Trilateral Spearman Katz Centrality Based Least Angle Regression for Influential Node Tracing in Social Network

P. Vimal Kumar1 · C. Balasubramanian1

Accepted: 16 August 2021 / Published online: 2 September 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract
With the epidemic growth of Online Social Networks (OSNs), a large scale research on information dissemination in OSNs has been made an appearance in contemporary years. One of the essential researches is Influence Maximization. Most research adopts community structure, greedy stage, and centrality measures, to identify the influence node set. However, the time consumed in analyzing the influence node set for edge server placement, service migration and service recommendation is ignored in terms of propagation delay. Considering the above analysis, we concentrate on the issue of time-sensitive influence maximization and maximize the targeted influence spread. To solve the problem, we propose a method called, Trilateral Spearman Katz Centrality-based Least Angle Regression for influential node tracing in social network. Besides, two algorithms are used in our work to find the influential node in social network with maximum influence spread and minimal time, namely Trilateral Statistical Node Extraction algorithm and Katz Centrality Least Angle Influence Node Tracing algorithm, respectively. Extensive experiments on The Telecom dataset demonstrate the efficiency and influence performance of the proposed algorithms on evaluation metrics, namely, sensitivity, specificity, accuracy, time and influence spread.

Keywords Social network · Node cover preprocessor · Trilateral · Spearman correlated · Katz centrality · Least angle regression

1 Introduction
With the swift evolution of network technology and familiarization of the Internet, social applications, to name a few, like WeChat, Weibo and Snapchat, moderately produce an enormous amount of network data. Several practical applications required to mine profitable information from the profitable network data. Since then, the investigation of Influence Maximization (IM) has become an intense topic of concern in social network, specifically in the recent few years due to its crucial part in an extensive span of fields, like, network monitoring, product/brand recommendation and so on.

* P. Vimal Kumar
mevimal2016@gmail.com

1 Department of CSE, PSR Engineering College, Sivakasi, India
Community Based Influence Maximization Algorithm (CBIMA) was proposed in [1] to solve the issues concerning influence maximization in social networks. With this objective, an influence maximization method was designed that in turn obtained influential nodes by means of community structure followed by which the influence distribution difference was presented. Here, to start with, network embedding-based community detection model was designed with which the entire social network was split into distinct high-quality communities.

Next, the solution for influence maximization was divided into candidate stage and the greedy stage. Here, in candidate stage, the candidate nodes were selected via a heuristic algorithm, and as far as the greedy stage was concerned seed nodes were selected via modular property-based Greedy algorithm. With this, better performance was said to be arrived at with minimum running time.

Community Finding Influential Node (CFIN) was designed in [2] was designed with the purpose of selecting optimal users on the basis of the community structure, which in turn maximized the influence spread in the networks. The CFIN was split into two stages. They were, selection of seed and community spreading in a localized manner.

In the first stage, seed nodes were extracted via community detection algorithm. Here, with the objective of reducing the computational complexity involved in selecting seed node, meaningful communities were obtained in a significant manner. Next, influence spread inside communities was determined with which the final seed nodes responsible for distribution was analyzed. With this both the coverage ratio and running time were said to be improved with better coverage ratios.

Despite improvements observed in the coverage ratio with minimum running time and higher coverage ratio for obtaining influential nodes, the accuracy and the sensitivity rate with higher influence spread was less focused. To address these issues in our work, Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) for influential node tracing in social network is proposed.

1.1 Motivation

Recently, some researches are made to discover influential node and offer user ranking based on number of followers the user has. Both social network structure and strength of influence among individuals evolve continuously. The existing Influential node tracing techniques is finding influential nodes for only one static social network. But real-world social networks are dynamic. Therefore, accurate and precise tracking of the influential nodes is required under a dynamic setting. In order to address these issues, Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) is proposed with the different processes to perform influential node tracing in social network.

1.2 Objectives

The main contributions of this paper are summarized as follows:

- To achieve influential node tracing in social network, a Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) is proposed. TSKC-LAR is proposed with the process of Node Cover Preprocessor model, Trilateral Statistical Node Extraction, Spearman Correlated Normalized Feature Selection, Katz Centrality Least
Angle Influence Node Tracing to maximize the accuracy and influence spread under a dynamic social network.

- To reduce the tracing time of influence maximization node, Node Cover Preprocessor model is employed to obtain the effective nodes and eliminate the ineffective nodes. Then, feature extraction or node extraction is presented by employing Trilateral Statistical Node Extraction model to obtain the optimal features or nodes.
- To identify the influential nodes, a Spearman Correlated Normalized Feature Selection model is applied. At last, Katz Centrality Least Angle Regression is utilized to the selected nodes for significant influential node tracing with higher rate of sensitivity and specificity.
- We evaluate the performance on large-scale telecom dataset. The experiment results confirm our theoretical findings and show that our proposed Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) method achieve better performance of both influence coverage and running time along with sensitivity, specificity and accuracy.

1.3 Organization of Paper

The manuscript continues as follows. Section 2 presents a considerable reference list for related work in the field. The results of this exploration account for the novelty of the multi-objective proposal of this work. Section 3 describes the Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) method. Section 4 provides the experimental setup, while Sect. 5 presents and discusses the main results obtained. Finally, conclusions are included in Sect. 6.

2 Related Works

Identifying the most influential nodes in social networks has extensive range of applications. Different types of methods have been in the recent years designed with the purpose of estimating the influential nodes on the basis of their structural location in social network. An algorithmic approach based on Map Reduce with centrality as a measure was proposed in [3] with the purpose of improving the computational efficiency. However, with natural life evolution and influence made by peer groups have resulted in temporal patterns. To address this issue, a temporal preference model was designed in [4] to detect network structure changes on the basis of the node centrality. With this, strong correlations were said to be established. Yet another influence ranking model based on Susceptible Infected Recovered was proposed in [5] to find influential nodes.

From the information transmission aspect, different types of social networks have different information transmission modes due to the function diversity and user structures. Due to this, social network has become a new path for viral marketing that is utilized in promoting products, innovations and opinions. An enhanced multi attribute k shell method was designed in [6] utilizing the iterative information in the decomposition process for ranking influential nodes.

Yet another deep influential evaluation model was proposed in [7] by considering complicated graphic structure. This in turn evaluated influence node in computationally efficient manner. However, with the inception of complex networks, this type of influence node
identification remains a major issue to be attended. A machine learning framework was
designed in [8] to evaluate the complicated relationship between neighboring and non-neigh-
boring nodes.

In the analysis of influence maximization in social networks, the speed with which the
information is spread reduces with two different factors, i.e., elevated time and distance.
Therefore, the examination of the transmission of information is of great importance to the
supervision and administration of public point of view.

An Influence Maximization algorithm was designed in [9] on the basis of the rate attenua-
tion propagation model. With this, both the accuracy and time were said to be improved. Two
new effective algorithms based on centrality measure and local measure were proposed in [10]
under the Independence Cascade (IC) and Linear Threshold (LT) models. With this, the time
complexity involved in influential node analysis was found to be reduced.

Conventional types of methods, like, centrality based and machine learning, only took
into considerations either the structures involved in the network or the features involved in
the design to measure the node significance. However, the influential significance has to be
determined by both considering the network structures and node features. To solve this issue, a
deep learning model was presented in [11] to identify the most influential nodes in a complex
network. Graph classification method was applied in [12] for identifying significant nodes.

In recent few years studies conducted on the influential community have identified com-
munities that involve large types of influential members. Several types of metrics are hence
involved in influencing and existing techniques however search for influential communities
by taking into account only one influence type. In [13], an efficient influential community
search method that identified the influential communities based on multiple influence criteria.
Yet another deep reinforcement learning was applied in [7] for influence maximization. In
the meanwhile, investigations made on prior analysis have specifically designed on the basis
of the static network topology. However, the user’s online/offline status and topic preference
made it a cumbersome process for conventional methods to be applied in real scenarios. Based
on the above analysis, time-sensitive influence maximization was concentrated on [14] and
accordingly influence maximization was made. An interchange greedy approach was applied
in [15] to maximize joint influence under single network. With this better performance was
said to be achieved in both influence coverage and running time.

To improve both influence spread and time efficiency, a Multipath Asynchronous Thresh-
old (MAT) model was designed in [16] by employing both neighboring and non-neighboring
influencers into account. A network representation learning model was proposed in [17] by
using diffusion based processes, therefore identifying influential nodes. A new optimization
model using independent cascade and linear threshold was designed in [18] to maximize influ-
ence spread. Based on the above research and analysis, the proposed method constructs a four
step processes to perform influential tracing in social network. That is to say, considering the
normalized spearman correlation and Katz Centrality Least Angle, network structure informa-
tion for influential node identification is maximized with minimum response time.

3 Proposed Work

This section explains the design and development of the TSKC-LAR for influential node
tracing in social network. Here, four different steps are effectively combined to design
an algorithm for influential nodes selection, covering service migration and service
recommendation.
Figure 1 given below shows the block diagram of Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) for influential node tracing in social network. As illustrated in the below figure, to identify the influential nodes in social network, four different processes are followed. They are preprocessing, feature extraction, feature selection and influential node tracing. First, with the Telecom dataset provided as input, ineffective nodes are removed by means of Node Cover Preprocessor model. Second, feature extraction or node extraction is performed by employing Trilateral Statistical Node Extraction model where only the optimal features or nodes are extracted. Next, feature selection of normalized nodes is used in identifying the influential nodes are selected by applying Spearman Correlated Normalized Feature Selection model. Finally, Katz Centrality Least Angle Regression is applied to the selected nodes for significant influential node tracing. The elaborate description of the proposed method is given below.

![Block diagram of trilateral spearman katz centrality-based least angle regression (TSKC-LAR)](image-url)

**Fig. 1** Block diagram of trilateral spearman katz centrality-based least angle regression (TSKC-LAR)
3.1 Node Cover Preprocessor Model

With the evolution of social networks and extensive utilization between social network users, the magnitude of the social network is surging day by day. So it makes certain issues in identifying the most influential nodes like, memory deficiency, slowdowns of processing and so on. In this work, to address this issue, a Node Cover Preprocessing (NCP) model is designed that provides a pre-processing sample prior to the execution of the indigenous influential node selection algorithm.

This NPC model in turn minimizes the memory and therefore enhancing the process of detecting influential nodes and accelerating overall processes. The proposed method uses the NCP as a pre-processing. Figure 2 given below shows the block diagram of Node Cover Preprocessing (NCP) model.

As shown in the above figure, let us consider the social network modeled as a directed network \( G = (V, E) \), where \( V \) represents the individual nodes (i.e., 600 nodes) or social network users while \( E \) represents the links between the individual nodes or link between the social network users. In addition, each edge \((a, b \in E)\) is related with a propagation probability \( \text{Prop}_{a,b}^G \) specifying the robustness of effect of individual node or social network user \( a \) on \( b \) with two different conditions, effective or potential \( C_{a,b} \in E \rightarrow \text{Eff/Pot} \).

On one hand the effective edge \( E_{\text{Eff}} \) refers to the relationship between online users, and on the other hand, the potential edge \( E_{\text{Pot}} \) refers to that at least one user on both sides of the edge is offline. Due to the nature of users’ online status and time constraints of information dissemination, network structure is distinct.

For two information \( \text{Ini}_i \) and \( \text{Ini}_j \), to be transmitted at different time intervals \( T_i \) and \( T_j \), then \( T_i \) denotes the effective time for information \( \text{Ini}_i \) obtained using the deadline of information \( t_{\text{Ini}_i}^{\text{end}} \) and start time of information \( t_{\text{Ini}_j}^{\text{start}} \), as given below.

\[
T_i = t_{\text{Ini}_i}^{\text{end}} - t_{\text{Ini}_j}^{\text{start}}
\]

Next, propagation probability is derived due to the impact of users on neighbors has considerable variability, the propagation probability decreases with increase in distance ‘d’. This is mathematically expressed as given below.

\[
\mu = \frac{1}{d^2}
\]

Finally, the propagation probability ‘Prop’ for a directed network graph ‘G’ of two different social network users ‘a’ and ‘b’ is mathematically expressed as given below.
From the above Eq. (3), the propagation probability for removing ineffective nodes in the network is obtained by integrating the gradual loss ‘μ’ and the information to be propagated between two nodes or users ‘a, b[Ti]’ respectively. The pseudo code representation of Node Cover Preprocessor model is given below.

**Algorithm 1 Node Cover Preprocessor Model**

| Input: Nodes ‘N = N1, N2, ..., Nn’ |
|-----------------------------------|
| Output: Effective nodes ‘EN = EN1, EN2, ..., ENn’ |
| 1: Begin               |
| 2: For each Nodes ‘N’ with a directed network ‘G = (V, E)’ |
| 3: For information ‘In1’ and ‘Inj’ to be transmitted at different time intervals ‘Ti’ and ‘Tj’ |
| 4: Evaluate effective time using (1) |
| 5: Evaluate gradual loss ‘μ’ using (2) |
| 6: Evaluate propagation probability using (3) |
| 7: Return (effective nodes) |
| 8: End for               |
| 9: End for               |
| 10: End                  |

**Algorithm 1 Explanation**

As given in the above Node Cover Preprocessor model, three different measures are evaluated with the objective of eliminating the ineffective nodes. First, the effective time is measured with the time obtained, the gradual loss based on the distance is evaluated. Finally, propagation probability is measured to obtain the effective nodes and eliminate the ineffective nodes.

### 3.2 Trilateral Statistical Node Extraction

In this section, with the elimination of ineffective nodes and selection of effective nodes, next node extraction is done by applying trilateral statistical function. Here, trilateral statistical function refers to the application of three different hypotheses and extracting the nodes based on these three hypotheses. For the purpose of influential node tracing in social network, a node in our work is defined as a set of points with three hypotheses that the number of points ‘P = P1, P2, ..., Pn’ must be above the threshold ‘T’ and must be within a predefined transmission range ‘TR’. Figure 3 given below shows the block diagram of Trilateral Statistical Node Extraction model.

As shown in the above figure, the preprocessed nodes (i.e., 500 nodes) provided as input, three hypotheses are applied in our work with the aid of statistical functions using the base station ‘BSi’ obtained via its location and the user ID ‘UIDi’ to extract nodes...
for identifying influential node in social network. The first hypothesis is expressed as given below.

\[ BS_i \leftarrow \sum_{i=1}^{n} \frac{\text{sumof}(UID_i)}{N} \]  \hspace{1cm} (4)

From the above Eq. (4), the base station ‘BS’ is recalculated by taking the average of its members (via ‘UID’) \( \frac{\text{sumof}(UID_i)}{N} \).

The second hypothesis is expressed as given below.

\[ BS_i \leftarrow \text{Add}(UID_i, P_i) \]  \hspace{1cm} (5)

From the above Eq. (5), the base station ‘BS’ saves the current user ‘UID’.

\[ BS_i \leftarrow \text{Add}(\text{Clear}(UID_i), P_i) \]  \hspace{1cm} (6)

From the above Eq. (6), the base station ‘BS’ is displaced with the current point ‘P’ by eliminating the current user ‘Clear(UID)’. In this way, by employing the location and significance of each node with respect to others, the most vital nodes required for identifying the influential node are extracted. The pseudo code representation of Trilateral Statistical Node Extraction is given below.

**Algorithm 2 Trilateral Statistical Node Extraction**
Algorithm 2 Explanation

As given in the above Trilateral Statistical Node Extraction algorithm, if the distance ‘Dis’ is smaller than the threshold distance ‘DisT’, then this means that the point ‘P_i’ is detected inside the transmission range ‘TR’ and hence the point index ‘i’ is added to prevailing user IDs ‘UID_i’ and the base station ‘BS_i’ with this corresponding location via latitude and longitude is recalculated by taking the mean location of its members (i.e., Users via ‘UID’). If the distance ‘Dis’ is larger than the distance threshold ‘DisT’ and the number ‘N’ of its members (i.e., Users via ‘UID’) is larger than the threshold ‘UIDT’, then the point ‘P_i’ is detected outside the transmission range ‘TR’ and the user points already stored in ‘UID_i’, therefore the node index ‘i’ is incremented by 1 to save the current user. Finally, the base station, ‘BS_i’ is displaced with current point ‘P_i’ and the ‘UID_i’ is cleared and then reinitiated with the index ‘i’.

3.3 Spearman Correlated Normalized Feature Selection

In this section, with the extracted nodes, features are selected by integrating spearman correlation and a normalization function. Figure 4 shows the block diagram of Spearman Correlated Normalized Feature Selection model.

As shown in the above figure, the green circle is the transmission range or the boundary of the node (i.e. user ID), the yellow circle is the node centre or the base station location (i.e., the base station) and the small points are locations of the users. For a
sample of size ‘\(n\)’, with ‘\(n\)’ representing the extracted features or nodes extracted ‘\(NE\)’, the ‘\(n\)’ raw scores ‘\(NE_i\)’, ‘\(NE_j\)’ are transformed to ranks ‘\(r_{NE_i}\)’, ‘\(r_{NE_j}\)’ and ‘\(r_s\)’ is evaluated as given below.

\[
r_s = \rho r_{NE_i} r_{NE_j} = \frac{\text{cov}(r_{NE_i}, r_{NE_j})}{\sigma_{r_{NE_i}} \sigma_{r_{NE_j}}} \tag{7}
\]

From the above Eq. (7), the rank ‘\(r_s\)’ is measured based on the pearson correlation coefficient ‘\(\rho\)’, covariance of rank of each extracted nodes ‘\(\text{cov}(r_{NE_i}, r_{NE_j})\)’, and the standard deviation of each extracted nodes ‘\(\sigma_{r_{NE_i}}, \sigma_{r_{NE_j}}\)’ respectively. With distinct integer formulation, the ranks to measure correlation between extracted nodes for selecting normalized nodes are expressed as given below.

\[
r_s = 1 - \frac{6 \sum [r(NE_i) - r(NE_j)]^2}{n(n^2 - 1)} \tag{8}
\]

From the above Eq. (8), ‘\(r(NE_i) - r(NE_j)\)’ refers to the difference between the ranks of two correlated extracted nodes ‘\(NE_i\)’ and ‘\(NE_j\)’ with ‘\(\cdot\)’ representing the total number of observations.

\[
L \left( \text{Prob}_{NE_j} \mid \text{Prob}_{BS} \right) = \frac{1 - r_s \left( \text{Prob}_{BS} \ast \text{Prob}_{NE_j} \right)}{1 - r_s \left( \text{Prob}_{BS} \right)} \ast \lambda \left| \text{Prob}_{NE_j} \right| \tag{9}
\]

This normalization is necessary because of the reason that as we advance through the algorithm, succeeding feature may explain less of the outcome of selected nodes ‘\(NS\)’ and so change in ‘\(1 - r_s \left( \text{Prob}_{BS} \ast \text{Prob}_{NE_j} \right)\)’ by splitting ‘\(NE_j\)’ are smaller. The consequence of this would be a coarser selection of base station, resulting in premature convergence and so the normalization ensures that this is not the case the feature is selected consistently by avoiding premature convergence at each stage of the algorithm. The pseudo code representation of Spearman Correlated Normalized Feature Selection is given below.
Algorithm 3 Spearman Correlated Normalized Feature Selection

| Input: Nodes Extracted ‘NE = NE₁, NE₂, ..., NEₙ’ |
|-----------------------------------------------|
| Output: Normalized node selection ‘NS = NS₁, NS₂, ..., NSₙ’ |

1: Initialize ‘ρ’
2: Begin
3: For each Nodes Extracted ‘NE’
4: Evaluate ranks for each extracted nodes using (7)
5: Evaluate distinct ranks using (8)
6: Perform normalization using (9)
7: Return (nodes selected)
8: End for
9: End

Algorithm 3 Explanation

As given in the above Spearman Correlated Normalized Feature Selection algorithm, the objective here remains in improving the accuracy involved in tracing the influential nodes in social network. This is achieved by applying two different functions, spearman correlation and normalization. First, by applying the spearman correlation the significance of influential node is shown and allows for further analysis of obtaining the influential node. Next, by employing normalization, premature convergence is avoided and hence has the probability of analyzing all the base stations for obtaining the influential node.

3.4 Katz Centrality Least Angle Influence Node Tracing

Finally, with the selected features or nodes, the actual influence node tracing in social network is performed by applying Katz Centrality to the Least Angle Regression model. Let ‘NS = NS₁, NS₂, ..., NSₚ’ be the predictors and ‘Y’ be the response and let ‘NS = [NS₁, NS₂, ..., NSₚ]’ represent the matrix with columns denoting the predictors. Let us also consider the regression coefficient as ‘β = (β₁, β₂, ..., βₚ)’, provide us with the estimator response ‘Y’ that is expressed as given below.

\[
Y = \sum_{j=1}^{p} NS_j \beta_j
\]  (10)

From the above Eq. (10), the estimator response ‘Y’, is obtained by employing the node selected ‘NS_j’ and the regression coefficient ‘β_j’ respectively. With this, Katz Centrality is applied to the estimator response for measuring the relative influence and this is expressed as given below.

\[
C_{Katz}(Y_i) = \alpha \sum_{j=1}^{n} NS_{ij} C_{Katz}(Y_j) + \beta
\]  (11)
From the above Eq. (11), the Katz Centrality \( C_{Katz} \) for measuring influence nodes based on neighboring and non-neighboring set is obtained based on the constant \( \alpha = 0.5 \), bias constant \( \beta \). Moreover, \( NS_{ji} \) take value ‘1’ if a predictor node is connected to node ‘j’ and take value ‘0’ if a predictor node is not connected to node ‘j’. Next, with the aid of LAR model, \( \beta \) is selected by minimizing total squared error subject as expressed below.

\[
\text{Min}(Y - Y'); \text{Subjectto} \sum_{j=1}^{p} |\beta_j| \leq t
\]

From the above Eq. (12), the estimator response is evaluated based on the minimization of total squared error, therefore considering the neighborhood and non-neighborhood set while identifying the influential nodes in social network. The pseudo code representation of Katz Centrality Least Angle Influence Node Tracing is given below.

**Algorithm 4 Katz Centrality Least Angle Influence Node Tracing**

| Input: Normalized node selected ‘NS = NS_1, NS_2, ..., NS_p’ |
|---|---|
| Output: Accurate and precise influence node tracing |

1: **Initialize** regression coefficient as \( \beta = (\beta_1, \beta_2, ..., \beta_p)' \),
2: **Begin**
3: **For** each Normalized node selected ‘NS’
4: **Evaluate** estimator response ‘Y’ using (10)
5: Apply Katz Centrality applied to the estimator response using (11) subject to constraint using (12)
6: **Return** (influential node)
7: **End for**
8: **End**

**Algorithm 4 Explanation**

As given in the above Katz Centrality Least Angle Influence Node Tracing algorithm, the objective remains in retrieving the influential node in social network by minimizing the total squared error subject to constraint while selecting toe predictors. This is achieved by first considering regression coefficient to accurately derive the relationship between the predictor variable (i.e., nodes selected) and the response (i.e., the influential node). This is obtained by second using a centrality factor considering both the neighboring and non-neighboring node while tracing the influential node.

**4 Experimental Setup and Qualitative Analysis**

**4.1 Experimental Settings**

Experimental evaluation of proposed Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) method and existing methods Community Based Influence
Maximization Algorithm (CBIMA) [1] and Community based algorithm for Finding Influential Nodes (CFIN) in complex social networks [2] are implemented in Java.

The experimental evaluation is performed with the Telecom Dataset http://sguangwang.com/TelecomDataset.html [19–22]. The dataset, provided by Shanghai Telecom consists of greater than 7.2 million records of effective accessing of the Internet via 3,233 base stations from 9,481 mobile phones acquired for a period of six months. This dataset is very commonly used for influential node selection serving topics such as edge server placement, service migration, service recommendation, etc. For the experimental consideration, six parameters, such as Month, Data, Start Time, End Time, Base Station Location, Mobile Phone ID are taken as input for influential node tracing in social network. The performances of proposed and existing methods are evaluated using different metrics such as sensitivity, specificity, response time, accuracy and influence spread with a number of nodes.

4.2 Qualitative Analysis

In this section the qualitative analysis with a sample of 10 nodes in social network is analyzed. First, Node Cover Preprocessing (NCP) model is applied to the sample nodes then the information to be transmitted at different time intervals is estimated as given below in Table 1 using the deadline of information and start time of information. Followed by which the propagation probability is evaluated based on the distance ‘d’.

Based on the propagation probability from (3), the effective nodes selected are ‘T1’, ‘T2’, ‘T4’, ‘T6’, ‘T9’ and ‘T10’ respectively. With the obtained effective nodes, most vital nodes required for identifying the influential node have to be extracted. This is performed by means of distance threshold ‘DisT’ and user threshold ‘UIDT’. Let the distance threshold be assumed to be ‘DisT = 4’ and user threshold to be assumed to be ‘UIDT = 3’ (Table 2).

Then from the first hypothesis, the base station ‘BSi’ is recalculated by taking the average of its members, i.e., given as below.

\[
\frac{1950 + 1900 + 2100 + 1450 + 1650}{6} = 1841.66
\]

| S. no | User ID | Using the deadline of information | Start time of information | Transmitted information time | Distance ‘d’ | Propagation probability |
|-------|---------|----------------------------------|---------------------------|-----------------------------|--------------|------------------------|
| 1     | T1      | 2250                             | 300                       | 1950                        | 2            | 0.25                   |
| 2     | T2      | 2250                             | 350                       | 1900                        | 2            | 0.25                   |
| 3     | T3      | 2250                             | 400                       | 1850                        | 3            | 0.11                   |
| 4     | T4      | 2250                             | 250                       | 2000                        | 2            | 0.25                   |
| 5     | T5      | 2250                             | 500                       | 1750                        | 4            | 0.06                   |
| 6     | T6      | 2250                             | 150                       | 2100                        | 2            | 0.25                   |
| 7     | T7      | 2250                             | 600                       | 1650                        | 4            | 0.06                   |
| 8     | T8      | 2250                             | 200                       | 2050                        | 1            | 1                      |
| 9     | T9      | 2250                             | 800                       | 1450                        | 2            | 0.25                   |
| 10    | T10     | 2250                             | 600                       | 1650                        | 2            | 0.25                   |
Then, the base station ‘BS$_i$’ saves the current user ‘UID$_i$’ based on the results of the second hypothesis results as ‘$T_4$’, ‘$T_6$’, ‘$T_9$’, ‘$T_{10}$’. The results of the optimal node extraction are given below (Table 3).

Then, extracted features or nodes extracted ‘NE’ is transformed to rank ‘$r_{NE}$’, ‘$r_{NE_j}$’ and \{displaystyle $X_{i}, Y_{i}$\} ‘$r_s$‘ using (7) and is given below.

Standard deviation of ‘$\sigma_{r_{NE_i}} = 70.71 (T_4, T_6)$’.

Standard deviation of ‘$\sigma_{r_{NE_j}} = 141.42 (T_9, T_{10})$’.

Covariance of $\text{cov} \left( r_{NE_i}, r_{NE_j} \right) = 5000$

The rank ‘$r_s$’ is measured based on the pearson correlation coefficient ‘$\rho$’, covariance of rank of each extracted nodes ‘$\text{cov} \left( r_{NE_i}, r_{NE_j} \right)$’, and the standard deviation of each extracted nodes ‘$\sigma_{r_{NE_i}}, \sigma_{r_{NE_j}}$’ as given below

$$r_s = \frac{5000}{70.71 * 141.42} = 0.50$$

Normalization for ‘NE$_j$’ is evaluated as given below.

$$1 - \frac{0.50(0.25 * 141.42)}{1 - 0.50(0.25)} * 0.01(141.42)$$

$$1 - \frac{17.6775}{1 - 0.125} * 1.41 = 2.70$$

Normalization for ‘NE$_i$’ is evaluated as given below.

$$1 - \frac{0.50(0.25 * 70.71)}{1 - 0.50(0.25)} * 0.01(70.71)$$

### Table 2 Preprocessed results

| User ID | Using the deadline of information | Start time of information | Transmitted information time | Propagation probability |
|---------|----------------------------------|---------------------------|----------------------------|------------------------|
| $T_1$   | 2250                             | 300                       | 1950                       | 0.25                   |
| $T_2$   | 2250                             | 350                       | 1900                       | 0.25                   |
| $T_4$   | 2250                             | 250                       | 2000                       | 0.25                   |
| $T_6$   | 2250                             | 150                       | 2100                       | 0.25                   |
| $T_9$   | 2250                             | 800                       | 1450                       | 0.25                   |
| $T_{10}$| 2250                             | 600                       | 1650                       | 0.25                   |

### Table 3 Feature extracted results

| User ID | Using the deadline of information | Start time of information | Transmitted information time | Propagation probability |
|---------|----------------------------------|---------------------------|----------------------------|------------------------|
| $T_4$   | 2250                             | 250                       | 2000                       | 0.25                   |
| $T_6$   | 2250                             | 150                       | 2100                       | 0.25                   |
| $T_9$   | 2250                             | 800                       | 1450                       | 0.25                   |
| $T_{10}$| 2250                             | 600                       | 1650                       | 0.25                   |
$
\frac{1 - 8.83875}{1 - 0.125} \times 0.77781 = 6.89$

So the features selected are from ‘NE$_i$ = $T_4$, $T_6$’ (Table 4)

Finally, the influential node is estimated based on the Katz Centrality using (11) and (12). Let us assume the regression coefficient ‘$\beta_j$’ to be between ‘(0.7 to 1.0)’ and constant ‘$\alpha = 0.5$’, then, the estimator response for node ID ‘$T_4$’ is given as below.

\[
0.5 \times 1 \times y + 0.7 = y \times 0.5 = -0.7
\]

\[
y = \frac{0.7}{0.5} = -1.4
\]

In a similar manner, the estimator response for node ID ‘$T_6$’ is given as below.

\[
0.5 \times 1 \times y + 0.8 = y \times 0.5 = -0.8
\]

\[
y = \frac{0.8}{0.5} = -1.6
\]

Therefore, the influential node is ‘$T_6$’.

5 Results and Discussion

5.1 Case 1: Sensitivity

In influence node tracing sensitivity is a measure of how well a test can identify true positives. In other words, the sensitivity is its potentiality to determine the influential node tracing correctly. To estimate it, we should calculate the ratio of true positive in influential node tracing cases. Mathematically, this can be stated as given below.

\[
\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (13)
\]

From the above Eq. (13), the sensitivity rate ‘Sensitivity’ is measured based on the true positive ‘TP’ (i.e., the number of nodes correctly identified as influential node if so) and the false negative ‘FN’ (i.e., the number of nodes incorrectly identified as normal node) respectively. The sensitivity of a test can assist in showing how well it can classify samples (i.e. nodes) that have the condition. In other words, a high sensitivity value means a test correctly classifies a sample. Table 5 given below shows the sensitivity rate of the proposed TSKC-LAR and existing methods, CBIMA [1] and CFIN [2].

| User ID | Using the deadline of information | Start time of information | Transmitted information time | Propagation probability |
|---------|----------------------------------|---------------------------|-----------------------------|------------------------|
| $T_4$   | 2250                             | 250                       | 2000                        | 0.25                   |
| $T_6$   | 2250                             | 150                       | 2100                        | 0.25                   |
Figure 5 given above shows the sensitivity measure for tracing the influential node in social network. A simulation of 5000 nodes with unique user ID (i.e., mobile phone) is considered and the sensitivity rate is measured accordingly. With ‘500’ nodes considered for simulation, ‘42’ number of nodes correctly identified as influential node and ‘10’ number of nodes incorrectly identified as normal node using TSKC-LAR, ‘40’ number of nodes correctly identified as influential node and ‘15’ number of nodes incorrectly identified as normal node using [1] and ‘38’ number of nodes correctly identified as influential node and ‘20’ number of nodes incorrectly identified as normal node using [2], the sensitivity rate was observed to be 80.76%, 72.72% and 65.5% respectively. From the simulation results it is inferred that the sensitivity rate though decreases with the increase in the number of nodes, the sensitivity rate using TSKC-LAR is found to be better than [1] and [2]. This improvement is due to the application of Node Cover Preprocessor algorithm. By applying this algorithm, initially the effective time is measured and then followed by

| Nodes | TSKC-LAR (%) | CBIMA (%) | CFIN (%) |
|-------|--------------|-----------|----------|
| 500   | 80.76        | 72.72     | 65.5     |
| 1000  | 80.25        | 71.35     | 64.85    |
| 1500  | 78.95        | 71.15     | 64.55    |
| 2000  | 78.35        | 71        | 64.15    |
| 2500  | 78.15        | 70.85     | 64       |
| 3000  | 77.45        | 70.65     | 63.95    |
| 3500  | 77.25        | 70.25     | 63.75    |
| 4000  | 77           | 70        | 63.55    |
| 4500  | 76.55        | 69.55     | 62.55    |
| 5000  | 75           | 69.25     | 62       |

Fig. 5 Graphical representation of sensitivity
which the gradual loss based on the distance is evaluated. Finally, propagation probability is estimated to arrive at the effective nodes and discard the ineffective nodes. With this the sensitivity rate using TSKC-LAR method is found to be improved by 10% compared to [1] and 22% compared to [2].

5.2 Case 2: Specificity

In influential node tracing, specificity is a measure of how well a test can identify true negatives. It refers to the percentage of the true negatives out of all the samples that do not have the condition (true negatives and false positives). Mathematically, this can be stated as given below.

\[
\text{Specificity} = \frac{TN}{TN + FP} \tag{14}
\]

From the above Eq. (14), the specificity rate ‘Specificity’ is measured based on the true negative ‘TN’ (i.e., number of nodes correctly identified as normal node) and the false positive ‘FP’ (i.e., the number of nodes incorrectly identified as influential node). A test with a high specificity value means that it is correctly classifies the nodes with the condition more often than a test with a lower specificity. Table 6 given below shows the specificity rate of the proposed TSKC-LAR and existing methods, CBIMA [1] and CFIN [2] (Table 6).

Figure 6 given above illustrates the specificity rate for different numbers of nodes for accessing the internet through base station. With this specificity rate evaluation the service migration and service recommendation can be made in an efficient manner. With ‘500’ nodes considered for simulation, ‘380’ number of nodes correctly identified as normal node and ‘15’ number of nodes incorrectly identified as influential node using TSKC-LAR and existing methods, ‘360’ number of nodes correctly identified as normal node and ‘20’ number of nodes incorrectly identified as influential node using CBIMA [1] and ‘340’ number of nodes correctly identified as normal node and ‘35’ number of nodes incorrectly identified as influential node using CFIN [2], the overall specificity was observed to be 96.2%, 94.73% and 90.66% respectively. From the results it is inferred that the specificity rate using TSKC-LAR is better than [1] and [2]. The specificity rate improvement is due to the application of propagation probability that evaluates the distance on the basis of the

| Nodes | Specificity (%) |
|-------|-----------------|
|       | TSKC-LAR | CBIMA | CFIN  |
| 500   | 96.2      | 94.73  | 90.66 |
| 1000  | 95.25     | 93.15  | 88.15 |
| 1500  | 95.15     | 92.55  | 87.35 |
| 2000  | 94.85     | 91.35  | 86    |
| 2500  | 94.35     | 90.45  | 85.25 |
| 3000  | 94        | 89.25  | 84.15 |
| 3500  | 93.95     | 89     | 84    |
| 4000  | 93.75     | 88.15  | 83.85 |
| 4500  | 93.25     | 88.05  | 83    |
| 5000  | 93        | 87     | 82    |
neighborhood factor therefore assisting in removing the ineffective nodes. With this the specificity of TSKC-LAR method is said to be improved by 4% compared to [1] and 11% compared to [2].

5.3 Case 3: Accuracy

The third parameter used for analysis is the rate of accuracy. The accuracy of a test is its potentiality to distinguish the normal and influential nodes correctly. To evaluate the accuracy of a test, the ratio of influential nodes correctly predicted as it is and the nodes for evaluation has to be considered. Mathematically, this can be stated as:

\[
\text{Acc} = \frac{\sum_{i=1}^{n} \frac{N_{CR}}{N_i}}{N_i} \times 100
\]  

From the above Eq. (15), the accuracy rate ‘Acc’ is evaluated based on the sample nodes involved in influential node tracing ‘N_i’ and the nodes correctly traced ‘N_{CR}’. It is expressed in terms of percentage (%). Table 7 given below shows the accuracy rate of the proposed TSKC-LAR and existing methods, CBIMA [1] and CFIN [2].

Figure 7 given above shows the accuracy involved in tracing the influential nodes in social network based on the trajectory of users. From the figure it is inferred that the accuracy rate is said to be inversely proportional to the nodes considered for simulation. In other words, increasing the nodes causes an increase in the overall nodes involved in influential node tracing and therefore reducing the accuracy rate. However, experimental simulation shows betterment using TSKC-LAR when compared to CBIMA [1] and CFIN [2]. With ‘500’ nodes considered for simulation and ‘480’ nodes correctly traced using TSKC-LAR, ‘465’ nodes correctly traced using [1] and ‘450’ nodes correctly traced using [2], the overall accuracy was found to be 96%, 93% and 90% respectively. The reason behind the improvement in the accuracy rate is due to the application of Trilateral Statistical Node
Extraction algorithm. By applying this algorithm, optimal nodes for influential node tracing is said to be extracted. Here, the point detected inside the transmission range along with the corresponding location via latitude and longitude is taken into account for obtaining optimal nodes. With this, the accuracy using TSKC-LAR is said to be improved by 5% compared to [1] and 9% compared to [2].

### 5.4 Case 4: Response Time

The response time in our work refers to the time consumed in identifying influential node in social network. Mathematically, this can be stated as:

---

**Table 7** Performance measure of accuracy

| Nodes | TSKC-LAR | CBIMA | CFIN |
|-------|----------|-------|------|
| 500   | 96       | 93    | 90   |
| 1000  | 95.85    | 92.15 | 89.35|
| 1500  | 95.25    | 92    | 89.15|
| 2000  | 95       | 89.85 | 88.15|
| 2500  | 94.35    | 89.35 | 86.35|
| 3000  | 94.15    | 89    | 86   |
| 3500  | 94       | 88.55 | 85.25|
| 4000  | 93.75    | 88.25 | 85   |
| 4500  | 93.55    | 88    | 84.15|
| 5000  | 93       | 87.15 | 84   |

**Fig. 7** Graphical representation of accuracy
From the above Eq. (16), the response time ‘RT’ is measured based on the sample nodes involved in simulation ‘Ni’ and the time consumed in node tracing ‘Time[NTracing]’. It is measured in terms of milliseconds (ms). Table 8 given below shows the response time of the proposed TSKC-LAR and existing methods, CBIMA [1] and CFIN [2].

![Graphical representation of response time](image)

**Fig. 8** Graphical representation of response time

\[ RT = \sum_{i=1}^{n} N_i \times \text{Time}[\text{NTracing}] \]  
(16)

From the above Eq. (16), the response time ‘RT’ is measured based on the sample nodes involved in simulation ‘Ni’ and the time consumed in node tracing ‘Time[NTracing]’. It is measured in terms of milliseconds (ms). Table 8 given below shows the response time of the proposed TSKC-LAR and existing methods, CBIMA [1] and CFIN [2].

Figure 8 given above illustrates the response time involved in tracing the influential nodes in social network to the distinct user ID obtained at different time intervals. However, for fair comparison same numbers of nodes and user IDs are utilized in all the three methods for measuring the response time. From the figure the response time is directly
proportional to the nodes, i.e., increasing the numbers of nodes causes increase in the node deployment and consequently a linear rise is said to be observed in the response time. However, with ‘500’ nodes considered for experimentation and the time consumed in node trace with single node involved in the network framework being ‘0.185ms’ using TSKC-LAR, the time consumed in node trace with single node involved in the network framework being ‘0.215ms’ using [1] and ‘0.230ms’ using [2]. From these results it is inferred that the response time is comparatively lesser using TSKC-LAR than [1] and [2]. The reason behind the improvement is due to the extraction of optimal nodes for influential node tracing via Trilateral Statistical function. By applying this function, three hypotheses are checked by using the location and node significance with respect to others, the most vital nodes required for identifying the influential node are extracted. With these nodes, the influential nodes are identified in minimum time using TSKC-LAR by 12% compared to [1] and 17% compared to [2].

5.5 Case 5: Influence Spread

Finally, the influence spread over the whole network is analyzed in this section. Influence maximization refers to the problem of identifying a small subset of nodes (seed nodes) in social network that maximize the spread of influence. Table 9 given below shows the influence spread using three methods, TSKC-LAR, CBIMA [1] and CFIN [2].

Figure 9 given above shows the influence spread for different seed set size in the range of 10 to 50. From the figure it is inferred that the influence spread is maximized in all the three methods and comparatively found better using TSKC-LAR method than CBIMA [1] and CFIN [2]. The influence spread maximization was achieved using TSKC-LAR method due to the application of Katz Centrality Least Angle Influence Node Tracing algorithm. By applying this algorithm, while retrieving the influential node in social network, the total squared error was minimized while selecting the predictors. Also, relationship between the predictor variable (i.e., nodes selected) and the response (i.e., the influential node) were estimated on the basis of regression factor. Finally, both the neighboring and non-neighboring node was involved while tracing the influential node based on the centrality factor. This in turn improved the influence spread using TSKC-LAR method by 26% compared to [1] and 54% compared to [2].

Table 10 illustrates the results comparison of TSKC-LAR method and existing [1, 2]. From the above table, the average value of proposed and existing methods is provided for five different metrics. The proposed TSKC-LAR method provides the Sensitivity of 78% and the sensitivity of existing methods are 71% and 64%. The proposed TSKC-LAR method provides the Specificity of 94% and Specificity of existing methods are 90% and

| Seed set size | Influence spread (number of nodes) | TSKC-LAR | CBIMA | CFIN |
|---------------|-----------------------------------|----------|-------|------|
| 10            | 900                               | 750      | 600   |
| 20            | 1200                              | 900      | 750   |
| 30            | 1400                              | 1150     | 1000  |
| 40            | 1750                              | 1300     | 1050  |
| 50            | 1900                              | 1550     | 1250  |
The response time of proposed TSKC-LAR method is 189 ms and the existing methods is 215 ms and 229 ms. The accuracy of proposed TSKC-LAR method is 94% and the existing methods is 90% and 87%. The influence spread of proposed TSKC-LAR is 1430 and the existing methods is 1130 and 930. From the above result analysis, it is evident that, the proposed TSKC-LAR method provides better performance in terms of Sensitivity, Specificity, Accuracy, Response Time and Influence spread than the existing methods.

## 6 Conclusion

An efficient Trilateral Spearman Katz Centrality-based Least Angle Regression (TSKC-LAR) for influential node tracing in social network is developed with objective of increasing the influence spread. The key objective of TSKC-LAR method is to ensure influence maximization and minimize response time during influence tracking. The objective of TSKC-LAR method is attained with application of Trilateral Statistical Node Extraction and Katz Centrality Least Angle Influence Node Tracing algorithm. First effective nodes were identified by means of Node Cover Preprocessor. Next, by employing Trilateral
Statistical Node Extraction algorithm optimal nodes were extracted via three hypotheses. Moreover, to avoid premature convergence, normalized node selection was performed by applying normalization and spearman correlation. Finally, influential node in social network was identified by minimizing the total squared error. The efficiency of TSKC-LAR method is estimated in terms of sensitivity, specificity, accuracy, influence spread and response time compared with state-of-the-art works. The simulation results shows that the TSKC-LAR method presents better performance with an enhancement of influence spread and minimization of response time when compared to the state-of-the-art works.

**Declarations**

**Conflicts of interest** None.

**References**

1. Zhang, Z., Li, X., & Gan, C. (2020). Identifying influential nodes in social networks via community structure and influence distribution difference. *Digital Communications and Networks*. Elsevier, Apr 2020 [Community Based Influence Maximization Algorithm (CBIMA)].
2. Khomami, M. M. D., Rezvanian, A., Meybodi, M. R., & Bagheri, A. (2020). CFIN: A community-based algorithm for finding influential nodes in complex social networks. *The Journal of Supercomputing*, 77, 2207–2236.
3. Naik, D., Behera, R. K., Ramesh, D., & Rath, S. K. (2020). Map-reduce-based centrality detection in social networks: An algorithmic approach. *Arabian Journal for Science and Engineering*, 45, 10199–10222.
4. Pereira, F. S., Gama, J., de Amo, S., & Oliveira, G. M. (2018). On analyzing user preference dynamics with temporal social networks. *Machine Learning*, 107(11), 1745–1773.
5. Zareie, A., Sheikhahmadi, A., Jalili, M., & Fasaei, M. S. K. (2020). Finding influential nodes in social networks based on neighborhood correlation coefficient. Elsevier.
6. Zhao, N., Bao, J., & Chen, N. (2020). *Ranking Influential Nodes in Complex Networks with Information Entropy Method*. Wiley.
7. Tian, S., Mo, S., Wang, L., & Peng, Z. (2020). Deep reinforcement learning-based approach to tackle topic-aware influence maximization. Springer.
8. Zhao, G., Jia, P., Huang, C., Zhou, A., & Fang, Y. (2020). A machine learning based framework for identifying influential nodes in complex networks. *IEEE Access*, 8, 65462–65471.
9. Li, W., Fan, Y., Mo, J., Liu, W., Wang, C., Xin, M., & Jin, Q. (2019). Three-hop velocity attenuation propagation model for influence maximization in social networks. Springer.
10. Alshahrani, M., Fuxi, Z., Sameh, A., Mekouar, S., & Huang, S. (2020). Efficient algorithms based on centrality measures for identification of top-K influential users in social networks. Elsevier.
11. Zhao, G., Jia, P., Zhou, A., & Zhang, B. (2020). *InfGCN: Identifying influential nodes in complex networks with graph convolutional networks*. Elsevier.
12. Ma, T., Wang, H., Zhang, L., Tian, Y., & Al-Nabhan, N. (2020). *Graph classification based on structural features of significant nodes and spatial convolutional neural networks*. Elsevier.
13. Seo, J. H., & Kim, M. H. (2021). Finding influential communities in networks with multiple influence types. *Information Sciences*, 548, 254–274.
14. Huixiu, M., Jiuxin, C., Tangfei, Y., & Liu, B. (2020). Topic based time-sensitive influence maximization in online social networks. *World Wide Web*, 23(3), 1831–1859.
15. Song, G., Li, Y., Chen, X., He, X., & Tang, J. (2016). Influential node tracking on dynamic social network: An interchange greedy approach. *IEEE Transactions on Knowledge and Data Engineering*, 29(2), 359–372.
16. Wang, W., & Street, W. N. (2018). Modeling and maximizing influence diffusion in social networks for viral marketing. *Applied network science*, 3(1), 1–26.
17. Wei, H., Pan, Z., Hu, G., Zhang, L., Yang, H., Li, X., & Zhou, X. (2018). Identifying influential nodes based on network representation learning in complex networks. *PLoS ONE*. https://doi.org/10.1371/journal.pone.0200091
18. Olivares, R., Muñoz, F., & Riquelme, F. (2020). A multi-objective linear threshold influence spread model solved by swarm intelligence-based methods. Elsevier.
19. Wang, S., Zhao, Y., Xu, J., Yuan, J., & Hsu, C. (2019). Edge server placement in mobile edge computing. Journal of Parallel and Distributed Computing, 127, 160–168.
20. Wang, S., Zhao, Y., Huang, L., Xu, J., & Hsu, C. (2019). QoS prediction for service recommendations in mobile edge computing. Journal of Parallel and Distributed Computing, 127, 134–144.
21. Guo, Y., Wang, S., Zhou, A., Xu, J., Yuan, J., & Hsu, C. (2020). User allocation-aware edge cloud placement in mobile edge computing. Software: Practice and Experience, 50(5), 489–502.
22. Wang, S., Guo, Y., Zhang, N., Yang, P., Zhou, A., & Shen, X. S. (2019). Delay-aware microservice coordination in mobile edge computing: A reinforcement learning approach. IEEE Transactions on Mobile Computing [https://ieeexplore.ieee.org/document/8924682

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

P. Vimal Kumar was born in Virudhunagar District, India, in 1985. He received the M.E. degree in computer science and engineering from Anna University, Chennai, India, in 2016, and he currently doing his Ph.D. degrees in Information and Communication Engineering in Anna University, Chennai. In 2016, he joined the Department of Computer Science and Engineering, P.S.R. Engineering College, as an Assistant Professor. His current research interests include data mining, bigdata analytics, and image mining.

C. Balasubramanian was born in Virudhunagar District, India. He received the M.E. degree in computer science and engineering from Annamalai University, India, in 2005, and he received his Ph.D. degrees in Information and Communication Engineering in Anna University, Chennai in 2011. Currently he is working as an Professor in the Department of Computer Science and Engineering, P.S.R. Engineering College. His current research interests include data mining, bigdata analytics, and image mining.