IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 26, No. 6, December, 2021, pp. 523-531
Journal homepage: http://iieta.org/journals/isi

Detection of Hot Topic in Tweets Using Modified Density Peak Clustering

Sarvani Anandarao*, Sweetlin Hemalatha Chellasamy

School of Computer Science and Engineering, Vellore Institute of Technology (VIT), Chennai 600127, Tamil Nadu, India

Corresponding Author Email: sarvani.anandarao@gmail.com

https://doi.org/10.18280/isi.260602

Received: 6 August 2021
Accepted: 10 November 2021

Keywords:
NLTK, TF-IDF vector model, density peak clustering, cosine similarity

ABSTRACT

Tweets based micro blogging is the most widely used social media to share the opinions in terms of short messages. Tweets facilitate business men to release the products based on the user interest which thereby produces more profits to their business. It also helps the government to monitor the public opinion which leads to better policies and standards. The large number of tweets on different topics are shared daily so, there is a need to identify trending topics. This paper proposes a method for automatic detection of hot topics discussed predominantly in social media by aggregating tweets of similar topics into manageable clusters. This produces hot topic detection irrespective of the current user location. A Modified Density Peak Clustering (MDPC) algorithm based hot topic detection is proposed. Local density of traditional Density Peak Clustering (DPC) is redefined by using the gaussian function in the calculation of d_c (threshold distance). The traditional DPC considering some random value as dc (threshold distance) this gives a negative impact on the cluster formation thereby return inappropriate clusters. This can be solved by using the MDPC. The MDPC algorithm works by taking the cosine similarity between the tweets as the input and produces clusters of similar tweets. The cluster having a greater number of tweets is considered as hot topic which is frequently discussed by most of the users on twitter. Events 2012 dataset is collected with streaming API. This contains tweets from 2012 to 2016. The dataset consists of 149 target events and 30 million tweets. Experimental result shows that the proposed algorithm performs better than the traditional algorithms such as density peak clustering, K-means clustering, and Spectral clustering. It has produced the accuracy of 97%.

1. INTRODUCTION

With the emergence of big data era, data has increased exponentially in all the areas. Twitter has attracted huge number of people to communicate and share the knowledge with one another. It has many special features such as posting short texts, images, videos and URLs. Twitter makes easy for people to post text, emotions, their daily activities etc. and thereby reflecting the variety of information. The twitter data consists of misspelling text, incomplete text along with the images, videos, text. Picking the trending topic from this high volume of data is the challenging task in producing the accurate output.

By detecting the trending topic, the user can understand the current tendency of the society, most discussed topic among the public, hot product recommendation and incidence detection. This can also help the enterprises to develop their respective businesses by analysis the people interest in their products. Government organizations can also implement the policy and standards. Based on the discussion between the users in the form of tweets, crime branch can gather the information about the crime scene and also based on the user location, crime branch can get the location information about the incident.

In earlier days, experimental study, questionnaires and interviews were conducted to get the trending topic. Traditionally single pass-based topic detection, Latent Dirichlet (LDA) model and graph-based method in topic detection. The conventional methods can bring accurate output only on lengthy text but not on twitter data since it occurs unexpectedly. The twitter data is very short and dynamic data. It is a combination of images, videos, URLs, so these methods cannot be applied. Twitter is a good platform for discussing the current topic, breaking news and some useful information. This leads to increase in huge and variety of data. For this reason, many researchers are interested to do their study on the twitter data.

Previously, in order to detect hot topic in online communities, the document clustering was used in which promising results were not found and also very less research has been done. Most Twitter users see trends tailored to them, via their location and who they follow. You cannot see what's trending in another city until you change your personal settings to match the desired city or monitor what's trending nationwide. This paper aims to identify the hot topic detection from the tweets available in twitter irrespective of the current location of the user.

The traditional Density Peak Clustering (DPC) is unable to bring out the accurate output when there are different clusters with different densities which works by considering some random value as dc (threshold distance). The dc is the important parameter in identifying the local density of the datapoint. The random dc value may not return accurate clusters. To avoid this, the local density of traditional DPC is modified by introducing the gaussian function for our application to calculate dc (threshold distance) shown in the

Eq. (1).

$$d_c' = \sum_{j \in X, j \neq i} \exp\left(-\frac{{d_{ij}}^2}{{d_c}^2}\right) \quad (1)$$

Calculation of d_c and d_{ij} used in our paper is shown in the Eq. (2).

$$d_c = Avg(d_{max}, d_{min}) \quad (2)$$

$$d_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - x_i)^2}$$

A Modified Density Peak Clustering (MDPC) algorithm based hot topic detection is proposed. The MDPC algorithm works by taking the cosine similarity between the tweets as the input based on which clusters are produced. The cluster which is having a greater number of tweets is considered as frequently discussed topic by most of the users on twitter.

The rest of the paper is organized as follows: Section 2 discusses the related work in the context of hot topic detection. Section 3 highlights the research contributions. Section 4 gives an overview of the clustering and its types. Section 5 shows the Modified Density Peak Clustering (MDPC) algorithm based hot topic detection. Section 6 presents the proposed system. Section 7 presents the experimental results and section 8 concludes the paper.

2. RELATED WORKS

Pohl et al. [1] discussed the social media clustering, indexing and the challenging task of identifying the events of real time data at the time of emergency. This paper shows the model which helps to classify the events based on the crisis related data. Learn and forget strategy, term frequency-inverse document frequency (TF-IDF) and incremental skewness were used. The overall model delivers a better opportunity to support emergency responders to construct a real-world emergency exercise.

Ai et al. [2] showed the usage of multidimensional sentence modelling and analysis of timeline in the detection of hot topic. To enhance the performance of the traditional system a methodology was developed using MapReduce operations which can handle huge data. Two important aspects considered in this methodology are a) micro-clustering using text selection, b) macro-clustering using topic selection.

Chen et al. [3] used automatic indexing and retrieval to get the relevant documents for a given user query. In this method, high order structure is formed by using the terms in the documents. Deerwester et al. [4] proposed a new approach for document indexing by using the factor analysis of count data.

Hofmann [5] showed the three-level structure for indexing and collection of various forms of data. This structure is formed based on the Bayesian network and it was named as Latent Dirichlet Allocation (LDA). Blei et al. [6] built a new probabilistic model to detect the topic by assigning the weights to features selected and using traditional vector model.

He et al. [7] showed the construction of the new model for topic detection. This model could update the new topic continuously by identifying the new patterns. Al Sumait et al. [8] design a model using word co-occurrence.

Newman et al. [9] proposed an enhanced single pass clustering method which uses Dirichlet allocation instead of conventional vector model. This picks the hot topic from blog. Huang et al. [10] designed a framework based on temporal features for clustering the hot topic. Chen et al. [11], also used LDA model to get the hot topic from the blog.

Ge et al. [12] used a pair of two terms and co-occurrence pattern to detect the hot topic from a small text. Lu et al. [13] devised an efficient algorithm build using Bayesian classifier to detect the hot topic over huge number of tweets online. Documents related to medical is taken as the dataset to measure the performance of proposed algorithm.

Fang et al. [14] proposed a novel methodology called multi-view clustering to pick the most discussed topic in tweets. The advantages of the proposed methodology are that it can show the relationship between the tweets and also can find the core keywords.

Huang et al. [15] proposed a new approach to detect the most discussed topic based on Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) on Vector Space Model (VSM). Here, TF-IDF, cosine similarity has been used to cluster the documents.

Yang et al. [16] used hierarchical agglomerative clustering to detect the events. But this technique is very expensive in terms of complexity. In order to reduce the complexity K-Means is introduced in topic detection [17], but with prior definition of number of clusters.

Yu et al. [18] discovered the most discussed topic from continuous display of news occurring with very less time gap. Time related data is taken as a dataset to check the performance of proposed algorithm., Cosine similarity was used to calculate similarity between the news. Tu et al. [19], which used the advantage of density peak clustering to detect the anomalies in the spectral images. The gaussian distance measure is used to compute the local density of test data points.

Xu et al. [20] designed an enhanced adaptive density peaks clustering (DPC) to detect the overlapping communities. This uses the linear fitting to choose the cluster centroids of DPC. The gaussian distance measure is used to compute the local density of test data points. A few other works [19-21] also uses gaussian distance measure to compute local density and density peak clustering is applied for detecting the hot topic from the collected tweets.

Twitter trend analysis [22] is done using Term Frequency-Inverse Document Frequency (TF-IDF), Combined Component Approach (CCA) and Biterm Topic Model (BTM). Density peak clustering along with nearest neighbour technique to identify the noise node is shown in the reference [23]. This doesn't rely on cut-off distance to identify the noise node. The cut-off distance computing using the Gini index in density peak clustering is shown in reference [24] and this is able to bring out better accuracy than the traditional density peak clustering. The work [25] used the online clustering for detection of trends in tweeter. The clusters are ranked based on the size and recency. The ranked clusters are persisted to Manhattan periodically for serving purposes. The Table 1 shows the recent research contribution in the area of hot topic detection.

In this hot topic detection and recommendation research work, a Modified Density Peak Clustering (MDPC) algorithm based hot topic detection is proposed. In recent years, many researchers have introduced various innovative techniques to detect hot topics in Twitter such as machine learning, data mining and swarm intelligence. The main advantage of using MDPC is that it extracts similar features of complex network data and divide the network dataset into several subsets with different clusters. Instead of using deep learning-based

algorithm, these subset in MDPC algorithm minimizes the imbalance of multi-class network data and improve the detection rate of the minority classes. When compared to clustering algorithms, deep learning is considered as a very challenging issue to make recommendations because of its big scale, dynamic corpus and variety of unobservable external factors. In addition to that, Gaussian kernel is utilized by the MDPC algorithm instead of using cut-off journal in which the former one effectively measures the distance by implicitly mapping the raw data into a high-dimensional feature space. The local density of the data point is calculated by this Gaussian kernel function.

Table 1. Existing work

Year	Technique	Dataset	Result discussion	Merit	Demerit
2021	Term Frequency-Inverse Document Frequency (TF-IDF), Combined Component Approach (CCA) and Biterm Topic Model (BTM)	Tweets were collected for 15 days from November 13 to November 28, 2018	-	Detect the future trends with small datasets	Finding the topics and terms within given topics
2020	Density peak clustering, spatial density background purification	Three real hyperspectral datasets such as Airport, Beach, and Urban scenes	97.62, 98.99, 99.54	Detection accuracy of this method outperforms other commonly used methods.	Works only on the image data.
2019	Extended adaptive density peaks clustering (EADP)	Twelve real-world datasets, including weighted and un-weighted datasets, labeled and unlabeled datasets	74.4857% on one of the datasets.	EADP could be used directly in social networks.	EADP incorporates only distance function
2018	A non-iterative algorithm called parallel two phase mic-mac hot topic detection (TMHTD). Micro-clustering and Macro-clustering	Sina Weibo which is the top microblog service in China	F-measure value of the TMHTD algorithm shows a 6% and 8% improvement over the general single-pass algorithm and the LDA algorithm, respectively	Less comparative study is made in the experimental section	Obtains valuable hot topics from the vast amount of digitized textual data
2017	multiview clustering	The topics and their corresponding tweets, which were distributed during 1–9 January 2012, were collected	F-measure-92.3, 92.6,	Twitter topic detection- integrate multirelations among tweets	No Significant change from the traditional method and no improvement in keyword extraction

3. RESEARCH CONTRIBUTION

- Hot topic of detection from the tweets irrespective of user current location using the refined traditional density peak clustering by Gaussian function.
- Refined the random value of d_c (threshold distance) using Gaussian function.
- A Modified Density Peak Clustering (MDPC) algorithm based hot topic detection is proposed.
- Compared our work with the DPC, K-means and spectral clustering 6 parameters.

4. CLUSTERING

4.1 K-means clustering

K-means [26-28] comes under unsupervised data clustering algorithm. In this initially k-centroids are defined to form k-clusters. K can be any numerical number. Datapoints are assigned to the nearest cluster centroids by using the distance measure.

Steps in K-means cluster:

Step 1: Random 'k' number of centroids are selected from the given data set.

Step 2: Assign every datapoint to the nearest cluster centroids.

Step 3: New centroids are calculated after completion of step 2.

Step 4: Step 2 and Step 3 gets iterated until a fixed centroids are obtained.

Step 5: Again, every datapoints are assigned to the nearest new cluster centroid and returns the final clusters.

4.2 Spectral clustering

This clustering technique [29] is independent of shape of the cluster and cluster centroid whereas K-means algorithm can produce accurate output only when shape of the cluster is spherical and round. The best centroid can be obtained only after many iterations in K-means. The connected datapoints belong one cluster even though they are far apart. The datapoints which are not connected belong to different cluster. Steps of algorithm are shown below.

Steps:

- Compute weighted adjacency matrix for the corresponding graph and name as $'W'$.
- Compute the Laplacian $'L'$.
- Calculate the eigenvectors for first $e_1, e_2, e_3, \ldots \ldots e_k$ in $'L'$.
- Construct a matrix 'U' of vectors $e_1, e_2, e_3, \ldots \ldots e_k$ as columns.
- let $y_i \in R$ be the vector of ith row of U, where i=1,2,3…..n.

- Apply the K-means to assign the points $y_i \in R$ to one of the cluster.

Output: Cluster $A_i, \ldots \ldots A_k$ with $A_i = \{j | y_j \in C_i\}$.

4.3 Density peak clustering algorithm

Density peak clustering algorithm [30, 31] is used to group the words (represented in vector format using word embedding) taken from all the tweets. The word with highest term frequency is taken as core points for the clustering.

The underlying structure of unlabelled data space are analysed by the clustering procedure which is a significant unsupervised method. Based on the attribute similarity, the clustering procedure organizes the data into groups, such that the data from different clusters differ from each other and the data within the same cluster have similar properties.

The vector format of words is given as the input to the clustering process. The density peak clustering works based on the denser clusters. Initially, the datapoints with higher density are identified. The higher density datapoints are the datapoints which are surrounded by maximum number of neighbouring datapoints. The datapoint with maximum density and relatively large distance are considered as cluster centroids. The datapoints are assigned to their nearest cluster by measuring the distance between cluster centroid and itself. The local density ρ_i is defined for every single data point x_i:

$$\rho_i = \sum_j \chi(d_{ij} - d_c)$$
$$\chi(x) = \begin{cases} 1, x < 0 \\ 0, others \end{cases} \quad (3)$$

In Eq. (3) d_c is the cut-off distance/user specified parameter. The distance between any two datapoints x_i and x_j are represented as d_{ij}. If the distance between any two data points x_i and x_j are less than d_c, then those two datapoints are grouped into one cluster. The number of such data points x_i is called density of data point x_i represented as ρ_i.

The $\delta_i$value is either taken as the minimum distance from x_i to any other data point if there exists data point with a local density > ρ (x_i) or the maximum distance from x_i to any other data point if there exists no data point with a local density > ρ (x_i). The calculation of δ_i is shown the below Eq. (4).

$$\delta_i = \begin{cases} \min_{j:\rho_j > \rho_i} (d_{ij}), x_i \text{ is not the higher density point} \\ \max_j (d_{ij}), x_i \text{ is the higher density point} \end{cases} \quad (4)$$

The steps of this algorithm are as follows:

Step 1: Compute the distance matrix for every datapoint.
Step 2: Define the cut-off distance d_c.
Step 3: Calculate the local density ρ for every data point with d_c parameter.

$$\rho_i = |A(i)| // \text{ Local density} \quad (5)$$

where, A(i) denotes number of datapoints whose distance to point 'i' is less than the d_c parameter. That is:

$$A(i) = \{ j \in X | d(i,j) < d_c \} \quad (6)$$

Step 4: Calculate the maximum distance between the datapoints.

$$\delta_i = \max_j d_{ij} \quad (7)$$

Step 5: For each datapoint x_i, calcule the γ_i for each datapoint x_i.

$$\gamma_i = \rho_i \, \delta_i \quad (8)$$

Step 6: Arrange all the γ in the decreasing order and select the cluster centers.
Step 7: Assign rest of the datapoints to their nearest cluster centroid.

According to DPC, centroid node has two major characteristics: one is relative high density, and the other is relatively large distance from other higher density node.

The traditional Density Peak Clustering (DPC) calculates the local density by using the random dc (threshold distance) value. The random dc value shows the negative impact on the cluster formation. To avoid this, the local density of traditional DPC is modified by introducing the gaussian function for our application. This returns the hot topic irrespective of the user current location.

5. MDPC ALGORITHM

Below is the algorithm for the modified density peak clustering. This MDPC shows the difference in the calculation of local density by introducing the gaussian function unlike considered a random value for local density in density peak clustering. The difference is showed in the Eqns. (8), (9).

Input: The set of tweets as datapoints and number of clusters.

Output: The set of clusters in decreasing order, the top cluster is the maximum size cluster with a greater number of tweets.

Step 1: Consider every datapoint x_i and calculate the local density with modified d_c'parameter in Eq. (8).

$$d_c' = \sum_{j \in X, j \neq i} \exp\left(-\frac{{d_{ij}}^2}{{d_c}^2}\right) \quad (9)$$

where, d_{ij} is the distance between any two points.

$$\rho_i' = |A(i)| // \text{ Local density} \quad (10)$$

where, $A(i)$ denotes number of datapoints whose distance to point 'i' is less than the d_c' parameter. That is:

$$A(i) = \{ i \in X | d(i,j) < d_c' \} \quad (11)$$

Step 2: Calculate the maximum distance between the datapoints.

$$\delta_i = \max_j d_{ij} \quad (12)$$

Step 3: For each datapoint x_i, calculate the γ_i for each datapoint x_i.

$$\gamma_i = \rho_i' \delta_i \quad (13)$$

Step 4: Arrange all the γ in the decreasing order and select the top 10 γ as the cluster centers, which indicate 10 clusters are formed.

Eq. (13) is used to calculate the numerical value of γ. Based on the numerical value obtained, γ are arranged in decreasing order. Arranging in decreasing order is only possible with the numerical values. So, the step3, plays an important role in arranging the γ values.

Step 5: After cluster centroid are selected, based on their distances, assign rest of the datapoints to their nearest cluster centroid.

Step 6: Return all the formed clusters along with their datapoint(tweets) and label.

6. PROPOSED METHOD

Main focus of our work is to detect the hot topic from the tweets irrespective of user current location. Here we have used the Modified density peak clustering to cluster the tweets and to get the largest cluster. The largest cluster is considered as the topic discussed by the most of twitter participants. The block diagram of the proposed work is shown in the Figure 1.

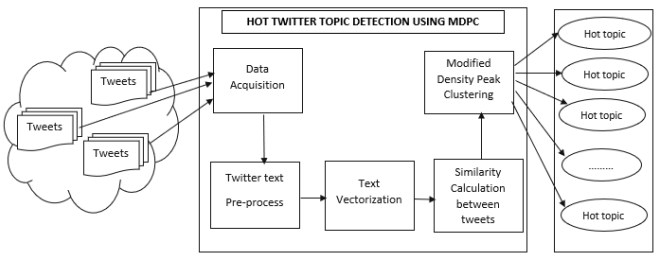

Figure 1. The block diagram of the proposed work

6.1 Collection of tweets

The EVENT 2012 dataset [32] is used in the experiment. A total of 30 million tweets are available, from these huge lists of tweets 159952 are labelled tweets according to the events. Twitter streaming API is used to collect the tweets. From the Wikipedia Current Event Portal and Amazon Mechanical Turk, 506 event types are collected. The collected datasets consist of the tweets discussed about Chemistry Nobel Prize and US presidential election results.

6.2 Text pre-processing

Based on the collected tweets, pre-processing is done. Pre-processing is defined by transforming the raw data into an understandable format. The text pre-processing eliminates the links, prepositions, abbreviations.

The following things are done in the pre-processing step: (i) punctuation removal, (ii) removal of stop-words such as, prepositions, articles, etc., (iii) removal of white spaces, unnecessary tabs and URLs, etc. and (iv) alteration of upper-case letters to lower case. The pre-processing is achieved by using the following procedures. In this pre-processing stage we don't remove the hashtags as these are most valuable parts of Tweets to assign a topic.

6.2.1 Elimination of stop words

The English words which do not give much meaning to a sentence is termed as stop words. These meaningless words are removed without changing the meaning of the sentence. The, he, have etc. are the examples for this word. Normalization is described by removing the repeated words. Some letters in the words are repeated many times in social media text. This type of words cannot be found in the dictionary words and highly complex to deal with such words. Here, if any character repeats continuously for more than two times, then that character can be replaced with one single character.

6.2.2 Stemming

Stemming means transforming the word into their root form. The root is a stem of a word which created the word. For example, the stem for the words "playing", "played", and "player" is "Play".

6.3 Vector format using TF-IDF

Word embedding tool is used to convert the words into vectors of real numbers. The similar meaning words incline to be close in the vector space. Using the word embedding, words can be represented using low dimensional vector instead of high dimensional vector whereas encoding method represents the word in high dimensional vector. The computational power and generalization of learning models are enhanced by neural network toolkits and dense representations.

After the successful completion of the pre-processing, tweets are converted into vector format, where TF-IDF vector representation model is used.

TF-IDF is a way to judge the topic of an article. This is done by the kind of words it contains. Here words are given weight so it measures relevance, not frequency. Wordcounts are replaced with TF-IDF scores throughout dataset.

Word2vec produces one vector per word, whereas TF-IDF produces a score. Word2vec is great for going deeper into the documents we have and helps in identifying content and subsets of content. Its vectors represent each word's context. (i.e., the n-gram of which it is a part). But, the biggest problem with word2vec is the inability to handle unknown or out-of-vocabulary (OOV) words. When compared to other methods, TF-IDF returns documents that are highly relevant to a particular query. Moreover, it is very easy to calculate the similarity between two documents.

6.3.1 TF-IDF vector representation

TF-IDF is used to represent the word into numerical format. The features are extracted for various NLP applications by using this technique. To apply statistical technique, the text must be transformed into numerical format. The numeric representation depicts the substantial characteristics of the text.

6.3.2 Term frequency

Term frequency represents the occurrence of word in each document. If the same word occurs a greater number of times in the document, then that word is considered as the most important word in that document.

The term frequency of a word is defined as follows:

$$tf(w) = \frac{Tweet.count(w)}{Total\ words\ in\ all\ tweets} \quad (14)$$

6.3.3 Inverse document frequency

Some word occurs very commonly in all the document and its contribution is very less in giving meaning. For example, like, make etc. are most common words that occur in every

document. The occurrence of these words is extremely high when compared with the key words in the document. To reduce this effect, these words must be removed. Inverse document frequency is used to remove most frequent words.

$$idf(w) = \log \frac{Total\ Number\ of\ tweets}{Number\ of\ tweets\ Containing\ word\ w} \quad (15)$$

The words like 'what', 'and' these words are common in every tweet/document. The common words in every tweet/document are not considered. These words can be picked out using the "Inverse document frequency (IDF)" shown in Eq. (15). IDF calculates the term with respect to a corpus of documents. The term which appears in every document/tweet doesn't provide any important information to identify the topic.

$IDF("and") = \log \frac{(100))}{(100)} \Rightarrow \log 1 \Rightarrow 0$ [If the term "and" appears in every document].

$IDF("and") = \log \frac{(100))}{(98)} \Rightarrow \log 1.02 \sim 0$ [If the term "and" appears in most of the document].

So, the terms whose value of IDF is 0 or nearly 0 is avoided and these terms are not considered.

6.3.4 Term frequency-inverse document frequency

If one term occurs regularly in a particular document, then the vector representation assigns high values for a given term. The IDF computed value will be zero, when the given term occurs in every single document. The Eq. (16) shows the calculation of TF-IDF.

$$tf - idf(w) = idf(w) * tf(w) \quad (16)$$

The word with highest TF-IDF value is the most important word in the document. Finally, the words in the tweets are represented as vector format and for each word in a tweet the TF–IDF value is calculated. Then by adding all the TF-IDF values of all the words in the tweet, we get value for each and every tweet. At the end of this step, we can get the value for all the tweets. These values are given as an input to the cosine similarity.

6.4 Similarity measure

In this paper, the cosine similarity measure is used to calculate the similarity between all the tweets. Here, cosine similarity matrix is formed by calculating the similarity among all the tweets.

Cosine similarity finds the dot product between two vectors. Cosine similarity produces the best accuracy in calculating the similarity between the texts and it is also low dimensional.

Cosine similarity is advantages because it returns the smaller angle, even if Euclidean distance between two similar documents is high. Smaller angle indicates high similarity.

For example, the two tweets are:

1. Julie loves me more than Linda loves me
2. Jane likes me more than Julie loves me

Count of words in both the tweets

me 2 2
Jane 0 1
Julie 1 1
Linda 1 0
likes 0 1
loves 2 1
more 1 1
than 1 1

The two vectors are, again:

a: [2, 0, 1, 1, 0, 2, 1, 1]
b: [2, 1, 1, 0, 1, 1, 1, 1]
Cos(a, b) = a . b / ||a|| * ||b||
a.b = 2*2+0*1+1*1+1*0+0*1+2*1+1*1+1*1 = 9
||a|| = sqrt[(2)^2+(0)^2+(1)^2+(1)^2+(0)^2+(2)^2+(1)^2+(1)^2]= 3.16
||b|| = sqrt[(2)^2+(1)^2+(1)^2+(0)^2+(1)^2+(1)^2+(1)^2+(1)^2] = 3.16
Cos(a, b) = 9 / (3.16 * 3.16) = 0.9.

6.5 Clustering

Density Peak Clustering (DPC) is one of the fastest clustering algorithms and requires less user input. Hence, we choose Density Peak Clustering algorithm in our work. Density peak clustering algorithm is the best at finding and can also work on arbitrary shape sample data sets.

Modified density peak clustering (MDPC) is used to detect the hot topic of discussion. This takes cosine similarity matrix as input and produces number of clusters. The similar tweets are grouped into a single cluster. The clusters having a greater number of tweets are considered as the interested topic discussed by the most of people. This interested topic is the hot topic of discussion.

The traditional DPC cannot bring the accurate output when we have different clusters with different densities and also random value of d_c (threshold distance) shows a negative impact on the cluster formation. So, instead of taking the random value of d_c, we have redefined the local density of traditional DPC by introducing the gaussian function in the calculation of d_c (threshold distance). Here the calculation of new d_c' parameter is shown in the below Eq. (17). The d_c calculation is shown in the Eq. (18).

$$d_c' = (-(d_{ij}/d_c) ** 2) \quad (17)$$

where, d_{ij} is the distance between any two datapoints.

$$d_c = \frac{Max-distance + Min-distance}{2} \quad (18)$$

Distances between all the datapoints are calculated. Among all the distance, the maximum distance is represented as $Max - distance$ and minimum distance is represented as $Min - distance$.

7. EXPERIMENTAL ANALYSIS

Python platform is used to conduct the experiment to detect the hot topic and also to get the related tweets to the user given query. Natural Language Tool Kit (NLTK) [33] tool is used to do the pre-processing. The performance of this approach is measured using accuracy, RMSE, execution time, precision, recall and F-measure. Finally, the obtained results are compared with existing clustering techniques such DPC, K-means clustering, Spectral clustering. This comparison indicates that the proposed method provides better result than other existing methods.

7.1 Result

Table 2. Cluster information

Cluster Number	Total number of Tweets in cluster	Corresponding label
1	817	Armed conflicts and attacks
2	574	Law and Crime
3	401	sport news
4	391	Politics and elections
5	241	Sport
6	182	Elections
7	179	Disasters and accidents
8	177	International relations
9	171	Arts and culture
10	105	Business and economy

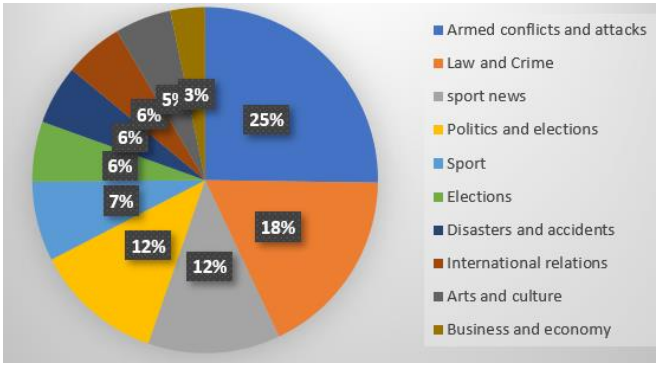

Figure 2. Cluster details

The proposed system returns top 10 clusters. The first cluster has a total of 817 tweets and its corresponding label as "Armed conflicts and attacks". The second cluster has a total of 574 tweets and its corresponding label as "Law and Crime". According to the dataset, hot topic of discussion is "**armed conflicts and attacks**" i.e., most of twitter participants are discussing about "**armed conflicts and attacks**". The cluster information along with total number of tweets in cluster and its corresponding label are shown in the Table 2 and graphical representation is shown in Figure 2 along with the percentage of contribution. The labels shown in the Table 2, dependents on the tweets chosen as dataset.

7.2 Evaluation measures

To check the performance of the proposed system we used the parameters such as Precision [24, 34], Recall [24], F1 measure [14, 35], RMSE [35, 36] and Accuracy [34-36].

7.2.1 Precision

Precision (P) measure is the ratio of all correctly predicted hot topic to the total number topic. Also, it is a fraction between the correctly predicted as hot topic to the sum of correctly predicted as hot topic and the incorrectly predicted as hot topic. It is also known as positive predictive value (PPV). The precision values for all the clusters are shown in the Table 3.

$$Pr\,e\,cision = \frac{TP}{FP+TP}$$

7.2.2 Recall

It is a fraction between the number of correctly predicted hot topic to the number of topics that would have been recommended. It is also known as true positive rate (TPR).

$$Recall = \frac{TP}{FN+TP}$$

7.2.3 F-measure

The F1 score or balanced F-score/F-measure is the harmonic mean of recall and precision.

$$F1_{Score} = \frac{2*Precision*Recall}{Precision+Recall}$$

7.2.4 Accuracy

Accuracy is the quantity of correctness of the recommendation.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Table 3. Precision value for all the clusters

Cluster Number	Total number of tweets correctly predicted to its label	Total number of tweets wrongly	Total number of tweets in cluster	Precision value (%)
1	788	29	817	96.45043
2	553	21	574	96.34146
3	387	14	401	96.50873
4	378	13	391	96.67519
5	233	8	241	96.6805
6	177	5	182	97.25275
7	172	7	179	96.08939
8	169	8	177	95.48023
9	161	10	171	94.15205
10	100	5	105	95.2381

Table 4. Comparison among MDPC, DPC, K-means, Spectral

Algorithm	Precision	Recall	F-measure	Accuracy	RMSE	Execution Time
MDPC	**96.0803**	**98.0018**	**94.2319**	**97.0016**	**2.0000**	**35.7282**
DPC	88.23	90.0	80.35	89.0	6.0	68.91
K-means	87.5	84.0	75.0	86.0	6.0	72.16
Spectral	77.77	84.0	67.74	80.0	12	73.12

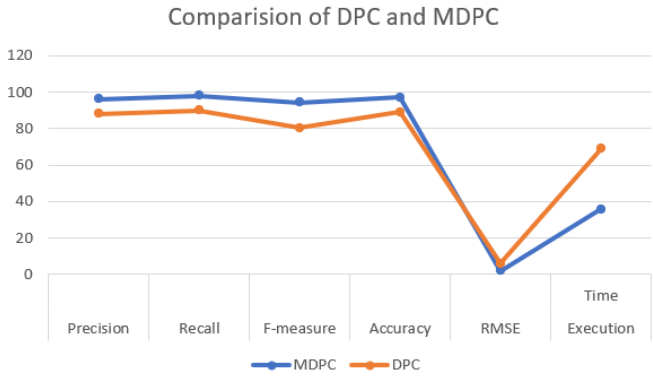

Figure 3. Comparison of DPC and MDPC

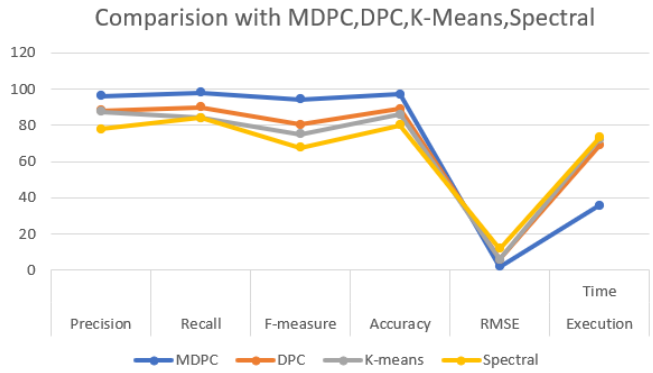

Figure 4. Comparison between MDPC, DPC, K-means, Spectral

The above Table 4 shows the accuracy, recall, precision, f-measure, RMSE, execution time of the proposed system along with other existing algorithms. The Figure 3 shows the comparison of DPC & MDPC, Figure 4 shows the comparison between MDPC, DPC, K-means, Spectral.

From the Table 4 it's clear that DPC is producing accuracy of 89.0% while the MDPC has produced 97.0016%. This shows that MDPC produced better accuracy than DPC while using the same dataset. The MDPC has shown the better performance than another algorithm.

The accuracy, recall, precision, f-measure, RMSE, execution time for DPC, K-means, Spectral are also calculated by using the same dataset. For k-means clustering we have considered k value as 10 and the traditional spectral clustering is used.

8. CONCLUSION

This paper proposes a method for detecting the hot topics in the tweets using the MDPC technique. In this paper, the pre-processing is done by converting uppercase to lowercase letters, removing punctuations, stop words and also stemming the tweets. The NLTK tool is utilized to achieve the pre-processing procedure. The word embedding is accomplished by using TF-IDF vector model, which represents the meaningful words into a vector of real numbers. Then cosine similarity matrix is constructed which is taken as input to the MDPC, which gives number of clusters along with their tweets and also it returns the related user tweets for the given user query(tweet) along with their label. The cluster with a greater number of tweets are considered as hot topic of discussion.

The proposed algorithm has shown the best performance in terms of accuracy, precision, F-measure, recall, RMSE and execution time when compared with the other algorithms such as DPC, K means, and Spectral clustering.

REFERENCES

[1] Pohl, D., Bouchachia, A., Hellwagner, H. (2016). Online indexing and clustering of social media data for emergency management. Neurocomputing, 172: 168-179. https://doi.org/10.1016/j.neucom.2015.01.084

[2] Ai, W., Li, K., Li, K. (2018). An effective hot topic detection method for microblog on spark. Applied Soft Computing, 70: 1010-1023. https://doi.org/10.1016/j.asoc.2017.08.053

[3] Chen, K.Y., Luesukprasert, L., Seng-cho, T.C. (2007). Hot topic extraction based on timeline analysis and multidimensional sentence modeling. IEEE Transactions on Knowledge And Data Engineering, 19(8): 1016-1025. https://doi.org/10.1109/TKDE.2007.1040

[4] Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R. (1990). Indexing by latent semantic analysis. Journal of the American Society for Information Science, 41(6): 391-407. https://doi.org/10.1002/(SICI)10974571(199009)41:6<391::AID-ASI1>3.0.CO;2-9

[5] Hofmann, T. (1999). Probabilistic latent semantic indexing. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50-57. https://doi.org/10.1145/312624.312649

[6] Blei, D.M., Ng, A.Y., Jordan, M.I. (2003). Latent Dirichlet allocation. The Journal of machine Learning research, 3: 993-1022.

[7] He, Q., Chang, K., Lim, E.P., Banerjee, A. (2010). Keep it simple with time: A reexamination of probabilistic topic detection models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(10): 1795-1808. https://doi.org/10.1109/TPAMI.2009.203

[8] AlSumait, L., Barbará, D., Domeniconi, C. (2008). On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In 2008 Eighth IEEE International Conference on Data Mining, pp. 3-12. https://doi.org/10.1109/ICDM.2008.140

[9] Newman, D., Bonilla, E.V., Buntine, W. (2011). Improving topic coherence with regularized topic models. Advances in Neural Information Processing Systems, 24: 496-504.

[10] Huang, B., Yang, Y., Mahmood, A., Wang, H. (2012). Microblog topic detection based on LDA model and single-pass clustering. In International Conference on Rough Sets and Current Trends in Computing, 7413: 166-171. https://doi.org/10.1007/978-3-642-32115-3_19

[11] Chen, Y., Amiri, H., Li, Z., Chua, T.S. (2013). Emerging topic detection for organizations from microblogs. In Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 43-52. https://doi.org/10.1145/2484028.2484057

[12] Ge, G., Chen, L., Du, J. (2013). The research on topic detection of microblog based on TC-LDA. In 2013 15th IEEE International Conference on Communication Technology, pp. 722-727. https://doi.org/10.1109/ICCT.2013.6820469

[13] Lu, Y., Zhang, P., Liu, J., Li, J., Deng, S. (2013). Health-related hot topic detection in online communities using text clustering. Plos One, 8(2): e56221. https://doi.org/10.1371/journal.pone.0056221
[14] Fang, Y., Zhang, H., Ye, Y., Li, X. (2014). Detecting hot topics from Twitter: A multiview approach. Journal of Information Science, 40(5): 578-593. https://doi.org/10.1177/0165551514541614
[15] Huang, S., Peng, X., Niu, Z., Wang, K. (2011). News topic detection based on hierarchical clustering and named entity. In 2011 7th International Conference on Natural Language Processing and Knowledge Engineering, pp. 280-284. https://doi.org/10.1109/NLPKE.2011.6138209
[16] Yang, Y., Pierce, T., Carbonell, J. (1998). A study of retrospective and on-line event detection. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 28-36. https://doi.org/10.1145/290941.290953
[17] Bouras, C., Tsogkas, V. (2010). Assigning web news to clusters. In 2010 Fifth International Conference on Internet and Web Applications and Services, pp. 1-6. https://doi.org/10.1109/ICIW.2010.8
[18] Yu, R., Zhao, M., Peng, C., He, M. (2014). Online hot topic detection from web news archive in short terms. In 2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 919-923. https://doi.org/10.1109/FSKD.2014.6980962
[19] Tu, B., Yang, X., Li, N., Zhou, C., He, D. (2020). Hyperspectral anomaly detection via density peak clustering. Pattern Recognition Letters, 129: 144-149. https://doi.org/10.1016/j.patrec.2019.11.022
[20] Xu, M., Li, Y., Li, R., Zou, F., Gu, X. (2019). EADP: An extended adaptive density peaks clustering for overlapping community detection in social networks. Neurocomputing, 337: 287-302. https://doi.org/10.1016/j.neucom.2019.01.074
[21] https://onlinelibrary.wiley.com/doi/full/10.1002/asi.24026, accessed on 20 October 2021.
[22] Khan, H.U., Nasir, S., Nasim, K., Shabbir, D., Mahmood, A. (2021). Twitter trends: A ranking algorithm analysis on real time data. Expert Systems with Applications, 164: 113990. https://doi.org/10.1016/j.eswa.2020.113990
[23] Tong, B. (2019). Density peak clustering algorithm based on the nearest neighbor. In 3rd International Conference on Mechatronics Engineering and Information Technology (ICMEIT 2019), pp. 665-670. https://doi.org/10.2991/icmeit-19.2019.106
[24] Wang, Z., Wang, Y. (2020). A new density peak clustering algorithm for automatically determining clustering centers. In 2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI), pp. 128-134. https://doi.org/10.1109/IWECAI50956.2020.00034
[25] https://figshare.com/articles/dataset/Twitter_event_data_sets_2012-2016_/5100460, accessed on 20 October 2021.
[26] Batchanaboyina, M.R., Devarakonda, N. (2019). An effective approach for selecting cluster centroids for the k-means algorithm using IABC approach. In 2019 IEEE 18th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC), pp. 379-388. https://doi.org/10.1109/ICCICC46617.2019.9146077
[27] Devarakonda, N., Anandarao, S., Kamarajugadda, R., Wang, Y. (2019). Unique dragonfly optimization algorithm for harvesting and clustering the key features. In 2019 IEEE 18th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC), pp. 422-434. https://doi.org/10.1109/ICCICC46617.2019.9146092
[28] Sarvani, A., Venugopal, B., Devarakonda, N. (2018). A refined K-means technique to find the frequent item sets. In Cognitive Science and Artificial Intelligence, pp. 45-53. https://doi.org/10.1007/978-981-10-6698-6_5
[29] Raju, Y., Devarakonda, N. (2021). A cluster medoid approach for cloud task scheduling. International Journal of Knowledge-based and Intelligent Engineering Systems, 25(1): 65-73. https://doi.org/10.3233/KES-210053
[30] Rodriguez, A., Laio, A. (2014). Clustering by fast search and find of density peaks. Science, 344(6191): 1492-1496. https://doi.org/10.1126/science.1242072
[31] Devarakonda, N., Kavitha, D., Kamarajugadda, R. (2021). Escape the traffic congestion using brainstorming optimization algorithm and density peak clustering. Ingénierie des Systèmes d'Information, 26(3): 285-293. https://doi.org/10.18280/isi.260305
[32] https://blog.twitter.com/engineering/en_us/a/2015/building-a-new-trends-experience, accessed on 20 October 2021.
[33] Bird, S. (2006). NLTK: The natural language toolkit. In Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions, 69-72. https://doi.org/10.3115/1225403.1225421
[34] Wang, P.P., Liu, P.Y., Wang, R., Zhu, Z.F. (2017). A recommendation algorithm based on density peak clustering and key users. In 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), pp. 468-473. https://doi.org/10.1109/FSKD.2017.8393314
[35] Yang, Y., Zheng, K., Wu, C., Niu, X., Yang, Y. (2019). Building an effective intrusion detection system using the modified density peak clustering algorithm and deep belief networks. Applied Sciences, 9(2): 238. https://doi.org/10.3390/app9020238
[36] Ahuja, R., Solanki, A., Nayyar, A. (2019). Movie recommender system using K-Means clustering and K-Nearest Neighbor. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 263-268. https://doi.org/10.1109/CONFLUENCE.2019.8776969