Journal of Big Data

RESEARCH

Open Access

Hybrid topic modeling method based on dirichlet multinomial mixture and fuzzy match algorithm for short text clustering

Mutasem K. Alsmadi[1*], Malek Alzaqebah[2,3], Sana Jawarneh[4], Ibrahim ALmarashdeh[1], Mohammed Azmi Al-Betar[5], Maram Alwohaibi[2,3], Noha A. Al-Mulla[2,3], Eman AE Ahmed[2,3] and Ahmad AL Smadi[6]

*Correspondence:
Mutasem K. Alsmadi
mkalsmadi@iau.edu.sa
[1]Department of MIS, College of Applied Studies and Community Service, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia
[2]Department of Mathematics, College of Science, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia
[3]Basic and Applied Scientific Research Center, Imam Abdulrahman Bin Faisal University, P. O. Box 1982, Dammam, Saudi Arabia
[4]Computer Science Department, Community College Dammam, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia
[5]Artificial Intelligence Research Center (AIRC), College of Engineering and Information Technology, Ajman University, Ajman, United Arab Emirates
[6]Department of Data Science and Artificial Intelligence, Zarqa University, Zarqa 13100, Jordan

Abstract

Topic modeling methods proved to be effective for inferring latent topics from short texts. Dealing with short texts is challenging yet helpful for many real-world applications, due to the sparse terms in the text and the high dimensionality representation. Most of the topic modeling methods require the number of topics to be defined earlier. Similarly, methods based on Dirichlet Multinomial Mixture (DMM) involve the maximum possible number of topics before execution which is hard to determine due to topic uncertainty, and many noises exist in the dataset. Hence, a new approach called the Topic Clustering algorithm based on Levenshtein Distance (TCLD) is introduced in this paper, TCLD combines DMM models and the Fuzzy matching algorithm to address two key challenges in topic modeling: (a) The outlier problem in topic modeling methods. (b) The problem of determining the optimal number of topics. TCLD uses the initial clustered topics generated by DMM models and then evaluates the semantic relationships between documents using Levenshtein Distance. Subsequently, it determines whether to keep the document in the same cluster, relocate it to another cluster, or mark it as an outlier. The results demonstrate the efficiency of the proposed approach across six English benchmark datasets, in comparison to seven topic modeling approaches, with 83% improvement in purity and 67% enhancement in Normalized Mutual Information (NMI) across all datasets. The proposed method was also applied to a collected Arabic tweet and the results showed that only 12% of the Arabic short texts were incorrectly clustered, according to human inspection.

Keywords Topic modeling, Dirichlet multinomial mixture, Levenshtein distance, Arabic tweets, Short text, Outliers, Fuzzy match

Introduction

Topic modeling methods have become very effective in finding hidden semantics and latent patterns in textual data [1–4]. With the rise of social media, clustering the short text tracked the researcher's attention to using topic modeling methods for extracting semantic subjects. The main challenges of clustering the short text are fewer word co-occurrence, sparseness problems, a limited number of words in each text, and difficulty in finding semantically related words [5].

Several model-based clustering methods have been proposed to address these challenges, demonstrating effective performance in handling issues associated with short text data [6]. Among the frequently employed methods for text clustering is Latent Dirichlet Allocation (LDA) [7–9]. Short text topic modeling methods are categorized into three groups based on their characteristics [5]: (i) Global Word Co-occurrences (GWC), (ii) Self-aggregation-based Methods (SA), and (iii) the Dirichlet Multinomial Mixture (DMM). GWC-based methods consider the closer two words to each other to be more relevant [10]. GWC finds the global word co-occurrences from the original corpus to predict the latent topics. While SA combines the short texts into long pseudo-documents to solve the sparseness problem [11].

Nigam et al. [12] proposed the Dirichlet Multinomial Mixture Model (DMM) based on the Expectation-Maximization method, assuming each text belongs to one topic. The success of DMM motivated Yu et al. [13] to incorporate DMM with feature selection. However, the proposed method showed a slow convergence. The authors in [9] proposed a DMM model with feature partition. Moreover, the Gibbs Sampling method is proposed based on DMM (GSDMM) for short text clustering [14], GSDMM needs the maximum number of possible clusters (k) to find the optimal number of clusters [5]. The main drawback of this method is the high computational cost (space and time) when assigning a high number to parameter k.

The aforementioned kind of unsupervised statistical learning is typically thought to work well, but only if the corpus is statistically enough. Short texts have sparse terms and noises [15, 16], rendering them insufficient data necessary for successful statistical learning [17]. In addition, determining the number of topics before execution is challenging since the number of topics in real cases is unknown [18]. Determining a large number of topics usually increases the complexity of the model and leads to unsatisfactory results [19]. So each text can be represented by a separate topic due to the very limited available word co-occurrence in short texts [20].

One of the interesting algorithms is the fuzzy matching algorithm (e.g., Levenshtein distance), which is characterized by its ability to match sentences to find the distances between a single text and other texts [21, 22]. In the work presented in [23], a fuzzy matching approach was proposed for data alignment by employing the Levenshtein distance and N-grams for string matching, especially when there is no exact match. Although the main problem of the fuzzy matching algorithm is the high computational complexity, particularly when large samples are presented in the dataset [24], the authors in [18] suggested a technique that uses a similarity metric that integrates syntactic and semantic information from the text, where the Levenshtein distance is used to measure how similar the feature vectors are to one another.

In this paper, a hybrid approach called Topic Clustering based on Levenshtein Distance (TCLD) algorithm is proposed, which effectively addresses the limitations associated with both topic modeling and fuzzy matching techniques. TCLD aims to optimize the number of topics generated through topic modeling, enhance the results of topic modeling, and improve the complexity of both topic modeling and Levenshtein distance calculations.

In addition, the majority of short text modeling efforts have been made for the English language. On the other hand, research on the Arabic language is limited and facing significant problems. However, most of the research on the Arabic language is utilized for sentiment analysis [25]. The work by [26] has combined the k-means and topic modeling for Arabic document clustering where a data set for Arabic classification [27, 28] was used (Not a short text dataset). Collecting Arabic data from social media produced huge short texts with spelling errors, homonyms, repeated text, and even some words that have different spellings, which makes it hard to find the latent topics from the short text data [29–31]. Therefore, the proposed method is tested using collected tweets based on specific geographic locations to validate this work. Since the Arabic language is challenging and there is no available dataset for short-text modeling, a new short-text Arabic dataset is established for short-text modeling.

The main contributions of this work can be summarized as follows:

1. Introducing a novel method that treats individual texts as singular units and compares them with others through a partial match involving more than one word from the same text.
2. Introducing a novel hybrid approach, Topic Clustering based on Levenshtein Distance (TCLD), aimed at optimizing the number of topics generated through topic modeling. The developed TCLD algorithm evaluates documents within topics and ensures accurate document placement within clusters.
3. Improving the efficacy of topic modeling methods by addressing their challenges in dealing with short texts, optimizing the topic count, and managing noisy data.
4. Establishing a standardized Arabic dataset by integrating the TCLD algorithm and manual annotation, thereby supporting research in short text topic modeling within the Arabic language context.

Two intrinsic/internal measures and four extrinsic/external measures are used to measure the effectiveness of the proposed method. To demonstrate the effectiveness of the proposed method, six English benchmark short-text datasets [5] were tested. Additionally, a case study on collected Arabic tweets has been performed as a challenging task to confirm the ability of the proposed method to deal with messy and unstructured data. The results show that the proposed method was able to produce good results on benchmark English datasets compared with state-of-the-art topic modeling techniques, such as LDA, Gibbs Sampling DMM (GSDMM), Latent-Feature Dirichlet Multinomial Model (LF-DMM), Biterm Topic Modeling (BTM), Generalized Polya Urn DMM (GPU-DMM), Pseudo-Document-Based Topic Modeling (PTM), and Self-Aggregation based Topic Modeling (SATM), with 83% improvement on overall datasets in terms of purity and 67% in term of NMI. The Arabic dataset shows only 12% of the short texts are miss-clustered based on manual checking.

The remainder of the paper is organized as follows: Sect. 2 presents a detailed literature review, followed by Methods and Materials in Sect. 3. The results and discussion are presented in Sect. 4. Finally, the paper concludes with Sect. 5.

Literature review

Clustering may be used to find the hidden patterns in complicated datasets that are exposed by several data points. Many clustering techniques have been developed for a range of applications [32–35]. Text clustering is an important problem in the field of natural language processing [36]. Clustering short texts is challenging due to the extremely low amount of word co-occurrence within such texts [20], this poses difficulties when employing the traditional topic modeling algorithms, such as probabilistic latent semantic analysis (PLSA) and Latent Dirichlet Allocation (LDA) [37], as these algorithms were primarily designed for long texts [5].

Model-based clustering algorithms stand out as the most effective approach for short-text clustering [6]. Among these is the Gaussian Mixture Model(GMM) [38]. GMM considers the features in data to be produced by a Gaussian distribution mixture, where each cluster is related to an exact Gaussian distribution. However, when applied to short text, GMM encounters challenges stemming from the inherent high dimensionality of text data [39]. Addressing this issue, Nigam et al. [12] introduced the Dirichlet Multinomial Mixture (DMM) model to mitigate the complexities associated with high-dimensional representations in text data. Building upon this approach, a more refined variation known as the Gibbs Sampling DMM (GSDMM) was later introduced in [14], this improvement in model design offers better convergence and the capability to automatically infer the optimal number of topics.

Different methods have been proposed for improving short text clustering based on topic modeling methods. Qiang et al. [39]. , proposed a model based on Pitman-Yor Process using the probabilities from the model of PitmanYor where each text selects one active cluster or creates a new cluster. Moreover, a pioneering dual word graph topic model was introduced to enhance short text clustering, aiming to extract topics by considering simultaneous word co-occurrence. This approach demonstrated a superior capability in generating more cohesive topics when compared to existing topic models [40]. Another method that integrates the topic modeling with the graph convolutional networks is also proposed in [41]. This method is based on an external knowledge graph, where WordNet and a pre-trained graph are used to deal with the short noisy text. An advanced hybrid topic modeling approach, combining Latent Dirichlet Allocation (LDA) and Latent Semantic Indexing (LSI), and incorporating visualization techniques, has been presented by [42]. This method highlights its effectiveness by successfully identifying health-related topics within healthcare data.

Within the confines of the short text dataset, a substantial portion of terms tends to co-occur merely once or twice. This prevalence of limited co-occurrences poses a challenge in determining the ideal number of clusters, potentially leading to suboptimal outcomes in the categorization of topics [43, 44]. Various strategies, such as text augmentation [45], topic modeling [46], and the Dirichlet Mixture Model [14, 47, 48], have been proposed to address the challenge of sparsity in short-text clustering. Moreover, In the realm of short-text clustering models, studies have delved into innovative approaches. For instance, the Dirichlet Multinomial Mixture model (GSDMM) was introduced for short text clustering [32] which infers the number of topics and achieves the best results compared to other topic modeling and clustering algorithms [9, 11]. The drawback of this model is the defined number of clusters which the higher number increases the complexity of the model and leads to unsatisfactory results [19].

Levenshtein distance algorithm has been applied in previous studies [24, 42], specifically for text clustering. The algorithm makes use similarity metric (not an exact match) that integrates syntactic and semantic information from the text, The main drawback of this algorithm is the high complexity when large samples are presented in the dataset [24].

This paper aims to overcome the limitations of topic modeling, particularly the difficulty in determining the optimal number of clusters, and the complexity of the Levenshtein distance algorithm, particularly evident with large datasets. To overcome these challenges, the Levenshtein distance is utilized to assess the outcomes of topic modeling. This approach allows for the evaluation of the similarity of feature vectors within each topic generated by topic modeling algorithms, focusing specifically on subsets rather than the entire dataset.

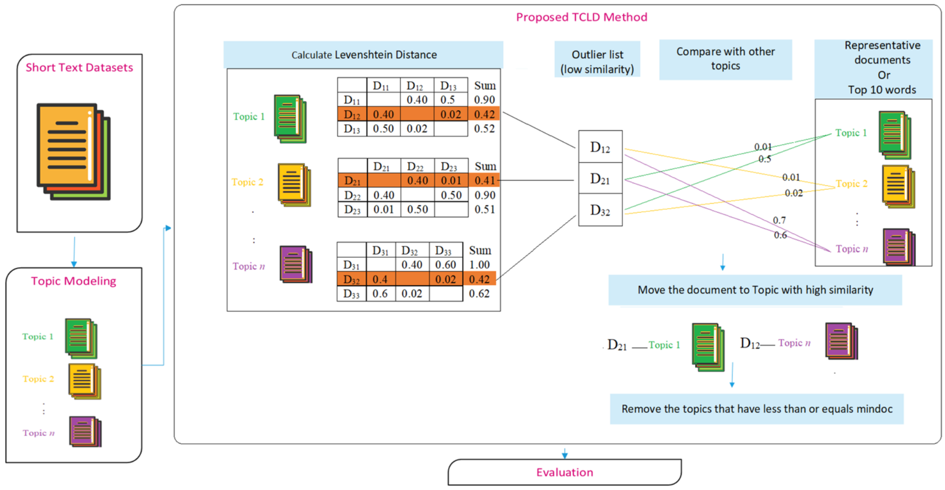
Materials and methods

The proposed method is illustrated in this section. Figure 1 shows the overall process of the proposed approach. The process starts with preparing the dataset, where the collected tweets are cleaned and preprocessed. In the next step, the topic modeling methods are applied to cluster the documents into a prespecified number of topics. Once the solution is produced, a topic clustering algorithm based on the TCLD approach is performed to improve the produced solution. Finally, the performance of the proposed method is evaluated. The following subsections explain in detail the proposed approach.

Short text datasets

English benchmark datasets

Six datasets are used in this paper to validate the proposed model and demonstrate the impacts of the proposed model compared with other models. The details of these datasets are shown in Table 1, where the table represents the number of topics and the number of documents, and the AvgLength and MaxLength related to the average and maximum length of each document, and the vocabulary size, respectively. It should be noted that these datasets have gone through preprocessing [5].


Short Text Datasets
Topic Modeling
Topic 1
Topic 2
Topic n
Proposed TCLD Method
Calculate Levenshtein Distance
Topic 1
Topic 2
Topic n

	D_{11}	D_{12}	D_{13}	Sum
D_{11}		0.40	0.5	0.90
D_{12}	0.40		0.02	0.42
D_{13}	0.50	0.02		0.52

	D_{21}	D_{22}	D_{23}	Sum
D_{21}		0.40	0.01	0.41
D_{22}	0.40		0.50	0.90
D_{23}	0.01	0.50		0.51

	D_{31}	D_{32}	D_{33}	Sum
D_{31}		0.40	0.60	1.00
D_{32}	0.4		0.02	0.42
D_{33}	0.6	0.02		0.62

Outlier list (low similarity)
Compare with other topics
Representative documents Or Top 10 words
D12
D21
D32
0.01
0.5
0.01
0.02
0.7
0.6
Topic 1
Topic 2
Topic n
Move the document to Topic with high similarity
D21 — Topic 1
D12— Topic n
Remove the topics that have less than or equals mindoc
Evaluation

Fig. 1 The proposed approach

Table 1 Summary of the English benchmark datasets

Dataset	#Topics	# Documents	AvgLength	MaxLength	Vocabulary size
SearchSnippets	8	12,295	14.4	37	5,547
StackOverflow	20	16,407	5.03	17	2,638
Biomedicine	20	19,448	7.44	28	4498
Tweet	89	2,472	8.55	20	5,096
GoogleNews	152	11,109	6.23	14	8,110
PascalFlickr	20	4,834	5.37	19	3,431

SearchSnippets: This dataset was selected from the outcomes of an online search transaction using predetermined terms from 8 distinct domains. Business, computers, culture-arts, education-science, engineering, health, politics-society, and sports are the eight areas, in that order.

StackOverflow: Kaggle.com has made the dataset available. 3,370,528 samples from July 31 to August 14, 2012, make up the raw dataset. Here, 20,000 question titles from 20 distinct tags are chosen at random by the dataset.

Biomedicine: The challenge information provided on the official BioASQ website is used in the field of biomedicine.

Tweet: There are 109 queries available for use in the 2011 and 2012 microblog tracks at the Text Retrieval Conference (TREC). After eliminating the searches for which there were no highly relevant tweets, the Tweet dataset had 89 clusters and 2,472 tweets in total.

Google News: The news articles are automatically grouped into clusters on the Google News website. On November 27, 2013, the Google News dataset was obtained from the Google News website, and 152 clusters worth 11,109 news items' titles and summaries were crawled.

PascalFlickr: A collection of captions from the PascalFlickr dataset [49] is used to assess the effectiveness of short text clustering [32].

Arabic tweets datasets

To the best of our knowledge, there is no available topic modeling short text Arabic dataset, since less work has been done for the Arabic language. In this study, the Arabic tweet datasets were collected using the Twitter API. Consequently, the dataset is made available for further investigation and can be accessed in the Data Availability and Materials section.

The Twitter API was used to collect Arabic tweets; tweets were downloaded using the Python package (tweepy). The language in the tweets was Arabic. It is important to note that the search procedure of Twitter API, returns a maximum of 18k tweets and only the data of the previous 6 to 9 days. Thus, the process was run several times, collecting 21,303 tweets based on specific geographic locations. Then, duplicated tweets were deleted, and the number of unique tweets became 17,753.

Table 2 shows a summary of the used dataset. It includes the number of topics, the number of documents, the average and maximum length per document, and the vocabulary size.

The preprocessing task for the collected tweets is a very important task that should be performed [29–31, 50] since the Arabic tweets are usually not standard and complicated due to the existence of slang, diacritical marks, elongations, spelling mistakes, dialect

Table 2 Summary of the collected Arabic dataset

Dataset	#Topics	# Documents	avg/max length	Vocabulary size
Full Arabic Tweets	unknowing	17,753	9.9/69	37,389

text, and extra characters. The employed preprocessing methods are detailed below, encompassing cleaning, normalization, and stemming [51–53].

1. Tweets Cleaning: Tweets contain many noises, including elongations, hashtags, emojis, links, non-Arabic text, digits, punctuation marks, and other marks. Thus, each Arabic tweet is read character-by-character and checks if it belongs to the Arabic aliphatic and white space or not, along with removing the URLs and advertisements.
2. Tweets Normalization: The objective of this step is to make the text more reliable by removing diacritical marks, extra white spaces, and duplicates, and also to convert each letter to its standard form [51, 52].
3. Light Stemmer for Arabic Words: It is used to convert the inflected Arabic word to a common canonical form (Stem) [54]. Following [55], a stemmer is utilized for stemming Arabic words without relying on a root dictionary.

Topic modeling

Two topic modeling methods used in this research are presented in this section: LDA and GSDMM [5, 39], They were chosen based on their successful performance. The problem of short-text topic modeling is defined for each method with the required notations.

Latent dirichlet allocation (LDA)

LDA is a probabilistic generative model of a corpus. The fundamental concept is that texts are modeled as random mixtures over latent topics, with each topic being defined by a distribution over words [56].

The process of LDA is represented as follows:

- For every topic *(t)* generate the Dirichlet distribution on words, $\phi k \sim$ Dirichlet (β).
- Generate a vector of Dirichlet distribution on topics $\theta_d \sim$ Dirichlet(α) For every document *(d)*.
- For every word w_n out of the N words:
- Select a random topic $t_d \sim$ Multinomial *(θ)*.
- Select a word *(w_n)* from *$p(w_n|t_d, \beta)$*, a multinomial probability that depends on the topic t_d.

The LDA graphical model is represented in Fig. 2, where the observable variable is highlighted [7].

Where:

ϕ is the topic word distribution.

θ is the document topic distribution with parameters α and β.

K is the predefined number of topics.

D is the short text corpus consisting of N words.

n_d is the number of words in document $\boldsymbol{d}$.

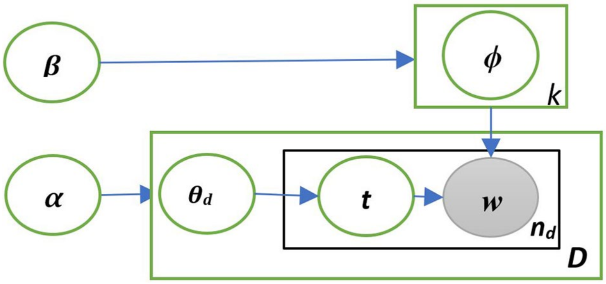
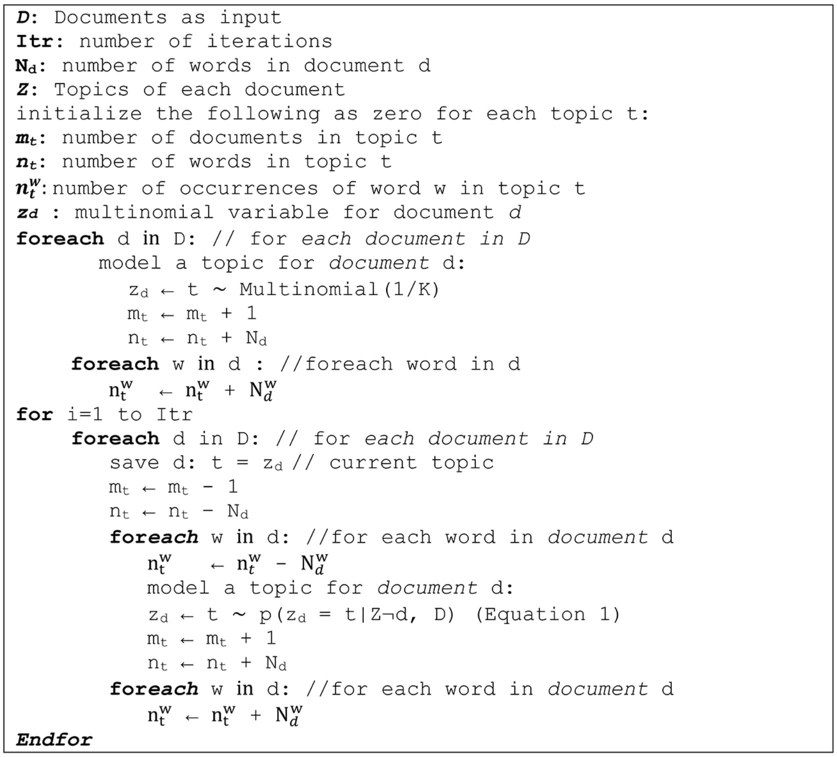


Fig. 2 LDA graphical model



```
D: Documents as input
Itr: number of iterations
N_d: number of words in document d
Z: Topics of each document
initialize the following as zero for each topic t:
m_t: number of documents in topic t
n_t: number of words in topic t
n_t^w: number of occurrences of word w in topic t
zd : multinomial variable for document d
foreach d in D: // for each document in D
        model a topic for document d:
          $z_d \leftarrow t \sim$ Multinomial(1/K)
          $m_t \leftarrow m_t + 1$
          $n_t \leftarrow n_t + N_d$
     foreach w in d : //foreach word in d
          $n_t^w \leftarrow n_t^w + N_d^w$
for i=1 to Itr
     foreach d in D: // for each document in D
        save d: t = $z_d$ // current topic
        $m_t \leftarrow m_t - 1$
        $n_t \leftarrow n_t - N_d$
        foreach w in d: //for each word in document d
           $n_t^w \leftarrow n_t^w - N_d^w$
           model a topic for document d:
           $z_d \leftarrow t \sim p(z_d = t|Z\neg d, D)$ (Equation 1)
           $m_t \leftarrow m_t + 1$
           $n_t \leftarrow n_t + N_d$
        foreach w in d: //for each word in document d
           $n_t^w \leftarrow n_t^w + N_d^w$
Endfor
```

Fig. 3 GSDMM [14]

Gibbs Sampling for Dirichlet Multinomial Mixture (GSDMM)

GSDMM has shown good performance in dealing with highly sparse short text. GSDMM uses Gibbs sampling to assign a topic to each short text. Dirichlet Multinomial Mixture Model (DMM) [12] is a probabilistic generative model for short text. Note that each short text is produced by a mixture model, and each mixture component has one cluster.

The whole process of GSDMM is pseudo-coded in Fig. 3. Firstly, the documents are assigned randomly to K clusters, then the following parameters are initialized: T, m_t, n_t, *and* n_t^w.

In each iteration, the topic t is reassigned for the document d based on the $p(zd=t|Z\neg d, D)$, which represents the conditional distribution, where all topics of all documents are represented by Z and $\neg d$ indicate the topic of document d is removed from Z. A topic t is reassigned to document d and the corresponding variables are updated every time (i.e., *Z*, m_t, n_t, *and* N_t^w. Finally, the nonempty topics in *Z* will be considered as the output of the model. The GSDMM conditional probability distribution is represented in Eq. 1.

$$P\left(z_d = t|Z_{\neg d}, D\right) \propto \frac{m_t, \neg d + \alpha}{N - 1 + K_\alpha} \frac{\prod_{w \in d} \prod_{j=1}^{n_d^w} (n_t^w, \neg d + \beta + j - 1)}{\prod_{i=1}^{n_d} (n_t, \neg d + V\beta + i - 1)} \tag{1}$$

$P\left(z_d = t|Z_{\neg d}, D\right)$ is derived from the Dirichlet Multinomial Mixture (DMM) model, DMM chooses Dirichlet distribution for ϕ and *θ*, respectively, as a pre-distribution with parameters *α* and *β*. A topic *zd* is sampled in DMM for document *d* by *θ* and then generates all words in document *d* from topic *zd* by ϕzd.

The proposed topic clustering based on levenshtein distance (TCLD)

After clustering the topics with both clustering algorithms and topic modeling, an improvement of the resulting set of topics is proposed using the Levenshtein Distance algorithm. The details of the Levenshtein Distance algorithm and the proposed approach are explained as follows:

Edit distance algorithms are commonly utilized in textual similarity recognition, featuring diverse edit operations such as insertions, deletions, and substitutions, each associated with its specific cost. The cost is determined by the number of steps required to transform one text into another, reflected in the edit distance score. Employing edit distance metrics for parsing and document analysis proves to be a suitable approach for quantifying semantic similarities between documents. The Levenshtein Distance (LD) algorithm was proposed by Vladimir Levenshtein [21] to calculate the similarity between sequences of words.

The LD algorithm is used to measure the separation of two sequences of words. The LD between two words is the minimum number of edits of a single character required to change one word into another.

Mathematically, the Levenshtein distance algorithm is illustrated in Eq. (2) [22].

$$D_{s1,s2}(i,j)\begin{cases} Max\left(i,j\right) & if min\left(i,j\right) = 0 \\ Min\begin{cases} D_{s1,s2}\left(i-1,j\right)+1 \\ D_{s1,s2}\left(i,j-1\right)+1 \\ D_{s1,s2}\left(i-1,j-1\right)+1_{s1_i \neq s2_j} \end{cases} & otherwise \end{cases} \tag{2}$$

Where the comparison is between the two strings; *s1* and *s2*, also, *i* and *j* are the current indexes to be evaluated, starting at |*s1*| and |*s2*| respectively. Moreover, the theoretical maximum of the Levenshtein distance is the longer string length, so if the strings have no common characters, all the characters in the shorter string have to be initially substituted, and then the rest must be added. This upper limit allows defining a similarity ratio between both strings *s1* and *s2* as in Eq. (3) [22, 57]:

$$D_{ratio}\left(s1,s2\right) = 1 - \frac{D_{s1,s2}\left(\left|s1\right|,\left|s2\right|\right)}{Max\left(\left(\left|s1\right|,\left|s2\right|\right)\right)} \tag{3}$$

TCLD is introduced to improve the clustered topics based on semantic similarity. After applying topic modeling, the semantic similarities between documents will be measured. Figure 1 serves as an illustrative example show the proposed TCLD method, starting from calculating Levenshtein Distance, and highlighting its role in identifying outliers (Documents deviating significantly from the norm). In this scenario, the dataset comprises nine documents distributed across three topics generated by topic modeling (denoted as Topic 1, Topic 2, and Topic 3). Each document is labeled as D_1(topic)1(document) (e.g., D_{11}, D_{12}, ..., D_{33}). The TCLD method involves the following steps:

1. Compute the Levenshtein distance ratio between all documents in each clustered topic.
2. The documents with an acceptable distance ratio are kept in the same topic, and the rest are saved in an outlier list.
3. Find the representative documents (or 10 top words) among all documents in each topic.
4. Compute the Levenshtein distance ratio between the documents in the outlier list with the representative documents (or 10 top words) in each topic. When an *acceptable distance ratio* is found, the document will be moved to the corresponding topic.
5. Finally, check the number of documents in all topics and mark the topics that have less than or equal *mindoc* documents as outliers, then repeat step 4 to handle these documents. *mindoc* is a threshold that represents the minimum number of documents needed to keep the topic. In this study, mindoc is set to 10.

The *acceptable distance ratio* means that the Levenshtein distance ratio between two documents is greater than or equal to 0.35. The proposed thresholds are assumptions tested and tuned in experiment analysis. The representative document is the document that has the greatest distance ratio from other documents in each topic.

Experimental design

All of the experiments have been performed on an Intel Core i5-6200U processor with 8 GB of RAM. The experiments are designed based on two main parts. i.e., the whole dataset number of collected tweets without the actual truth-labels and the dataset with actual truth-labels. The experiment steps can be summarized as follows:

1. Determine the initial number of clusters based on the elbow method and GSDMM.
2. Apply LDA and GSDMM.
3. Apply the proposed method based on the TCLD as explained in Sect. 3.3.
4. Evaluate the clustering performance based on intrinsic measures, because actual truth-labels are not available.
5. Generate the dataset with truth-labels.
6. Evaluate and fine-tune the dataset with truth-labels using eyeballing, intrinsic evaluation metrics, and human judgments.
7. Apply and evaluate the new dataset using LDA and GSDMM.
8. Apply the proposed TCLD with two methods of dealing with the documents in the outlier list, where the documents in the outlier list are compared with the representative document in each topic (TCLD-RD) and with the top 10 words from each topic (TCLD-TW).
9. Evaluate and compare TCLD-RD and TCLD-TW.

Performance evaluation

Clustering performance is evaluated by relying on intrinsic and extrinsic measures. Intrinsic measures inspect the separation and compaction of the clusters and do not require ground truth labels. In this work, the Silhouette score (SI) and Davies-Bouldin Index (DBI) [58, 59] are used as intrinsic or internal measures. The indication value of SI ranges between 1 and -1, where 1 is the best value, the worst is -1, and the overlapping clusters are indicated by values close to 0. On the other hand, DBI is a ratio calculated between the cluster scatter and the separation of the cluster, where a lower value of DBI indicates better clustering.

As aforementioned, the extrinsic and external measures require the ground truth labels to be compared with the predicted clusters. In this work, the Normalized Mutual Information (NMI), Adjusted Mutual Information (AMI), Homogeneity score (HS), and purity are used as extrinsic measures [9, 14]. Where the higher values of NMI, AMI, HS, and purity indicate a better clustering quality.

Results and discussion

We present the experimental results in this section, aligning with the experimental design outlined in Sect. 3.4. This presentation aims to illustrate the robustness and efficacy of the proposed method for short-text clustering, using six benchmark datasets reported in Sect. 3.1.1. The parameter settings are tuned and the final settings are as follows: LDA parameters are determined to be $alpha = 0.05$ and $beta = 0.01$. and GSDMM parameters are set $k = 100$, $alpha = 0.1$, and $beta = 0.1$. For both models, the number of iterations is set to 100.

Results on benchmark datasets

Table 3 shows the performance with clustering algorithms of LDA and GSDMM relying on purity, Homogeneity, NMI, AMI, SI, and DBI. Besides the results of the proposed Levenshtein Distance for improving the Topic Clustering (TCLD). The proposed TCLD is applied to the output from GSDMM and LDA. The results show the comparison of two methods dealing with the documents in the outlier list, as follows: (a) using the top 10 words from each topic (TCLD-TW). (b) using the representative document in each topic (TCLD-RD).

The results using the clustering metrics are shown in Table 3. GSDMM demonstrates superior performance, as indicated by the valuation metrics. The results of both GSDMM and LDA are further enhanced with the application of the proposed TCLD technique. This improvement is attributed to the TCLD approach, which evaluates each document in existing topic modeling solutions, thereby enhancing clustering results. Additionally, GSDMM with TCLD-TW performs better than GSDMM with TCLD-RD, demonstrating that moving a document from the outlier list to the appropriate topic may be accomplished by comparing the outlier list with the top 10 words from each topic. In this study, the impact of the acceptable distance ratio on the performance of TCLD was explored, taking into consideration the execution time, the time used to generate the results of the TCLD-TW is acceptable because the processing will be for each topic individually.

Figures 4 and 5 depict the results of TCLD-TW GSDMM and TCLD-TW LDA when varying the acceptable distance ratio from 30 to 60 measured by SI on the six datasets.

Table 3 Results of all models on benchmark datasets

Dataset		GSDMM	LDA	TCLD-TW		TCLD-RD	
				GSDMM	LDA	GSDMM	LDA
SearchSnippets	Purity	0.81	0.349	**0.854**	0.376	0.805	0.349
	Homogeneity	0.694	0.118	**0.754**	0.137	0.668	0.118
	NMI	0.431	0.087	**0.433**	0.096	0.38	0.087
	AMI	**0.426**	0.082	0.421	0.091	0.367	0.082
	SI	-0.226	-0.397	**0.022**	-0.001	0.021	0.001
	DBI	26.038	30.019	**3.798**	10.178	4.376	7.915
	Time(sec)	-	-	**42.984**	99.532	58.714	76.433
StackOverflow	Purity	0.625	0.299	**0.633**	0.299	0.621	0.298
	Homogeneity	0.529	0.219	**0.544**	0.219	0.529	0.219
	NMI	0.478	0.237	**0.496**	0.237	0.483	0.236
	AMI	0.473	0.232	**0.492**	0.232	0.478	0.231
	SI	-0.22	-0.101	-0.003	**0.008**	-0.003	0.006
	DBI	40.319	42.036	8.241	6.922	8.133	**6.805**
	Time(sec)	-	-	39.67	67.958	**39.448**	70.537
Biomedicine	Purity	0.503	0.147	**0.51**	0.162	0.509	0.158
	Homogeneity	0.427	0.072	**0.433**	0.078	0.432	0.075
	NMI	0.378	0.075	**0.379**	0.075	0.378	0.073
	AMI	**0.372**	0.07	**0.372**	0.07	0.371	0.068
	SI	-0.324	-0.12	**-0.002**	-0.003	**-0.002**	**-0.002**
	DBI	30.575	39.114	7.377	14.739	7.383	13.65
	Time(sec)	-	-	**80.873**	270.829	81.829	218.393
Tweet	Purity	0.815	0.597	**0.869**	0.646	0.843	0.61
	Homogeneity	0.856	0.587	**0.892**	0.633	0.875	0.603
	NMI	0.873	0.633	**0.896**	0.671	0.884	0.648
	AMI	0.848	0.569	**0.872**	0.61	0.859	0.585
	SI	-0.049	-0.039	0.066	0.032	**0.065**	0.032
	DBI	9.657	10.814	**2.964**	4.948	3.357	4.943
	Time(sec)	-	-	**3.197**	4.947	3.528	4.312
GoogleNews	Purity	0.702	0.243	**0.803**	0.248	0.799	0.247
	Homogeneity	0.81	0.297	**0.867**	0.294	0.863	0.298
	NMI	0.852	0.396	**0.886**	0.383	0.881	0.394
	AMI	0.835	0.363	**0.869**	0.348	0.863	0.36
	SI	-0.042	-0.012	0.067	0.012	**0.068**	0.013
	DBI	15.971	23.107	**3.7**	7.683	3.846	6.951
	Time(sec)	-	-	34.616	59.894	**33.904**	55.094
PascalFlickr	Purity	0.341	0.111	**0.401**	0.114	0.344	0.113
	Homogeneity	0.334	0.056	**0.386**	0.054	0.332	0.055
	NMI	0.364	0.073	**0.425**	0.066	0.36	0.067
	AMI	0.355	0.063	**0.415**	0.056	0.346	0.058
	SI	-0.089	-0.046	0	0.001	**0.004**	0.002
	DBI	17.744	21.605	**5.127**	9.986	5.627	9.557
	Time(sec)	-	-	**8.049**	14.188	9.479	13.081

This evaluation matrix was chosen based on the fact that the model should not reveal the true value of the topics. Observing the results, both TCLD-TW GSDMM and TCLD-TW LDA demonstrate effectiveness within a favorable range when the ratio is between 0.4 and 0.5. When the ratio is insufficient, there will be no outliers in the cluster. The higher the ratio, the greater the number of outliers extracted. When SI stabilized, the ratio was between 0.4 and 0.5, and if the ratio increased, the results became worse. This indicates that the model accepts samples with small distances, and this worsens the results.

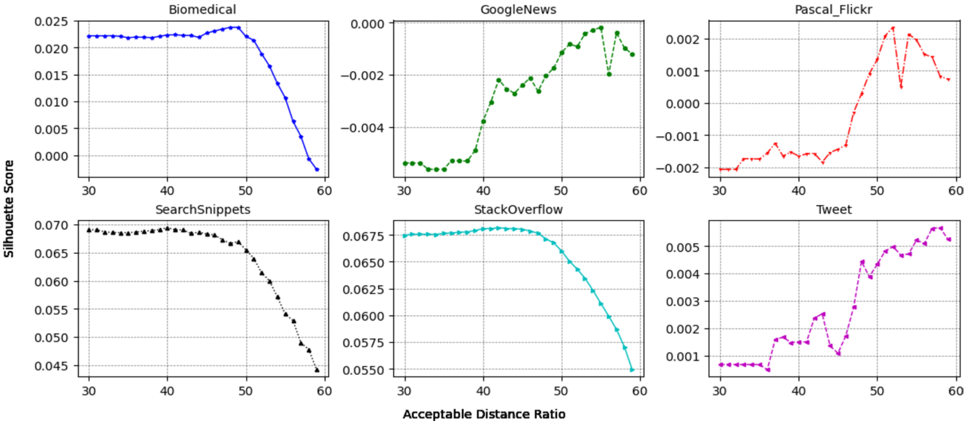


Fig. 4 Influence of the Acceptable Distance Ratio Using TCLD-TW GSDMM

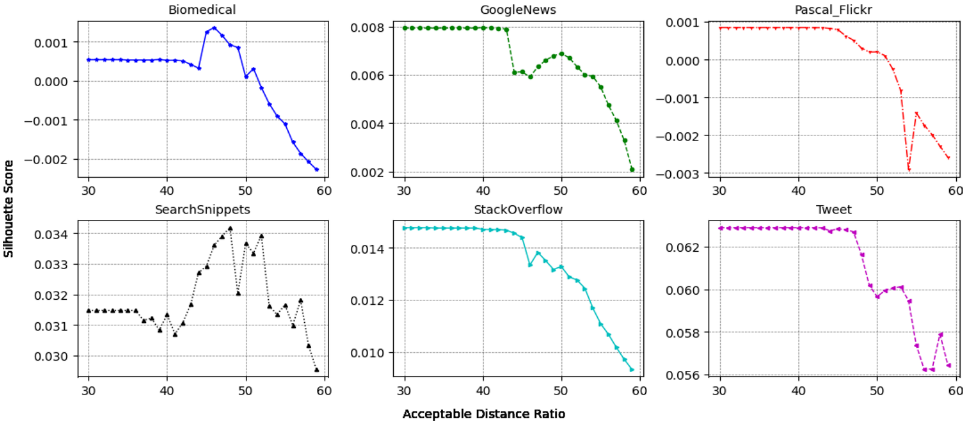


Fig. 5 Influence of the Acceptable Distance Ratio Using TCLD-TW LDA

Comparison with the state-of-the-art topic models

We further compared the results of the best-performing algorithm TCLD-TW GSDMM, with other topic modeling. According to the findings of Qiang et, al [5]. , the approaches that showed the most promise for short-text topic modeling include the latent-feature Dirichlet multinomial model (LF-DMM), generalized Polya urn DMM (GPU-DMM), and other topic modeling conducted in the survey such as Biterm Topic Modeling (BTM), Self-aggregation based topic modeling (SATM) and pseudo-document-based topic modeling (PTM). Table 4 shows the results measured by purity and NMI on the six datasets.

As can be observed from Tables 4 and 83% of the results achieved by the TCLD-TW GSDMM outperform other models in terms of purity and 67% in terms of NMI. The computed mean values of purity and NMI confirm that the proposed method improves the results in both purity and NMI when considering purity, the TCLD-TW GSDMM achieved the best mean value followed by in order BTM, LF-DMM, GPU-DMM, PTM then SATM. Also, in terms of NMI the TCLD-TW GSDMM achieved the best mean value followed by in order LF-DMM, BTM, GPU-DMM, PTM then SATM.

Table 4 Results of all models on six datasets

Dataset	Evaluation Metrics	LF-DMM	GPU-DMM	BTM	SATM	PTM	TCLD-TW
							GSDMM
SearchSnippets	Purity	0.762	0.751	0.765	0.459	0.674	**0.854**
	NMI	**0.579**	0.561	0.566	0.205	0.457	0.433
StackOverflow	Purity	0.518	0.511	0.537	0.505	0.481	**0.633**
	NMI	0.443	0.429	0.456	0.366	0.442	**0.496**
Biomedicine	Purity	0.421	0.433	0.458	0.384	0.425	**0.51**
	NMI	0.348	0.366	0.38	0.27	0.353	**0.433**
Tweet	Purity	0.856	0.83	0.814	0.392	0.839	**0.869**
	NMI	0.843	0.81	0.808	0.507	0.846	**0.896**
GoogleNews	Purity	0.828	0.818	**0.849**	0.654	0.807	0.803
	NMI	**0.875**	0.852	**0.875**	0.76	0.866	0.867
PascalFlickr	Purity	0.381	0.395	0.392	0.237	0.359	**0.401**
	NMI	0.365	0.37	0.368	0.186	0.336	**0.425**
Mean value	*Purity*	*0.628*	*0.623*	*0.636*	*0.439*	*0.598*	***0.678***
	NMI	*0.576*	*0.565*	*0.576*	*0.382*	*0.55*	***0.592***

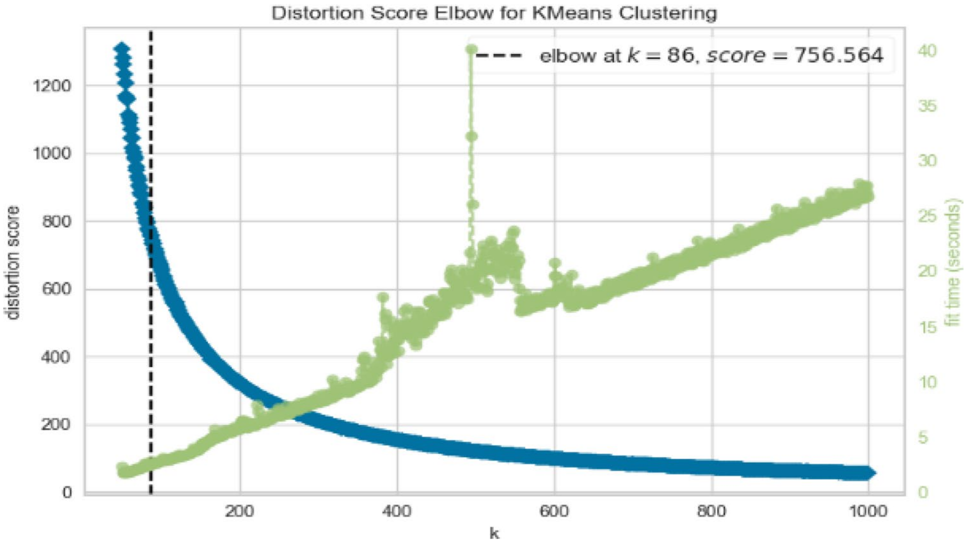


Fig. 6 Distortion Score Elbow for K-Means Clustering

Case study: arabic dataset

The results of the proposed method are presented in this section. Two Arabic datasets were used in this work. The dataset with unknown actual truth labels and the dataset with known actual truth labels are as follows:

Results on the dataset without the actual truth-labels

This study determines the initial number of topics based on K-Means clustering. The Elbow method is used to determine the range of clusters. Figure 6 shows the sum of squared distance and the number of clusters (in the range of 2–1000).

After identifying the initial number of topics, the GSDMM method was applied, where the parameter settings for LDA and GSDMM models were trained according to the following parameters: alpha: 0.1 and beta: 0.01. Testing both algorithms was conducted with two different values of iterations: 50 and 100. These hyper-parameters are executed

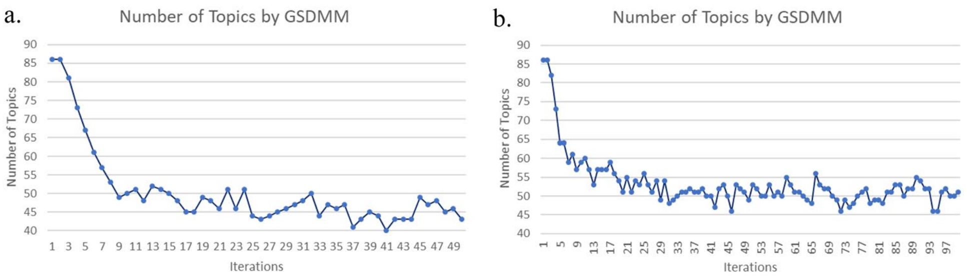


Fig. 7 Number of Topics Generated by GSDMM Using 50 and 100 Iterations. (**a**) Silhouette Score for GSDMM. (**b**) Davies-Bouldin Index for GSDMM

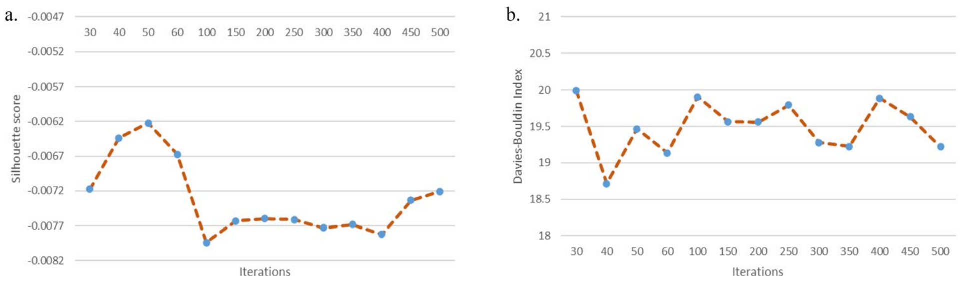


Fig. 8 Silhouette Score and Davies-Bouldin Index of GSDMM with 500 Iterations. (**a**) Silhouette score of GSDMM with 500 iterations. (**b**) Davies-Bouldin Index of GSDMM with 500 iterations

to train the model. Really low alpha values mean that a document may have one topic, while for a large alpha value, the topics become uniform.

GSDMM can determine the number of clusters in the dataset and requires only an upper bound K [60]. The number of clusters in GSDMM works based on the Movie Group Process [14], where the documents are randomly assigned to *K* topics at initialization. For each document, the probability distribution list over the k topics is generated. At every iteration, each document will be assigned to a topic that satisfies one or both of the following conditions:1) The new topic is empty, or it has more documents than the current topic. 2) The new topic has documents with a higher number of word occurrences. The number of topics generated by GSDMM is shown in Fig. 7.

As shown in Fig. 7a, the number of topics is not stable, and it is hard to determine the number of topics based on a single method. Even when the number of iterations increases (Fig. 7b), the number of topics changes according to the number of iterations, so if the algorithm is stopped at any point along the x-axis in Fig. 7a, b and a different number of topics is obtained. As illustrated in Fig. 6a, utilizing 40 iterations results in 43 topics with a Silhouette score of -0.0062 and a Davies-Bouldin Index of 19.46. Conversely, Fig. 8b shows that utilizing 30 iterations yields 51 topics with a Silhouette score of -0.0064 and a Davies-Bouldin Index of 18.71. In this case, even the internal measures are not able to determine the number of topics.

The results of using GSDMM and LDA with different numbers of topics and different numbers of iterations are represented in Table 5. It can be seen that the performance of both topic modeling methods is better when the number of iterations is 50 compared with 100 iterations. In addition, upon comparing the results based on the number of topics, the optimal outcomes were observed when the number of topics was set at 51, outperforming configurations with 43 and 86 topics.

Table 5 Comparative results using various number of topics

#Iterations	#Topics		GSDMM	LDA
50	43	SI	-0.0817	-0.0709
		DBI	14.1254	15.7441
	51	SI	**-0.0002**	**-0.0007**
		DBI	**10.0631**	**8.1954**
	86	SI	-0.0160	-0.0029
		DBI	15.5930	9.5644
100	43	SI	-0.0817	-0.0258
		DBI	14.1254	20.0367
	51	SI	-0.0018	-0.0145
		DBI	12.4406	20.3461
	86	SI	-0.0789	-0.0709
		DBI	12.8088	15.7441

Table 6 LDA and GSDMM with TCLD

LDA- TCLD		GSDMM - TCLD
SI	0.5488	**0.750806**
DBI	1.01957	**0.840491**

The findings revealed that the models do not produce good outcomes. Because there is not enough word co-occurrence, the documents are sparse, and the number of words in each document is limited. After manually checking the produced topics, it was observed that the documents were not effectively grouped, leading to suboptimal results. This outcome is attributed to the large number of topics represented. Some topics are incorrectly grouped with other documents; hence, an enhancement of the clustering results is proposed, specifically TCLD (see Sect. 3.3).

At this stage, the proposed TCLD is applied. Utilizing the output from the topic modeling, the Levenshtein distance ratio is calculated for each document within a topic. The Levenshtein distance serves to measure the difference between documents. This method is employed to filter out irrelevant and duplicated documents in the cluster, considering a Levenshtein distance ratio of less than 15%. Subsequently, the identified irrelevant documents are assigned to the outlier list (see Sect. 3.3). The outlier list has contained irrelevant documents and has been excluded from the dataset when the documents do not have an acceptable distance ratio (50% in this work) compared to the representative document in each topic. The representative document determines that the document has a greater distance ratio compared with other documents in each topic. Finally, the topics that have less than or equal to *mindoc* documents are marked as outliers. The computational time cost of this method is acceptable because the execution will be in parallel with all topics. The final results using the proposed method are represented in Table 6.

Table 6 shows that the SI and DBI are improved and the number of topics is reduced. Finally, the labeled dataset with the most topics that can be represented is summarized in Table 7. This dataset has been validated using eyeballing, intrinsic evaluation metrics, and human judgments.

Figure 9 presents the distribution of documents over the nine topics, where the figure gives the number of documents that belong to each topic.

The number of topics produced by the proposed approach was able to find nine topics with 1,421 documents among the 17,753 documents that exist in the collected tweets.

Table 7 The Summary of the Datasets with Identified Topics

Dataset	#Topics	# Documents	avg/max length	Vocabulary size
Generated Arabic Tweets dataset	9	1,421	12.5/39	6,680

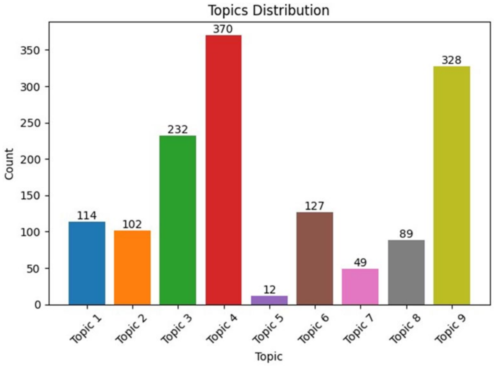


Fig. 9 The distribution of documents over the selected 9 topics

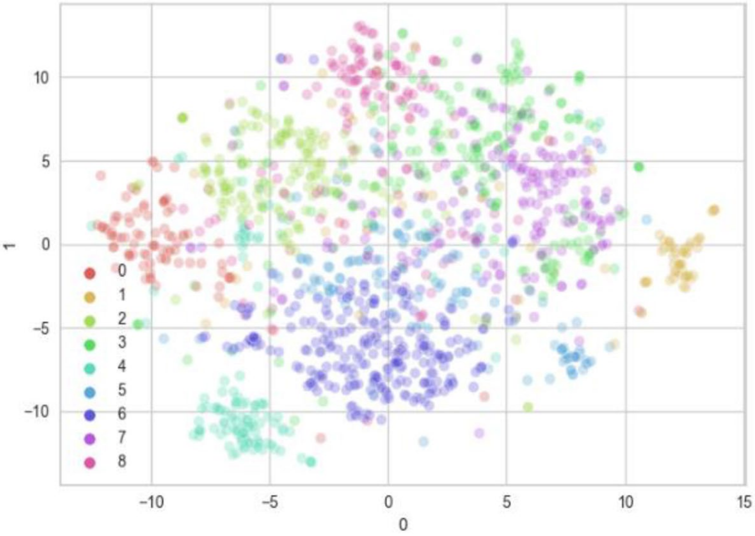


Fig. 10 Number of documents in each topic

The rest of the documents either do not have enough documents to produce a topic where the minimum number of documents is 10 to keep the topic. Figure 10 shows the TSNE dimensionality reduction method for visualization of the documents' clusters.

Results with known actual truth-labels

In this section, the dataset with ground-truth labels and the generated dataset from the previous section is utilized to validate the results of the proposed algorithm based on extrinsic evaluation measures. These measures evaluate clustering performance by comparing the ground truth-labels and predicted labels.

The methods used to evaluate the dataset are LDA and GSDMM for topic modeling. Table 8 shows the performance of clustering algorithms relying on purity, homogeneity, NMI, and AMI. The results illustrate that the method of GSDMM has superior performance. There is approx. 15% enhancement in the NMI and AMI scores when the results

Table 8 The performance of the clustering algorithms with TCLD.

	GSDMM	**LDA**	**TCLD-RD**		**TCLD-TW**	
			GSDMM	**LDA**	**GSDMM**	**LDA**
Purity	0.774	0.427	0.803	0.695	***0.814***	0.707
Homogeneity	0.704	0.153	0.644	0.548	***0.651***	0.558
NMI	0.432	0.144	0.606	0.519	***0.622***	0.528
AMI	0.283	0.134	0.602	0.513	***0.618***	0.521

of GSDMM are compared with the LDA model. Also, Table 5 shows the results of the proposed Levenshtein Distance for the Improvement of Topic Clustering (TCLD).

The proposed TCLD is applied to the output from GSDMM and LDA. The results show the comparison of two methods of dealing with the documents in the outlier list, as follows:

- TCLD-RD compares documents in the outlier list with the representative document in each topic.
- TCLD-TW compares documents in the outlier list with the top 10 words from each topic.

As shown in Table 8, the external measures indicate that the results of GSDMM and LDA are improved using the proposed TCLD method because the TCLD method evaluates each document in the current solutions provided by the topic modeling and enhances the clustering output. Moreover, GSDMM with TCLD-TW performs better than GSDMM with TCLD-RD, which means comparing the outlier list with the top 10 words of each topic can move the document correctly from the outlier list to the relevant topic.

TCLD-TW yielded 76 documents as outliers. Upon manual inspection, only 52 documents were correctly identified as outliers. As illustrated in Table 8, the optimal outcome was achieved when comparing the performance of the algorithms against clustering algorithms and TCLD. GSDMM emerged as the best performer and observed superior results with TCLD-TW compared to TCLD-RD.

Conclusion

In this paper, a new method for enhancing the topic modeling methods is proposed. The proposed topic clustering based on Levenshtein distance (TCLD) focuses on refining the position of each clustered document. The main objective of TCLD is to overcome the problem of irrelevant documents within each topic by either relocating them to more suitable topics or identifying them as outliers. This strategy allows the effective tackling of the challenge of an unknown number of topics and the effect of noises in the dataset. As a result, this strategy ensures the elimination of topics lacking sufficient document representation, leading to an accurate determination of the appropriate number of topics. Six English benchmark datasets have been used to evaluate the proposed methods and compare them to other topic modeling approaches, including LDA, GSDMM, LF-DMM, BTM, GPU-DMM, PTM, and SATM.

The proposed TCLD incorporates different approaches to comparing the distances between the documents by finding the representative document (TCLD-RD) or 10 top words from each topic (TCLD_TW). The results showed that TCLD_TW showed better performance compared with TCLD-RD, resulting in an 83% improvement in overall dataset performance concerning purity and a 67% improvement in terms of NMI compared to other topic modeling techniques.

Testing the proposed method on the collected Arabic tweets showed that the two topic modeling methods were unable to cluster the documents effectively, as presented in the result section. This might be due to the existence of irrelevant documents; the documents are sparse, and the number of words in each document is limited. TCLD was able to overcome this limitation, and finding roughly the correct number of topics eliminated the outliers. As a result, only 12% of the short tests in the Arabic datasets, according to human inspection, are incorrectly clustered. where TCLD_TW identified 76 tweets as outliers, but manual validation revealed that only 52 are outliers. Based on the results found, applying this model to more complex data and improving its results by accepting the movement of the texts between topics based on an objective function will be our future work direction.

Acknowledgments
Many thanks to the Deanship of Scientific Research at the Imam Abdulrahman Bin Faisal University for supporting this research.

Author contributions
All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Mutasem K. Alsmadi, Sana Jawarneh, Malek Alzaqebah, Ibrahim ALma-rashdeh, Mohammed Azmi Al-Betar, Maram Alwohaibi, Noha A. Al-Mulla, Eman AE Ahmed and Ahmad AL Smadi. The first draft of the manuscript was written by Mutasem K. Alsmadi and Malek Alzaqebah and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding
This work was supported by the Deanship of Scientific Research, Imam Abdulrahman Bin Faisal University, Grant number BASRC-083-2020.

Data availability
and Materials.
The datasets generated during and/or analysed during the current study are available in the [github] repository, [https://github.com/MlkZaq/arabic-short-text-clustering-datasets].

Declarations

Ethics approval and consent to participate
Not Applicable.

Consent for publication
Authors give consent for publication.

Competing interests
The authors declare no competing interests.

Received: 6 May 2023 / Accepted: 26 April 2024
Published online: 09 May 2024

References

1. Hirchoua B, Ouhbi B, Frikh B. Topic modeling for short texts: a novel modeling method. AI and IoT for Sustainable Development in Emerging Countries. Springer; 2022. pp. 573–95.
2. Singla M, Dutta M. Deep Classifier for News Text Classification Using Topic Modeling Approach, in *International Conference on Innovative Computing and Communications*, 2022, pp. 139–147: Springer.
3. Rani S, Kumar M. Topic modeling and its applications in materials science and engineering. Mater Today: Proc. 2021;45:5591–6.
4. Shah AM, Yan X, Tariq S, Ali M. What patients like or dislike in physicians: analyzing drivers of patient satisfaction and dissatisfaction using a digital topic modeling approach. Inf Process Manag. 2021;58(3):102516.
5. Qiang J, Qian Z, Li Y, Yuan Y, Wu X. Short text topic modeling techniques, applications, and performance: a survey. IEEE Trans Knowl Data Eng, 2020.
6. Ghali BE, El Qadi A. Context-aware query expansion method using Language models and latent semantic analyses. Knowl Inf Syst. 2017;50(3):751–62.
7. Blei DM, Ng AY, Jordan MI, Latent dirichllocation, et al. 2003.
8. Griffiths TL, Steyvers M. Finding scientific topics, *Proceedings of the National Academy of Sciences*, vol. 101, no. suppl 1, pp. 5228–5235, 2004.

9. Huang R, Yu G, Wang Z, Zhang J, Shi L. Dirichlet process mixture model for document clustering with feature partition. IEEE Trans Knowl Data Eng. 2012;25(8):1748–59.
10. Hussain SF, Bisson G. Text categorization using word similarities based on higher order co-occurrences, in *Proceedings of the* 2010 *SIAM International Conference on Data Mining*, 2010, pp. 1–12: SIAM.
11. Qiang J, Chen P, Wang T, Wu X. Topic modeling over short texts by incorporating word embeddings, in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2017, pp. 363–374: Springer.
12. Nigam K, McCallum AK, Thrun S, Mitchell T. Text classification from labeled and unlabeled documents using EM. Mach Learn. 2000;39(2):103–34.
13. Yu G, Huang R, Wang Z. Document clustering via dirichlet process mixture model with feature selection, in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2010, pp. 763–772.
14. Yin J, Wang J. A dirichlet multinomial mixture model-based approach for short text clustering, in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 233–242.
15. Wang X, McCallum A. Topics over time: a non-markov continuous-time model of topical trends, in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 424–433.
16. Alzaqebah M, et al. Cyberbullying detection framework for short and imbalanced arabic datasets. J King Saud University-Computer Inform Sci. 2023;35(8):101652.
17. Chen Y, Zhang H, Liu R, Ye Z, Lin J. Experimental explorations on short text topic mining between LDA and NMF based schemes. Knowl Based Syst. 2019;163:1–13.
18. Comito C, Forestiero A, Pizzuti C. Word embedding based clustering to detect topics in social media, in *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, 2019, pp. 192–199: IEEE.
19. Li AQ, Ahmed A, Ravi S, Smola AJ. Reducing the sampling complexity of topic models, in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 891–900.
20. Murshed BAH, Mallappa S, Abawajy J, Saif MAN, Al-Ariki HDE, Abdulwahab HM. Short text topic modelling approaches in the context of big data: taxonomy, survey, and analysis. Artif Intell Rev. 2023;56(6):5133–260.
21. Levenshtein VI. Binary codes capable of correcting deletions, insertions, and reversals. Soviet physics doklady. Volume 10. Soviet Union; 1966. pp. 707–10.
22. Zhang S, Hu Y, Bian G. Research on string similarity algorithm based on Levenshtein Distance, in 2017 *IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2017, pp. 2247–2251: IEEE.
23. Singh SP, Kumar A, Singh L, Mishra A, Sharma S. Strategy of Fuzzy Approaches for Data Alignment, in *Proceedings of International Conference on Computational Intelligence*, 2022, pp. 299–310: Springer.
24. Logan R. Optimized Levenshtein Distance for Clustering third-generation sequencing data. Northeastern University; 2021.
25. Omar A, Mahmoud TM, Abd-El-Hafeez T, Mahfouz A. Multi-label arabic text classification in online social networks. Inform Syst. 2021;100:101785.
26. Alhawarat M, Hegazi M. Revisiting K-means and topic modeling, a comparison study to cluster arabic documents. IEEE Access. 2018;6:42740–9.
27. Abuaiadah D, Sana JE, Abusalah W. On the impact of dataset characteristics on arabic document classification. Int J Comput Appl, 101, 7, 2014.
28. Alwehaibi A, Bikdash M, Albogmi M, Roy K. A study of the performance of embedding methods for arabic short-text sentiment analysis using deep learning approaches. J King Saud University-Computer Inform Sci, 2021.
29. Hegazi MO, Al-Dossari Y, Al-Yahy A, Al-Sumari A, Hilal A. Preprocessing Arabic text on social media, *Heliyon*, vol. 7, no. 2, p. e06191, 2021.
30. Oueslati O, Cambria E, HajHmida MB, Ounelli H. A review of sentiment analysis research in arabic language. Future Generation Comput Syst. 2020;112:408–30.
31. Elbarougy R, Behery G, El Khatib A. A proposed natural language processing preprocessing procedures for enhancing arabic text summarization. Recent advances in NLP: the case of Arabic Language. Springer; 2020. pp. 39–57.
32. Finegan-Dollak C, Coke R, Zhang R, Ye X, Radev D. Effects of creativity and cluster tightness on short text clustering performance, in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 654–665.
33. Wu X, et al. Top 10 algorithms in data mining. Knowl Inf Syst. 2008;14(1):1–37.
34. Mojahed A, de la Iglesia B. An adaptive version of k-medoids to deal with the uncertainty in clustering heterogeneous data using an intermediary fusion approach. Knowl Inf Syst. 2017;50(1):27–52.
35. Jahan M, Hasan M. A robust fuzzy approach for gene expression data clustering. Soft Comput. 2021;25(23):14583–96.
36. Bilancia M, Di Nanni M, Manca F, Pio G. Variational Bayes estimation of hierarchical Dirichlet-multinomial mixtures for text clustering. Comput Stat, pp. 1–37, 2023.
37. Lu Y, Mei Q, Zhai C. Investigating task performance of probabilistic topic models: an empirical study of PLSA and LDA. Inf Retr. 2011;14:178–203.
38. Reynolds DA. Gaussian mixture models. Encyclopedia Biometrics. 2009;741:659–63.
39. Qiang J, Li Y, Yuan Y, Wu X. Short text clustering based on Pitman-Yor process mixture model. Appl Intell. 2018;48(7):1802–12.
40. Wang Y, Li X, Zhou X, Ouyang J. Extracting Topics with Simultaneous Word Co-occurrence and Semantic Correlation Graphs: Neural Topic Modeling for Short Texts, in *Findings of the Association for Computational Linguistics: EMNLP 2021*, 2021, pp. 18–27.
41. Van Linh N, Bach TX, Than K. A graph convolutional topic model for short and noisy text streams. Neurocomputing. 2022;468:345–59.
42. Rajendra Prasad K, Mohammed M, Noorullah R. Visual topic models for healthcare data clustering. Evol Intel. 2021;14(2):545–62.
43. Aggarwal CC, Zhai C. A survey of text clustering algorithms. Mining text data. Springer; 2012. pp. 77–128.
44. Weisser C, et al. Pseudo-document simulation for comparing LDA, GSDMM and GPM Topic models on short and sparse text using Twitter data. Comput Stat. 2023;38(2):647–74.
45. Zheng CT, Liu C, San Wong H. Corpus-based topic diffusion for short text clustering, *Neurocomputing*, vol. 275, pp. 2444–2458, 2018.
46. Cheng X, Yan X, Lan Y, Guo J. Btm: topic modeling over short texts. IEEE Trans Knowl Data Eng. 2014;26(12):2928–41.

47. Yin J, Wang J. A model-based approach for text clustering with outlier detection, in 2016 *IEEE 32nd International Conference on Data Engineering (ICDE)*, 2016, pp. 625–636: IEEE.
48. Kumar J, Shao J, Uddin S, Ali W. An online semantic-enhanced Dirichlet model for short text stream clustering, in *Proceedings of the 58th annual meeting of the association for computational linguistics*, 2020, pp. 766–776.
49. Rashtchian C, Young P, Hodosh M, Hockenmaier J. Collecting image annotations using amazon's mechanical turk, in *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's Mechanical Turk*, 2010, pp. 139–147.
50. Naseem U, Razzak I, Eklund PW. A survey of pre-processing techniques to improve short-text quality: a case study on hate speech detection on twitter. Multimedia Tools Appl. 2021;80(28):35239–66.
51. Darwish K, Magdy W, Mourad A. Language processing for arabic microblog retrieval, in *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2012, pp. 2427–2430.
52. Mubarak H, Darwish K. Using Twitter to collect a multi-dialectal corpus of Arabic, in *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, 2014, pp. 1–7.
53. Alkhatib M, Barachi ME, Shaalan K. An arabic social media based framework for incidents and events monitoring in smart cities. J Clean Prod. 2019;220:771–85.
54. Elnagar A, Einea O. Brad 1.0: Book reviews in arabic dataset, in 2016 *IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, 2016, pp. 1–8: IEEE.
55. Taghva K, Elkhoury R, Coombs J. Arabic stemming without a root dictionary, in *International Conference on Information Technology: Coding and Computing (ITCC'05)-Volume II*, 2005, vol. 1, pp. 152–157: IEEE.
56. Frey BJ, Dueck D. Clustering by passing messages between data points, *science*, vol. 315, no. 5814, pp. 972–976, 2007.
57. Landing C, Tahvili S, Haggren H, Langkvis M, Muhammad A, Loufi A. Cluster-based parallel testing using semantic analysis, in 2020 *IEEE International Conference on Artificial Intelligence Testing (AITest)*, 2020, pp. 99–106: IEEE.
58. Yahyaoui H, Own HS. Unsupervised clustering of service performance behaviors. Inf Sci. 2018;422:558–71.
59. Petrovic S. A comparison between the silhouette index and the davies-bouldin index in labelling ids clusters, in *Proceedings of the 11th Nordic workshop of secure IT systems*, 2006, vol. 2006, pp. 53–64: Citeseer.
60. Wang B, Liakata M, Zubiaga A, Procter R. A hierarchical topic modelling approach for tweet clustering, in *International Conference on Social Informatics*, 2017, pp. 378–390: Springer.

Publisher's Note