Hindawi
Scientific Programming
Volume 2021, Article ID 7842631, 15 pages
https://doi.org/10.1155/2021/7842631

Research Article

Opinion Texts Clustering Using Manifold Learning Based on Sentiment and Semantics Analysis

Sajjad Jahanbakhsh Gudakahriz,[1] **Amir Masoud Eftekhari Moghadam**,[1] **and Fariborz Mahmoudi**[2]

[1]*Faculty of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*
[2]*Advanced Analytics Department, General Motors Company, Warren, MI, USA*

Correspondence should be addressed to Amir Masoud Eftekhari Moghadam; eftekhari@qiau.ac.ir

Received 3 July 2021; Revised 11 September 2021; Accepted 19 October 2021; Published 25 October 2021

Academic Editor: Sikandar Ali

Nowadays, opinion texts are quickly published on websites and social networks by various users in the form of short texts and also in high volumes and various fields. Because these texts reflect the opinions of many users, their processing and analysis, such as clustering, can be very useful in a variety of applications including politics, industry, commerce, and economics. High dimensions of the text representation decrease efficiency of clustering, and an effective solution for this challenge is reducing dimensions of texts. Manifold learning is a powerful tool for nonlinear dimension reduction of high-dimensional data. Therefore, in this paper, for increasing efficiency of opinion texts clustering, by manifold learning, dimensions of the represented opinion texts are reduced based on sentiment and semantics, and their intrinsic dimensions are extracted. Then, the clustering algorithm is applied to dimension-reduced opinion texts. The proposed approach helps us to cluster opinion texts with simultaneous consideration of sentiment and semantics, which has received very little attention in the previous works. This type of clustering helps users of opinion texts to obtain more useful information from texts and also provides more accurate summaries in applications, such as the summarization of opinion texts. Experimental results on three datasets show better performance of the proposed approach on opinion texts in terms of important measures for evaluating clustering efficiency. An improvement of about 9% is observed in terms of accuracy on the third dataset and clustering based on sentiment and semantics.

1. Introduction

In recent years, social networks and microblogs have expanded widely and are good platforms for sharing users' opinions in the form of short texts in various fields [1]. There are too many people on social networks who easily publish their opinions in various fields including politics, industry, trade, and economics [2]. Analysis of these opinion texts can be very important and influential in decision-making and policy-making in diverse fields. Due to the very high volume of the produced texts, it is not easy for users of these data to find what they need [3]. Therefore, there is a need for different techniques to automatically process these texts and extract necessary information [4]. These techniques perform the required analysis on texts in order to extract public opinions and determine different polarities of opinions in political, social, economic, and other contexts [5]. Opinion mining is an emerging and ever-dynamic field and has too many challenges. Some of the general challenges of this field are given in [6].

Clustering is one of the challenges and techniques used for the analysis of opinion texts. Clustering is a descriptive unsupervised technique in data mining, in which data are grouped into clusters so that similar samples are placed inside a cluster with respect to specific measures, and dissimilar or less similar ones should be placed in other clusters [7]. The main strength of clustering is that it can detect clusters without prior knowledge [8]. Opinion texts clustering can be used in applications, such as text summarization [9], topic detection [10], sentiment analysis [11],

question-answering systems [12], and recommender systems [13].

For opinion texts clustering, it is necessary to represent texts in the vector form [14]. In general, methods used for the representation of texts are divided into two categories, statistics-based and semantics-based methods. In statistics-based methods, the text is represented based on the number of repetitions of each word in the text. Common methods in this category include term frequency-inverse document frequency (TF-IDF) [15] and latent semantic analysis (LSA) [16]. Important advantages of statistics-based representation methods are intuitive representation, simple representation, and language-independent representation. Important advantages of statistics-based representation models are intuitive, simple, and language-independent representation. Important disadvantages of statistics-based representation models are loss of all word order information, loss of word meaning, high dimensionality, and data sparsity problems [17]. In the semantics-based method, the text is represented based on semantics and syntactic of words. Common methods in this category include Word2Vec [18] and Doc2Vec [19]. Important advantages of semantics-based representation models are using semantics for representation, low dimensionality representation, and no data sparsity problems. Important disadvantages of semantics-based representation models are the need for a semantics dictionary, language-dependent systems, and computation cost [17]. As opinion texts are short and sparse, this sparse representation and their high dimensions have posed a major challenge to the clustering of such texts [20]. Using Word2Vec and Doc2Vec as text representation models solves the problem of sparse display of short texts, and to some extent, it also solves the problems regarding the representation of high-dimensional texts [21], but these methods represent the text with 200–500 dimensions where there is still the problem of high dimensions.

High-dimensional data increase computational time and memory required for processing, and also, the existence of noise and the low number of samples compared to high-dimensional data influences processing efficiency negatively. Thus, for reducing the problems of such data, dimension reduction is considered [22]. Dimension reduction methods take the sample x with D features and generate the sample y with d features, in which $d << D$.

$$(x_1 \cdot x_2, \ldots, x_n)\varepsilon R^D \xrightarrow{f} (y_1 \cdot y_2, \ldots, y_n)\varepsilon R^d. \quad (1)$$

Dimension reduction is one of the first steps in efficient data analysis and is an important preprocessing step in many fields of information processing including data retrieval, pattern recognition, text processing, data visualization, and data compression. Also, reducing text dimensions can increase clustering efficiency on texts [8, 23]. In terms of clustering with/without dimension reduction, an overview of opinion texts clustering is presented in Figure 1.

In this paper, opinion texts are clustered using nonlinear dimension reduction methods. Manifold learning, as a nonlinear dimension reduction method, is used, and opinion texts are reduced in terms of their dimensions based on sentiment and semantics. Then, the clustering algorithm is applied to these dimension-reduced texts. Reducing the dimension of the represented opinion texts makes it possible to optimally represent these texts based on sentiment and semantics. For this purpose, dimensions of these texts are reduced based on the sentiment and semantics of the text, and their intrinsic dimensions are extracted by applying the ISOMAP algorithm to the texts represented in Doc2Vec format. Then, the K-Means algorithm [24] is utilized for the given low-dimensional texts, and clustering is performed. Nonlinear dimension reduction methods are often more powerful than linear methods, because the relationship between intrinsic and measured features may be much richer than a linear relationship between them. Among the different nonlinear dimension reduction methods, manifold learning methods on various datasets have desirable results. Due to the fact that the relationship between the intrinsic features of opinion texts and the features in these texts is nonlinear, in this paper, to reduce the dimension of texts, manifold learning will be used.

This paper has two main contributions:

(1) A new approach is presented to reduce dimension and representation of opinion texts based on sentiment and semantics by manifold learning

(2) This approach helps to cluster opinion texts with high efficiency and simultaneous consideration of sentiment and semantics after representation of opinion texts in a low-dimensional manner based on sentiment and semantics

The rest of the paper is organized as follows. In Section 2, the related and previous literature on the field of this study is reviewed. In Section 3, manifold learning is explained and the ISOMAP algorithm is investigated as one of the common manifold learning algorithms and is used to reduce dimensions. Section 4 describes the proposed method for opinion texts clustering in detail. In Section 5, the simulations performed to evaluate the efficiency of the proposed method and relevant results are presented. Finally, the conclusion is provided in Section 6.

2. Related Works

Dimension reduction is a successful method for opinion texts clustering. In this section, first, the previous methods proposed for opinion texts clustering by reducing their dimensions are reviewed. It is noteworthy that manifold learning, as a nonlinear dimension reduction method, has not been used for clustering of high-dimensional opinion texts yet, but it has been applied to represent the texts with reduced dimensions. In the second part of this section, the previous methods presented for reducing text dimension using manifold learning will be reviewed.

2.1. Opinion Texts Clustering by Dimension Reduction. High dimensions of data are a barrier to extracting useful information, and dimension reduction techniques are used as a successful method to overcome this challenge. In the

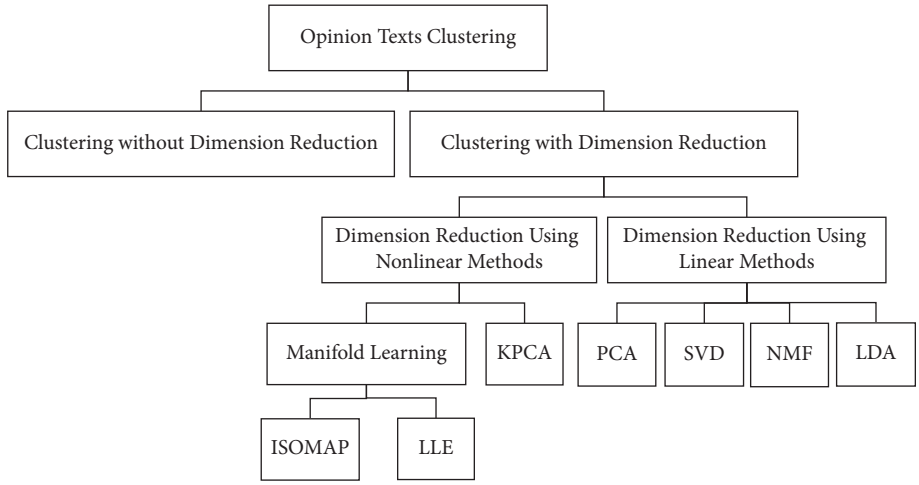


Figure 1: An overview of opinion texts clustering with/without dimension reduction.

previous works presented for opinion texts clustering using dimension reduction, mostly linear dimension reduction methods including principal component analysis (PCA), singular value decomposition (SVD), nonnegative matrix factorization (NMF), and linear discriminant analysis (LDA) have been used, and also in some cases, feature selection method has been applied. In [25–30], first, dimensions of the opinion texts were reduced by linear dimension reduction methods, and then, clustering was performed. In [25], the TF-IDF model was used to represent the text for clustering; then, dimensions were reduced using the SVD and NMF methods. Finally, clustering was performed on the given dimension-reduced texts by the K-Means algorithm. Simulation on the dataset of 20 newsgroups showed an improvement in clustering performance when dimension reduction methods of SVD and NMF were utilized. In [26], the short text clustering-linear discriminant analysis (SKP-LDA) method was presented for the clustering of short Chinese texts. This method has been proposed to solve the problem of low efficiency of sentiment analysis and semantics extraction and, as a result, low clustering accuracy. In the proposed method, first, a word bag was defined based on the synchronicity of sentimental words, and then, a definition of topic-specific words and words related to the subject was provided. For improving the quality of semantics analysis, information from topic-specific words and topic-related words was entered into the LDA model. Finally, 30 high-ranking special word sets obtained by the LDA model were clustered by the K-Means algorithm. In [27], different graph-based clustering methods were presented based on dimension reduction for the clustering of microblogs. The results of simulation done on several datasets collected from the microblogs showed that the proposed methods performed better than the standard and classical text clustering algorithms. Shortness, noisy nature, and the large volume of texts in microblogs are among the main challenges of clustering these texts. In [28], for clustering of short texts, first, the texts and their descriptions were converted into a low-density presentation using the convolutional neural network (CNN). Then, through this representation, the similarity between each text and other comments was measured. On the other hand, given that short texts must be strongly related to their descriptions, the differences and similarities between the related and unrelated descriptions were taken into account in CNN training. Finally, representation was expanded with multidimensional properties obtained from location, time, and other information. This provides a strong representation of short texts and increases clustering efficiency. In [29], another method was presented for clustering of microblog texts, which uses the feature selection technique as a dimension reduction method. In the proposed method, after preprocessing of the texts, the LDA model is used as a topic modeler. This model is used to specify a property that is compatible with the topic of a database and provides a list of the reduced properties that can be used to represent any topic in the entire database. Finally, for determining the cluster of each tuple, the hamming distance between each subject properties vector and the tuple is measured, and the cluster with the minimum distance is selected as the final cluster for the tuple. This process continues until the entire database is clustered. The proposed method was applied to the microblog database related to four incidents and the simulation results showed better performance in the proposed method compared to several existing clustering techniques. In [30], a hybrid dimension reduction method was presented for opinion texts clustering. In the proposed method, dimension reduction was performed by combining feature selection and feature extraction methods. For this purpose, first, two sets of features were selected within the primary features by two attribute selection methods. Then, these two feature sets were merged using one of the three methods of Union, Intersection, and Modified Union. In the Union method, all the properties of the two sets were selected. In the Intersection method, only common features between the two feature sets were selected. In the Modified Union method, first, the Union method was applied on the properties of two top-ranked sets of features, and then, the Intersection method was utilized for the other features. Next, the integrated feature set was reduced by the PCA feature extraction

method, and the final feature set was clustered using the K-Means algorithm. Finally, the sentiment score of each cluster was calculated using the SentiWordNet database.

2.2. Dimension Reduction of Texts Using Manifold Learning. Manifold learning is one of the successful methods of dimension reduction that has been used in various studies to reduce the dimension of texts. In [31], a new method was provided for the representation of biomedical sentences based on manifold learning. In the proposed method, first, biomedical sentences were represented using the trained sentence representation method. Then, a neighborhood representation graph was constructed using manifold learning to determine the local geometric structure of the sentence. In this way, the basic rules of the sentences were revealed and reembedded in sentence representation using manifold learning. As a result, the geometric structure of relationships between sentence representations was effectively described, having a positive effect on the efficiency of subsequent operations performed on representation. In [32], a simple and effective approach was introduced to represent words. This approach performs in a postprocessing manner and removes top components from all words. This simple operation can be used to embed words in downstream tasks or as initialization to teach task-specific embeds. In [33], manifold learning was used to reduce the dimension of texts while maintaining their semantic distances. The main goal was preserving the semantic connections between the texts and reducing dimensions from high to low. The proposed method uses manifold learning and reduces dimension by preserving mutual information between the texts. This method has been used to summarize texts. Large volumes, on the one hand, and large dimensions, on the other hand, have posed a major challenge to operations, such as the classification and clustering of big data. In [34], a new method was proposed to reduce dimensions of the big data. This method uses ISOMAP and LLE algorithms and performs dimension reduction. For evaluating efficiency of the proposed method, SVM and Random Forest classification algorithms were used, and the given dimension-reduced data were classified, confirming good performance of the proposed dimension reduction method. Word embedding methods seek to discover a space based on Euclidean measure to map words into vectors, which is performed based on cooccurrence of words in a corpus. Embedding words may misjudge the similarity between words. For solving this problem, a method was proposed in [35] to reembed the pretaught words embedded by a step in the manifold learning. This method also takes into account geometric information of the words that helps to better estimate the similarity between the words. In [36], statistical manifold learning was used to represent texts. The proposed method is an effective text learning framework whose main purpose is using the hidden topics to represent and measure texts. Assuming that words with the same topic follow the same Gaussian distribution, the texts are represented as a combination of themes. In [37], a word representation method was introduced for retrieving semantic space criterion. This method integrates existing word representation algorithms and applies the manifold learning algorithm to them. For this purpose, the corpus with word cooccurrences was compared with semantic similarity of words, and it was shown that cooccurrence of words is consistent with the Euclidean semantic space hypothesis.

Table 1 summarizes the related works reviewed in Sections 2.1 and 2.2, so that it helps better compare studies. In cases where the dimension reduction method is written manifold learning, the used algorithm is not specified in the relevant papers and includes all manifold learning algorithms.

3. Manifold Learning

In general, there are two methods to reduce dimensions of high-dimensional data: feature selection and feature extraction. In the feature selection method, some features are selected from the initial features. To do this, features that have more potency to distinguish samples are chosen. In the feature extraction method, some new features are generated based on the initial features, so that the number of features is less than the original ones. In feature extraction, the generated features are new features in another space, and no correlation can be found between the initial and generated features.

Dimension reduction based on feature extraction methods is divided into two categories: linear and nonlinear [38]. Linear methods are suitable in cases where there is a linear relationship between the data, which is also considered as a limitation of these methods. They cannot manage nonlinear and complex real-world data. The most popular linear dimension reduction methods are PCA [39], LDA [40], NMF [41], and SVD [42]. Nonlinear dimension reduction methods, often known as manifold learning, are used in cases where there is no linear and seemingly significant relationship between the data, and they can overcome the limitations of linear methods [43].

The goal of manifold learning is mapping a set of high-dimensional data to a set of low-dimensional data. The reason for naming this method as "manifold learning" is that this method tries to make a manifold from a set of learning points. This method is a powerful tool to reduce the nonlinear dimension of data, in which intrinsic parameters of the system, as the main factors in distinguishing data from each other, are identified and the whole set is placed on a manifold that expresses the actual relationship of the parameters. Thus, the relationship between data is expressed in a low-dimensional space [38]. If the primary data set has D dimensions, the manifold learning problem would be determining the position of each record of the primary data in a space d, where d is much smaller than D. Figure 2 shows an overview of the main purpose of manifold learning.

ISOMAP [45] and LLE [46] are the most famous manifold learning algorithms. The main idea in these methods is reducing system dimensions based on maintaining the relationship between data, which can be expressed as maintaining distance. In this paper, the

Table 1: Summary of related works.

	Paper	Dimension reduction method	Details of the work
Opinion texts clustering by dimensionality reduction	[25]	SVD and NMF	Texts represented by TF-IDF, reduced by SVD and NMF, clustered by K-Means
	[26]	LDA	Texts modeled and reduced by LDA, clustered by K-Means
	[27]	LDA and TF-IDF	Texts modeled and reduced by LDA and TF-IDF, clustered by graph-based methods
	[28]	CNN	Texts and their descriptions converted to low-density representation by CNN
	[29]	Feature selection	Texts reduced by feature selection, modeled by LDA
	[30]	Feature selection and extraction	Texts reduced by a combination of feature selection and extraction methods, clustered by K-Means
Dimension reduction of texts using manifold learning	[31]	Manifold learning	Biomedical sentences reduced by manifold learning methods
	[32]	Manifold learning	Words represented in a simple and effective approach by manifold learning
	[33]	LLE	Texts reduction by LLE with maintaining semantics distances
	[34]	LLE and ISOMAP	Big data dimension reduction by LLE and ISOMAP
	[35]	Manifold learning	Words embedding by manifold learning
	[36]	Manifold learning	Texts representation by statistical manifold learning
	[37]	SVD	Word representation for retrieving semantic space criterion by SVD

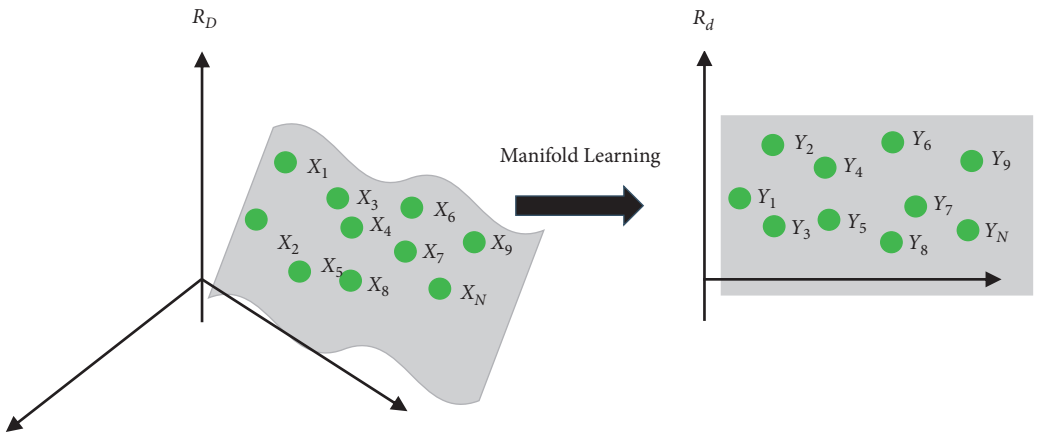


Figure 2: The main purpose of manifold learning [44].

ISOMAP algorithm is used to reduce text dimensions, which is explained in the future.

The ISOMAP algorithm as a part of the global method is among the first manifold learning algorithms, and it preserves geometric features of the input data set and reduces dimensions nonlinearly by maintaining the distance between the data. This algorithm is an extended type of linear dimension reduction method of multidimensional scaling (MDS) algorithm and is a classic method for embedding heterogeneous information in a Euclidean space. The ISOMAP algorithm maps high-dimensional data to low-dimensional ones by maintaining the distance between each data pair and consists of three general steps: (1) finding neighboring points for each of data points, (2) calculating the geodesic distance for the data points, and (3) implementation of MDS algorithm (Algorithm 1). In Step 1, k-nearest neighbors are calculated for each data point. To do this, first, a matrix of distance or similarity is created between the data points and then, based on this matrix, k-nearest neighbors to each data are found and a corresponding matrix is created. The shortest path between two points on the graph is the geodesic distance between the data points, which can be determined in Step 2 by dynamic algorithms, such as Dijkstra's and Floyds¯ algorithms. Finally, in Step 3, the MDS algorithm is applied and the next dimension reduction step is performed. The MDS algorithm maps high-dimensional data to low-dimensional ones. After calculating the geodesic distance and storing it in matrix D, equation (2) will be established.

$$X^T X = -\frac{1}{2}\left[I - \frac{1}{N}1_N 1_N^T\right] D \left[I - \frac{1}{N}1_N 1_N^T\right]. \quad (2)$$

Then, using equation (3), eigenvalues and eigenvectors of the matrix $X^T X$ are calculated. In this equation, λ is the diagonal matrix and U is the orthogonal matrix.

$$X^T X = U\lambda U^T. \quad (3)$$

Finally, the necessary low-dimensional mapping to the space is performed by maintaining the eigenvectors corresponding to d largest eigenvalues equation.

$$Y = \text{diag}\left(\sqrt{\lambda_1} \cdot \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_d}\right)[U_1 \cdot U_2, \ldots, U_d]. \quad (4)$$

4. Proposed Method

As mentioned in the previous sections, the purpose of this study is to improve opinion texts clustering by reducing their dimensions using manifold learning. Therefore, the proposed method mainly focuses on opinion texts clustering with simultaneous consideration of sentiment and semantics. For this purpose, after preparing the texts for clustering, first, their dimensions are reduced using the ISOMAP algorithm, and then, clustering is performed. The dimension reduction step will be performed in three modes and based on semantics, sentiment, and a combination of them. In fact, dimension reduction based on a combination of sentiment and semantics helps us to cluster opinion texts in terms of both metrics of sentiment and semantics, which is necessary for many applications, while in the previous works, very little attention has been paid to this important issue. Figure 3 shows an overview of the proposed method. This method is described in detail in future sections.

4.1. Acquisition of Opinion Texts and Preprocessing. In the acquisition step, the opinion texts to be clustered are received in the form of datasets. The datasets used in this paper have two main fields. The first field is a text field and contains the opinion written by an individual on a website or social network. In fact, this field is analyzed, and clustering is done on it. The second field is a label field and contains a label of an actual cluster of opinion texts. In fact, the actual cluster to which the text field belongs is located in this field. For performing clustering, the text field is processed and clustered, but during clustering, the cluster field is not used, and it is used after the completion of clustering to evaluate clustering performance.

The main purpose of text preprocessing is to minimize structural and writing errors in the text. Due to the shortness of opinion texts, abbreviations, irregular expressions, and infrequent words are widely used in these texts, and these cases produce high noise levels in these texts and influence the quality of text analysis. For solving this challenge, different preprocessing techniques should be used [2, 47]. In the proposed method, preprocessing was performed with high sensitivity, and an attempt was made to reduce the noise of the opinion texts to a great extent and also bring the texts as close as possible to the structured texts. The preprocessing approach performed in the presented method consists of two categories. The first category removes the unwanted and noisy elements from texts. The methods used for this purpose are removing duplicate letters, chunks, emojis, emails, tweets' signs, hashtags, extra spaces, and special characters. The second category of methods tries to get the texts out of the unstructured state and bring them into the structured state as much as possible. Methods used for this purpose are converting acronyms, removing contents in parentheses, lowercasing, removing stop words, and lemmatizing.

4.2. Dimension Reduction of the Opinion Texts by ISOMAP Algorithm. After preprocessing of the texts, they are converted into vector form. For this purpose, the Doc2Vec model was used in the proposed method. Although the Doc2Vec model takes the representation form of texts out of sparse mode and greatly reduces the dimension of the text, it still has high dimensions. In the proposed method, the text represented with the Doc2Vec model has 300 dimensions. For increasing the efficiency of analysis of these texts, it is necessary to reduce text dimensions. For decreasing dimensions, first, intrinsic dimensions of the texts need to be estimated. For this purpose, a scree plot is created, in which the index of text elements is placed on the horizontal axis and the corresponding eigenvalue is placed on the vertical axis, and a curve is drawn based on these values. Then, elbow points are marked on the curve, each of which can be a candidate for the intrinsic dimension of the texts. After estimating the intrinsic dimension, the ISOMAP algorithm is used to reduce text dimensions, which has three main steps. The first and most important step is building a similarity matrix. In the proposed method, for constructing a neighborhood graph, a similarity matrix of the texts is extracted first. In this paper, the opinion texts were clustered based on semantics, sentiment, and a combination of them. Therefore, in building a similarity matrix, this matrix is created based on the method with which dimensions are reduced.

In the first mode, for clustering of texts based on semantic, a similarity matrix is constructed based on the semantic distance between the texts. The metric used to measure the semantic distance is the Euclidean distance equation.

$$\text{Euclidean distance}\left(t_i \cdot t_j\right) = \sqrt{\sum_{k=1}^{n}\left(t_{ik} - t_{jk}\right)^2}, \quad (5)$$

where t_i. t_j are two opinion texts represented by Doc2Vec model, n is the length of Doc2Vec vectors, and t_{ik} is the ith element of kth opinion text in Doc2Vec vector.

In the second mode, a similarity matrix is constructed based on the sentiment distance between the texts. The valence aware dictionary for sentiment reasoning (VADER) tool [48] was used to measure the sentiment of each text. VADER is a lexicon and rule-based sentiment analysis tool that is specifically compatible with sentiments expressed on social media. This tool is sensitive to both polarity (positive/negative) and intensity (strength) of opinion texts. VADER has been included in the NLTK package and can be applied directly to the unlabelled opinion texts. For constructing a sentimental similarity matrix, the first degree of the

Input: Initial dataset with D dimensions
Output: Dimension-reduced dataset with d dimensions ($d << D$)
Algorithm:
Construct the weighted graph G from the distances pairwise for all points in the input and find the graph $G\_$ by applying the nearest neighbor algorithm on the graph G.
Compute the shortest path graph $G\_\_$ between all pairs of nodes from graph $G\_$. This might be done by the all-pairs Dijkstra's or by the Floyd Warshall algorithm.
Use $G\_\_$ to construct the p-dimensional embedding using the MDS algorithm. In other words, the MDS method can now be used to construct a representation in subspaces of the Rn.

Algorithm 1: The pseudocode of ISOMAP algorithm.

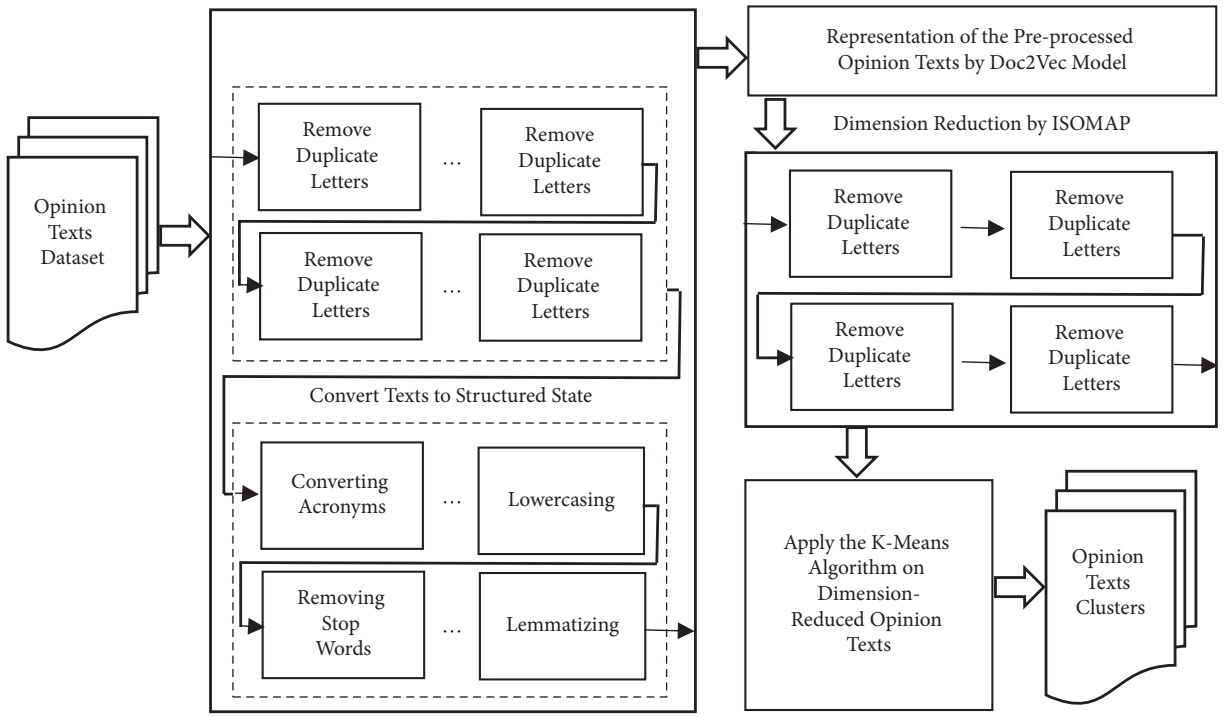


Figure 3: The block diagram of the proposed method.

sentiment of each text is determined using the VADER tool, and then, the sentimental distance between all the texts is measured by calculating the difference in the degree of the sentiment of the desired texts. The degree of sentiment indicates the amount of sentiment in the text, and we can understand the degree of positivity/negativity of the topics of each text [3].

In the third mode, where our main goal is clustering the texts based on sentiment and semantics of the texts, in constructing a similarity matrix, first, the semantics distance of the texts is calculated using the cosine similarity metric (equation (6)) and is stored in a matrix. Then, the sentimental distance between the texts is measured using the VADER method and is stored in another matrix. Considering that, in this mode, clustering is done by considering semantics and sentiment at the same time and it is necessary to combine the semantics and sentiment of the texts, in combining these two metrics, the use of the cosine similarity metric shows better efficiency.

$$\text{Cosine similarity}\left(t_i \cdot t_j\right) = \frac{t_i \cdot t_j}{\left\|t_i\right\|\left\|t_j\right\|} = \frac{\sum_{k=1}^{n} t_{ik} t_{jk}}{\sqrt{\sum_{k=1}^{n} t_{ik^2}} \sqrt{\sum_{k=1}^{n} t_{jk^2}}}. \tag{6}$$

where $t_i \cdot t_j$ are two opinion texts represented by Doc2Vec model, n is the length of Doc2Vec vectors, and t_{ik} is the ith element of kth opinion text in Doc2Vec vector.

Eventually, the final similarity matrix is created by combining these two matrices using Algorithm 2. In this algorithm, DSM is a matrix that stores the semantic distance between texts and DSN is a matrix storing the sentiment distance between texts. The two matrices are combined according to the procedure specified in the algorithm, and the matrix forms the final similarity, i.e., D. In combining the

Input: Opinion texts preprocessed and represented by Doc2Vec Model (TF)
Output: Dimension-reduced opinion texts based on semantics and sentiment (RF)
Definitions:
TF: Opinion texts represented by Doc2Vect Model
RF: Dimension-reduced opinion texts
DSM: Semantics distance matrix between opinion texts by a cosine similarity metric
DSN: Sentiment distance matrix between opinion texts by VADER method
D: Final distance between opinion texts based on semantics and sentiment
NG: Neighborhood graph between opinion texts
Min_DSM, Max_DSM: Minimum and Maximum semantics distance between opinion texts
Min_DSN, Max_DSN: Minimum and Maximum sentiment distance between opinion texts
Algorithm:
Estimate intrinsic dimension of opinion texts
Create DSM and DSN matrices based on TF
Calculate Min_DSM, Max_DSM, Min_DSN and Max_DSN
D = DSM
for i in range rows of D matrix
  for j in range columns of D matrix
    if (DSN[i][j] ≥ (Max_DSM/1.5))
      $D[i][j]$ = Max_DSM
    if (DSN[i][j] < (Max_DSN/1.5) and DSN[i][j] ≥ (Max_DSN/2.0))
      $D[i][j]$+ = Max_DSM/2.0
    if (DSN[i][j] < (Max_DSN/2.0) and DSN[i][j] ≥ (Max_DSN/4.0))
      $D[i][j]$− = (Max_DSM/14.0)
    if (DSN[i][j] < (Max_DSN/4.0))
      $D[i][j]$− = (Max_DSM/7.0)
Create NG graph based on D matrix
Apply MDS algorithm on NG and product RF

Algorithm 2: The pseudocode of the proposed method.

Input: N data that must be clustered and k as the number of clusters
Output: k clusters of input data
Algorithm:
Randomly determine k data as the centroids of the clusters
Repeat
  Assign each data to its closest centroid
  Compute the new centroid of each cluster
Until The cluster centroids do not change

Algorithm 3: The pseudocode of the K-Means clustering algorithm.

semantics and sentiment similarity matrices to construct a distance matrix, the main effect will be the sentiment similarity matrix, where, in different states of this matrix, the semantics matrix puts its effect on this matrix. The numbers obtained in the combination of these matrices are obtained based on experiments.

After constructing the similarity matrix between the texts, the neighborhood graph of the texts is constructed, and then, the MDS algorithm is applied to this graph and the dimension-reduced vectors are obtained.

4.3. Clustering of Dimension-Reduced Opinion Texts. After performing dimension reduction on the opinion texts, the final step of clustering will take place. In this step, the dimension-reduced texts stored as vectors are clustered using the clustering algorithm. The clustering algorithm used in this paper is the K-Means algorithm, which is a very famous and widely used clustering algorithm and has high efficiency in text clustering. The pseudocode for the K-Means algorithm is given in Algorithm 3.

5. Experiments and Evaluation

Simulations were performed to show the effectiveness of the proposed approach on the clustering performance of opinion texts. The used datasets, performance measures, and simulation results are presented in the following.

5.1. Datasets. For increasing the validity of the evaluation of the proposed approaches, various datasets should be used.

Therefore, three diverse datasets are used in the simulation. The used datasets are shown in Table 2.

The "Search Snippets" dataset contains 2280 texts collected by Google search, which includes 8 clusters, namely, Business, Computers, Culture-Art-Entertainment, Education-Science, Engineering, Health, Politics-Society, and Sports, according to the semantics of the texts. The "Twitter Dataset" includes 2000 tweets collected from Twitter, which are tagged based on sentiment into positive and negative clusters and are related to various topics, such as Sports, Saints, Funny Images, etc. The "Twitter-Sentiment-Corpus-3" dataset contains 3424 tweets, which are collected from Twitter on 4 topics and are clustered into three clusters of positive, negative, and neutral based on sentiment and into four clusters of Apple, Google, Tagged Facebook, and Microsoft according to semantics. In this paper, 1091 tweets from this dataset were selected that had positive or negative sentiment and are used in three labeled modes. In the first mode, 1091 tweets included four clusters of Apple, Google, Facebook, and Microsoft based on semantics. In the second mode, 1091 tweets included two positive and negative clusters based on sentiment. In the third mode, considering semantics and sentiment simultaneously, 1091 tweets included 8 clusters of Apple-Positive, Apple-Negative, Google-Positive, Google-Negative, Facebook-Positive, Facebook-Negative, Microsoft-Positive, and Microsoft-Negative.

5.2. Evaluation Measures. In this paper, various measures were used to evaluate the efficiency of clustering methods including accuracy, precision, recall, *F*-score, adjusted rand index (ARI), normalized mutual information (NMI), completeness, and homogeneity.

Accuracy. This measure determines which percentage of the texts are properly clustered and placed in their respective clusters. In other words, accuracy is the closeness of the predicted clusters to true clusters.

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}. \tag{7}$$

F-Score. It determines the harmonic mean of precision and recall and helps to have a trade-off between precision and recall. If one is strengthened and the other is weakened, this metric quickly decreases.

$$F - \mathrm{score} = \frac{2 \times \mathrm{precision} \times \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}. \tag{8}$$

ARI. This measure is the corrected-for-chance version of the rand index (RI) and determines the degree to which real and clustered labels match with each other.

$$\mathrm{ARI} = \frac{\mathrm{RI} - \mathrm{expected\,RI}}{\mathrm{Max}\ (\mathrm{RI}) - \mathrm{expected\,RI}}. \tag{9}$$

In this regard, the value of RI is calculated using

$$\mathrm{RI} = \frac{\mathrm{number\ of\ agreeing\ pairs}}{\mathrm{number\ of\ pairs}}. \tag{10}$$

NMI. This measure calculates the statistical similarity between the created clusters and predefined labels.

$$\mathrm{NMI} = \frac{I(C;K)}{(H(C) + H(K))/2}. \tag{11}$$

Here, C is a random variable representing the initialization of points inside the cluster, and K is a random variable representing the group label. Also, $I(C;K) = H(C) - H(C/K)$ is the amount of mutual information between the variables C and K, $H(C)$ is the expansion of the variables C, and $H(C/K)$ is the expansion of the variable C, given K.

Completeness. This measure determines the degree of completeness of the clustering algorithm. A complete clustering is achieved when each cluster has the data belonging to the same cluster.

$$\mathrm{Completeness} = 1 - \frac{H(C/K)}{H(C)}. \tag{12}$$

For calculating some of the above measures, TP, TN, FP, and FN are required. The definition and method of calculating these values are given as follows:

TP: if a pair of observations in the same category is in the same cluster, the clustering result for this pair is indicated by TP

TN: if a pair of observations in two separate categories is also placed in two separate clusters, the clustering result for this pair is shown with TN

FP: if a pair of observations in two separate categories fits into a cluster, the clustering result for this pair is indicated with FP

FN: if a pair of observations in the same category is placed in two clusters by mistake, the clustering result for this pair is also shown with FN

5.3. Experimental Results and Discussion. In this section, results of evaluating the efficiency of the proposed method on each of the datasets are given. Recently, various methods have been proposed for clustering; for example, in [49], successful methods for clustering have been proposed, but these methods are not for opinion texts clustering. On the other hand, the K-Means is the most successful and common clustering method used in opinion texts clustering; therefore, this algorithm is used as a basic algorithm. For a more accurate evaluation of the proposed method, this method was compared with the basic algorithm used (K-Means), the basic algorithm along with linear dimension reduction methods including PCA and SVD algorithms, which have been widely used in previous opinion texts clustering researches [50]. Also, the proposed method was compared with the basic algorithm along with the nonlinear dimension reduction method (KPCA algorithm).

Table 2: Details of the used datasets.

Dataset	Description	Number of texts	Clustering based on	References
Search Snippets	Google Web Search Transactions	2280	Semantics	[13]
Twitter Dataset	Twitter tweets	2000	Sentiment	[2]
Twitter-Sentiment-Corpus-3	Twitter tweets	1091	Semantics and sentiment	[2]

5.3.1. Intrinsic Dimension of Opinion Texts. For determining the intrinsic dimension of opinion texts, first, the intrinsic dimension of the texts is estimated by the proposed method as explained in Section 4.2. Given that several intrinsic dimensions were estimated on the datasets, for obtaining the final intrinsic dimension, experiments are performed and accuracy of clustering on each dataset is obtained. Then, the final dimensions are determined based on the highest obtained accuracy. The results of experiments performed on different datasets are shown in Figure 4, in which the horizontal axis shows the estimated intrinsic dimension and the vertical axis shows the value of accuracy obtained for each estimated intrinsic dimension.

5.3.2. Results. Since the Search Snippets dataset is labeled only by semantics, only semantic-based dimension reduction was applied to this dataset and the results are shown in Table 3. It should be noted that each clustering method was performed 50 times, and the average of each measure is given in tables. Table 3 shows the results regarding the performance of the proposed method and the compared methods based on the Search Snippets dataset. As can be seen, in this dataset, dimension reduction does not have a positive effect on the performance of clustering algorithms and the best performance was related to the K-Means algorithm without using dimension reduction.

Given that the Twitter Dataset is only labeled based on sentiment, the sentiment-based dimension reduction was applied to this dataset, and the results are shown in Table 4. As can be seen in Table 4, in this dataset, the proposed method has a positive effect on opinion texts clustering and better results are obtained.

The Twitter-Sentiment-Corpus-3 dataset is labeled based on semantics, sentiment, and a combination of them. Therefore, three-dimension reduction modes were applied to this dataset, and the results are shown in Tables 5–7. Table 5 shows the simulation results done on the Twitter-Sentiment-Corpus-3 dataset that are related to the mode, in which the dataset is labeled based on semantics, and in the proposed method, dimension reduction is done only based on semantics. As can be seen, in this mode, the proposed method also shows the best performance compared to other methods.

Table 6 shows the simulation results done on the Twitter-Sentiment-Corpus-3 dataset. These results are related to the mode, in which the dataset is labeled based on sentiment, and in the proposed method, dimension reduction was done only based on sentiment. As can be seen in Table 6, in this mode, the proposed method also improves clustering performance.

Table 7 shows the simulation results done on the Twitter-Sentiment-Corpus-3 dataset. These results are related to the mode, in which the dataset is labeled based on sentiment and semantics. Therefore, in the proposed method, dimension reduction was done based on sentiment and semantics. In this mode, the proposed method has also a positive effect on opinion texts clustering.

Given that accuracy is one of the most important measures in evaluating the efficiency of clustering methods, Figure 5 shows a comparison between the accuracy of the proposed method and other methods.

5.3.3. Discussion. According to the results presented in Tables 3–7 and Figure 5, dimension reduction using the proposed method improved clustering performance, except in the Search Snippets dataset. This result can be due to the fact that this dataset contains texts collected from Google search, which are structured and somewhat devoid of abbreviations, irregular expressions, and infrequent words, and that these texts contain the minimum amount of noise. Meanwhile, other datasets contain tweets collected from Twitter, which are unstructured and include noise, and in these datasets, dimension reduction has led to improved clustering efficiency.

The simulation results done on the Twitter Dataset show that the proposed method caused an improvement of about 2.5% in accuracy as well as other measures and an improvement of about 8% in the NMI. The results of this dataset show that the proposed method can have a positive effect on opinion texts clustering. The simulation results done on the Twitter-Sentiment-Corpus-3 dataset demonstrate that the performance of the proposed method is reliable. Table 5 shows the results of clustering based on semantics. According to the results, the proposed method has a positive effect on opinion texts collected from social networks. In this case, the proposed method causes an improvement of about 10% in accuracy and an improvement of about 17% in the NMI. The results presented in Table 6 also confirm the good and acceptable performance of the proposed method in the sentiment-based clustering. In this case, the proposed method caused an improvement of about 21% in accuracy.

As stated previously, the main purpose of the proposed method is opinion texts clustering based on sentiment and semantics, the results of which are given in Table 7. As can be seen in Table 7, in this case, the proposed method also shows its positive performance and causes an improvement of about 9 and 1% in accuracy and NMI, respectively. This good performance of the proposed method can be attributed to some reasons. First, it reduces dimension on the texts, leading to the optimal representation of the texts and

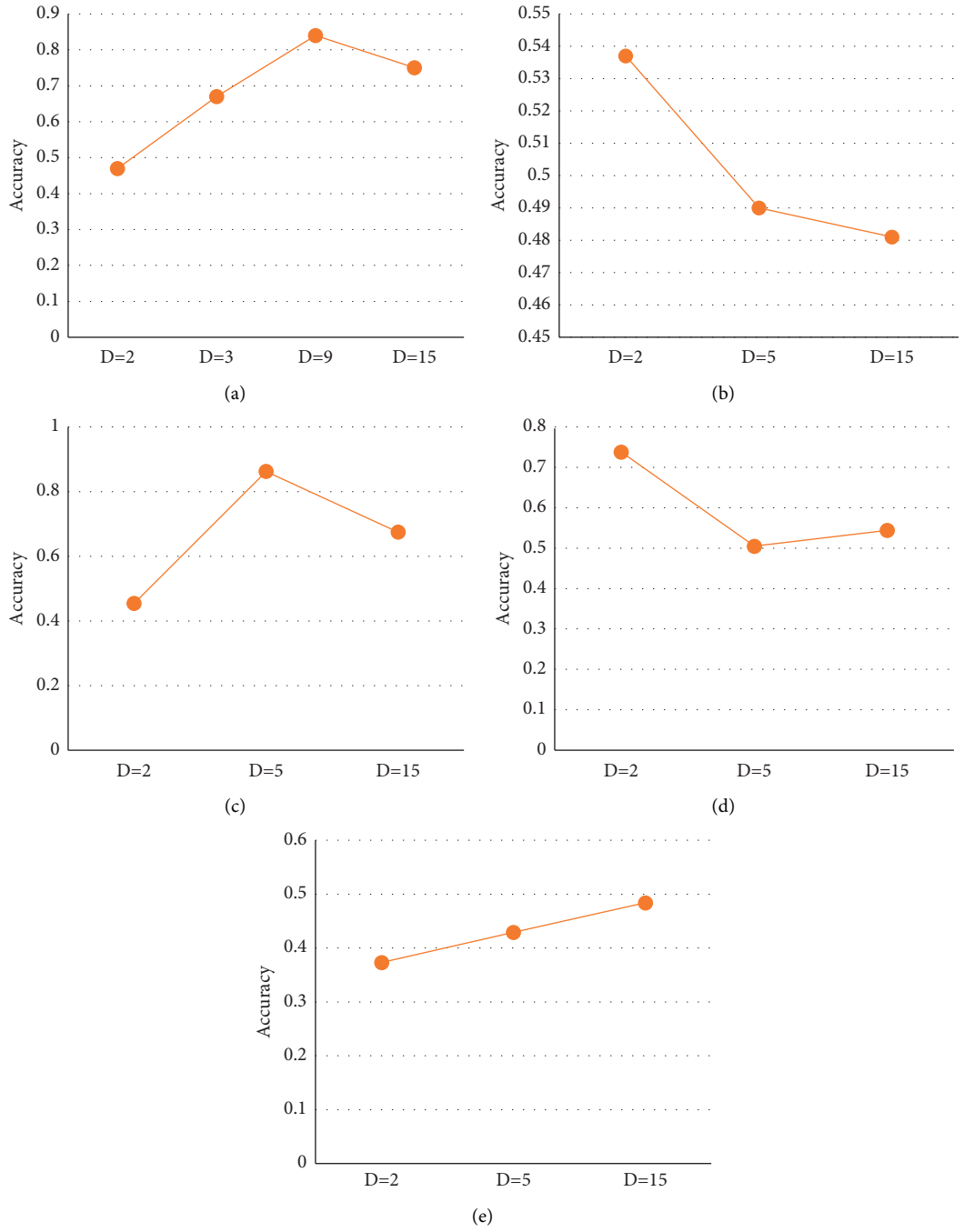


Figure 4: The accuracy results on three datasets based on estimated intrinsic dimension (the accuracy is in range 0, 1. 0 indicates the lowest accuracy and 1 indicates the highest accuracy). (a) First dataset. (b) Second dataset. (c) Third dataset (labels based on semantics). (d) Third dataset (label based on sentiment). (e) Third dataset (labels based on semantics and sentiment).

Table 3: Clustering results on Search Snippets dataset.

Methods	Evaluation measures				
	Accuracy	*F*-score	ARI	NMI	Completeness
K-Means	0.861	0.858	0.711	0.719	0.718
K-Means with PCA	0.829	0.827	0.651	0.666	0.665
K-Means with SVD	0.819	0.816	0.634	0.648	0.647
K-Means with KPCA	0.829	0.827	0.651	0.666	0.665
Proposed method	0.840	0.839	0.670	0.686	0.686

Table 4: Clustering results on Twitter Dataset.

Methods	Evaluation measures				
	Accuracy	*F*-score	ARI	NMI	Completeness
K-Means	0.51	0.505	0.0277	0.0215	0.0218
K-Means with PCA	0.498	0.494	0.0277	0.0214	0.0216
K-Means with SVD	0.506	0.501	0.0267	0.0207	0.0210
K-Means with KPCA	0.513	0.509	0.0286	0.0221	0.0224
Proposed method	0.537	0.536	0.1360	0.1010	0.1010

Table 5: Clustering results on Twitter-Sentiment-Corpus-3 (cluster labels based on semantics).

Methods	Evaluation measures				
	Accuracy	*F*-score	ARI	NMI	Completeness
K-Means	0.760	0.748	0.480	0.469	0.457
K-Means with PCA	0.747	0.738	0.454	0.450	0.438
K-Means with SVD	0.745	0.736	0.449	0.446	0.434
K-Means with KPCA	0.696	0.692	0.368	0.379	0.369
Proposed method	0.862	0.843	0.691	0.636	0.624

Table 6: Clustering results on Twitter-Sentiment-Corpus-3 (cluster labels based on sentiment).

Methods	Evaluation measures				
	Accuracy	*F*-score	ARI	NMI	Completeness
K-Means	0.500	0.498	0.0334	0.0238	0.0239
K-Means with PCA	0.522	0.520	0.0337	0.0240	0.0241
K-Means with SVD	0.511	0.509	0.0353	0.0253	0.0254
K-Means with KPCA	0.510	0.500	0.0323	0.0230	0.0231
Proposed method	0.738	0.736	0.227	0.1700	0.1710

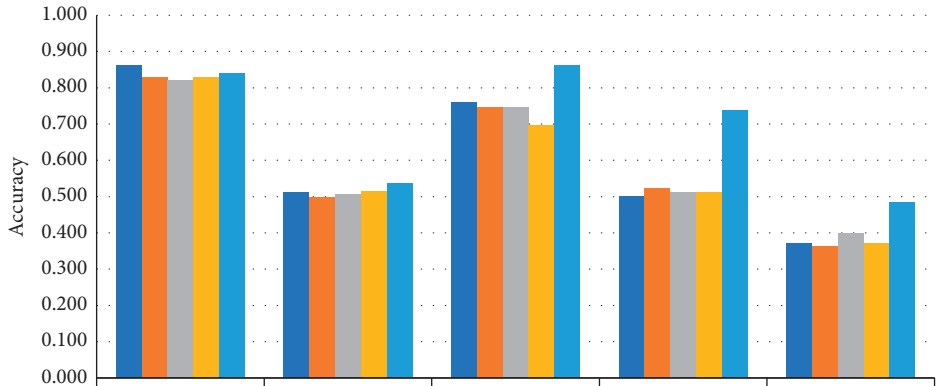
Table 7: Clustering results on Twitter-Sentiment-Corpus-3 (cluster labels based on semantics and sentiment).

Methods	Evaluation measures				
	Accuracy	*F*-score	ARI	NMI	Completeness
K-Means	0.371	0.337	0.339	0.485	0.482
K-Means with PCA	0.363	0.335	0.315	0.473	0.463
K-Means with SVD	0.399	0.350	0.314	0.444	0.439
K-Means with KPCA	0.370	0.336	0.314	0.473	0.464
Proposed method	0.484	0.390	0.438	0.493	0.506

subsequently making a high efficiency in processing on the texts, and also as observed, dimension reduction results in higher quality in clustering. Second, dimension reduction is based on the purpose of clustering, which is also useful in increasing the accuracy of clustering. Here, depending on the used dataset, dimension reduction was based on sentiment, semantics, and a combination of both, which also increased the quality of clustering. The results and discussion may be presented separately, or in one combined section, and may optionally be divided into headed sections.

6. Conclusions

In this paper, a new approach for opinion texts clustering was presented. According to the previous studies, dimension reduction causes an optimal representation of the texts and has a positive effect on the efficiency of machine learning algorithms. Therefore, in this paper, manifold learning was used to reduce dimension and to extract intrinsic dimensions of opinion texts. Using the ISOMAP algorithm, as one of the global manifold learning algorithms, first, dimensions of texts were reduced based on semantics and sentiments, and then, the texts were clustered using the K-Means algorithm. In the dimension reduction phase, three reduction modes were performed. In the first case, dimension reduction was performed based on semantics. In the second mode, it was performed based on the sentiment, and finally, in the third mode, it was done based on both semantics and sentiment. The simulation results done on the three datasets show that the proposed method does not have acceptable performance on the structured or noise-free texts but it improves clustering performance on tweets collected from



	First Dataset	Second Dataset	Third Dataset (Semantic)	Third Dataset (Sentiment)	Third Dataset (Sentiment and Semantic)
K-Means	0.861	0.510	0.760	0.500	0.371
K-Means with PCA	0.829	0.498	0.747	0.522	0.363
K-Means with SVD	0.819	0.506	0.745	0.511	0.399
K-Means with KPCA	0.829	0.513	0.696	0.510	0.370
Proposed Method	0.840	0.537	0.862	0.738	0.484

Figure 5: The accuracy of clustering methods on three datasets.

Twitter that contain noise and are unstructured. The third mode of dimension reduction, in addition to improving the efficiency of clustering, allows us to cluster opinion texts with high efficiency while simultaneously considering semantics and sentiment, which has received very little attention in the previous works. An improvement of about 9% is observed in terms of accuracy, about 4% in terms of *F*-Score, about 10% in terms of ARI, about 1% in terms of NMI, and about 2% in terms of completeness on the third dataset and clustering based on sentiment and semantics. High-precision clustering based on semantics and sentiment helps us to summarize opinion texts with high quality.

In future works, progress can be achieved in terms of both dimension reduction algorithms and methods. In the dimension reduction section, other manifold learning algorithms, such as the LLE, can be used. Also, the initial dimension reduction can be done using linear dimension reduction methods or feature selection methods, and then, the final dimension reduction can be performed using manifold learning algorithms.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] M. R. Bouadjenek, H. Hacid, and M. Bouzeghoub, "Social networks and information retrieval, how are they converging? a survey, a taxonomy and an analysis of social information retrieval approaches and platforms," *Information Systems*, vol. 56, pp. 1–18, 2016.

[2] H. Rehioui and A. Idrissi, "New clustering algorithms for twitter sentiment analysis," *IEEE Systems Journal*, vol. 14, no. 1, pp. 530–537, 2020.

[3] R. Harakawa, S. Takimura, T. Ogawa, M. Haseyama, and M. Iwahashi, "Consensus clustering of tweet networks via semantic and sentiment similarity estimation," *IEEE Access*, vol. 7, pp. 116207–116217, 2019.

[4] S. Jahanbakhsh Gudakahriz, A. M. Eftekhari Moghadam, and F. Mahmoudi, "An experimental study on performance of text representation models for sentiment analysis," *Journal of Information Systems and Telecommunication*, vol. 8, no. 1, pp. 45–52, 2020.

[5] N. Ghali, M. Panda, A. E. Hassanien, A. Abraham, and V. Snasel, "Social networks analysis: tools, measures, and visualization," *Computational Social Networks*, Springer, Berlin, Germany, 2012.

[6] S. S. Khan, M. Khan, Q. Ran, and R. Naseem, "Challenges in opinion mining, a comprehensive review," *A Science and Technology Journal*, vol. 33, no. 11, pp. 123–135, 2018.

[7] L. Rokach and O. Maimon, *Clustering Methods*, Springer, Boston, MA, USA, 2005.

[8] A. Fahad, N. Alshatri, Z. Tari et al., "A survey of clustering algorithms for Big data: taxonomy and empirical analysis," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 3, pp. 267–279, 2014.

[9] P. Verma and A. Verma, "A review on text summarization techniques," *Journal of Scientific Research*, vol. 64, no. 1, pp. 251–257, 2020.

[10] C. Vicient and A. Moreno, "Unsupervised topic discovery in microblogging networks," *Expert Systems with Applications*, vol. 42, no. 17, pp. 6472–6485, 2015.

[11] M. T. AL-Sharuee, F. Liu, and M. Pratama, "Sentiment analysis: dynamic and temporal clustering of product reviews," *Applied Intelligence*, vol. 51, no. 1, pp. 51–70, 2020.

[12] S. Momtazi and D. Klakow, "A word clustering approach for language model-based sentence retrieval in question answering systems," in *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 1911–1914, ACM, Hong Kong, China, November 2009.

[13] Y. Vikash, S. Rati, T. Aprna, and M. Anamika, "A new approach for movie recommender system using K-means clustering and PCA," *Journal of Scientific and Industrial Research*, vol. 80, pp. 159–165, 2021.

[14] S. Kongwudhikunakorn and K. Waiyamai, "Combining distributed word representation and document distance for short text document clustering," *Journal of Information Processing Systems*, vol. 16, no. 2, pp. 277–300, 2020.

[15] A. Aizawa, "An information-theoretic perspective of TF-IDF measures," *Information Processing and Management*, vol. 39, no. 1, pp. 45–65, 2003.

[16] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[17] Y. Zhang, B. Xu, and T. Zhao, "Convolutional multi-head self-attention on memory for aspect sentiment classification," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 4, pp. 1038–1044, 2020.

[18] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of the 1st International Conference on Learning Representations (ICLR 2013)*, vol. 3, Scottsdale, Arizona, USA, May 2013.

[19] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pp. 1188–1196, Beijing, China, June 2014.

[20] C. C. Aggarwal and C. Zhai, "A survey of text clustering algorithms," *Mining Text Data*, Springer, Boston, MA, USA, 2012.

[21] J. Li, G. Huang, C. Fan, Z. Sun, and H. Zhu, "Key word extraction for short text via word2vec, doc2vec, and textrank," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 27, no. 3, pp. 1794–1805, 2019.

[22] C. He, Z. Dong, R. Li, and Y. Zhong, "Dimensionality reduction for text using LLE," in *Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering*, IEEE, Beijing, China, October 2008.

[23] L. M. Abualigah, A. T. Khader, M. A. Al-Betar, and O. A. Alomari, "Text feature selection with a robust weight scheme and dynamic dimension reduction to text document clustering," *Expert Systems with Applications*, vol. 84, pp. 24–36, 2017.

[24] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, Berkeley, CA, July 1967.

[25] R. Kumbhar, S. Mhamane, H. Patil, S. Patil, and S. Kale, "Text document clustering using K-means algorithm with dimension reduction techniques," in *Proceedings of the Fifth International Conference on Communication and Electronics Systems (ICCES)*, pp. 1222–1228, IEEE, Coimbatore, India, June 2020.

[26] D. Wu, R. Yang, and C. Shen, "Sentiment word co-occurrence and knowledge pair feature extraction based LDA short text clustering algorithm," *Journal of Intelligent Information Systems*, vol. 56, no. 1, pp. 1–23, 2021.

[27] S. Dutta, N. Saha, A. K. Das, and S. Ghosh, "Clustering model for microblogging sites using dimension reduction techniques," *International Journal of Information System Modeling and Design*, vol. 10, no. 2, pp. 26–45, 2019.

[28] S. Wu, H. Zhang, C. Xu, and T. Guo, "Text clustering on short message by using deep semantic representation," *Advances in Intelligent Systems and Computing*, vol. 760, pp. 133–145, 2018.

[29] S. Dutta, S. Ghatak, A. K. Das, M. Gupta, and S. Dasgupta, "Feature selection-based clustering on micro-blogging data," *Advances in Intelligent Systems and Computing*, vol. 711, pp. 885–895, 2018.

[30] C. M. Nebu and S. Joseph, "A hybrid dimension reduction technique for document clustering," *Advances in Intelligent Systems and Computing*, vol. 424, pp. 403–416, 2015.

[31] D. Zhao, J. Wang, H. Lin et al., "Sentence representation with manifold learning for biomedical texts," *Knowledge-Based Systems*, vol. 218, Article ID 106869, 2021.

[32] J. Mu and P. Viswanath, "All-but-the-top: simple and effective postprocessing for word representations," in *Proceedings of the 6th International Conference on Learning Representations*, pp. 1–25, Vancouver, BC, Canada, April 2018.

[33] Y. Zhen, Y. Fei, F. Kefeng, and H. Jian, "Text dimensionality reduction with mutual information preserving mapping," *Chinese Journal of Electronics*, vol. 26, no. 5, pp. 919–925, 2017.

[34] R. Salem, "A manifold learning framework for reducing high-dimensional big text data," in *Proceedings of the 12th International Conference on Computer Engineering and Systems (ICCES)*, pp. 347–352, IEEE, Cairo, Egypt, December 2017.

[35] S. Hasan and E. Curry, "Word re-embedding via manifold dimensionality retention," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 321–326, Association for Computational Linguistics, Copenhagen, Denmark, September 2017.

[36] B. Jiang, Z. Li, H. Chen, and A. G. Cohn, "Latent topic text representation learning on statistical manifolds," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5643–5654, 2018.

[37] T. B. Hashimoto, D. Alvarez-Melis, and T. S. Jaakkola, "Word embeddings as metric recovery in semantic spaces," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 273–286, 2016.

[38] P. Gifani, H. Behnam, and Z. Alizade Sani, "Analysis of echocardiography Images using manifold learning," *Iranian Journal of Biomedical Engineering*, vol. 4, pp. 149–160, 2010.

[39] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, pp. 417–441, 1993.

[40] E. K. Tang, P. N. Suganthan, X. Yao, and A. K. Qin, "Linear dimensionality reduction using relevance weighted LDA," *Pattern Recognition*, vol. 38, no. 4, pp. 485–493, 2005.

[41] V. Klema and A. Laub, "The singular value decomposition: its computation and some applications," *IEEE Transactions on Automatic Control*, vol. 25, no. 2, pp. 164–176, 1980.

[42] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pp. 267–273, ACM, Toronto, Canada, July 2003.

[43] D. Lunga, S. Prasad, M. M. Crawford, and O. Ersoy, "Manifold-learning-based feature extraction for classification of hyperspectral data: a review of advances in manifold

learning," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 55–66, 2014.

[44] E. Golchin and K. Maghooli, "Overview of manifold learning and its application in medical data set," *International Journal of Biomedical Engineering and Science*, vol. 1, no. 2, pp. 23–33, 2014.

[45] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[46] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[47] A. Chandra Pandey, D. Singh Rajpoot, and M. Saraswat, "Twitter sentiment analysis using hybrid cuckoo search method," *Information Processing and Management*, vol. 53, no. 4, pp. 764–779, 2017.

[48] C. J. Hutto and E. Gilbert, "VADER: a parsimonious rule-based model for sentiment analysis of social media text," in *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, pp. 216–225, Ann Arbor, MI, USA, July 2014.

[49] C. Wang, W. Pedrycz, Z. Li, and M. Zhou, "Residual-driven fuzzy C-means clustering for image segmentation," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 4, pp. 876–889, 2021.

[50] A. A. Mohamed, "An effective dimension reduction algorithm for clustering Arabic text," *Egyptian Informatics Journal*, vol. 21, no. 1, pp. 1–5, 2020.