Algorithm 4 PTM Generative Process

```
Sample $\psi \sim Dirichlet(\lambda)$
For each topic $k \in \{1 \ldots K\}$
    Sample $\varphi_k \sim Dirichlet(\beta)$
For each Pseudo-document $d_l'$ :
    Sample $\theta_l \sim$ Dirichlet($\alpha$)
For each short text $d_s \in \mathcal{D}$ :
    Sample a Pseudo-document $l \sim$ Multinomial($\psi$)
    For each word $w \in \left\{w_{d,1}, w_{d,1}, \ldots, w_{d,n_d}\right\}$ in $d_s$:
        ∘ Sample a topic $z \sim$ Multinomial($\theta_l$)
        ∘ Select a word $w \sim$ Multinomial ($\varphi_z$)
```

5) GLOBAL AND LOCAL WORD EMBEDDING-BASED TOPIC MODELING (GLTM)

The GLTM [68] trains global embedding from a huge external dataset with a suitable encoding of continuous Skip-Gram method with Negative Sampling (SGNS) for getting local word embedding. This model can extract semantic relatedness among words using both local and global word embeddings in short texts, which the Gibbs sampler can exploit to increase the semantic topic coherence throughout the inference process. Then, the spike-and-slab prior is employed in this model to extract the sparse topic structure for every ST. The Dual-Sparsity topic method is adopted, which specifies a weak smoothing prior for the spike-and-slab structure and a smoothing prior for topic distribution. The spike-and-slab prior can efficiently separate the smoothness of probability distribution and sparsity [78]. Algorithm 5 describes the GLTM generation process in detail.

where the set of STs is denoted by $\mathcal{D}$, the short text (d) length is denoted by N_d, the set of topics is indicated by K. The multinomial distribution on topics of short text d is represented by θ_d, γ is the beta prior for ψ_d, Bernoulli distribution on topic selectors of short text D is denoted by ψ_d. Multinomial distribution on words of topic k is denoted by φ_k. The weak smoothing prior for topic distribution is denoted by b, α is the smoothing prior for topic distribution. The indicator of topic k in short text d is referred to $\psi_{d,k}$. $w_{d,i}$ is the i-th word in short text d. The topic index of word $w_{d,i}$ is referred to $z_{d,i}$.

In the GLTM model, the collapsed Gibbs Sampling is employed to conduct the estimated inference. The latent variables required to be sampled the topic assignment of words z as well as the topic indicators in documents ψ. The maximum posterior estimation (MAP) is used to estimate the three variables parameters like ψ, θ, and φ. The details of model inference can be referred to [68].

Algorithm 5 GLTM Generative Process

```
1.  For every topic $k \in \{1, \ldots, K\}$
2.      Draw word distribution for topic
        $k : \varphi_k \sim \mathrm{Dirichlet}(\beta)$
3.  For each document d $\in \{1, \ldots, |\mathcal{D}|\}$
4.      Draw Bernoulli distribution $\psi_d \sim Beta\,(\gamma)$
5.      For every topic $k \in \{1, \ldots, K\}$
6.          ∘ Draw topic indicator $\text{ɏ}_{d,k} \sim Bernoulli(\psi_d)$
7.      Draw topic distribution for document $d$,
                $\theta_d \sim Dirichlet(\alpha \text{ɏ}_d + b)$
8.      For every word position $i \in \{1, \ldots, N_d\}$
9.          ∘ Draw a topic $z_{d,i} \sim \mathrm{Multinomial}\,(\theta_{\mathrm{d}})$
10          ∘ Draw a textual word
            $w_{d,i} \sim \mathrm{Multinomial}\left(\varphi_{z_{d,i}}\right)$
```

6) FUZZY TOPIC MODELING (FTM)

FTM [69] is a clustering-based topic modelling approach which is based on the fuzzy concept perspective of extracting and discovering the latent topics from the STs corpus. FTM is developed to alleviate the data sparsity problems over STs. In this model, the BOW approach is used to compute the global and local frequencies of terms. Then, Principal Component Analysis (PCA) is utilized to minimize the high dimensionality features. After that, the Fuzzy C-mean technique (FCM) is used to cluster short text and extract the themes from STs data, supposing each cluster as an extracted topic. The overall process for FTM is described in algorithm 6. More detail about this model is presented in [69].

V. EXPERIMENTAL ANALYSIS

This section discusses the experimental analysis and evaluates the performance of the proposed SMDCM and studies its effects on different STTM models: LDA, PTM, BTM, WNTM, GLTM, and FTM in terms of accuracy, Topic Coherence (TC), NMI, and Purity. For evaluation, we utilized two real-world short text social media datasets: real-world Cyberbullying Twitter (RW-CB-Twitter) and Cyberbullying Mendeley (CB-MNDLY). The proposed preprocessing and data cleansing phases have been explained in detail in the SMDCM model in sub-section IV-A. After that, concerning the feature extraction process, the features are extracted using TF-IDF, BOW, and GloVe. The Feature extraction techniques are used based on the technique utilized in the original papers of the considered topic modelling models. Lastly, the STTM models are applied to discover the topics. The effects of the SMDCM over STTM were noted with k different numbers of topics such as $k = \{5, 20, \ \textit{and}\ 40\}$.

A. EXPERIMENTAL SETUP

This subsection provides the experimental setup of this research, like SMDCM configurations and the parameters setting of the considered short text topic models.

1) SMDCM CONFIGURATIONS

The selected models are carried out utilizing Python 3.7.4 programming with an Anaconda IDE-Sypder environment and Java. The suggested SMDCM has been incorporated with many dictionaries, tools, and libraries like an English dictionary and an acronym and slang dictionaries, which provides a collection of all abbreviations and their versions and slang as lookup dictionaries for the purpose of transformation. Besides some of the needed libraries like Gensim [79], Scikit-learn [80], NumPy, pandas, NLTK [81], and Tweepy. In addition to that, we have utilized packages like "SpellChecker", which provides some methods such as "Correction" and "Spell" to correct and check the spelling mistakes in the respective datasets. For the experiment, we construct a set of SMDCM settings and compare the differences and similarities to depict the functionality of the SMDCM. Here, we select two scenarios of settings for both RW-CB-Twitter and CB-MNDLY datasets. The first scenario is called "with baseline" techniques, consisting of only some techniques like tokenizatiom, lowercase conversion, punctuation removal, and stopwords elimination. The second scenario is the suggested preprocessing techniques, known as "with SMDCM" techniques, which consists of four stages with all the proposed tasks (as depicted in Figure 1-B), starting with (1) Filtering short texts, (2) Noise removal, including URL Elimination, Mentions and Hashtags Elimination, Emoji and Emoticon Transformation, and Punctuation & Symbol (3) Out of Vocabulary (OOV) cleaning, such as concatenated words splitting, contraction replacement, elongated words transformation, slangs modifications, and spelling correction (4) Posts Transformation including lowercase conversion, tokenization, stop words removal, and stemming.

2) PARAMETERS SETTING

The parameters setting of all the considered short text topic modelling approaches are set as given in the original articles. The number of iterations is fixed to 1000 for all the approaches. The value of α is set 0.05 for LDA and $\alpha = 50/K$ for both BTM and GLTM, whereas we fixed $\alpha = 0.1$ with WNTM and PTM models. The value of the λ hyperparameter is fixed at $\lambda = 0.1$ and $\lambda = 0.5$ with PTM, and GLTM approaches, respectively. We fixed $\beta = 0.01$ for all the following models LDA, BTM, PTM, WNTM, and GLTM. The value of the sliding window for WNTM is set to 10. We fixed the number of pseudo-documents for PTM to 1000. We evaluate the effects of the SMDCM over the considered topic discovery models with k different numbers of topics such as $\mathrm{k} = \{5, 20, \ \mathrm{and}\ 30\}$.

B. DATASETS

In this subsection, we explain in brief the utilized datasets in the analysis of the experiments. The evaluation is performed over two social media datasets: the publicly available Cyberbullying Mendeley dataset collected by Elsafoury [82] we refer to as (CB-MNDLY), and the

Algorithm 6 Fuzzy Topic Modelling (FTM) model

```
Functions BOW (), IDFS: IDFS (), IDF (): IDF, E (): Entropy (), U (): Unary, FCM (), PCA ()
Preprocessing of data
BOW(CleanTextData)
Compute the LTW
Compute the Global term weighting (GTW)
Eliminate the high dimensionality effect on GTW models utilizing PCA ()
$FCM\left(E, F, n, d_i, f_i, \mu_{i,j}\right)$
Compute the probability of short texts for each GTW
Compute the probability of documents in topics. $P\left(Z_j \mid Y_k\right)$,
    $P\left(Z_j, Y_k\right) = P\left(Y_k \mid Z_j\right) \times P\left(Z_j\right) \quad (22)$
Normalize $P(Z, Y)$ for each topic utilizing $P\left(Z_j \mid Y_k\right) = \frac{P\left(Z_j, Y_k\right)}{\sum_{j=1}^{n} P\left(Z_j, Y_k\right)} \quad (23)$
Compute the words in short texts (documents) probability $P\left(X_i \mid Z_j\right)$.
    $P\left(X_i \mid Z_j\right) = \frac{P\left(X_i, Z_j\right)}{\sum_{i=1}^{m} P\left(X_i, Z_j\right)} \quad (24)$
Compute the words in topics probability $P\left(X_i \mid Y_k\right)$
    $P\left(X_i \mid Y_k\right) = \sum_{j=1}^{n} P\left(X_i, Z_j\right) \times P\left(Z_j \mid Y_k\right) \quad (25)$
```

TABLE 3. The statistics of utilized datasets.

SN.	Name of Datasets	# Short Text	# Topics (labels)	Data Sources
1	CB-MNDLY	50,000	6	Kaggle, Twitter, YouTube, Talk pages, and Wikipedia.
2	RW- CB-Twitter	20,000	5	Twitter

other Real-world Cyberbullying Twitter (RW-CB-Twitter) dataset. Table 3 shows the statistics of these datasets. The descriptions of these datasets are provided in the upcoming subsections.

1) RW-CB-TWITTER DATASET

This dataset is collected from the Twitter social media platform by selecting some cyberbullying key terms such as whale, bitch, LGBTQ, fucking, idiot, sucker, fuck, pussy, nigger, poser, moron, etc., using API Twitter streaming as recommended by the authors in psychology literature [83], and [84]. Besides, some other key terms related to racism as recommended by [85], such as black, hate, Islamic, threat, Islam, terrorist, attack, racism, ban, and kill. The number of gathered tweets included in the RW-CB-Twitter dataset is 435764 tweets. We selected 20000 tweets randomly after deleting irrelevant tweets and re-tweet and utilized them in this research for the evaluations. This dataset is expanded to the collected dataset utilized in [14] and classified it into five classes: Not-bullying, sexism, racism, aggressive, and insult.

2) CB-MNDLY DATASET

This dataset is freely available in the data repository of Mendeley[2] for research purposes. It is collected by Elsafoury [82] from various SM sources such as Kaggle, Twitter, YouTube, Talk pages, and Wikipedia. The different types of cyberbullying like aggression, racism, hate speech, insults, sexism, and toxicity, are included in this corpus (dataset); each of them is kept in a separate file and categorized as bullying and not-bullying. We combined these files to generate a new dataset we refer to as the CB-MNDLY dataset, composed of 6 data classes, including Insult, racism, sexism, aggression, toxicity, and not-bullying. The CB-MNDLY dataset contains 448880 short texts. We selected 50,000 short texts out of the combined dataset and used them in this work to evaluate the effect of data cleansing on short text topic discovery models.

C. EVALUATION METRICS

In this subsection, we introduce the evaluation metrics for evaluating the effects of the proposed SMDCM techniques on the STTMs. To provide a good assessment, we evaluate all the considered models from many perspectives utilizing various metrics such as topic coherence evaluation, two other evaluations like short text clustering evaluation, including purity and NMI, and short text classification evaluation, such as accuracy. The descriptions of these metrics are explained as follows:

[2] https://data.mendeley.com/datasets/jf4pzyvnpj/1

1) TOPIC COHERENCE (TC)

TC is a metric utilized to assess the quality of extracted topics. For every topic k of post generated, the *TC* is employed to the top N words $(W_1, \ldots\ldots, W_N)$. We chose 10 top-most words as a sliding window in the experiment. It computed the semantic score of a particular topic by assessing the semantic similarity degree of the topic's high-scoring words. To compute TC, we require an external dataset (e.g. Wikipedia) to score pairs of words using the term co-occurrence. Here, the TC is computed using Normalized PMI (NPMI) [86] instead of PMI [87] as provided below in Eq. (26), where the score (w_j, w_l) denotes the NPMI.

$$Topic\ Coherence\ (K) = \frac{2}{N(N-1)} \sum_{j=1}^{N-1} \sum_{l=j+1}^{N} score(w_j, w_l) \tag{26}$$

$$score\left(w_j, w_l\right) = \frac{log\frac{P(w_j, w_l)+\varepsilon}{P(w_j)P(w_l)}}{-logP(w_j, w_l)} \tag{27}$$

2) SHORT TEXT CLUSTERING EVALUATION METRICS (NMI AND PURITY)

Short text clustering is a significant application of STTM. We select the maximum value from its topic probability distribution for each social media post as the cluster label. Then, the golden and cluster labels are compared using the clustering evaluation metrics NMI and Purity.

a: PURITY

The purity metric is utilized to evaluate the ratio of an appropriate number of correctly clustered posts (short texts) to all the labelled posts (golden label) in the corpus. The value of purity lies between 0 and 1. It is defined as in Eq. (28).

$$Purity = \frac{1}{N} \sum_{i=1}^{|A|} \sum_{j=1}^{|B|} max \left| a_i \cap b_j \right| \tag{28}$$

where the total posts in the corpus (dataset) is denoted to N. The group of clusters is denoted as $A = \left\{a_1, \ldots, a_{|A|}\right\}$, and the group of ground-truth (labelled) clusters in datasets can be represented as $B = \left\{b_1, \ldots, a_{|B|}\right\}$.

b: NMI [88]

It is a metric used to calculate the Mutual Information $I\ (A, B)$ shared between A and B, whose range is normalized to [0,1]. Where H (A) and H (B) are the entropy metrics of clusters and classes, respectively. The NMI is formulated as given in Eq. (29).

$$NMI\ (A, B) = \frac{2 * I(A, B)}{[H\ (A) + H(B)]} \tag{29}$$

$$I\ (A, B) = \sum_{i=1}^{|A|} \sum_{j=1}^{|B|} \left[P(a_i \cap b_j) log \frac{P\left(a_i \cap b_j\right)}{P\left(a_i\right) P\left(b_j\right)} \right]$$

$$= \sum_{i=1}^{|A|} \sum_{j=1}^{|B|} \left[\frac{\left|a_i \cap b_j\right|}{M} log \frac{M \left|a_i \cap b_j\right|}{|a_i|\left|b_j\right|} \right] \tag{30}$$

$$H\ (A) = -\sum_{i=1}^{|A|} P(a_i) log P(a_i)$$

$$= -\sum_{i=1}^{|A|} \frac{|a_i|}{M} log \frac{|a_i|}{M} \tag{31}$$

$$H\ (B) = -\sum_{j=1}^{|B|} P(b_j) log P(b_j)$$

$$= -\sum_{j=1}^{|B|} \frac{\left|b_j\right|}{M} log \frac{\left|b_j\right|}{M} \tag{32}$$

The final formula of the NMI can be defined as given in Eq. (33)

$$NMI\ (A, B) = \frac{\sum_{i=1}^{k} \sum_{j=1}^{p} \left[\frac{\left|a_i \cap b_j\right|}{M} log \frac{M\left|a_i \cap b_j\right|}{|a_i|\left|b_j\right|} \right]}{\left[\sum_{i=1}^{k} \frac{|a_i|}{M} \log \frac{|a_i|}{M} + \sum_{j=1}^{p} \frac{\left|b_j\right|}{M} log \frac{\left|b_j\right|}{M} \right] \Big/ 2} \tag{33}$$

3) SHORT TEXT CLASSIFICATION EVALUATION

Each social media post can be represented by document-topic distribution $P\left(z \mid D\right)$. Text classification can be used to evaluate the topic modelling performance. Therefore, we select accuracy as a measure for short text classification. Accuracy can be expressed as the ratio of an appropriate correctly all predicted observations to the total predictions [89], where the higher accuracy indicates that the learned themes are more representative and discriminative. We utilize the SVM classifier for this task. The classification accuracy is calculated with five fold cross-validation over both CB-MNDLY and RW-CB-Twitter datasets. It is computed as defined in Eq. (34).

$$Accuracy = \frac{T_p + T_N}{T_p + T_N + F_P + F_N} \tag{34}$$

4) PERFORMANCE IMPROVEMENT RATE (PIR)

In this study, the PIR is defined as the improvement rate of STTM models' performance with SMDCM compared to STTM models without SMDCM. It can be expressed as given in Eq. (35).

$$PIR = \frac{|A-B|}{B} * 100 \tag{35}$$

$$A = \sum_{j=k_i}^{n} Perf_M \left(STTM\ with\ SMDCM_j\right) \tag{36}$$

$$B = \sum_{j=k_i}^{n} Perf_M \left(STTM\ with\ Baseline_j\right) \tag{37}$$

where $k_i = \{5, 20,\ and\ 30\}\ and\ i = 1, 2,\ and\ 3$, M is the performance metric, n denotes the number of k^{th} times.

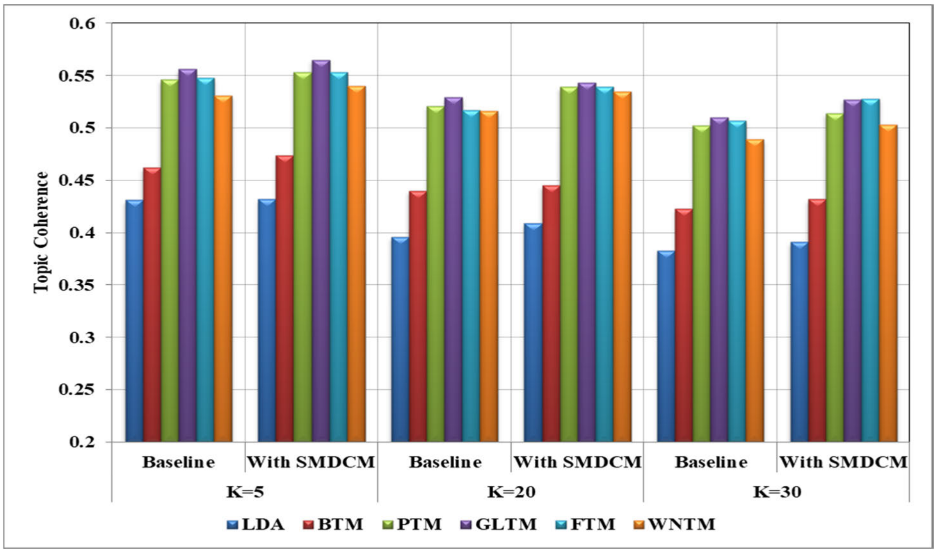

FIGURE 2. Topic coherence results with $k = \{5, 20, 30\}$ topics on the RW-CB-Twitter dataset with baseline and SMDCM.

D. RESULTS AND DISCUSSION

In this subsection, we discuss the results of the effects of preprocessing and data cleansing techniques on the short text topic modelling from three perspectives in terms of four metrics such as topic coherence, classification accuracy, clustering (purity, NMI). In addition, we show the Performance Improvement Rate (PIR) of the STTM with SMDCM scenario over STTM with baseline scenario on two cyberbullying datasets: RW-CB-Twitter dataset and CB-MNDLY dataset, in terms of all performance metrics.

1) TOPIC COHERENCE EVALUATION RESULT WITH SMDCM AND BASELINE TECHNIQUES

This subsection investigates the effects of preprocessing (SMDCM) over short text topic modelling in terms of topic coherence *(TC)* metric on both RW-CB-Twitter and CB-MNDLY datasets. In the case of the RW-CB-Twitter dataset, all the considered STTM models operated on this dataset, and the evaluation has been performed with various topics such as $k = \{5, 20, \ and\ 30\}$. When $k = 5$, the topic coherence values of GLTM, FTM, and WNTM are 0.565, 0.553, and 0.540 with the SMDCM scenario, respectively. Where the GLTM yields good topic coherence compared to other STTM models. The GLTM, FTM, and WNTM have 0.556, 0.548, and 0.531 of topic coherence without SMDCM (with only baseline scenario). Similarly, when the number of topics is $k = 30$, the FTM has yielded a high topic coherence of 0.528 with the SMDCM model and 0.507 without SMDCM. We observed that the preprocessing (SMDCM) effects on short text topic modelling results in discovering topics in terms of topic coherence metric, as depicted in Figure 2. In addition, we have investigated the impacts of the SMDCM over STTM models on the CB-MNDLY dataset. Figure 3 depicts the results of topic coherence of the considered topic models with $k = \{5, 20, 30\}$ topics on the CB-MNDLY dataset with and without data cleansing model and show the effects of the SMDCM over the short text topic discovery. In case $k = 5, 20, \ and\ 30$, the WNTM yielded the best result compared to other models of 0.634, 0.600, and 0.593 of topic coherence with SMDCM and followed by the GLTM, which yields 0.629, 0.592, and 0.579 of topic coherence with preprocessing in case of $k = 5, 20, \ and\ 30$, respectively. Whereas the topic coherence decreases without preprocessing SMDCM, as depicted in Figure 3. The interesting observation is that the GLTM yields a high topic coherence value of 0.565 with SMDCM when k = 5 over the RW-CB-Twitter dataset, whereas the WNTM has got 0.634 of topic coherence value which is the best topic coherence value when $k = 5$ with SMDCM over all the models with and without SMDCM in case of CB-MNDLY dataset. We can conclude that social media data cleansing (SMDCM) affects the performance of the short text topic discovery models, as presented in Figures 2 and 3.

Table 4 depicts the PIR of the short text topic modelling models with SMDCM over short text topic modelling models with baseline over both RW-CB-Twitter and CB-MNDLY datasets in terms of topic coherence. The PIR is computed as formulated in Eq. (35). In case of RW-CB-Twitter, the PIRs of the STTM models LDA, BTM, PTM, GLTM, FTM, and WNTM are 1.82%, 1.96%, 2.36%, 2.51%, 3.05%, and 2.73%, respectively. Similarly, the PIRs of the STTM models with SMDCM scenario over the same Models without SMDCM (with baseline scenario) are 4.70%, 5.65%, 4.97%, 5.04%, 4.28%, and 3.63%, respectively. This improvement proves the effectiveness of the SMDCM on short text topic modelling.

2) ACCURACY EVALUATION RESULTS WITH SMDCM AND BASELINE TECHNIQUES

Here, in this subsection, six topic modelling approaches, such as LDA, PTM, BTM, WNTM, GLTM, and FTM, have been

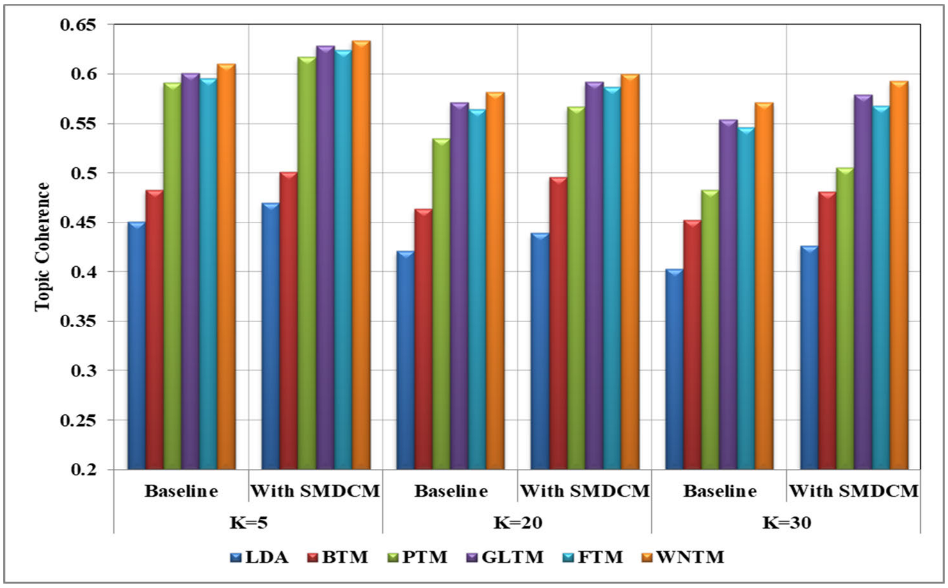

FIGURE 3. Topic coherence results with $k = \{5, 20, 30\}$ topics on CB-MNDLY dataset with baseline and SMDCM.

TABLE 4. The overall performance improvement rate (%) of STTM with SMDCM over STTM with baseline in terms of topic coherence.

Models	RW-CB-Twitter Dataset						CB-MNDLY Dataset					
	LDA	BTM	PTM	GLTM	FTM	WNTM	LDA	BTM	PTM	GLTM	FTM	WNTM
$\sum_{j=k_i}^{n}(Topic\ Coherence\ With\ SMDCM)_j$	1.232	1.351	1.606	1.635	1.62	1.578	1.335	1.478	1.689	1.813	1.779	1.827
$\sum_{j=k_i}^{n}(Topic\ Coherence\ With\ Baseline)_j$	1.21	1.325	1.569	1.595	1.572	1.536	1.275	1.399	1.609	1.726	1.706	1.763
PIR over LDA with Baseline	**1.82**	-	-	-	-	-	**4.70**	-	-	-	-	-
PIR over BTM with Baseline	-	**1.96**	-	-	-	-	-	**5.65**	-	-	-	-
PIR over PTM with Baseline	-	-	**2.36**	-	-	-	-	-	**4.97**	-	-	-
PIR over GLTM with Baseline	-	-	-	**2.51**	-	-	-	-	-	**5.04**	-	
PIR over FTM with Baseline	-	-	-	-	**3.05**	-	-	-	-	-	**4.28**	
PIR over WNTM with Baseline	-	-	-	-	-	**2.73**	-	-	-	-	-	**3.63**

run over two cyberbullying datasets: RW-CB-Twitter and CB-MNDLY, along with and without the proposed SMDCM model. The evaluations have been performed with k different numbers of topics such as $k = \{5, 20, \textit{and } 30\}$ topics. Figure 4 shows the results of accuracy with $k = \{5, 20, 30\}$ topics on the RW-CB-Twitter dataset with and without SMDCM and show the effects of the SMDCM over the short text topic discovery. When $k = 5$, the WNTM and FTM yield good results of 77.42% and 77.43% of accuracy with preprocessing (SMDCM), whereas they have got 75.65% and 75.33% of accuracy without SMDCM, respectively, as depicted in Figure 4. The performance improvement rate over WNTM and FTM without the SMDCM model or with the baseline preprocessing is 2.34% and 2.78% when $k = 5$. In contrast, the classification accuracy of LDA with SMDCM is 72.41% which is the lowest accuracy compared to all other models with SMDCM when $k = 5$ topics. Similarly, when the number of topics is $k = 30$, the WNTM has got high accuracy of 78.53% with the SMDCM model (preprocessing) and 77.85% with the baseline (without SMDCM). We conclude that the WNTM is the best model choice with Social media data cleansing SMDCM; in contrast, the accuracy result decreases with baseline (without SMDCM) over the RW-CB-Twitter dataset. Besides, we have studied another dataset named Cyberbullying Mendeley (CB-MNDLY) to investigate the effectiveness of SMDCM over short text topic modelling approaches. Figure 5 provides the accuracy results with $k = \{5, 20, 30\}$ topics on the CB-MNDLY dataset with and without SMDCM (Preprocessing). In case $k = 30$, the GLTM achieved the best result of 81.87% accuracy and followed the WNTM model, which yields 81.31% with (SMDCM) preprocessing. Whereas the GLTM and WNTM have 79.75% and 78.89% of accuracy without SMDCM, as depicted in Figure 5. Similarly, when $k = 20$, the GLTM has the highest accuracy with SMDCM compared to other short text topic modelling methods. In contrast, in case $k = 5$, the WNTM has the best accuracy result, followed by the GLTM, which achieves 79.54% and 78.96% of accuracies,

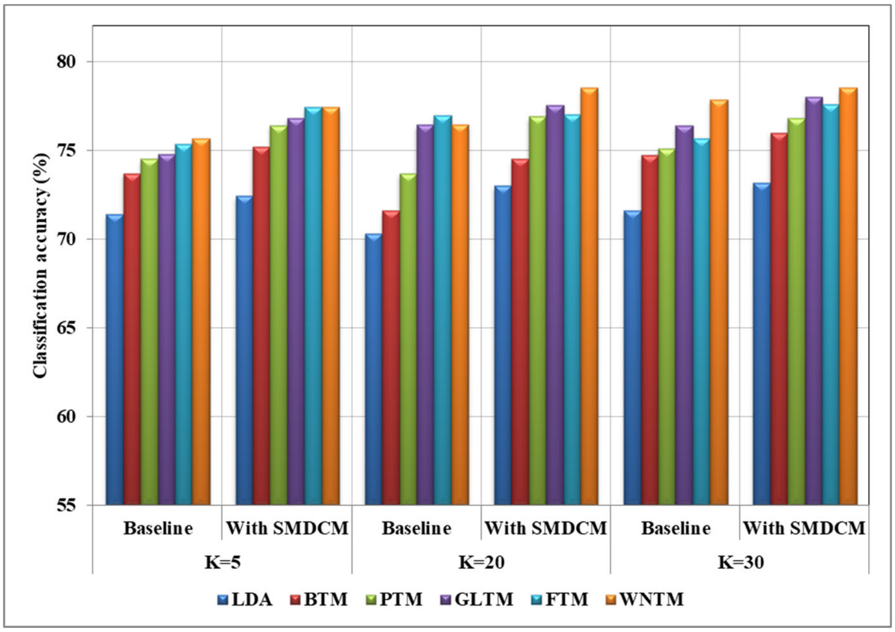

FIGURE 4. Accuracy results with k = {5, 20, 30} topics on RW-CB-Twitter dataset with baseline and SMDCM.

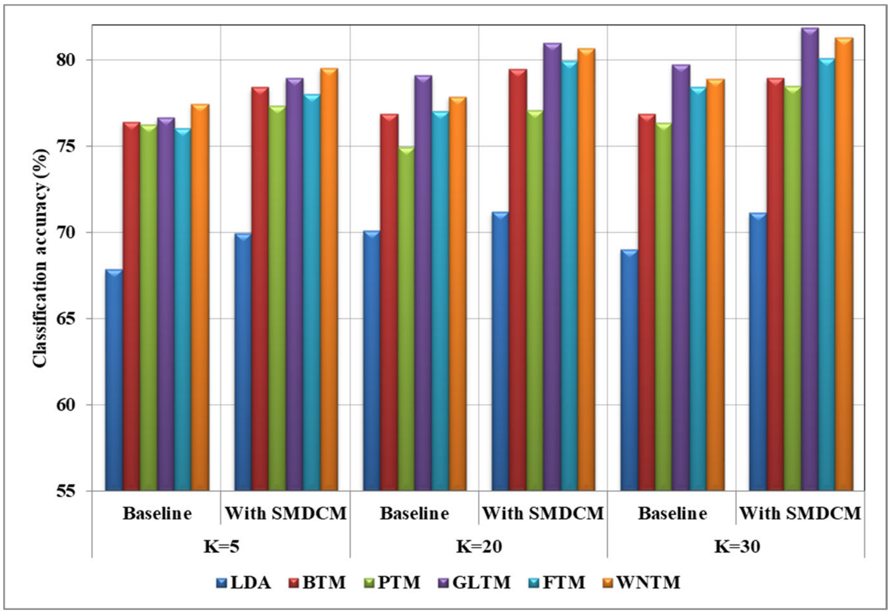

FIGURE 5. Accuracy results with *k* = {5, 20, 30} topics on CB-MNDLY Dataset with baseline and SMDCM.

respectively. In general, we conclude that the GLTM and WNTM have the best results and the SMDCM preprocessing effects on short text topic modelling performance, as shown in Figure 5.

The Performance Improvement Rate (PIR %) of STTM models with SMDCM over STTM models with baseline over both RW-CB-Twitter and CB-MNDLY datasets in terms of accuracy is provided in Table 5. The PIR is computed based on Eq. (35). In case of RW-CB-Twitter, the PIRs of the STTM models LDA, BTM, PTM, GLTM, FTM, and WNTM are 2.46%, 2.59%, 3.04%, 2.08%, 1.80%, and 1.96%, respectively. Similarly, the PIRs of the STTM models with SMDCM scenario over the same models with baseline scenario over the CB-MNDLY dataset are 2.56%, 2.89%, 2.33%, 2.67%, 2.84%, and 3.12%, respectively. This improvement proves the effectiveness of the SMDCM on STTM models.

3) PURITY EVALUATION RESULTS WITH SMDCM AND BASELINE TECHNIQUES

This subsection studies the effects of suggested SMDCM over short text topic modelling on both RW-CB-Twitter and CB-MNDLY datasets in terms of short text clustering

TABLE 5. The overall performance improvement rate (%) of STTM with SMDCM over STTM with baseline in terms of accuracy.

Models	RW-CB-Twitter Dataset						CB-MNDLY Dataset					
	LDA	BTM	PTM	GLTM	FTM	WNTM	LDA	BTM	PTM	GLTM	FTM	WNTM
$\sum_{j=k_i}^{n}(\text{Accuracy}\ \ With\ SMDCM)_j$	218.55	225.7	230.1	232.39	232.04	234.48	212.28	236.81	232.83	241.83	238.06	241.54
$\sum_{j=k_i}^{n}(\text{Accuracy}\ \ With\ Baseline)_j$	213.3	220.01	223.31	227.65	227.93	229.97	206.98	230.15	227.53	235.53	231.48	234.21
PIR over LDA with Baseline	**2.46**	-	-	-	-	-	**2.56**	-	-	-	-	-
PIR over BTM with Baseline	-	**2.59**	-	-	-	-	-	**2.89**	-	-	-	-
PIR over PTM with Baseline	-	-	**3.04**	-	-	-	-	-	**2.33**	-	-	-
PIR over GLTM with Baseline	-	-	-	**2.08**	-	-	-	-	-	**2.67**	-	-
PIR over FTM with Baseline	-	-	-	-	**1.80**	-	-	-	-	-	**2.84**	-
PIR over WNTM with Baseline	-	-	-	-	-	**1.96**	-	-	-	-	-	**3.12**

TABLE 6. The overall performance improvement rate (%) of STTM with SMDCM over STTM with baseline in terms of purity.

Models	RW-CB-Twitter Dataset						CB-MNDLY Dataset					
	LDA	BTM	PTM	GLTM	FTM	WNTM	LDA	BTM	PTM	GLTM	FTM	WNTM
$\sum_{j=k_i}^{n}(\text{Purity}\ With\ SMDCM)_j$	2.09	2.189	2.273	2.281	2.273	2.278	2.215	2.432	2.413	2.512	2.498	2.482
$\sum_{j=k_i}^{n}(\text{Purity}\ With\ Baseline)_j$	2.01	2.15	2.198	2.205	2.203	2.225	2.17	2.394	**2.361**	2.456	2.439	2.43
PIR over LDA with Baseline	**3.98**	-	-	-	-	-	**2.07**	-	-	-	-	-
PIR over BTM with Baseline	-	**1.81**	-	-	-	-	-	**1.59**	-	-	-	-
PIR over PTM with Baseline	-	-	**3.41**	-	-	-	-	-	**2.20**	-	-	-
PIR over GLTM with Baseline	-	-	-	**3.45**	-	-	-	-	-	**2.28**	-	-
PIR over FTM with Baseline	-	-	-	-	**3.18**	-	-	-	-	-	**2.42**	-
PIR over WNTM with Baseline	-	-	-	-	-	**2.38**	-	-	-	-	-	**2.14**

(Purity). In the case of the RW-CB-Twitter dataset, all the considered STTM models operated on this dataset, and we evaluated the STTM models with different topics such as $k = \{5, 20,\ and\ 30\}$. When $k = 5$, the purity values of WNTM, GLTM, and FTM are 0.775, 0.768, and 0.766 with SMDCM, respectively. where the WNTM offers good purity compared to other topic modelling approaches. Whereas the WNTM, GLTM, and FTM have got 0.754, 0.742, and 0.735 with baseline techniques. Similarly, when the number of topics is $k = 30$, the GLTM yields high purity of 0.754 with the SMDCM model (preprocessing) and 0.722 with baseline (without SMDCM). We noted that the SMDCM effects on STTM result in discovering topics in terms of clustering purity metric, as depicted in Figure 6.

In addition, we have investigated the effectiveness of the SMDCM over short text topic discovery on the CB-MNDLY dataset. Figure 7 shows the results of purity of the considered topic modelling models with $k = \{5, 20, 30\}$ topics on the CB-MNDLY dataset with SMDCMand baseline techniques and show the effects of the SMDCM over the short text topic discovery. In case $k = 5, 20$, and 30, the GLTM yields the best result compared to other models of 0.853, 0.842, and 0.817 purity with SMDCM, respectively, and followed by the FTM, which yields 0.849, 0.838, and 0.811of purity with preprocessing SMDCM in case of $k = 5, 20,\ and\ 30$, respectively. Whereas the purity decreases when investigated without SMDCM (baseline), as depicted in Figure 6. We can conclude that social media data cleansing (SMDCM) can impact the performance of the STTM models, as presented in Figures 6 and 7.

This paragraph analyzes and discusses the PIR (%) of the considered STTM models with the social media data cleansing model over the same models with the baseline scenario in terms of NMI. Table 6 presents the PIR of STTM models with SMDCM over STTM with baseline scenario on both CB-MNDLY and RW-CB-Twitter datasets. It can be concluded from PIR values of purity that the proposed SMDCM positively affects the Short Text topic Modeling approaches over both social media Cyberbullying datasets, as shown in Table 6.

4) NMI EVALUATION RESULTS WITH SMDCM AND BASELINE TECHNIQUES

In this sub-section, we study the effectiveness of the SMDCM in terms of NMI results over STTM models LDA, PTM, BTM, WNTM, GLTM, and FTM on the RW-CB-Twitter and CB-MNDLY datasets. In the RW-CB-Twitter dataset, the evaluations have been performed with k different numbers

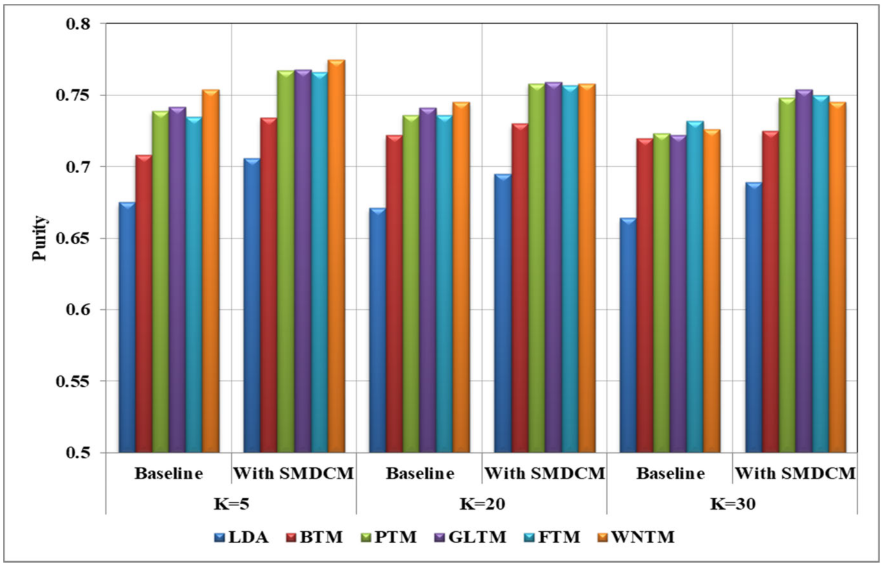

FIGURE 6. Purity results with $k = \{5, 20, 30\}$ topics on RW-CB-Twitter dataset with baseline and SMDCM.

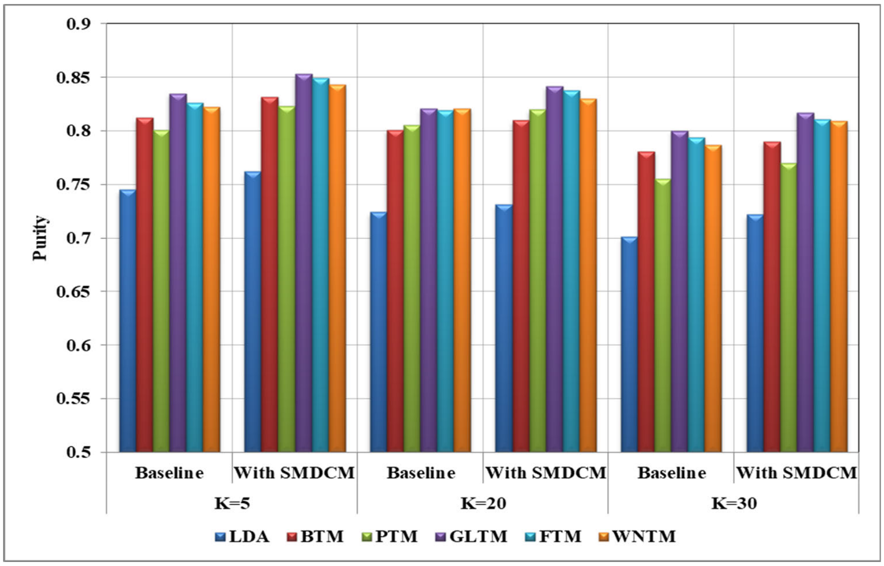

FIGURE 7. Purity results with $k = \{5, 20, 30\}$ topics on CB-MNDLY dataset with baseline and SMDCM.

of topics such as $k = \{5, 20, \textit{ and } 30\}$. Figure 8 shows the results of NMI on the RW-CB-Twitter dataset and shows the effectiveness of the SMDCM over the STTM models. When $k = 5, 20, \textit{ and } 30$, the WNTM yields good results of 0.695, 0.688, and 0.678 of NMI with SMDCM techniques, respectively, whereas without SMDCM, the NMI values of WNTM when $k = 5, 20, \textit{ and } 30$ are 0.672, 0.663, and 0.649, respectively. Followed the GLTM and FTM, which achieved the second and third-best results of NMI with all the different topics. Here, the performance improvement rates (%) of WNTM with SMDCM over WNTM without SMDCM when $k = 5, 20, \textit{and } 30$ are 3.42%, 3.77%, and 4.47%, respectively. In contrast, the NMI values of LDA with SMDCM when $k = 5, 20,$ and 30 are 0.626, 0.607, and 0.615, respectively, which are the lowest NMI values compared to all other models with SMDCM, whereas the NMI values of LDA without SMDCM when $k = 5, 20,$ and 30 are 0.601, 0.585, and 0.600, respectively. We conclude that the WNTM is found to be the best model choice with social media data cleansing SMDCM; in contrast, the NMI decreases without SMDCM.

In addition, we have studied the CB-MNDLY dataset to investigate the effectiveness of SMDCM over short text topic modelling approaches. Figure 9 provides the NMI results with $k = \{5, 20, 30\}$ topics on the CB-MNDLY dataset with baseline and SMDCM (Preprocessing). In the case of $k = 5, 20, \textit{ and } 30$, the GLTM has achieved the best results of 0.849, 0.851, and 0.838 of NMI with SMDCM, while the values of NMI of GLTM with baseline when $k = 5, 20, \textit{ and } 30$ are 0.827, 0.835, and 0.818, respectively, as depicted in Figure 9. Followed that the WNTM

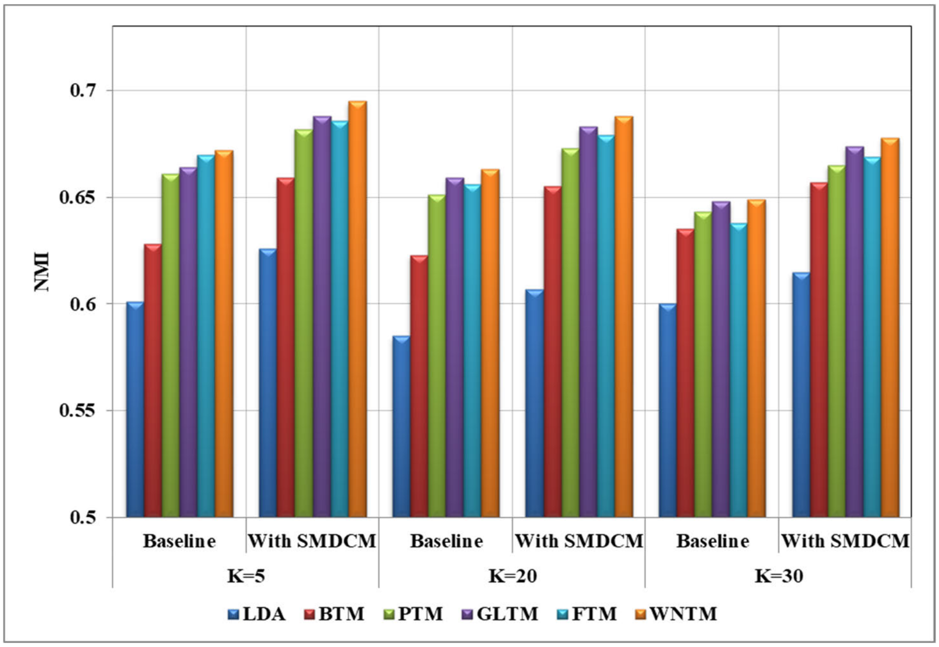

FIGURE 8. NMI results with $k = \{5, 20, 30\}$ topics on RW-CB-Twitter with baseline and SMDCM.

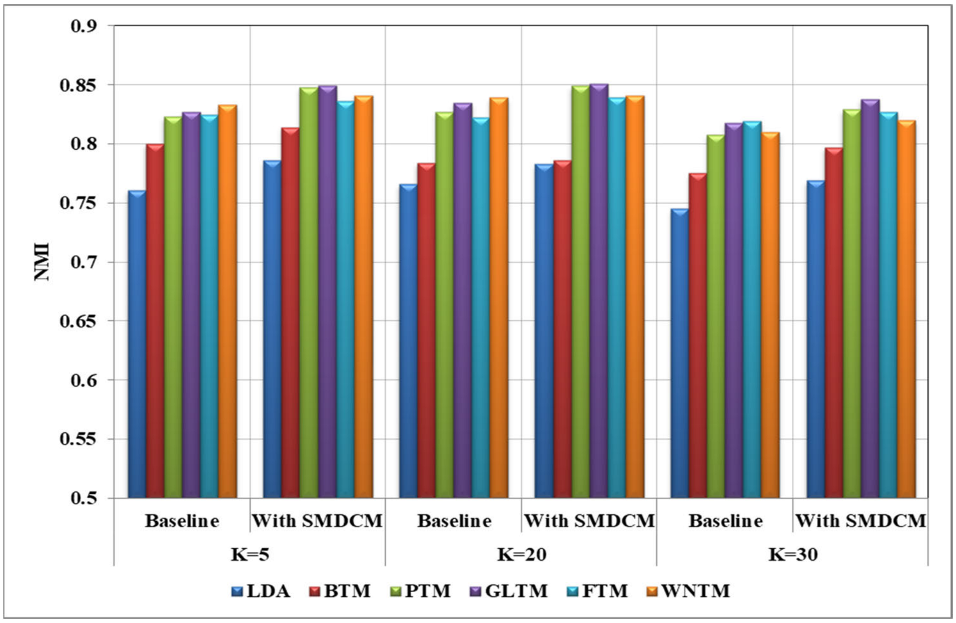

FIGURE 9. NMI results with $k = \{5, 20, 30\}$ topics on CB-MNDLY dataset with baseline and SMDCM.

achieved 0.841 and 0.841 of NMI with (SMDCM) preprocessing when $k = 5$ and 20 over the CB-MNDLY dataset. The FTM has achieved the second-best value of NMI 0.827 with SMDCM when $k = 30$ as depicted in Figure 9. The interesting observation is that the GLTM has got 0.851 of NMI value which is the best NMI value when $k = 20$ with SMDCM over all the models with baseline and SMDCM in the case of the CB-MNDLY dataset, while the WNTM yields high NMI value of 0.695 with SMDCM when $k = 5$ over RW-CB-MNDLY dataset. In general, we conclude that the NMI values increase with SMDCM (preprocessing) and decrease somewhat without SMDCM, and we investigate the SMDCM effects on short text topic modelling performance.

Here, we discuss the PIR (%) of the STTM model with proposed SMDCM over the same considered STTM with baseline scenario in terms of NMI metric on both datasets. For RW-CB-Twitter, the short text topic modelling approaches LDA, BTM, PTM, GLTM, FTM, and WNTM with SMDCM scenario generate 3.47%, 4.51%, 3.32%, 3.75%, 3.56%, and 3.88% of NMI improvements over the same models with the baseline scenario, respectively. In the case of using

TABLE 7. The overall performance improvement rate (%) of STTM with SMDCM over STTM with baseline in terms of NMI.

Models	RW-CB-Twitter Dataset						CB-MNDLY Dataset					
	LDA	BTM	PTM	GLTM	FTM	WNTM	LDA	BTM	PTM	GLTM	FTM	WNTM
$\sum_{j=k_i}^{n}(\text{NMI } \mathit{With\ SMDCM})_j$	1.848	1.971	2.02	2.045	2.034	2.061	2.338	2.397	2.527	2.538	2.502	2.502
$\sum_{j=k_i}^{n}(\text{NMI } \mathit{With\ Baseline})_j$	1.786	1.886	1.955	1.971	1.964	1.984	2.272	2.359	2.458	2.48	2.466	2.482
PIR over LDA with Baseline	**3.47**	-	-	-	-	-	**2.90**	-	-	-	-	-
PIR over BTM with Baseline	-	**4.51**	-	-	-	-	-	**1.61**	-	-	-	-
PIR over PTM with Baseline	-	-	**3.32**	-	-	-	-	-	**2.81**	-	-	-
PIR over GLTM with Baseline	-	-	-	**3.75**	-	-	-	-	-	**2.34**	-	-
PIR over FTM with Baseline	-	-	-	-	**3.56**	-	-	-	-	-	**1.46**	-
PIR over WNTM with Baseline	-	-	-	-	-	**3.88**	-	-	-	-	-	**0.81**

the CB-MNDLY dataset, the LDA, BTM, PTM, GLTM, FTM, and WNTM models with Social Media Data Cleansing Model (SMDCM) produce 2.90%, 1.61%, 2.81%, 2.34%, 1.46%, and 0.81% NMI improvements over these models with baseline. The overall performance improvement rate (%) of STTM with SMDCM over STTM with baseline in terms of NMI is presented in Table 7. From the results and PIRs values, we conclude that the suggested SMDCM positively affects the performance of STTM.

VI. CONCLUSION

As the use of Twitter data in topic modeling is increasing, improving the quality of social media data before processing it to derive value and insight from social media datasets represents an important and challenging requirement. This paper introduced a model called SMDCM for addressing the data quality problem in social media. Moreover, it investigated the impact of SMDCM on the performance of short text topic modelling (STTM) using six models: LDA, WNTM, BTM, PTM, GLTM, and FTM. Extensive experiments were conducted with various scenarios over two social media datasets: RW-CB-Twitter and CB-MNDLY for evaluating the quality of data, as well as the quality of topic for each scenario, utilizing different short text topic modelling algorithms in terms of purity, NMI, accuracy, and topic coherence. The experimental results showed that the STTM performance highly depends on data cleansing (SMDCM) techniques and the used dataset's nature. It can be concluded that SMDCM has an impact on the performance of TM and the quality of data. The results proved the efficiency of the GLTM and WNTM over the other STTM models when applying the SMDCM techniques, which achieved optimum topic coherence and high accuracy values on the RW-CB-Twitter and CB-MNDLY datasets.

ACKNOWLEDGMENT

Belal Abdullah Hezam Murshed would like to thank the Department of Studies in Computer Science and the Chairman of the Department, Prof. D. S. Guru at the University of Mysore in India for all the facilities offered for this research, also would like to thank the Department of Computer Science, University of Amran, Amran, Yemen, for their support and all the facilities provided for this research work, and also would like to thank his brother Dr. Samir Abdullah Hizam [ENT consultant], for his constant support financially and morally. The help of Maliha Omar is highly appreciated.

REFERENCES

[1] F. Arolfo, K. C. Rodriguez, and A. Vaisman, "Analyzing the quality of Twitter data streams," *Inf. Syst. Frontiers*, vol. 24, no. 1, pp. 349–369, Feb. 2022, doi: 10.1007/s10796-020-10072-x.

[2] S. A. Curiskis, B. Drake, T. R. Osborn, and P. J. Kennedy, "An evaluation of document clustering and topic modelling in two online social networks: Twitter and reddit," *Inf. Process. Manag.*, vol. 57, no. 2, Mar. 2020, Art. no. 102034, doi: 10.1016/j.ipm.2019.04.002.

[3] J. Abawajy, "Comprehensive analysis of big data variety landscape," *Int. J. Parallel, Emergent Distrib. Syst.*, vol. 30, no. 1, pp. 5–14, 2015, doi: 10.1080/17445760.2014.925548.

[4] L. Cai and Y. Zhu, "The challenges of data quality and data quality assessment in the big data era," *Data Sci. J.*, vol. 14, p. 2, May 2015, doi: 10.5334/dsj-2015-002.

[5] C. Zhang, S. Lu, C. Zhang, X. Xiao, Q. Wang, and G. Chen, "A novel hot topic detection framework with integration of image and short text information from Twitter," *IEEE Access*, vol. 7, pp. 9225–9231, 2018, doi: 10.1109/ACCESS.2018.2886366.

[6] B. Alkouz, Z. A. Aghbari, and J. H. Abawajy, "Tweetluenza: Predicting flu trends from Twitter data," *Big Data Mining Anal.*, vol. 2, no. 4, pp. 273–287, Dec. 2019, doi: 10.26599/BDMA.2019.9020012.

[7] L. Martin-Domingo, J. C. Martín, and G. Mandsberg, "Social media as a resource for sentiment analysis of airport service quality (ASQ)," *J. Air Transp. Manag.*, vol. 78, pp. 106–115, Jul. 2019, doi: 10.1016/j.jairtraman.2019.01.004.

[8] U. Kursuncu, M. Gaur, U. Lokala, K. Thirunarayan, A. Sheth, and I. B. Arpinar, "Predictive analysis on Twitter: Techniques and applications," in *Emerging Research Challenges and Opportunities in Computational Social Network Analysis and Mining*, N. Agarwal, N. Dokoohaki, S. Tokdemir, Eds. Cham, Switzerland: Springer, 2019, pp. 67–104.

[9] A. Kumar and G. Garg, "Sentiment analysis of multimodal Twitter data," *Multimedia Tools Appl.*, vol. 78, no. 17, pp. 24103–24119, Sep. 2019, doi: 10.1007/s11042-019-7390-1.

[10] J. H. Abawajy, M. I. H. Ninggal, and T. Herawan, "Privacy preserving social network data publication," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 1974–1997, 3rd Quart., 2016, doi: 10.1109/COMST.2016.2533668.

[11] T. Lynn, P. Rosati, B. Nair, and C. M. A. Bhaird, "An exploratory data analysis of the crowdfunding network on Twitter," *J. Open Innov., Technol., Market, Complex.*, vol. 6, no. 3, p. 80, Sep. 2020, doi: 10.3390/joitmc6030080.

[12] J. A. Caetano, H. S. Lima, M. F. Santos, and H. T. Marques-Neto, ''Using sentiment analysis to define Twitter political users' classes and their homophily during the 2016 American presidential election,'' *J. Internet Services Appl.*, vol. 9, no. 1, p. 18, Dec. 2018, doi: 10.1186/s13174-018-0089-0.

[13] M. Mendoza, B. Poblete, and I. Valderrama, ''Nowcasting earthquake damages with Twitter,'' *EPJ Data Sci.*, vol. 8, no. 1, pp. 1–23, Dec. 2019, doi: 10.1140/epjds/s13688-019-0181-0.

[14] B. A. H. Murshed, J. Abawajy, S. Mallappa, M. A. N. Saif, and H. D. E. Al-Ariki, ''DEA-RNN: A hybrid deep learning approach for cyberbullying detection in Twitter social media platform,'' *IEEE Access*, vol. 10, pp. 25857–25871, 2022, doi: 10.1109/ACCESS.2022.3153675.

[15] H. Q. Vu, G. Li, R. Law, and Y. Zhang, ''Tourist activity analysis by leveraging mobile social media data,'' *J. Travel Res.*, vol. 57, no. 7, pp. 883–898, Sep. 2018, doi: 10.1177/0047287517722232.

[16] C. Salvatore, S. Biffignandi, and A. Bianchi, ''Social media and Twitter data quality for new social indicators,'' *Social Indicators Res.*, vol. 156, nos. 2–3, pp. 601–630, Aug. 2021, doi: 10.1007/s11205-020-02296-w.

[17] M. A. Qureshi, M. Asif, M. F. Hassan, A. Abid, A. Kamal, S. Safdar, and R. Akber, ''Sentiment analysis of reviews in natural language: Roman Urdu as a case study,'' *IEEE Access*, vol. 10, pp. 24945–24954, 2022, doi: 10.1109/ACCESS.2022.3150172.

[18] H. Kamaludin, H. Mahdin, and J. H. Abawajy, ''Filtering redundant data from RFID data streams,'' *J. Sensors*, vol. 2016, pp. 1–7, Jan. 2016, doi: 10.1155/2016/7107914.

[19] H. Mahdin and J. Abawajy, ''An approach for removing redundant data from RFID data streams,'' *Sensors*, vol. 11, no. 10, pp. 9863–9877, Oct. 2011, doi: 10.3390/s111009863.

[20] S. M. Al-Ghuribi, S. A. Noah, and S. Tiun, ''Various pre-processing strategies for domain-based sentiment analysis of unbalanced large-scale reviews,'' in *Proc. Int. Conf. Adv. Intell. Syst. Inform. (AISI)*, vol. 1261, 2021, pp. 204–214, doi: 10.1007/978-3-030-58669-0_19.

[21] Z. Jianqiang and G. Xiaolin, ''Comparison research on text pre-processing methods on Twitter sentiment analysis,'' *IEEE Access*, vol. 5, pp. 2870–2879, 2017, doi: 10.1109/ACCESS.2017.2672677.

[22] R. Churchill, L. Singh, and C. Kirov, ''A temporal topic model for noisy mediums,'' in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining* (Lecture Notes in Computer Science), vol. 10938. Cham, Switzerland: Springer, 2018, pp. 42–53.

[23] S. M. Al-Ghuribi, S. A. M. Noah, and S. Tiun, ''Unsupervised semantic approach of aspect-based sentiment analysis for large-scale user reviews,'' *IEEE Access*, vol. 8, pp. 218592–218613, 2020, doi: 10.1109/ACCESS.2020.3042312.

[24] V. Srividhya and R. Anitha, ''Evaluating preprocessing techniques in text categorization,'' *Int. J. Comput. Sci. Appl.*, vol. 47, no. 11, pp. 49–51, 2010. [Online]. Available: http://sinhgad.edu/ijcsa-2012/pdfpapers/1_11.pdf

[25] H. M. Zin, N. Mustapha, M. A. A. Murad, and N. M. Sharef, ''The effects of pre-processing strategies in sentiment analysis of online movie reviews,'' in *Proc. AIP Conf.*, vol. 1891, 2017, Art. no. 020089, doi: 10.1063/1.5005422.

[26] A. Chinnov, P. Kerschke, and C. Meske, ''An overview of topic discovery in Twitter communication through social media analytics full paper,'' in *Proc. 21st Amer. Conf. Inf. Syst.*, 2015, pp. 1–10. [Online]. Available: http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1274&context=amcis2015

[27] M. A. Qureshi, M. Asif, M. F. Hassan, G. Mustafa, M. K. Ehsan, A. Ali, and U. Sajid, ''A novel auto-annotation technique for aspect level sentiment analysis,'' *Comput., Mater. Continua*, vol. 70, no. 3, pp. 4987–5004, 2022, doi: 10.32604/cmc.2022.020544.

[28] B. A. H. Murshed, S. Mallappa, O. A. M. Ghaleb, and H. D. E. Al-Ariki, *Efficient Twitter Data Cleansing Model for Data Analysis of the Pandemic Tweets* (Studies in Systems, Decision and Control), vol. 348. Cham, Switzerland: Springer, 2021, pp. 93–114, doi: 10.1007/978-3-030-67716-9_7.

[29] A. Krouska, C. Troussas, and M. Virvou, ''The effect of preprocessing techniques on Twitter sentiment analysis,'' in *Proc. 7th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2016, pp. 1–5, doi: 10.1109/IISA.2016.7785373.

[30] F. Sun, A. Belatreche, S. Coleman, T. M. McGinnity, and Y. Li, ''Pre-processing online financial text for sentiment classification: A natural language processing approach,'' in *Proc. IEEE Conf. Comput. Intell. Financial Eng. Econ. (CIFE)*, Mar. 2014, pp. 122–129, doi: 10.1109/CIFEr.2014.6924063.

[31] R. Duwairi and M. El-Orfali, ''A study of the effects of preprocessing strategies on sentiment analysis for Arabic text,'' *J. Inf. Sci.*, vol. 40, no. 4, pp. 501–513, 2014, doi: 10.1177/0165551514534143.

[32] Y. A. Alhaj, J. Xiang, D. Zhao, M. A. A. Al-Qaness, M. A. Elaziz, and A. Dahou, ''A study of the effects of stemming strategies on Arabic document classification,'' *IEEE Access*, vol. 7, pp. 32664–32671, 2019, doi: 10.1109/ACCESS.2019.2903331.

[33] R. Mamoun and M. Ahmed, ''Arabic text stemming: Comparative analysis,'' in *Proc. Conf. Basic Sci. Eng. Stud. (SGCAC)*, Feb. 2016, pp. 88–93, doi: 10.1109/SGCAC.2016.7458011.

[34] A. Noaman and S. Al-ghuribi, ''A new approach for Arabic text classification using light stemmer and probabilities,'' *Int. J. Academic Res.*, vol. 4, no. 3, pp. 114–122, 2012.

[35] R. M. Sallam, H. M. Mousa, and M. Hussein, ''Improving Arabic text categorization using normalization and stemming techniques,'' *Int. J. Comput. Appl.*, vol. 135, no. 2, pp. 38–43, 2016, doi: 10.5120/ijca2016908328.

[36] R. Mehrotra, S. Sanner, W. Buntine, and L. Xie, ''Improving LDA topic models for microblogs via tweet pooling and automatic labeling,'' in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2013, pp. 889–892, doi: 10.1145/2484028.2484166.

[37] L. Hong and B. D. Davison, ''Empirical study of topic modeling in Twitter,'' in *Proc. 1st Workshop Social Media Anal. (SOMA)*, 2010, pp. 80–88, doi: 10.1145/1964858.1964870.

[38] J. Weng, E.-P. Lim, J. Jiang, and Q. He, ''TwitterRank: Finding topic-sensitive influential Twitterers,'' in *Proc. 3rd ACM Int. Conf. Web Search Data Mining (WSDM)*, 2010, pp. 261–270, doi: 10.1145/1718487.1718520.

[39] H. Lakkaraju, I. Bhattacharya, and C. Bhattacharyya, ''Dynamic multi-relational Chinese restaurant process for analyzing influences on users in social media,'' in *Proc. IEEE 12th Int. Conf. Data Mining*, Dec. 2012, pp. 389–398, doi: 10.1109/ICDM.2012.54.

[40] W. X. Zhao, J. Jiang, J. Weng, J. He, E. P. Lim, H. Yan, and X. Li, ''Comparing Twitter and traditional media using topic models,'' in *Proc. Eur. Conf. Inf. Retr.* Berlin, Germany: Springer, 2011, pp. 338–349, doi: 10.1007/978-3-642-20161-5_34.

[41] X. Cheng, X. Yan, Y. Lan, and J. Guo, ''BTM: Topic modeling over short texts,'' *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 12, pp. 2928–2941, Dec. 2014, doi: 10.1109/TKDE.2014.2313872.

[42] Y. Zuo, J. Wu, H. Zhang, H. Lin, F. Wang, K. Xu, and H. Xiong, ''Topic modeling of short texts: A pseudo-document view,'' in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 2105–2114, doi: 10.1145/2939672.2939880.

[43] X. Quan, C. Kit, Y. Ge, and S. J. Pan, ''Short and sparse text topic modeling via self-aggregation,'' in *Proc. 24th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2015, pp. 2270–2276.

[44] T. L. Griffiths and M. Steyvers, ''Finding scientific topics,'' *Proc. Nat. Acad. Sci. USA*, vol. 101, no. 1, pp. 5228–5235, 2004, doi: 10.1073/pnas.0307752101.

[45] M. J. Denny and A. Spirling, ''Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it,'' *Political Anal.*, vol. 26, no. 2, pp. 168–189, Apr. 2018, doi: 10.1017/pan.2017.44.

[46] A. Schofield, M. Magnusson, and D. Mimno, ''Pulling out the stops: Rethinking stopword removal for topic models,'' in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics*, vol. 2, 2017, pp. 432–436, doi: 10.18653/v1/E17-2069.

[47] R. Churchill and L. Singh, ''TextPrep: A text preprocessing toolkit for topic modeling on social media data,'' in *Proc. 10th Int. Conf. Data Sci., Technol. Appl.*, 2021, pp. 60–70, doi: 10.5220/0010559000600070.

[48] B. A. H. Murshed, S. Msallappa, J. Abawajy, M. A. N. Saif, H. D. E. Al-Ariki, and H. M. Abdulwahab, ''Short text topic modelling approaches in the context of big data: Taxonomy, survey, and analysis,'' *Artif. Intell. Rev.*, 2022, doi: 10.1007/s10462-022-10254-w.

[49] D. M. Blei, A. Y. Ng, and M. I. Jordan, ''Latent Dirichlet allocation,'' *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[50] L. Thompson and D. Mimno, ''Authorless topic models: Biasing models away from known structure,'' in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 3903–3914. [Online]. Available: https://aclanthology.org/C18-1329

[51] D. M. Blei and J. D. Lafferty, ''Dynamic topic models,'' in *Proc. 23rd Int. Conf. Mach. Learn. (ICML)*, vol. 148, 2006, pp. 113–120, doi: 10.1145/1143844.1143859.

[52] G.-B. Chen and H.-Y. Kao, ''Word co-occurrence augmented topic model in short text,'' *Intell. Data Anal.*, vol. 21, pp. 55–70, Apr. 2017, doi: 10.3233/IDA-170872.

[53] A. Fang, C. Macdonald, I. Ounis, P. Habel, and X. Yang, ''Exploring time-sensitive variational Bayesian inference LDA for social media data,'' in *Proc. Eur. Conf. Inf. Retr.* Cham, Switzerland: Springer, 2017, pp. 252–265.

[54] K. B. R. Sharath, W. Kuochen, and S. Shi-Min, ''Corpus-based topic derivation and timestamp-based popular hashtag prediction in Twitter,'' *J. Inf. Sci. Eng.*, vol. 35, no. 3, pp. 675–696, 2019, doi: 10.6688/JISE.201905_35(3).0011.

[55] N. Ni, C. Guo, and Z. Zeng, ''Public opinion clustering for hot event based on BR-LDA model,'' in *Proc. Int. Conf. Intell. Inf. Process.* Cham, Switzerland: Springer, 2018, pp. 3–11.

[56] Y. Zuo, J. Zhao, and K. Xu, ''Word network topic model: A simple but general solution for short and imbalanced texts,'' *Knowl. Inf. Syst.*, vol. 48, no. 2, pp. 379–398, Aug. 2016, doi: 10.1007/s10115-015-0882-z.

[57] F. Wang, R. Liu, Y. Zuo, H. Zhang, H. Zhang, and J. Wu, ''Robust word-network topic model for short texts,'' in *Proc. IEEE 28th Int. Conf. Tools Artif. Intell. (ICTAI)*, Nov. 2016, pp. 852–856, doi: 10.1109/ICTAI.2016.0132.

[58] M. Jiang, R. Liu, and F. Wang, ''Word network topic model based on Word2Vector,'' in *Proc. IEEE 4th Int. Conf. Big Data Comput. Service Appl. (BigDataService)*, Mar. 2018, pp. 241–247, doi: 10.1109/BigDataService.2018.00043.

[59] D. Wu, M. Zhang, C. Shen, Z. Huang, and M. Gu, ''BTM and GloVe similarity linear fusion-based short text clustering algorithm for microblog hot topic discovery,'' *IEEE Access*, vol. 8, pp. 32215–32225, 2020, doi: 10.1109/ACCESS.2020.2973430.

[60] J. Huang, M. Peng, P. Li, Z. Hu, and C. Xu, ''Improving biterm topic model with word embeddings,'' *World Wide Web*, vol. 23, no. 6, pp. 3099–3124, Nov. 2020, doi: 10.1007/s11280-020-00823-w.

[61] Y. Zuo, C. Li, H. Lin, and J. Wu, ''Topic modeling of short texts: A pseudo-document view with word embedding enhancement,'' *IEEE Trans. Knowl. Data Eng.*, early access, Apr. 14, 2021, doi: 10.1109/TKDE.2021.3073195.

[62] J. Feng, Y. Rao, H. Xie, F. L. Wang, and Q. Li, ''User group based emotion detection and topic discovery over short text,'' *World Wide Web*, vol. 23, no. 3, pp. 1553–1587, May 2020, doi: 10.1007/s11280-019-00760-3.

[63] J. Yin and J. Wang, ''A Dirichlet multinomial mixture model-based approach for short text clustering,'' in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 233–242, doi: 10.1145/2623330.2623715.

[64] J. Yin and J. Wang, ''A text clustering algorithm using an online clustering scheme for initialization,'' in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1995–2004, doi: 10.1145/2939672.2939841.

[65] C. Li, Y. Duan, H. Wang, Z. Zhang, A. Sun, and Z. Ma, ''Enhancing topic modeling for short texts with auxiliary word embeddings,'' *ACM Trans. Inf. Syst.*, vol. 36, no. 2, pp. 1–30, Sep. 2017, doi: 10.1145/3091108.

[66] C. Li, H. Wang, Z. Zhang, A. Sun, and Z. Ma, ''Topic modeling for short texts with auxiliary word embeddings,'' in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2016, pp. 165–174, doi: 10.1145/2911451.2911499.

[67] Z. Liu, T. Qin, K.-J. Chen, and Y. Li, ''Collaboratively modeling and embedding of latent topics for short texts,'' *IEEE Access*, vol. 8, pp. 99141–99153, 2020, doi: 10.1109/ACCESS.2020.2997973.

[68] W. Liang, R. Feng, X. Liu, Y. Li, and X. Zhang, ''GLTM: A global and local word embedding-based topic model for short texts,'' *IEEE Access*, vol. 6, pp. 43612–43621, 2018, doi: 10.1109/ACCESS.2018.2863260.

[69] J. Rashid, S. M. A. Shah, and A. Irtaza, ''Fuzzy topic modeling approach for text mining over short text,'' *Inf. Process. Manag.*, vol. 56, no. 6, Nov. 2019, Art. no. 102060, doi: 10.1016/j.ipm.2019.102060.

[70] C. Joakim, ''Explore Python, machine learning, and the NLTK library,'' IBM Dev. Work., Tech. Rep., 2012.

[71] M. F. Porter, ''An algorithm for suffix stripping,'' *Program*, vol. 40, no. 3, pp. 211–218, Jul. 2006, doi: 10.1108/00330330610681286.

[72] S. M. Al-Ghuribi and S. Alshomrani, ''A simple study of webpage text classification algorithms for Arabic and English languages,'' in *Proc. Int. Conf. IT Converg. Secur. (ICITCS)*, Dec. 2013, pp. 1–5, doi: 10.1109/ICITCS.2013.6717784.

[73] R. Brochier, A. Guille, and J. Velcin, ''Global vectors for node representations,'' in *Proc. World Wide Web Conf. (WWW)*, 2019, pp. 2587–2593, doi: 10.1145/3308558.3313595.

[74] J. Pennington, R. Socher, and C. Manning, ''Glove: Global vectors for word representation,'' in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[75] S. Kaur, P. Kumar, and P. Kumaraguru, ''Automating fake news detection system using multi-level voting model,'' *Soft Comput.*, vol. 24, no. 12, pp. 9049–9069, Jun. 2020, doi: 10.1007/s00500-019-04436-y.

[76] I. Muneer and R. M. A. Nawab, ''Cross-lingual text reuse detection at sentence level for English–Urdu language pair,'' *Comput. Speech Lang.*, vol. 75, Sep. 2022, Art. no. 101381, doi: 10.1016/j.csl.2022.101381.

[77] T. P. Minka and J. Lafferty, ''Expectation-propogation for the generative aspect model,'' 2012, *arXiv:1301.0588*.

[78] T. Lin, W. Tian, Q. Mei, and H. Cheng, ''The dual-sparse topic model: Mining focused topics and focused terms in short text,'' in *Proc. 23rd Int. Conf. World Wide Web (WWW)*, 2014, pp. 539–550, doi: 10.1145/2566486.2567980.

[79] R. Rehurek and P. Sojka, ''Software framework for topic modelling with large corpora,'' in *Proc. LREC Workshop New Challenges NLP Frameworks*, 2010, pp. 46–50.

[80] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. Vanderplas, A. Joly, B. Holt, and G. Varoquaux, ''API design for machine learning software: Experiences from the scikit-learn project,'' 2013, *arXiv:1309.0238*.

[81] S. Bird, E. Klein, and E. Loper, *Natural Language Processing With Python: Analyzing Text With the Natural Language Toolkit*. Sebastopol, CA, USA: O'Reilly Media, 2009.

[82] F. Elsafoury, ''Cyberbullying datasets,'' Tech. Rep., 2020, doi: 10.17632/jf4pzyvnpj.1.

[83] K. Cortis and S. Handschuh, ''Analysis of cyberbullying tweets in trending world events,'' in *Proc. 15th Int. Conf. Knowl. Technol. Data-Driven Bus.*, Oct. 2015, pp. 1–8, doi: 10.1145/2809563.2809605.

[84] L. Cheng, J. Li, Y. Silva, D. Hall, and H. Liu, ''PI-bully: Personalized cyberbullying detection with peer influence,'' in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 5829–5835, doi: 10.24963/ijcai.2019/808.

[85] Z. Zhang, D. Robinson, and J. Tepper, ''Detecting hate speech on Twitter using a convolution-GRU based deep neural network,'' in *Proc. Eur. Semantic Web Conf.* (Lecture Notes in Computer Science), vol. 10843, A. GangemiAnna, A. L. Gentile, A. G. Nuzzolese, S. Rudolph, M. Maleshkova, H. Paulheim, J. Z. Pan, and M. Alam, Eds. Cham, Switzerland: Springer, 2018, pp. 745–760.

[86] M. Röder, A. Both, and A. Hinneburg, ''Exploring the space of topic coherence measures,'' in *Proc. 8th ACM Int. Conf. Web Search Data Mining*, Feb. 2015, pp. 399–408, doi: 10.1145/2684822.2685324.

[87] B. A. H. Murshed, H. D. E. Al-Ariki, and S. Mallappa, ''Semantic analysis techniques using Twitter datasets on big data: Comparative analysis study,'' *Comput. Syst. Sci. Eng.*, vol. 35, no. 6, pp. 495–512, 2020, doi: 10.32604/csse.2020.35.495.

[88] J. Singh and A. K. Singh, ''NSLPCD: Topic based tweets clustering using node significance based label propagation community detection algorithm,'' *Ann. Math. Artif. Intell.*, vol. 89, no. 3, pp. 371–407, 2021, doi: 10.1007/s10472-020-09709-z.

[89] H. M. Abdulwahab, S. Ajitha, and M. A. N. Saif, ''Feature selection techniques in the context of big data: Taxonomy and analysis,'' *Appl. Intell.*, vol. 52, pp. 13568–13613, 2022, doi: 10.1007/s10489-021-03118-3.

BELAL ABDULLAH HEZAM MURSHED received the B.Sc. degree (Hons.) in computer science & information system from the University of Taiz, Yemen, in 2008, and the M.Sc. degree (Hons.) in computer science from the University of Mysore, Mysore, India, in 2016. He is currently a Ph.D. Research Scholar at the Department of Studies in Computer Science, University of Mysore. He is also a Teaching Assistant with the Faculty of Engineering and Information Technology, Amran University, Yemen. He was awarded three gold medals from the University of Mysore as a result of obtaining the first rank in the master's degree. His research interests include data mining, machine learning, NLP, artificial intelligence, topic modeling, optimization algorithms big data, and the Internet of Things.

JEMAL ABAWAJY (Senior Member, IEEE) received the B.Sc. degree from St. F. X University, Canada, the M.Sc. degree from Dalhousie University, Canada, and the Ph.D. degree from the Ottawa-Carleton Institute of Technology, Canada, all in computer science. He is currently a Full Professor with the Faculty of Science, Engineering and Built Environment, School of Information Technology, Deakin University, Australia. He is currently working on the Editorial Board of *Electronic Commerce Research* (Springer) and the *International Journal of Parallel, Emergent and Distributed Systems* (Taylor & Francis), and an Associate Editor of many journals, such as *Sensors* (MDPI). He is the author or coauthor of ten books, more than ten conference volumes, more than 453 refereed papers in conferences, book chapters, and journals, such as the IEEE Transactions on Computers and IEEE Transactions on Fuzzy Systems. He was a member of the organizing committees for over 300 international conferences serving in various capacities, including the chair and the general co-chair. He has delivered over 50 keynote and international seminars. He has also supervised numerous Ph.D. students to completion, and actively involved in various funded research, supervising postdoctoral, research assistants, and visiting scholar in the area of cloud computing, big data analytics, the IoT, cybersecurity, and decision support systems. He is on the editorial boards of many journals. He is a Senior Member of IEEE Computer Society, IEEE Technical Committee on Scalable Computing, IEEE Technical Committee on Dependable Computing and Fault Tolerance, and IEEE Communication Society.

SURESHA MALLAPPA received the M.Sc. degree from the University of Mysore, Mysore, the M.Phil. degree from DAVV, the M.Tech. degree from IIT Kharagpur, and the Ph.D. degree from IISc, Bengaluru. He is currently working as a Professor with the Department of Studies in Computer Science, University of Mysore. He has 31 years of teaching experience in computer science at the postgraduate level in various universities. He has published more than 80 research papers in reputed international and national journals and conferences. He has supervised numerous Ph.D. students. His research interests include dynamic web caching, database systems, image search engines, e-governance, data mining, big data, opinion mining, and cloud computing. He has also taught many courses in foreign universities as part of teaching assignments.

MUFEED AHMED NAJI SAIF received the B.Sc. degree in computer applications from Osmania University, Hyderabad, India, in 2010, and the M.Sc. degree in information technology from Bharathiar University, Coimbatore, India, in 2012. He is currently a Ph.D. Research Scholar with the Department of Computer Applications, Sri Jayachamarajendra College of Engineering (Affiliated to VTU), Mysore, India. He is also working in cloud computing under the guidance of Dr. S. K. Niranjan. He has more than six years of experience in both industry and teaching. His research interests include software engineering, distributed systems, cloud computing, networking, data mining, and big data.

SUMAIA MOHAMMED AL-GHURIBI received the B.Sc. degree (Hons.) in computer science from Taiz University, Yemen, in 2008, the M.S. degree in computer science from King Abdulaziz University, Jeddah, Saudi Arabia, in 2014, and the Ph.D. degree from the Universiti Kebangsaan Malaysia (UKM), in 2021. Her research interests include natural language processing, web mining, sentiment analysis, and recommender systems.

FAHD A. GHANEM received the B.Sc. degree in computer science from Hodeidah University, Yemen, in 2011, and the M.Sc. degree in computer science from the University of Mysore, India, in 2019. He is currently working as a full-time Research Scholar toward the Ph.D. under the guidance of Dr. M. C. Padma, with the Department of Computer Science and Engineering, PES College of Engineering (Affiliated to University of Mysore), Mandaya, India. He is also a Faculty Member of Hodeidah University, Hodeidah, Yemen. His research interests include data mining, big data analytics, and machine learning.