Final version of the accepted paper. Cite as: *"M. Abulaish, S. Sharma, and M. Fazil, A Multi-Attributed Graph-Based Approach for Text Data Modeling and Event Detection in Twitter, In Proceedings of the Social Networking Workshop, 11th International Conference on Communication Systems and Networks (COMSNETS), Banglore, India, pp. 1-6, Jan. 7-11, 2019."*

A Multi-Attributed Graph-Based Approach for Text Data Modeling and Event Detection in Twitter

Muhmmad Abulaish, SMIEEE
Department of Computer Science
South Asian University, Delhi, India
Email: abulaish@ieee.org

Sielvie Sharma
Department of Computer Engineering
Jamia Millia Islamia, Delhi, India
Email: sielvie@outlook.com

Mohd Fazil
Department of Computer Science
Jamia Millia Islamia, Delhi, India
Email: mfazil@ieee.org

***Abstract*—The popularity of the microblogging sites like `Twitter` is increasing exponentially in which users are allowed to post short messages (*aka* tweets) using a maximum of 280 characters, mainly for news sharing and events updates. Besides, textual data, `Twitter` data also contains multi-dimensional connections among the users if they follow each other or have common followers/followees. Similarly, multi-dimensional connections exist among the tweets if they contain common *hashtags*, *mentions*, etc. In recent years, `Word2Vec` is being extensively used to analyze textual data, and it has shown promising results in many domains. In this paper, we propose a multi-attributed graph-based approach for text data modeling and event detection in Twitter. To this end, we generate a multi-attributed social graph (`MASG`), in which nodes, representing tweets, are labelled with numeric vectors obtained through `Word2Vec` model, and edges represent the structural relationships among the tweets and they can also be labeled with numeric vectors. Thereafter, `MASG` is converted into a similarity graph using a distance function, and Markov clustering algorithm is applied over the similarity graph to identify different clusters, where each cluster corresponds to a particular event. The proposed approach is evaluated over real-world `Twitter` datasets using standard evaluation metrics including `TPR` and `FPR`. It is also compared with some baseline methods and performs significantly better.**

***Index Terms*—Social Network Analysis, Text Data Modeling, Word2Vec, Multi-Attributed Social Graph, Markov Clustering, Event Detection.**

I. Introduction

Online Social Networks (OSNs) are the modern communication media that are easy to use and facilitate real-time communication. The use of OSNs is generating massive amount of *user-generated content*, which can be used to extract interesting and useful insights. There are different OSNs for different purposes, e.g., `Facebook` is generally used to connect to family members, friends, and acquaintances to get in touch with them, `Instagram` is used for pictures and videos sharing, `Twitter` is used to share thoughts and views on recent events and incidents, and `YouTube` is used for uploading and downloading videos. Among various OSNs, `Twitter` is one of the most popular microblogging networks, which allows its users to follow other users of the network to get subscription of their content and activities. `Twitter` monitors and analyzes users discussion and displays a list of top-10 discussed hashtags and keywords for each region. `Twitter` is mainly used by the users for news sharing and event updates, to express views and thoughts about recent events and incidents, to propagate and promote campaigns, and so on [1]. Therefore, `Twitter` data can be analyzed to identify the real-time events and users discourse. However, user-generated content in OSN are informal, ambiguous, noisy, and multi-lingual. In case of `Twitter`, contents are more informal and noisy due to limit on the length of tweets. Therefore, event identification from `Twitter` data is a non-trivial and complicated task. In addition, words context further complicates the problem because same word in social media can be used at different places in different contexts. Therefore, monitoring and tracking of events and sub-events to observe the changes in users discourse is one of challenging and important problems. It has numerous applications in different situations, such as products rating prediction and recommendation, open-source intelligence, events monitoring, and so on.

Existing literatures have a number of event detection approaches based on classification and clustering techniques [2], [3], [4], [5]. However, existing approaches are generally based on feature engineering task to classify the posts/tweets as whether they are related to a particular event or not. In addition, existing approaches are generally based on the temporal and spatial information of the posts, and they use term frequencies for events tracking, ignoring the contextual similarity of the terms. Moreover, clustering-based approaches have utilized only content-based similarity and ignored the structural information for detecting events in OSN [6].

In this paper, we propose a graph-theoretic data modeling and clustering approach for detecting events in Twitter. The proposed approach uses the graph-theoretic concepts for generating a multi-attributed social graph (`MASG`) to model the textual and structural information embedded within the tweets. The textual information makes the proposed approach a context-aware method by exploiting the embeddings of the representative terms of the tweets, avoiding the sparseness problem, which is generally associated with the *tf-idf* based approaches. On the other hand, structural information based on the overlapping hashtags and mentions among the tweets represent the clique forming behavior of the tweets, which are related to the same event. In proposed `MASG`, nodes represent tweets and edges represent hashtags- and mentions-based relationship. A novel distance function proposed in one of our previous works [7] is applied over the `MASG` to convert it into a similarity graph. The constructed similarity graph is passed to a graph-based clustering algorithm, Markov Clustering (MCL), to identify clusters representing the underlying events. The proposed approach is experimentally evaluated over real-world Twitter datasets using standard evaluation metrics including TPR and FPR.

The rest of the paper is organized as follows. Section II presents a brief review of the existing literatures on events detection, especially in online social networks. Section III describes the basic concepts, such as multi-attributed social graph and word embedding that are used in the proposed approach. Section IV presents the functioning details of the proposed approach. Section V-C presents a detailed description of the experimental setup and evaluation results. Finally, section VI concludes the paper and presents future directions of research.

II. Related Works

OSNs have given rise to multiple research directions, such as predictive analytics, event detection, outlier detection. Similarly, a number of approaches for event detection have been proposed in [6], [8], [9]. Azam et al. [2] used standard data mining techniques for analysis and classification of events in Twitter data. They used the topic modeling technique, latent Dirichlet allocation, to identify significant terms from tweets. Thereafter, they constructed a weighted graph in which nodes represent tweets and weighted edge represent the degree of relevance association among the tweets. Finally, they used a clustering algorithm to identify clusters representing the underlying events.

A survey paper authored by Panagiotou et al. [10] discussed various event detection approaches in OSNs. In [4], the authors proposed an approach for detection of targeted events like earthquake. First, they learned a classification model based on various textual features for event-based tweets labeling, and thereafter used Kalman filtering and particle filtering to estimate locations of the events. Alsaedi et al. [6] presented a clustering-based approach for detecting events in Arabic tweets. They used textual information, ignoring the structural relationships among the tweets. McMinn et al. [8] proposed an entity-based event detection and event merging technique and evaluated it on a Twitter dataset.

It can be observed from the discussion mentioned above that most of the researchers have used only textual information, ignoring the structural information, for detecting events in OSNs. The proposed approach is an attempt to model textual and structural information of OSN data as a multi-attributed social graph and apply graph-based clustering techniques for events detection.

III. Preliminaries

This section presents a brief description of two major concepts – *multi-attributed social graph* and *word embedding* that are the building blocks of our proposed approach.

a) Multi-attributed social graph: The multi-attributed social graph (MASG) is defined as a multi-dimensional weighted undirected graph, in which vertices represent tweets and labelled with multi-dimensional real-valued vectors generated using word2vec, and edges represent the structural associations among the vertices. There may exist multiple edges between the same pair of vertices. Mathematically, an MASG can be defined as $G =< V, E, \mathcal{F}_v, \mathcal{F}_e >$, where V represents the set of vertices, $E \subseteq V \times V$ represents the set of edges between the vertices, $\mathcal{F}_v$ is a vertex-labeling function $\mathcal{F}_v : V \rightarrow \mathcal{R}^n$, which maps every vertex of G to an *n*-dimensional real-valued vector, and $\mathcal{F}_e$ is an edge-labeling function $\mathcal{F}_e : E \rightarrow \mathcal{R}^m$, which maps every edge to an *m*-dimensional real-valued vector. Therefore, in an MASG, each vertex $v \in V$ is represented as an *n*-dimensional vector $\overrightarrow{v} = \{v_1, v_2, \ldots, v_n\}$ representing the vertex attributes, and each edge e between a nodes pair (u, v) is represented as an m-dimensional vector $\overrightarrow{e}(u, v) = \{e_1, e_2, \ldots, e_m\}$ representing the structural relationship between the vertices.

b) Word embedding: Word embedding maps every word from 1-dimensional vector space to a higher dimensional vector space by replacing the word with a numeric vector, which represents the position of the word in the higher dimensional vector space [11]. This numeric vector incorporates the contextual and semantic information of the word with respect to the co-occurring words in a given corpus. For example, if two words, say w_i and w_j, are contextually similar, then the vector distance between their embeddings, say e_i and e_j, will be low, representing the fact that both words are placed in close vicinity in the vector space. Instead of using existing word embeddings like GloVe [12], in this work, we have learned word embeddings using the popular Word2Vec [13] model over a large tweets corpus.

IV. Proposed Approach for Text Data Modeling and Event Detection

This section presents a detailed description of the proposed multi-attributed graph-based approach for text data modeling and event detection in Twitter. Figure 1 presents the workflow of the proposed approach, which mainly consists of *data crawling and pre-processing*, *keywords extraction and word embeddings learning*, *multi-attributed social graph generation*, and *similarity graph construction and clustering*. A detailed description of these processes are presented in the following sub-sections.

A. Data Crawling and Pre-processing

This section presents the description of data crawling and pre-processing steps for generating datasets to evaluate the proposed approach for text data modeling and event detection approach. The tweets related to different events are extracted from Twitter using the REST API provided by Twitter. The crawled datasets include tweets content and their associated meta-data, such as, geolocation, time, tweet-id, and user name. A more detailed description, along with the statistics of the crawled datasets is presented in *the experimental setup*

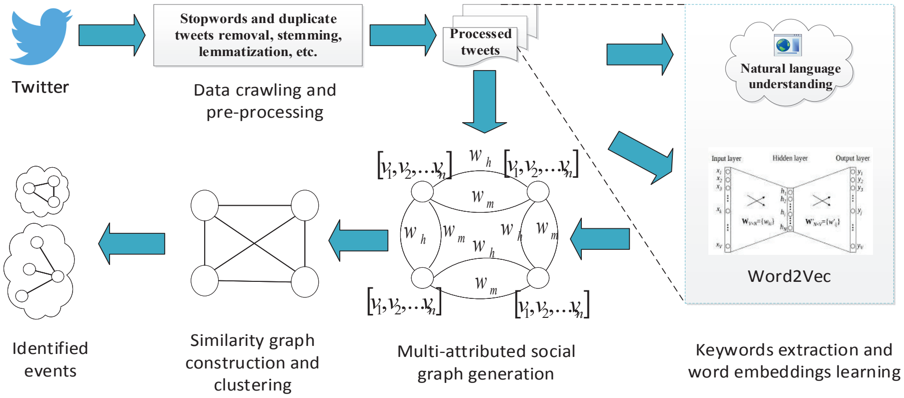
Twitter
Stopwords and duplicate tweets removal, stemming, lemmatization, etc.
Data crawling and pre-processing
Processed tweets
Natural language understanding
Input layer
Hidden layer
Output layer
Word2Vec
Keywords extraction and word embeddings learning
$[v_1, v_2, \ldots v_n]$
w_h
w_m
Multi-attributed social graph generation
Similarity graph construction and clustering
Identified events

Fig. 1. Work-flow of the proposed multi-attributed graph-based text data modeling and event detection approach

and results section of this paper. Data pre-processing is a major and essential step to decrease data redundancy, noise and complexity while maintaining the correctness of the data. The crawled datasets are pre-processed to remove stopwords, duplicates, and informal structures to convert them into a unified and meaningful format. In the data pre-processing step, first duplicate tweets were removed to avoid redundancy, and we found that a significant number of tweets were removed in this process. Thereafter, stop-words and punctuation symbols were also removed from the datasets using NLTK toolkit.

B. Keywords Extraction and Word Embeddings Learning

In the proposed approach, processed tweets are modeled as MASG, in which tweets are represented by the nodes, and each node is labeled by the average embedding vector of the embeddings of the representative keywords of the underlying tweet. Therefore, in the proposed approach, keywords extraction and embeddings learning processes are very vital. Keywords from the tweets corpus are extracted using natural language understanding (NLU) service, an IBM tool for natural language processing to extract entities, keywords, sentiment analysis, etc. The extracted keywords are the representative terms of the corpus describing the underlying events. The extracted keywords also have relevance score representing their relative importance in the underlying corpus. For word ebeddings, instead of using existing GloVe [12] or other word embedding models, we have learned 50-dimensional word embeddings using the Word2Vec [13] model over a large tweets corpus.

C. Multi-Attributed Social Graph Generation

This section presents a detailed description of the MASG construction process. As discussed earlier, an MASG can have multiple edges between vertices and multiple labels for each vertex. In the proposed approach, pre-processed tweets are modeled as MASG, in which vertices represent the tweets and edges represent the structural relationship among the tweets. The MASG is defined as $G(V, E, \mathcal{F}_v, \mathcal{F}_e)$, where V is set of vertices representing the tweets, $E \subseteq V \times V$ is set of edges representing the structural relationship between the tweets, $\mathcal{F}_v$ is an vertex-labeling function that maps each vertex of the graph to an n-dimensional real-valued word embedding, and $\mathcal{F}_e$ is an edge-labeling function that maps edges to an m-dimensional real-valued vector, based on the degree of hashtags and mentions overlap among the tweets. In the MASG, each vertex is labeled with the representative keywords that are present in the underlying tweet. Thereafter, average of the embeddings of all keywords of node is used to generated the real-valued label for the node. Similarly, edges between a pair of nodes are labeled using the structural similarity between the respective tweets, representing the structural association between them. In the proposed approach, there are two types of edges between every tweets pair, one based on common *hashtags* and other based on common *mentions*, representing two types of structural relationships between the tweets. In the proposed approach, edge vector representing different structural similarity between a pair of tweets is calculated as the fraction of overlapping *hashtags* and *mentions* between the tweets. A brief description of vertex and edge vectors, and corresponding weight calculation process is presented in the following sub-sections.

1) Edge Vector: To find different structural relationship among the vertices (tweets), edge vectors for an edge connecting a pair of tweets is generated based on the overlapping *hashtags* and *mentions* between the tweets. The structural similarity based on the overlapping *hashtags* between a pair of tweets T_i and T_j is represented using W_h and calculated using equation 1, where $H(T_i)$ represents the *hashtags* set of the i^{th} tweet T_i and $H(T_j)$ represents the *hashtags* set of the j^{th} tweet T_j. Similarly, structural similarity based on the overlapping *mentions* is represented using W_m and calculated using equation 2, where $M(T_i)$ and $M(T_j)$ represent the *mentions* set of i^{th} and j^{th} tweets, respectively.

$$W_h(T_i, T_j) = \frac{|H(T_i) \cap H(T_j)|}{|H(T_i) \cup H(T_j)|} \tag{1}$$

$$W_m(T_i, T_j) = \frac{|M(T_i) \cap M(T_j)|}{|M(T_i) \cup M(T_j)|} \tag{2}$$

$$W(i,j) = \alpha W_h(T_i,T_j) + \beta W_m(T_i,T_j) \quad (3)$$

2) Vertex Vector: In MASG, vertex vector represents the vertex attributes which can be used to find the attribute-based similarity between two vertices. The vertex vector for ecah vertex of the MASG is generated as an average of the real-valued embeddings of the keywords found in the respective tweet. As discussed earlier, keywords from the tweets corpus are extracted using NLU, and a 50-dimensional word embedding for each keyword is generated by learning the Word2Vec model [11] over the tweets corpus. Thereafter, table lookup is applied to find the containment of keywords in each tweet and the embedding of all keywords found in a tweet are averaged to generate the label of the respective vertex as a 50-dimensional real-valued vector.

D. Similarity Graph Construction and Clustering

On the basis of vertex and edge vectors calculated in the previous sections, this section finds the similarity between every pair of vertices of the MASG to map it into a similarity graph. In the process, aggregate similarity, incorporating the textual and structural similarity between every pair of nodes, say u and v, is calculated using a novel similarity function defined in one of our previously published research work [7]. The similarity function along with other related functions are re-produced in equations 4 to 7 for reference purpose. In these equations, γ is a real constant, which is used to control the degree of monotonicity of the λ function, and its value is determined empirically; n and m represent the dimensions of the vertex and edge vectors, respectively. The values of n and m are taken as 50 and 2, respectively in our experiment.

$$sim(u,v) = 1 - \frac{\Delta(u,v)}{\max\limits_{x,y \in V}\{\Delta(x,y)\}} \quad (4)$$

$$\Delta(u,v) = \sqrt{\lambda} \times \left(\sum_{i=1}^{n}(u_i - v_i)^2\right)^{1/2} \quad (5)$$

$$\lambda = \frac{1}{(1+\omega(u,v))^{\gamma}} \quad (6)$$

$$\begin{aligned}\omega(u,v) &= \alpha_1 e_1(u,v) + \alpha_2 e_2(u,v) + \cdots + \alpha_m e_m(u,v) \\ &= \sum_{i=1}^{m} \alpha_i e_i(u,v)\end{aligned} \quad (7)$$

Once the MASG is converted into the similarity graph using the process discussed above, we applied Markov clustering over it to find the clusters, where each cluster corresponds to a particular event. Markov clustering is a fast and scalable graph-based iterative clustering technique to decompose graph data into clusters. *Expansion* and *inflation* are the two major operations in Markov clustering which are performed iteratively, wherein *expansion* allows the flow of connection in different parts of the graph and *inflation* is responsible for strengthening and weakening of the current connections. The Markov clustering can be applied on both weighted and unweighted graphs.

V. Experimental Setup and Results

In this section, we present the experimental setup and evaluation results of the proposed multi-attributed graph-based approach for text data modeling and event detection in Twitter. Starting with a brief description of the datasets and evaluation metrics, we present experimental evaluation results of the proposed approach in the following sub-sections.

A. Datasets

To empirically evaluate the proposed approach, we crawled Twitter data using the trending hashtags of three important Indian events – *Budget 2018*, *Kasganj riot*, and *PNB scam* that occurred during the time-period 29^{th} January 2018 to 26^{th} February 2018. Table I presents the statistics of the dataset curated from the Twitter using a crawler, which was implemented in Python using REST API. Since many tweets were crawled more than once, we applied a filtering process using tweets' ids to remove all duplicate tweets from the dataset. As a result, the final dataset contains 75293 unique tweets. Table II presents the statistics of the filtering process and event-wise total number of tweets in the final dataset. All experiments were performed on a machine with 2.0 GHz Intel Core i3 processor and 16G RAM. Due to the limited computing resources, it was not feasible to do experiment over the whole dataset containing 75293 tweets. Therefore, we applied stratified random sampling over the whole dataset to generate two small datasets – one containing 750 tweets obtained by randomly selecting 250 tweets from each event category (hereafter known as *dataset-1*), and another containing 1350 tweets obtained by randomly selecting 450 tweets from each event category (hereafter known as *dataset-2*). All experimental evaluations and comparative analyses were performed over these two datasets.

B. Evaluation Metrics

The proposed approach is evaluated using standard data mining metrics – True Positive Rate (TPR) and False Positive Rate (FPR). TPR represents the fraction of correctly clustered tweets into their respective clusters, as defined in equation 8, where TP (*true positives*) represents the number of correctly clustered tweets and P represents the total number of tweets of a particular cluster. On the other hand, FPR represents the fraction of wrongly clustered tweets from other clusters' tweets, as defined in equation 9, where FP (*false positives*) represents the number of wrongly clustered tweets from other clusters, and N represents the total number of tweets in other clusters.

$$TPR = \frac{TP}{P} \quad (8)$$

$$FPR = \frac{FP}{N} \quad (9)$$

TABLE I
STATISTICS OF THE CRAWLED DATASET FROM TWITTER

Events	**Hashtags**	**Time period**	**#Raw tweets**
Kasganj riot	#chandangupta, #justice4chandan, #kasganj, #kasganjclashes	29.01.2018 to 06.02.2018	26,729
Budget 2018	#budget, #budget2018, #budgetsession	01.02.2018 to 10.02.2018	1,11,838
PNB Scam	#niravmodi, #PNBFraud, #PNBScam	16.02.2018 to 26.02.2018	1,33,643

TABLE II
STATISTICS OF THE FINAL DATASET REATINED AFTER DUPLICATE REMOVAL AND FILTERING PRROCESS

Events	**#Raw tweets**	**#Duplicate tweets**	**#Retweets**	**#Unique tweets**
Kasganj riot	26,729	18	23,547	3,170
Budget 2018	1,11,838	262	80,142	31,434
PNB Scam	1,33,643	4,637	88,317	40,689
Total	2,72,210	4,917	1,92,006	75,293

TABLE III
CONTINGENCY TABLE REPRESENTING TP AND FP VALUES OF THE CLUSTERING RESULTS OBTAINED FROM TWO DIFFERENT DATASETS

Cluster id	**Dataset-1**			**Dataset-2**		
	Budget2018	*Kasganj riot*	*PNB scam*	*Budget2018*	*Kasganj riot*	*PNB scam*
C_1	243	0	0	441	0	0
C_2	7	250	3	0	407	0
C_3	0	0	247	9	43	450

TABLE IV
PERFORMANCE EVALUATION OF THE PROPOSED APPROACH IN TERMS OF TPR AND FPR OVER TWO DIFFERENT DATASETS

Cluster id	**Dataset-1**		**Dataset-2**	
	TPR	*FPR*	*TPR*	*FPR*
C_1	0.972	0.000	0.980	0.000
C_2	1.000	0.020	0.904	0.000
C_3	0.988	0.000	1.000	0.058
Average	0.987	0.007	0.961	0.019

C. Evaluation Results

In this section, we present the evaluation results obtained over both datasets discussed earlier. We applied NLU over the datasets for extracting keywords. Thereafter, we generated 50-dimensional real-valued vectors for each keyword through learning Word2Vec model over our complete tweets dataset. After keywords extraction and real-valued vectors generation, we constructed MASGs for both datasets using the process discussed earlier, and converted them into similarity graphs. Finally, we applied Markov clustering over the similarity graphs to generate clusters. Table III presents the clustering results in the form of contingency table obtained by applying the proposed approach over both datasets. In this table, cluster ids C1, C2, and C3 are used to represent the *Budget2018*, *Kasganj riot*, and *PNB scam* clusters, respectively, and the numeric values in the cells can be interpreted accordingly. For example, the numeric values in the second column of table III represents the fact that out of 250 *budget2018*-related tweets, 243 were correctly clustered and joined the cluster C1 (i.e., the value of TP for C1 is 243); 7 tweets were wrongly clustered to C2 (increasing the FP count of C2 by 7). Similarly, other column values of this table can be interpreted. Using the TP and FP values shown in table III, we have evaluated the performance of the proposed approach in terms of TPR and FPR values over both datasets that are shown in table IV. It can be observed from table IV that the proposed approach is successful to cluster most of the tweets correctly, and very few instances are mapped to wrong clusters. As a results, the average *TPR* value over both datasets of the proposed approach is 97.4%, whereas average FPR value over both datasets is very low, i.e., 1.3%.

D. Comparative Analysis

This section presents comparative analysis of the proposed approach with two baseline methods. In the first baseline approach, edge vector representing the mentions- and hashtags-based similarity values between the nodes is neglected and similarity graph is constructed based on only vertex vectors. Similarly, in the second baseline approach, vertex vectors are ignored and similarity graph is constructed based on the edge vectors only. The proposed approach is compared with the baseline methods using the two datasets described in section V-A, and it is evaluated using the metrics defined in section V-B. The performance comparison results of the proposed approach with the two baseline methods are presented in table V. It can be observed from this table that the proposed approach performs significantly better than the baseline methods. The second row of the table has same value for both the evaluation metrics on both the datasets. It is due to the fact that while taking only vertex component, all the tweets of three events are grouped into a single cluster for both datasets, thereby, having 100% TPR for one event and 0% for the remaining two events,

TABLE V
PERFORMANCE COMPARISION RESULTS IN TERMS OF TPR AND FPR OVER TWO DIFFERENT DATASETS

Approach	Dataset-1		Dataset-2	
	TPR	*FPR*	*TPR*	*FPR*
Proposed approach	0.987	0.007	0.961	0.019
Proposed approach\edge component	0.333	0.333	0.333	0.333
Proposed approach\vertex component	0.841	0.0	92.66	0.0

resulting in average TPR as 33.3%. Similarly, 100% FPR for one event and 0% for the remaining two events, resulting in average FPR as 33.3%. On the other hand, while considering only edge component for similarity graph construction, more than 10 clusters are obtained for both the datasets having one major cluster for each event, whereas remaining tweets of each events are clustered in separate clusters. Although, tweets are not misclassified in the clusters of other events, they form several small and separate clusters, leading to zero FPR value.

VI. Conclusion and Future Work

In this paper, we have proposed a multi-attributed graph-based approach for text data modeling and event detection in Twitter. We have also provided a mathematical formulation of a multi-attributed social graph (MASG and shown its application to model the textual and structural information of the textual data, using the concept of Word2Vec and structural relationships between the text documents. The multi-attributed social graph is defined as an extension of the undirected weighted graph in which both nodes and edges can have labels as multi-dimensional real-valued vectors. We have also presented the process of converting MASG into a similarity graph. Finally, Markov clustering is applied over the similarity graph to decompose it into multiple clusters, each one describing a particular event. Though, we have used Markov clustering in this study, other graph-based clustering algorithms can also be used to cluster the similarity graph. The proposed approach is evaluated over real Twitter datasets, and the evaluation results in terms of TPR and FPR are encouraging. At present, we are working towards quantitative evaluation of the proposed approach over larger-scale datasets collected from different online social platforms. We are also working towards the comparison of the proposed approach with some of the state-of-the-art approaches for events detection in online social media.

References

[1] M. Fazil and M. Abulaish, “A hybrid approach for detecting automated spammers in twitter,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2707–2719, 2018.

[2] N. Azam, M. Abulaish, and N. A.-H. Haldar, “Twitter data mining for events classification and analysis,” in *Soft Computing and Machine Intelligence (ISCMI), 2015 Second International Conference on*. Hong Kong, China: IEEE Computer Society, 2015, pp. 79–83.

[3] H. Becker, M. Naaman, and L. Gravano, “Beyond trending topics: Real-world event identification on twitter,” Barcelona, Spain, 2011, pp. 438–441.

[4] T. Sakaki, M. Okazaki, and Y. Matsuo, “Earthquake shakes twitter users: real-time event detection by social sensors,” in *Proceedings of the 19th international conference on World Wide Web*, Raleigh, USA, 2010, pp. 851–860.

[5] R. Li, K. H. Lei, R. Khadiwala, and K. C. Chang, “Tedas: a twitter based event detection and analysis system,” Washington, USA, 2012, pp. 1273–1276.

[6] N. Alsaedi, P. Burnap, and O. Rana, “Sensing real-world events using arabic twitter posts,” Cologne, Germany, 2016, pp. 515–518.

[7] M. Abulaish and Jahiruddin, “A novel weighted distance measure for multi-attributed graph,” in *Proceedings of the 10th India Compute Conference*. Bhopal, India: ACM, 2017, pp. 39–47.

[8] A. J. McMinn and J. M. Jose, “Real-time entity-based event detection for twitter,” in *Proceedings of the 6th International Conference on Experimental IR Meets Multilinguality, Multimodality, and Interaction*. Toulouse, France: Springer-Verlag Berlin, Heidelberg, 2015, pp. 65–77.

[9] A. Schulz, B. Schmidt, and T. Strufe, “Small-scale incident detection based on microposts,” in *Proceedings of the International Conference on Hyper Text*. Guzelyurt, Northern Cyprus: ACM, 2015, pp. 3–12.

[10] N. Panagiotou, I. Katakis, and D. Gunopulos, “Detecting events in online social networks: Definitions, trends and challenges,” in *Solving Large Scale Learning Tasks: Challenges and Algorithms (Lecture Notes in Computer Science)*, 2016, pp. 42–84.

[11] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Proceedings of International Conference on Neural Information Processing Systems*. Nevada, USA: ACM, 2013, pp. 3111–3119.

[12] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: ACL, 2014, pp. 1532–1543.

[13] Y. Goldberg and O. Levy, “word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method,” *arXiv preprint arXiv:1402.3722*, 2014.