Information Processing and Management 57 (2020) 102321

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Effective influence estimation in twitter using temporal, profile, structural and interaction characteristics

Sakshi Agarwal*, Shikha Mehta

Computer Science & Information Technology, Jaypee Institute of Information Technology, Noida, India

ARTICLE INFO

Keywords:
Influence diffusion
Temporal features
Interactions features
Structural features
Independent cascade
Linear threshold

ABSTRACT

Influence diffusion is extensively studied in social networks for product or service promotion and viral-marketing applications. This paper proposes two models for social influence estimation, namely Time Decay Features Cascade Model (TDF-C) and Time Decay Features Cascade Threshold Model (TDF-CT). These models overcome three main existing challenges - first, measure the strength of user's influence as an influencer; second, identify the set of users influenced by an influencer; third, estimate the time frame of the influence. TDF-C is an M-TAP based diffusion model, which learns influence probabilities between users using four types of features, namely temporal, interaction, structural, and profile features, and uses Independent Cascade (IC) model for influence estimation. TDF-CT is an extension of the TDF-C model, which uses temporal and interaction features to calculate the diffusion through the Progressive Feedback Estimation (PFE) model in place of IC model. PFE model is a fusion of two diffusion models, i.e., Linear Threshold (LT) and Independent Cascade. TDF-CT handles the limitations of the contemporary diffusion models, i.e., IC and LT. The efficacy of proposed models is evaluated with respect to existing models Independent Cascade (IC), Time Constant Cascade (TC-C), Time Decay Cascade (TD-C), and Time-Depth Decay Cascade (TDD-C). Experimental evaluation over two benchmark datasets namely Darwin and MelCup17 reveal that proposed models are able to make the predictions very close to the real-time in a given time frame. TDF-CT and TDF-C are most suitable for applications requiring high precision and high recall, respectively. Results of spread shape establish the efficacy of models to spread the influence with good coverage of the social network. Results are obtained with improved accuracy by up to 39%.

1. Introduction

With the rapid growth of users on social network platforms, websites like Twitter, MySpace, and Facebook have become the primary medium for product promotion. The rising quantity of information spread through these portals is pushing the users to compete for being influential, i.e., social influence. Social influence is defined as a change in the activities of an individual due to another person's intentional or unintentional actions and activities (Gass, 2015). For Example, in viral marketing, organizations try to identify the most influential users of social network portals to promote their product. If these users like the product and endorse it in their social circle, it is expected that many of their friends will also purchase/opt the product and promote the same to their social-circle. This process of promotion continues, and a large number of people get awareness about the product. This process is called Social Influence Maximization. Hence, the objective of Influence Maximization (IM) is to discover the most influential users of the

* Corresponding author.
E-mail addresses: sakshi.officialid@gmail.com (S. Agarwal), mehtshikha@gmail.com (S. Mehta).

https://doi.org/10.1016/j.ipm.2020.102321
Received 29 January 2020; Received in revised form 1 May 2020; Accepted 2 June 2020
Available online 30 June 2020

system such that information propagation can be maximized. Influence maximization in social networks is a well-known problem (Kempe, Kleinberg & Tardos, 2015). IM is a two-step process. It begins with the initial set of users known as seeds and selection of the features that are used to study the influence of seed users over other users of the networks. The second step of IM is the estimation of information spread using a diffusion model that estimates the number of users influenced by the selected seed users. Thus, the effectiveness of the IM is based on three components: seed users, features set, and diffusion model.

There are various heuristics available for seed selection, such as random, high degree, single discount, general greedy (Agarwal & Mehta, 2018). Similarly, there are various features available to study influence from one user to another user. In literature, a variety of features have been applied to study influence between users on social networks such as user's related personal information, time-dependent information, location-based information, and user's social circle related information (Agarwal & Mehta, 2019). Therefore, these features of the social network are categorized into four major types: structural features, temporal features, profile features, and interaction features, as shown in Table I.

These features are used for estimating the link influence strength between the users. This link strength is subsequently used in the diffusion models to estimate the influence spread in online social networks (Aldous, An & Jansen, 2019; Noekhah, Binti Salim & Zakaria, 2020; Ramos, Boratto & Caleiro, 2019; Sun & Tang, 2011). Kempe et al. (2015) presented simple uniform influence probabilistic models Linear Threshold (LT) and Independent Cascade (IC) to estimate the influence diffusion. In their work, they have used only the profile feature (Degree) to estimate the link influence strength between the user pairs. Mei, Zhong & Yang (2015) learned the role of features in influence analysis. They discovered that not only the profile features such as the number of friends, the ratio of followers to friends; interaction features such as retweets or mentions are also useful to predict link influence strength. Mei et al. calculated the link influence strength according to users' action history, including tweet, favorite, mention/reply, and retweet (Mei, Zhao & Yang, 2017). They presented an R-J cascade model for the diffusion estimation that is an extension of the basic IC model. Similarly, Chen, Lakshmanan & Castillo (2013) presented the degree-weighted (profile feature) and action log-based (interaction feature) model to estimate influence diffusion. In their work, they developed a diffusion model for influence spread estimation that is based on the LT and IC. Jendoubi, Martin, Liétard, Hadji & Yaghlane (2017) measured the link influence strength using structural features and profile features, i.e., the importance of the user in the network structure and the popularity of users' tweets (messages), respectively. They compared their presented model with the LT, IC, and their variants such as IC with uniform edge probabilities and IC with trivalency edge probabilities (TV ICM). Zhang, Zhao, Yang, Paris & Nepal (2019) identified that time is an essential feature (temporal feature) of the social network in the influence diffusion. They used temporal features along with the interaction features in their presented diffusion models. They presented three extended versions of IC, i.e., Time Constant Cascade Model (TC-C), Time Decay Cascade Model (TD-C), and Time-Depth Decay Cascade Model (TDD-C). All presented models mainly focused on the diffusion estimation only. Their work oversight the estimation of link influence strength, i.e., estimation of the user's influence as an influencer and only used profile features to determine link influence strength. The interaction and temporal features were used for diffusion estimation in TDD-C and TD-C models, whereas structural features are overlooked by all the presented models. Overall, various limitations of existing models are as follows:

1 Independent Cascade (IC) and Linear Threshold Model (LT) play a vital role in the field of the diffusion for influence estimation and form the base for the various recent diffusion models (Bozorgi, Haghighi, Zahedi & Rezvani, 2016; Kempe et al., 2015; Saxena & Saxena, 2019). However, these two base models have the following assumptions, which are not appropriate with respect to real-world applications:
   a Influence is an immutable process. Once a user is converted to influenced state from non-influenced state, it cannot become non-influenced again;
   b Link influence strength is considered as a static probability such that probability is same for all timestamps and does not decrease with time.
   c IC model and LT model support one of the influence forms, i.e., Single strong influence or collaborative influence, respectively. If only one person influences a person, then it is known as the single strong influence. Similarly, if a person is influenced by the combined efforts of other people, it is called collective influence. For example, a person in the office can be either influenced by his group of associates (collective influence) or by his boss (single strong influence), but not by both forms at the same time. While, in a real-world scenario, a person can be influenced by both forms at the same time.
   d Both LT and IC models consider diffusion as a binary problem, i.e., a user is either influenced or non-influenced. It is based on the assumption that once a consumer adopts the product, he influences other non-adopters to adopt (or abandon) the product at all later times. However, Diffusion can also be defined as a multi-class problem as done in compartmental models (Kermack & McKendrick, 1927). These models are used to predict patterns of disease spread like susceptible-infected-recovered (SIR). Another model, Bass-SIR (Fibich, 2016) considers that diffusion is a three-class problem, i.e., **S**usceptible (who are non-influenced), **I**nfectious (influenced users), and **R**ecovered or neutral individuals. More often than not, but adopters become "*neutral*" from influencing other people after some time like a person who recovers from a disease and loses the power to infect others (Fibich, 2016). Therefore, there is no consideration of the neutral state by LT and IC models.
2 Secondly, Profile feature is the commonly used feature among other features and in some cases, their combinations (such as profile + interaction, temporal + profile, and profile + structural). However, consideration of all four features in the same frame for influence analysis is ignored so far.

Table I
Categorization of features in Social Network.

	Structural Features	Interaction Features	Profile Features	Temporal Features
Examples	• Members of a group • Users from the Same City • Set of Users who retweeted the same tweet	• A User tagged another user in a picture • A User Tagged his friend in the tweet • A User linked his friend's post	• Number of followers • Number of friends • Age of a User • Geographic Location	• Time when a user shared a picture • Time when a user commented/ retweeted on a tweet • Distribution of user activity across the day

1.1. Research objectives and contributions

Based on the research gaps, two main challenges or research questions, which need to be considered while detecting and estimating social influence, are:

RQ1: How to calculate the user's strength of influence as an influencer.
RQ2: How many users will be influenced by an influencer, and what is the time frame of the influence?

Contribution 1: To address RQ1, this work proposes two influence models; namely Time Decay Features Cascade Model (TDF-C) and Time Decay Features Cascade Threshold Model (TDF-CT). The first proposed model TDF-C focuses on the link strength estimation, which calculates dynamic influence probabilities between users using Modified Topical Affinity Propagation (M-TAP) (Agarwal & Mehta, 2018). M-TAP is an extension of the Topical Affinity Propagation (TAP) model (Tang, Sun, Wang & Yang, 2009). M-TAP extends the functionality of the TAP model by including both temporal feature and interaction features along-with the structural features, and profile features. Therefore, the proposed model uses profile features, structural features, temporal features, and interaction features to calculate the strength of influence from one user to another user, i.e., **Time Decay Features Cascade Model (TDF-C).** For estimation of the influence diffusion in the social network, TDF-C uses the IC model.

Contribution 2: To resolve RQ2, a hybrid approach is proposed, i.e., **Time Decay Features Cascade Threshold Model (TDF-CT)**. TDF-CT is the extension of TDF-C, which uses the fusion of the Linear Threshold model and the Independent Cascade model instead of only IC. Therefore, TDF-CT applies the hybrid approach that can handle the limitations of the existing influence diffusion models, as mentioned above. Thus, the proposed interaction and temporal features-based hybrid (LT + IC) diffusion model is proposed to determine the set of users, influenced by an influencer 'v' within 0-t timestamp. To the best of our knowledge, minimal work has been done in the literature, which models the characteristics of LT and IC in a single frame while handling their limitations.

The performance of proposed algorithms is evaluated against two real-world datasets, i.e., "Darwin" and "MelCup17" (Zhang et al., 2019) over metrics like spread shape, precision, recall, and accuracy with respect to contemporary models. The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents the notations used in this paper and explains the contemporary influence diffusion models in detail. In Section 4, the proposed methods for influence strength calculation and influence diffusion modeling are discussed. After this, the evaluation metrics are discussed in Section 5. The experimental results are given and analyzed in Section 6. The conclusion is given in Section 7.

2. Related work

Social networks are being used by millions of persons who share posts and messages on various topics. The information is analyzed to foretell or discover actions and events in the real world. In order to predict and maximize the diffusion behavior, it is essential to discover and measure the spread of the information appropriately in the social network (Baig, Shuib & Yadegaridehkordi, 2019). Numerous theories and computational models have been presented for various influence-based applications (Liu, Jin & Shen, 2019; Najafabadi, Mohamed & Onn, 2019; Tang, 2017; Tong, 2019; Wen, Peng & Shuai, 2018). A review of current state-of-art research for influence analysis on the social network is presented in Table II.

Alp et al. developed an Influence Factorization model for IM, which identifies topic-specific experts (Alp & Öğüdücü, 2019). Kuhnle, Alim, Li, Zhang & Thai (2018) studied the IM problem on social platforms that are linked together through common users. D'Angelo, Severini & Velaj (2019) and Tong, Wu, Tang & Du (2016) applied the profile feature of the social network to study the Influence Maximization problem using the Independent Cascade model. He, Han, Ji, Du & Li (2019) also used the profile features along with the IC model to maximize the influence spread while considering the positive and negative influence impact within the social network. Asim, Malik, Raza & Shahid (2019) presented the SNTrust model to find trusted/non-trusted users in a social network. Saito et al. handled the diffusion spread prediction with respect to complex networks (Saito, Nakano & Kimura, 2008). Their approach is based on the IC model that estimated the influence probabilities between the users using the Expectation-Maximization problem. The computational complexity of their method was high for large networks since influence probabilities were calculated in each iteration of the model. Zhu et al. developed a heuristic algorithm (IS-LSS & IS-LSS) based on a competitive Independent Cascade to examine the diffusion in social networks (Zhu et al., 2019). Xuan et al. (2019), Iñiguez, Ruan, Kaski, Kertész & Karsai (2018), and Liu, Qu, Chen, Hanjalic & Wang (2018) developed the Independent Cascade based models to study the influence diffusion in social

Table II
Review of current state-of-art for Influence Analysis.

Author Reference	Type of Features	Diffusion model	Evaluation Measures	Application	Advantages/Limitation
Xin et al., 2020 (Xin & Wu, 2020)	Profile, Temporal	Multi-feature SVM Model (MF-SVM)	Accuracy, Precision, Recall, F1-measure	Friend Recommendation	Useful for link strength calculation Only, Can be extended as a diffusion model
Fani et al., 2020 (Fani et al., 2020)	Structural, Temporal	–	Accuracy, Precision, Recall, F1-measure	User Community Detection	–
Tong et al., 2019 (Tong, 2019)	Profile	Independent Cascade	Diffusion size	Influence Maximization	The presented solution selects one seed each time which the author suggested to do in batch mode as future work
He et al., 2019 (He et al., 2019)	Profile	Independent Cascade	Diffusion Size	Minimum-sized Positive Influential User Selection	–
Zhang et al., 2019 (Zhang et al., 2019)	Profile, Interaction, Temporal	Independent Cascade, Time Constant Cascade Model (TC−C), Time Decay Cascade Model (TD-C), and Time-Depth Decay Cascade Model (TDD-C)	Diffusion Size, Diffusion Shape, Precision, Recall	Influence Diffusion	High precision & Recall
Alp et al., 2019 (Alp & Öğüdücü, 2019)	Profile	Influence Factorization and its variations	Diffusion Size	Influence Maximization	Able to identify topic-specific experts
Asim et al., 2019 (Asim et al., 2019)	Profile	SNTrust Model	Pearson's Correlation Analysis	Identification of influential users in Consulting Company, Blog Catalog, and Facebook	Datasets used in the experiments are small as compared to social network datasets.
Zhu et al., 2019 (Zhu et al., 2019)	Profile, Structural	Heuristic algorithm IS-LSS & IS-LSS+ based on competitive Independent Cascade	Running time of models	Influence Blocking Maximization, Product Promotion, Rumor Control	Influence spread by obtained results cannot be guaranteed
Kuhnle et al., 2018 (Kuhnle et al., 2018)	Profile	Linear Threshold, Independent Cascade, ISF	Diffusion Size	Multiplex Influence Maximization	High execution time
Xuan et al., 2019 (Xuan et al., 2019)	Profile, Interaction	Independent Cascade	Mathematical modeling	Influence Diffusion	Experiments are performed over synthetic data only
Lie et al., 2017 (Liu et al., 2018)	Temporal, Interaction	Independent Cascade	Diffusion Size, Diffusion Shape	Information Diffusion	–
Iñiguez et al., 2018 (Iñiguez et al., 2018)	Profile	Independent Cascade, Watts Model	Diffusion Size	Innovation Diffusion	–
D'Angelo et al., 2019 (D'Angelo et al., 2019)	Profile	Independent Cascade	Mathematical Modeling	Influence Maximization	No experimental analysis on real-world datasets
Tong et al., 2016 (Tong et al., 2016)	Profile	Independent Cascade	Diffusion Size	Influence Maximization	With a limited number of iterations of the model, influence spread cannot be guaranteed

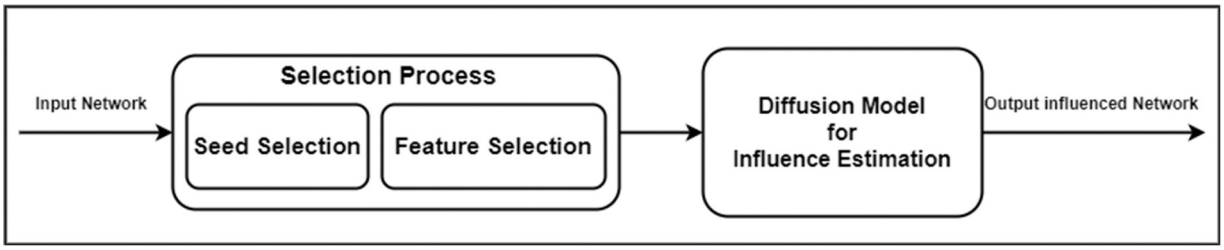

Fig. 1. Basic Influence Maximization Model.

Table III
List of Symbols.

NOTATIONS	DETAIL
S	Collection of the estimated link influence strength corresponding to each link
I	Symbolizes i^{th} user in V
y_i	User i's agent
e_{ij}	A directed link from user 'i' to another user 'j'
w_{ij}	link_weight of the link e_{ij}
S_{ij}	The score of the influence from user 'i' to user 'j'
T_{ij}	Probability of influence user 'j' thinks have on the user 'i'
R_{ij}	Influence probability user 'j' approves to receive for itself by user 'i'
NBS(i)	Collection of neighbors of user 'i'

networks. For the selection of users, they applied the various graph heuristics such as K-core, closeness centrality, eigenvector centrality, and page rank. Goyal, Bonchi & Lakshmanan (2010) presented a diffusion model that learns the edge influence probabilities through the action history log. Saxena & Kumar (2019) also used the action history log in their work. They presented two action log-based models, namely the Activity-based Independent Cascade model and Activity-based Linear Threshold model, based on IC and LT model, respectively. Both models selected the initial seed users using the UAC-Rank algorithm. Overall, they emphasized on the interactions between the users to analyze the diffusion spread in the social network. Fani et al. (2020) studied the role of features of the social network to identify and extract similar users in the social networks. They explored that contemporary community detection techniques apply either link analysis or content analysis and ignore the temporal analysis. In their work, they applied the multi-model embedding technique, which utilizes the temporal features of the social network. Shin, Jian, Driscoll & Bar (2018) examined the diffusion of information over Twitter using three components of the social network, i.e., content, temporal, and source of information. From their study, they found an interesting pattern about the information over the social network that the rumor visits the same users several times, whereas the real information does not. Hoang & Mothe (2018) presented a framework to predict whether a tweet will be retweeted or not and how far it will be gone in the network. In their model, they used the user-based, time-based, and content-based diffusion model and demonstrated the usability of the features through the experimental results. Xin & Wu (2020) studied the structure and content features of location-based social networks and applied these features to identify the strength of the link between two users. They studied the four different feature modeling: pure check-in data-based user modeling, geographical information-based user modeling, spatiotemporal information-based user modeling, and geo-social information-based user modeling.

From Table II, it can be observed that the majority of the researchers have utilized profile features only, and few have used temporal, structural, and interaction features. To the best of our knowledge, the strength of all features together is yet to be explored. Further, the Independent Cascade (IC) and the Linear Threshold (LT) model are the backbones of the diffusion process. Therefore, it is hypothesized that the hybridization of IC and LT models would help to estimate diffusion more efficiently.

3. Background: Preliminary concepts and diffusion models

The main objective of the Influence Maximization (IM) model (Fig. 1) is to discover the most influential users and measure the spread of the information by them appropriately such that information propagation can be maximized (Kempe et al., 2015). The process begins primarily with the selection of seed users and identification of features to be used in order to study the influence of seeds users over other users of the networks. Subsequently, information spread is estimated using the diffusion model to compute the number of users influenced by the initial seed users. Thus, the efficacy of an IM model depends on the quality of seed users, features set, and diffusion model.

This work focuses on two components of the IM:- first is features of the social network, which are used to compute the link influence strength between users, and second is the diffusion model, which calculates how many users are influenced by the set of seed users. In this work, seed users are selected randomly from the network. These selected seed users are fixed input for each model, which is used in this work to maintain the impartiality and non-discriminatory of the experiments. Further, this section briefly

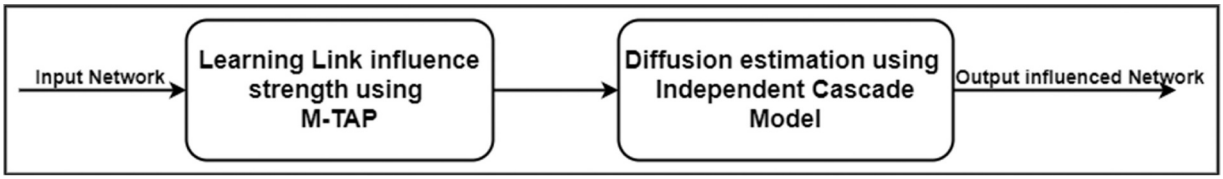

Fig. 2. Process flow of the proposed TDF-C model.

discusses the preliminary concepts of influence diffusion and existing diffusion models.

3.1. Preliminary concepts

The work presented two diffusion models TDF-C, and TDF-CT for effective influence estimation in social networks G. The Following are the key definitions of the presented work.

Definition 1. *G is defined by a tuple (V, E), where V represents the set of users in G, E is the set of all links (u,v),* i.e., $u,v \in V$. *The total number of users and the total number of links in the network are represented by* $N_v = |V|$ *and* $N_e = |E|$, *respectively.*

Definition 2. *Action* log *history (A) is the set of actions performed by each user in the previous timestamps.*

Definition 3. *Feature_set (F) is the set of feature vectors corresponding to each user in G.* f_a *is the feature vector* (f_{a1}, f_{a2}, f_{a3}, …, f_{an}) *having n different feature values corresponding to user 'a' in the network.*

Some other notations used in this paper are summarized in Table III.

3.2. Diffusion models

Various diffusion models used for influence estimation in the literature are based on one of the oldest and core models, i.e., Independent Cascade (IC) (Peng et al., 2018). Few recent extensions have been amended in IC, i.e., Time Constant Cascade Model, Time Decay Cascade Model, and Time-Depth Decay Cascade Model (Zhang et al., 2019).

Independent Cascade Model (IC): This model is based on the two parameters, i.e., influence threshold θ and link influence strength $P(i, j)_{IC}$. $P(i, j)_{IC}$ of link e_{ij} denotes how much influence coming from user 'i' to user 'j'. A user 'j' is moved from non-influenced to influenced category only when the influence strength from the influenced neighbor is higher than the value of θ. The threshold θ is an independent and identically distributed uniform random variable on [0, 1].

Time Constant Cascade Model (TC-C): Functioning of the TC-C model is independent of time. The link strength probability $P_t(i, j)_{TC-C}$ of a link e_{ij} does not rely on the time t which implies that at every timestamp t ($t \leq t + 1$) after user 'i' takes action, $P_{t+1}(i, j)_{TC-C}$ remains the same as $P_t(i, j)_{TC-C}$.

Time Decay Cascade Model (TD-C): TD-C model is based on the concept of time decay. In social network applications such as twitter, the interaction occurs within a small period, and the number of interactions decays quickly. Therefore, the number of interactions is analyzed against the time to discover at what time user 'i' influence user 'j', i.e., $P_n(i, j)_{TD}$. In this paper, the interaction period is divided into 60 slots. Each slot is of 5 min for time t (0 to 300 mins).

Time-Depth Decay Cascade Model (TDD-C): TDD-C model is based on the concept of the depth decay (Zhang et al., 2019). The link influence strength $P_n(j, k)_{TDD}$ depends upon the shortest path length from user 'i' to user 'j', i.e., D(i,j). Where user 'i' is the root user who influenced the user 'j'. Let root user 'i' influenced the user 'j' after n time intervals then $P_n(j, k)_{TDD}$ calculates the link strength probability from user 'j' to its neighbor 'k' using the following equation:

$$P_n(j, k)_{TDD} = \delta^{D(i,j)} P_n(j, k)_{TD} \quad (1)$$

Where δ is a decay factor, ranging between [0, 1].

4. Proposed models

The first proposed model, namely Time Decay Features Cascade Model (TDF-C), is a diffusion model that learns link influence strength between users using the M-TAP algorithm and estimates the diffusion through the IC model. The second proposed model is Time Decay Features Cascade Threshold model (TDF-CT), which also uses the M-TAP model to learn link influence strengths like TDF-C used. Except that TDF-CT estimates the diffusion through the Progressive Feedback Estimation (PFE) model instead of IC model. The PFE model is a fusion of two diffusion models, i.e., Linear Threshold (LT) and Independent Cascade. The process flow of the TDF-C and TDF-CT are shown in Fig. 2 and Fig. 3, respectively.

Both models are divided into two components: (i) Learning link Influence Strengths; (ii) diffusion model. The detailed description of both components is as follow:

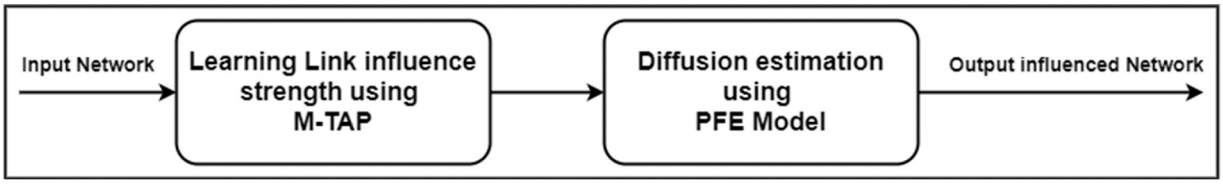

Fig. 3. Process flow of the proposed TDF-CT model.

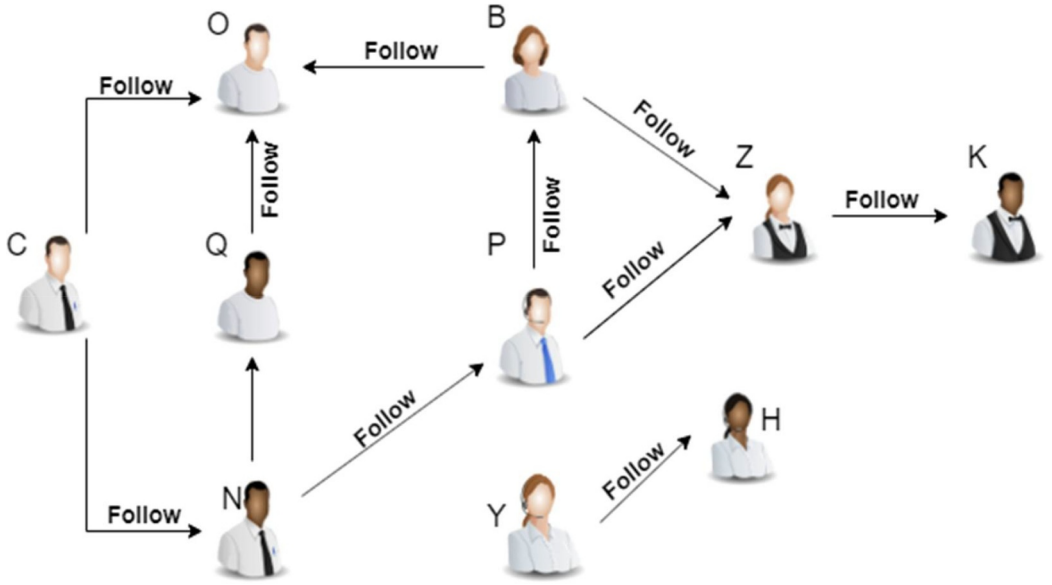

Fig. 4. Example follower network G.

Table IV
Sample Action History Log (A) for network G.

	Interaction Features			Temporal Features
Action ID	Action Taken By	Action Taken On	Action Type	Time-stamp
1	O	–	POST	t_a
2	B	O	LIKE	t_{a+1}
3	C	O	RETWEETED	t_{a+2}
4	P	O	QUOTE	t_{a+3}
5	Q	O	REPLY	t_{a+2}
6	N	P	POST The Similar Content	t_{a+4}
7	B	–	POST the different opinion than A	t_h
8	P	B	LIKE	t_{h+2}
9	N	B	REPLY	t_{h+3}
10	C	C	QUOTE THE PREVIOUS POST	t_d
….	…	…	…	…
….	…	…	…	…

4.1. Learning link influence strengths using m-tap

The first phase of the TDF-C and TDF-CT is link influence strength estimation. Graph $G = \{V, E\}$, Feature_set(F), and the action history log(A) are the inputs for the algorithm M-TAP. The structure of these inputs is described through an example. Fig. 4 shows the follower network G of the example, where $V = \{O, B, C, P, Q, Z, K, H, N, Y\}$ represents the users in the network and $E = \{e_{BO}, e_{QO}, e_{CO}, e_{PB}, e_{BZ}, e_{PZ}, e_{ZK}, e_{CN}, e_{NQ}, e_{NP}, e_{YH}\}$ represents the follow edges.

Similarly, Tables IV and V represent the action history log (A) and feature_set (F) for network G, respectively. Action log history comprises the set of actions performed by the users in the previous timestamps is represented as A. Each tuple in action history log consists of four components: 'action taken by', 'action taken on', 'action type' and 'time of action'.

Feature_set is the set of feature vectors with respect to each user in the network that represented as F. Feature_set F is composed of

Table V
Sample feature_set (F) of network G.

	Profile Features				Structural Features		
User ID	Number of followers	Location of the user	…	…	Count & Set of other users reacted to the action of the user	…	…
O	3 (B, C, Q)	Abc	…	…	4 (B, C, P, Q)	…	…
B	1 (P)	Abc	…	…	2 (P, N)	…	…
C	0	Xyz	…	…	1 (N)	…	…
P	1 (N)	Abc	…	…	1 (N)	…	…
Q	…	…	…	…	…	…	…
….	…	…	…	…	…	…	…

two types of features: profile features and structural features, as shown in Table V. In the example, two profile features are used, i.e., 'number of followers' and 'location of the user'. Similarly, 'count and the set of the users reacted' is the structural feature that is used in the example.

Primarily, the algorithm takes feature_set F of the network G, and action history logs A as input. Subsequently, it computes link_weight for each link e_{ij} in network G using the Jaccard Index as defined by Eq. (2). Jaccard Index determines the similarity between two sets, i.e., the fraction of the sizes of the intersection of the sets to the union of the sets. The mathematical representation of the link_weight of link e_{ij} is as follows:

$$\text{link\_weight}\ (i, j) = \frac{|\ f_i \cap f_j|}{|f_i| + |f_j| - |f_i \cap f_j|} \tag{2}$$

Where the link_weight(i,j) of each directed link is the Jaccard index and calculated with respect to each profile and structural features of the feature vector f_i and feature vector f_j such as no. of followers, set of users re-tweeted on the same tweet.

After computing the link_weight of each link, the dominant_user function g(i, y_i) is calculated using eq. 3

$$g(i,\ y_i) = \begin{cases} \frac{w_{iy_i}}{\sum_{j \in NBS(i)} (w_{ij} + w_{ji})} y_i \neq i \\ \frac{\sum_{j \in NBS(i)} (w_{ji})}{\sum_{j \in NBS(i)} (w_{ij} + w_{ji})} y_i = i \end{cases} \tag{3}$$

Where, y_i represents the user agent for user 'i' such that y_i {NBS(i)∪i} with highest link_total out of all features i.e.

$$\text{Link\_Total}(i)_f = \sum_{k \in\ NBS(i)} w_{ik} \tag{4}$$

Using the dominant_user function g, the L_{ij} is calculated. L_{ij} is the logarithm of normalized combined dominant_user function for a link e_{ij} defined by Eq. (5).

$$L_{ij} = \log \frac{g(i,\ y_i)|_{y_i=j}}{\sum_{k \in NBS(i) \cup \{i\}} g(i,\ \ y_i)|_{y_i=k}} \tag{5}$$

The value of L_{ij} is used to compute the values of R_{ij}, T_{jj}, $T_{ij,}$ and S_{ij} using Eqs. (6)Eqn 7Eqn 8, and –(9), respectively.

$$R_{ij} = L_{ij} - \max_{k \in NBS(j)} \{L_{ik} + T_{ik}\} \tag{6}$$

$$T_{jj} = \max_{k \in NBS(j)} \min\{R_{kj}, 0\} \tag{7}$$

$$T_{ij} = \min\Big(\max\{A_{jj}, 0\} - \min\{R_{jj}, 0\} - \max_{k \in NBS(j) \setminus \{i\}} \min\{R_{kj}, 0\}\Big),\ \ i \in NBS(j) \tag{8}$$

$$S_{ij} = \frac{1}{1 + \ e^{-(A_{ji} + T_{ji})}} \tag{9}$$

The value of S_{ij} for each link e_{ij}, refers to the estimated link influence strength corresponding to each link in network G.

Using Algorithm 1 (M-TAP), the influence network $G^1(V, E, S^0, S^1)$ is generated by calculating two values with respect to each link of the network, i.e., S^0 and S^1. Where S^0 is the set of link influence strengths and S^1 is the set of link non-influencing strengths.

Fig. 5 shows the generated Influence Network G^1 using the structural, temporal, profile, and interaction features of network G. In the proposed models, the link influence strengths between any link e_{ij} is not only dependent upon the direct neighbors. As shown in Fig. 5 by the dotted lines, M-TAP also incorporates the amount of influence coming from their indirect peer-2-peer connections. $S^0_{(O,\ P)}$ represents the indirect link influence strength from user 'O' to user 'P', i.e., indirectly moving through the path O → B → P. Similarly, $S^1_{(O,\ N)}$ represents the indirect link non-influence strength from user 'O' to user 'N', i.e., indirectly moving through the paths O → P → N and O → Q → N. Therefore, the proposed model also considers the influences coming from non-neighboring users.

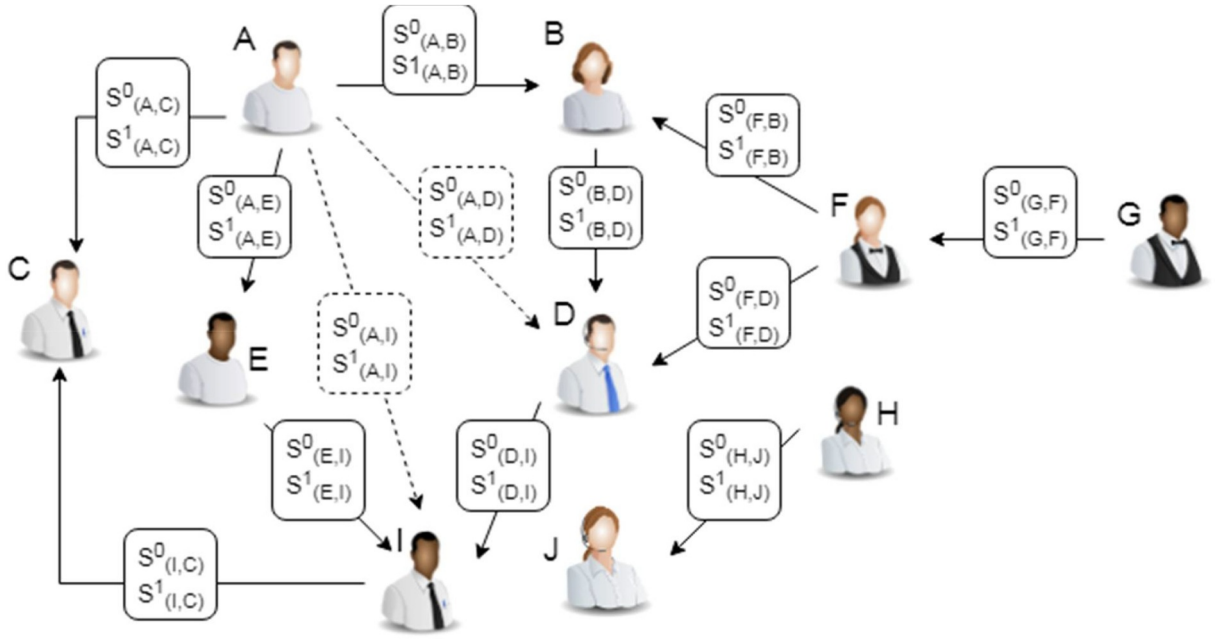

Fig. 5. Influence Network G^1 using the structural, temporal, profile, and interaction features of network G.

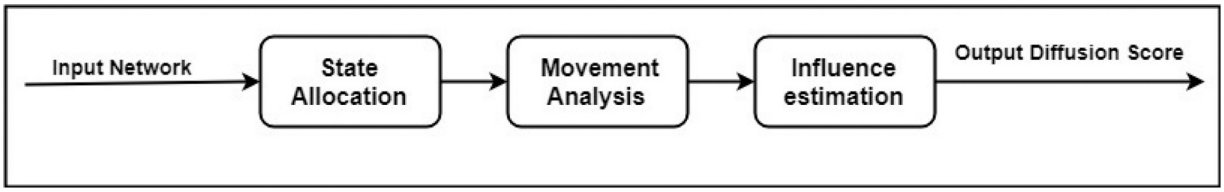

Fig. 6. Process flow of the PFE model.

4.2. Diffusion estimation using progressive feedback estimation (PFE) model

The second phase of the TDF-CT is to estimate diffusion using the PFE Model. PFE model is a hybrid model that combines the LT and IC as a single unit to estimate the diffusion spread in the network. In this paper, PFE considers the action history log A and network $G^1(V, E, S^0, S^1)$, which computed in the first phase using M-TAP as an input. The stepwise process of PFE is shown in Fig. 6.

PFE is a three-step process: first is state allocation, second is movement analysis, and third is diffusion estimation. The detailed explanation of all these steps is as follow:

4.2.1. State allocation

State allocation is the first step of the PFE that represents the current state of each user 'i' of the network G^1. A user's state is categorized into three classes: influenced, non-influenced, or neutral. The 'neutral' state represents the users who recover from this diffusion process and lose the potential to influence/non-influenced the others. For example, an initiating author creates a research group and begins a research discussion on a topic by posting a thread on her wall; the author becomes influenced. The users who have a certain level of interest in a topic will read the post, comment on the thread, and join the group. New joiners and readers will post other threads, thereby influencing others to join the group by their posts. After a certain period, some authors may lose interest and stop participating in discussions. Such users who lose the potential to influence others are considered as neutral users. Therefore, the state of users may change from one to another at any time in the network. In the algorithm, State(i,t) computes the current state of the user 'i' after t intervals using the following equation:

$$\text{State}(i,\ t) = \begin{cases} \text{influenced} & \text{if,user`i'isgettingenoughinfluencelinkstrengthaftertimet} \\ \text{non}-\text{influenced} & \text{if, user`i'isgettingenoughnon} - \text{influencelinkstrengthaftertimet} \\ \text{neutral} & \text{if,user`i'losesthepowertoinfluence/non}-\text{influencedtheothersaftertime} \end{cases} \tag{10}$$

Following are the examples from Table IV to explain how this State Allocation happens:

Initially, User 'O' posted something 'x' on his wall at time t_a. Therefore, only User 'O' has the State(O, t_a) = "influenced", and the rest of the users are in the "neutral" state. In the next timestamp t_{a+1}, when the user 'B' liked the post 'x', the State(B, t_{a+1}) will be

changed to “influenced” from “neutral”. Similarly, till the time t_{a+2}, the value of State(P, t_{a+2}) remains “neutral”. Since there is no reaction from the user ‘P’. However, at time t_{a+3}, State(P, t_{a+3}) changed to “influenced” as user ‘P’ quote the post of user ‘O’.

Similarly, when user ‘B’ post something ‘y’ against the post ‘x’ at time t_h. The state of the user ‘B’ changed to “non-influenced” with respect to action ‘x’. Thus, at time t_{h+1}, user ‘P’ gets both types of influence, i.e., to get influence from user ‘O’ and to not get influenced by user ‘B’. Such that, $t_a < t_h$. Therefore, the State(P, t_{h+1}) changed to “neutral”, since user ‘P’ might lose interest in the topic due to both views. The state of the user ‘P’ again changed to “influenced”, afterward when the user ‘P’ like the post ‘y’.

4.2.2. Movement analysis

A user in the network has been influenced either by a single user (according to the IC model), by the group of users (according to the LT model), by both ways (IC & LT), or not yet been attempted by any user to influence as per the action log history (NONE). Therefore, the actions of the user ‘i’ in the time interval of [0-t] are analyzed using the action history log to identify the influence pattern using function M(i,t), i.e., movement analysis.

$$M(i,t) = \begin{cases} IC & \text{influenced by IC only,} \\ LT & \text{influenced by LT only,} \\ \text{Can't Say} & \text{BOTH/NONE} \end{cases} \tag{11}$$

Hence, PFE hybridizes the functionality of both the IC and LT models. Following are the examples from Table IV to explain how the Movement Analysis works:

Initially, user ‘O’ posted something ‘x’ on his wall at time t_a. Therefore, only User ‘O’ can influence its neighbors. In the next timestamp t_{a+1}, when the user ‘B’ liked the post ‘x’, user ‘B’ was getting influence from the single user. Therefore, the movement of user ‘B’, i.e., M(B, t_{a+1}) is “IC”.

At timestamp t_{a+4}, when the user ‘N’ posts the similar content as post ‘x’. User ‘N’ was getting influence from the users ‘P’ and user ‘Q’. It can be observed that collective efforts motivated the user ‘N’ to post similar content as ‘x’. Therefore, the movement of user ‘N’, i.e., M(N, t_{a+4}) is “LT”.

For the users such as ‘Z’, when no history log is available, or both “LT” and “IC” type log is available. The movement is considered as “Can't Say”, i.e., M(Z, t_{a+4}) is “Can't Say”.

4.2.3. Diffusion estimation

The last step of the PFE model is to estimate the influence diffusion such that the number of users influenced by any post ‘x’ within the period [t_a, t_{a+n}] since user ‘i’ shared the post ‘x’. [Refer to Table IV and Fig. 5] Initially, user ‘O’ posted something ‘x’ on his wall at time t_a. Therefore, only the user ‘O’ is added to the diffusion set D at the beginning of the diffusion process. After this, the diffusion effect is checked with respect to each neighbor ‘j’ such that $j \in NBS(O)$ OR {B, C, Q}. The diffusion effect is analyzed based on the movement analysis value. If M(B, t_{a+1}) is “IC”, then link influence strength from each influenced neighbor is checked individually such that a single neighbor can influence user ‘B’ or not. Similarly, if M(B, t_{a+1}) is “LT”, then the link influence strength of the all influenced neighbors is checked collectively. If M(B, t_{a+1}) is “Can't Say”, diffusion is analyzed by both models sequentially, i.e., LT and IC. The diffusion analysis is done with respect to network G^1, and the state and movement of every user are recalculated at each timestamp. If the State(B, t_{a+1}) of the user ‘B’ is “influenced”, then user ‘B’ is added to the diffusion set D. Similarly if the State(B, t_{a+1}) of the user ‘B’ is “non-influenced”, user ‘B’ is removed from the diffusion set D. This process of diffusion estimation is continued till no new update is observed from timestamp t_n to t_n+1. Finally, the diffusion score, i.e., Dscore(O) is computed. Algorithm 2 depicts the PFE in detail.

5. Evaluative parameters

The performance of the proposed models is analyzed with respect to three significant characteristics, i.e.

- Maximization of true predictions
- Minimization of false predictions
- Exploration of the complete search space

Thus the various parameters used to evaluate the proposed models based on the predicted diffusion pattern of the influence are:

- **Predicting size:** The total number of users gets influenced in the diffusion process is called the influence spread size (Zhang et al., 2019). The difference between the actual influence spread size **$Dscore_A(i)$** and predicted influence spread size **Dscore(i)** is called the normalize Mean Squared Error (NMSE) of the prediction size.

$$NMSE = \frac{1}{|V|} * \frac{(Dscore(i) - Dscore_A(i))^2}{Dscore(i)} \tag{12}$$

- **Accuracy, Precision, Recall, and F_1 Score** (Alkhodair, Ding, Fung & Liu, 2020): To measure the quality of the diffusion model accuracy of the results are calculated, i.e., how close the predicted influenced network to the ground truth as follows:

- Accuracy is the ratio of the correctly predicted influenced users to the complete set of users in the network.
- Precision is the ratio of the correctly predicted influenced users to all predicted influenced users in the results.
- Recall is the ratio of the correctly predicted influenced users to all who are influenced users in reality.
- The F_1 Score is the harmonic mean of precision and recall.

Information diffusion may or may not involve the cost factor. Therefore, precision and recall are used to evaluate the suitability of the diffusion model with respect to different applications. Precision tends to involve direct costs and Recall, tends to involve opportunity costs. Precision is more appropriate for applications that require less False Positives in trade-off to more False Negatives. Meaning, getting a False Positive is very costly as compared to False Negative.

- **Predicting Shape:** To study the shape of the influence cascade, the influence spread pattern is traced within the network. The spread pattern illustrates the false prediction spread and the true prediction spread with respect to each diffusion model. This influence spread analysis determines the reach of the influence in the network, i.e., whether influence spread reached to every corner of the network or not. For the influence cascade shape, the depth distribution of the influenced users is plotted in the influenced graph. The depth of the influence in the network is represented through the shortest path length from the seed user to influenced user and non-influenced user both.

6. Experiments and analysis

Meticulous experiments were performed in order to evaluate the performance of the proposed model for influence diffusion with respect to conventional diffusion models,[1] i.e., IC, Time Constant Cascade Model, Time Decay Cascade Model, and Time-Depth Decay Cascade Model.

6.1. Datasets

There are only two open datasets[2] that are suitable for this work, as only these datasets have included rich interactions, i.e., “Darwin” and “MelCup17” (Zhang et al., 2019). Both datasets are from Twitter: one is based on users’ location, and the other one is based on an event. Each dataset is collected into two phases: First phase data is used for training, and second phase data is used for testing purposes. Table VI summarized the statistics of the action and interaction components of the datasets, and Table VII summarized the statistics of the user networks generated from the datasets. These datasets include “tweet”, “retweet”, “reply”, and “quote”, which are named as “statuses”.

“Darwin” is a location-based dataset in which total 1265 users data is collected whose location was “Darwin, Northern Territory, Australia” from date 16/11/2017 to 27/12/2017. In this period, total 69,323 statuses were collected including 19,164 tweets, 60,481 “likes” of these statuses and 129,804 messages/actions posted/taken by 1265 users. Total 6793 (excluding self-interactions) interactions among 437 users are captured, which creates an interaction network of 2080 links (as shown in Table VII). Total 3880 users following information are also captured that create the following network of 58,503 links. Using the interaction network and following network, an influence network is created, which includes 3883 users and 58,799 edges. Similarly, a dataset of the same users from period 30/12/2017 to 26/01/2018 is used as the testing data. Total 47,749 statuses were collected including 13,656 tweets, 40,550 “likes” of these statuses and 88,229 messages/actions posted/taken by 1130 users. Total 3127 (excluding self-interactions) interactions among 375 users are available in the dataset.

Similarly, “MelCup17” is an event-based dataset from the period 06/11/2017 to 08/11/2017 covering three days around the 2017 Melbourne Cup Day. The results of the Melbourne Cup came out on 07/11/2017. Data before the results are used in training, and the data produced after the results of the Melbourne Cup are used for the testing. In this data, each user has at least six statuses. Details of the “MelCup17” dataset are given in Table VI and Table VII.

6.2. Experimental setup

In this work, the value of time T is ranging from 0 to 300 min with the 3 min time interval. Parameters that are used to implement TC-C, TD-C, and TDD-C are set as follows: β is fixed as 0.5, p1 is fixed as 0.001, and the δ is 0.15 and 0.13 for the “Darwin” & “MelCup17” datasets, respectively. Testing data contained users with more than two entries in action history logs and executed each model up to 50 iterations. Both datasets are balanced such that the number of influenced users is higher than the number of non-influenced users in both datasets, i.e., “Darwin” & “MelCup17”.

6.3. Analysis of results

The performance of the models is evaluated on four parameters over two open twitter datasets as follows:

[1] All the approaches used in this paper are implemented in Python version 2.7 with hardware configuration as macOS operating system with 8GB RAM and 1.8GHz Intel Core i5 processor.

[2] Datasets are available at https://bit.ly/2UlQ3xW.

Table VI
Description of the Action and Interaction Dataset.

Dataset		Action data				Interaction data	
		Status	Likes	Actions	Users	Interaction	Users
Darwin	Training	69,323	60,481	129,804	1265	6793	437
	Testing	47,749	40,550	88,229	1130	3127	375
MelCup17	Training	12,977	11,024	24,001	1484	12,894	1391
	testing	10,567	9266	19,833	1374	7670	1132

Table VII
User Link Statistics of Dataset.

Dataset	Following n/w		Interaction n/w		Influence n/w	
	Users	Links	Users	Links	Users	Links
Darwin	3880	58,503	437	2080	3883	58,799
MelCup17	1484	70,756	1391	7403	1484	72,710

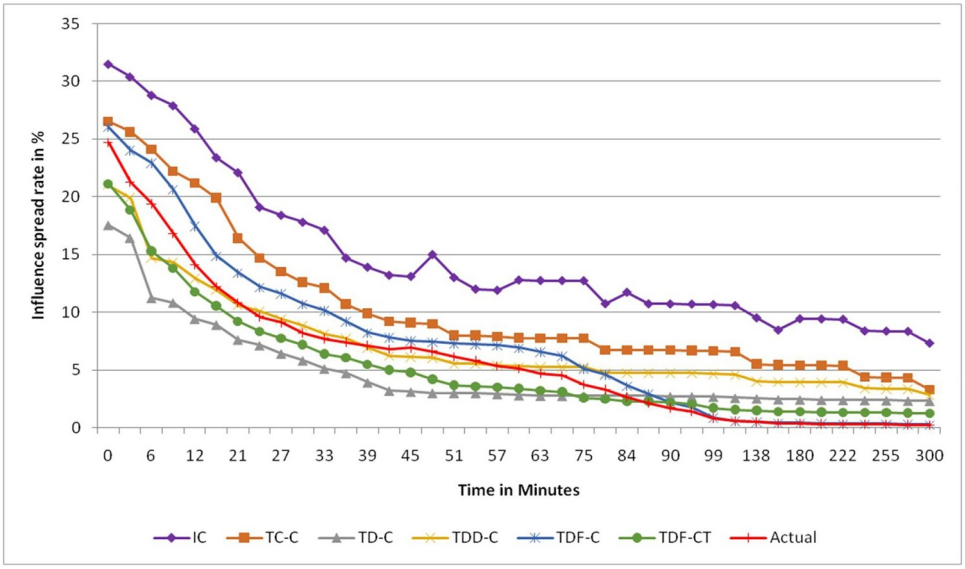

Fig. 7. Comparison of predicted size for Darwin Dataset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

6.3.1. Predicting size

The first evaluation parameter is the prediction of the cascade size **Dscore(i)**. The number of users, a user 'i' influences when initiating an action, is the most straight-forward measure of how well the predictions fit what actually happened. Figs. 7 and 8 show the experimental results of diffusion models for the "Darwin" dataset and the "Melcup17" dataset, respectively. For example, a product p is initially promoted to the set of k users (seeds) at time t, and these seeds influenced the N other people in the network to use the product P by time $t + h$ (actual spread). The influence spread estimated by a diffusion model is considered as the optimal influence spread, if the predicted spread is not very less or more than the actual spread. Hence, the performance of the models is compared using the influence spread rate.

As shown in Fig. 7, the actual influence spreading rate of the newly-influenced user in the actual scenario is decreasing with respect to time that is represented by the red color line with a plus sign. It is clear from Fig. 7 that for the first 60 timestamps, the actual influence spread rate is more than 5%. After this, it is continuously decreasing, and nearly 0% influence spread rate is noticed after the 140th timestamps. Following observations are derived from the experimental results (shown in Fig. 7):

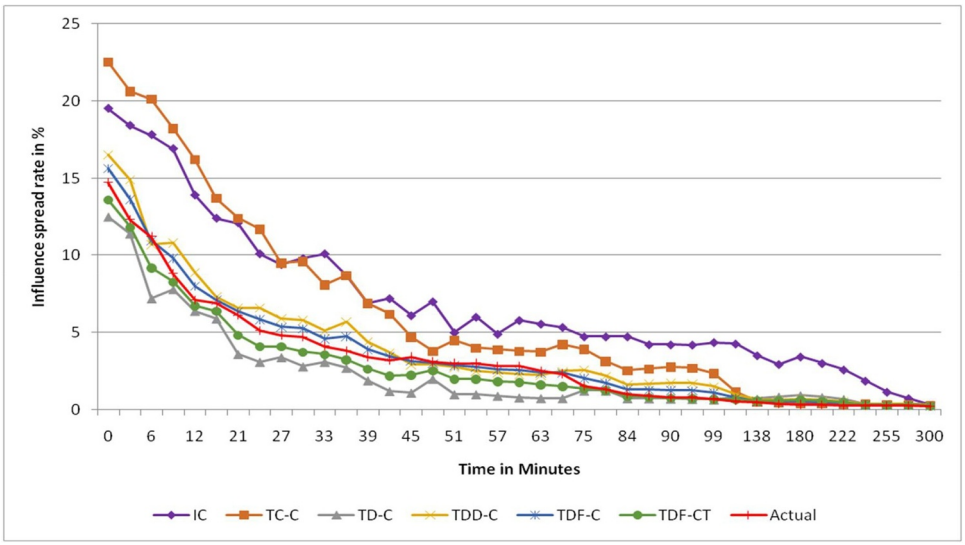

Fig. 8. Comparison of predicted size for Melcup17 Dataset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

- IC model predicted approximately 10% of the higher influence spread rate after the 160th timestamp when the actual influence spread rate is approximately 0%.
- TDD-C and TC-C also displayed a high influence rate at the cost of false predictions.
- TDF-C depicts the higher influence spread rate as compared to the actual influence spread rate in the initial timestamps. After the 95th timestamp, the influence spread rate predicted by the TDF-C is approximately equal to the actual influence spread rate.
- TD-C and TDF-CT showed the lower influence spread rate as compared to the actual influence spread rate that may lead to less number of false predictions. However, it also increases the probability of True-Negatives.

Similarly, it is clear from Fig. 8 that for the first 30 timestamps, the actual influence spread rate is more than 5%. After this, it is continuously decreasing, and nearly 0% influence spread rate is noticed after the 130th timestamps. Similar, observations are observed from the experimental results of Melcup17 dataset (shown in Fig. 8):

- The IC and TC-C model's influence spread rate is higher as compared to other models, i.e., the high false predictions.
- TD-C depicts the lower influence spread rate as compared to the actual influence spread rate.
- TDF-C and TDF-CT model's influence spread rate is the closest to the actual influence spread rate as compared to the other models.

Overall, TDF-C and TDF-CT demonstrated the optimal influence spread rate as compared to other models for both datasets. After analyzing the influence spread rate observed by the models. Next, Table VIII shows the normalize Mean Squared Error (NMSE) of the prediction size for each model with respect to both datasets.

Table VIII
Approximation error with respect to the actual influenced size.

				Normalize Mean Squared Error (NMSE) in percentage
Name of the Dataset	Melcup17 Dataset	***Name of the models***	IC	22
			TC—C	18
			TD-C	19
			TDD-C	12
			TDF-C	4.5
			TDF-CT	5
	Darwin Dataset	***Name of the models***	IC	28
			TC—C	23
			TD-C	14
			TDD-C	18
			TDF-C	5
			TDF-CT	4

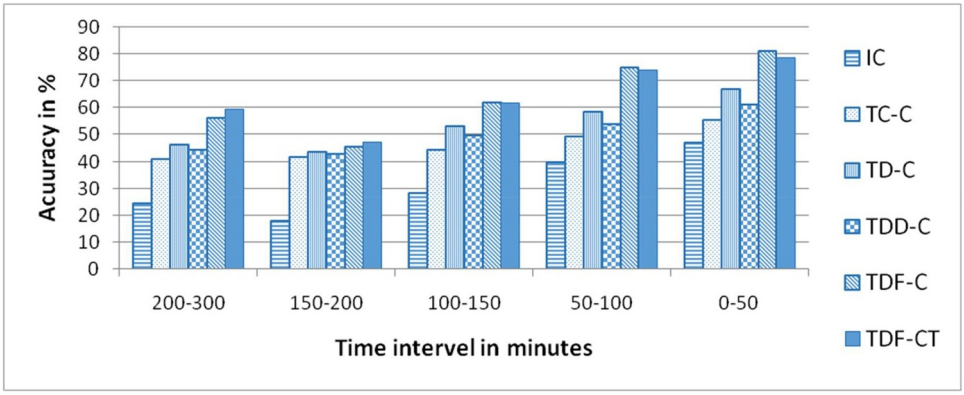

Fig. 9. Accuracy analysis of diffusion models for Darwin Dataset.

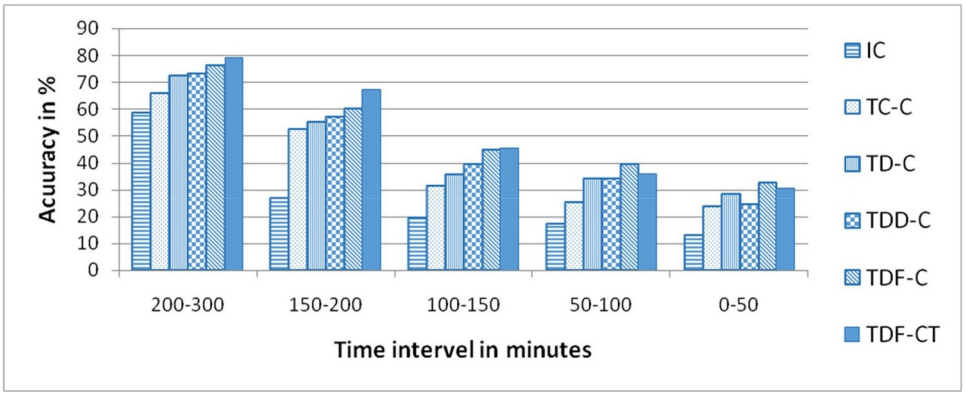

Fig. 10. Accuracy analysis of diffusion models for Melcup17 Dataset.

It can be observed from Table VIII that the prediction size of the proposed models TDF-C and TDF-CT are optimal as compared to the other diffusion models. Similar trends have been observed in Fig. 7, and Fig. 8 that the influence spread rate of TDF-C and TDF-CT are close to actual influence spread rate. Since the influence spread rate and size cannot ensure the goodness of the diffusion model alone as the number of predictions does not ensure the correctness of the model always. Therefore, the accuracy of the predictions is analyzed with respect to each diffusion model.

6.3.2. *Accuracy, precision, recall, and F_1 score*

Fig. 9 and Fig. 10 illustrated the accuracy results for the "Darwin" dataset and the "Melcup17" dataset, respectively. Table IX shows the accuracy improvement in percentage achieved by the proposed models as compared to the contemporary models, i.e., how much higher accuracy is obtained by the proposed models. It is clear from Table IX that proposed models TDF-C and TDF-CT

Table IX
Accuracy Improvement in percentage.

				Proposed Models									
				TDF–C					TDF–CT				
				The time interval in Minutes					*The time interval in Minutes*				
				201–300	151–200	101–150	51–100	0–50	201–300	151–200	101–150	51–100	0–50
Name of the Dataset	Darwin Dataset	***Name of the contemporary models***	IC	31.6	27.8	**33.9**	31.3	29.2	**34.7**	29.3	33.5	28.1	31.4
			TC−C	15.4	14.1	**27.6**	22.8	21.6	18.5	**22.8**	17.2	21.6	22.8
			TD-C	9.7	11.9	10.9	**16.6**	14.1	12.8	**15.4**	8.6	10.4	11.3
			TDD-C	11.6	13	12.4	21	**18.9**	14.7	14.5	12	**19.8**	17.1
	Melcup17 Dataset	***Name of the contemporary models***	IC	17.7	**33.4**	25.4	21.9	19.7	20.8	**38.5**	26.1	18.7	17.7
			TC−C	10.5	12.6	**14**	13.7	8.7	13.6	**14.7**	14.4	10.8	6.7
			TD-C	9.9	12.1	**10.2**	15.1	14.2	12	**12.2**	9.9	11.9	12.2
			TDD-C	9.2	10.1	9.5	10.4	**10**	6.3	**10.2**	6.2	8.2	6

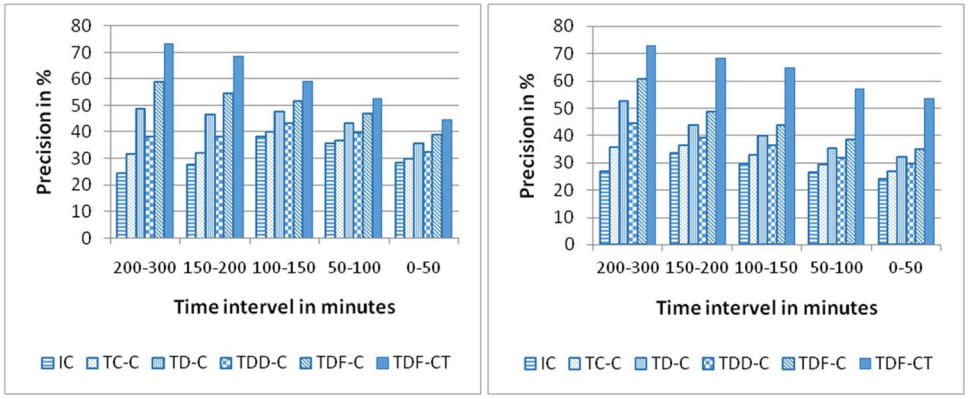

Fig. 11. Precision analysis of diffusion models.

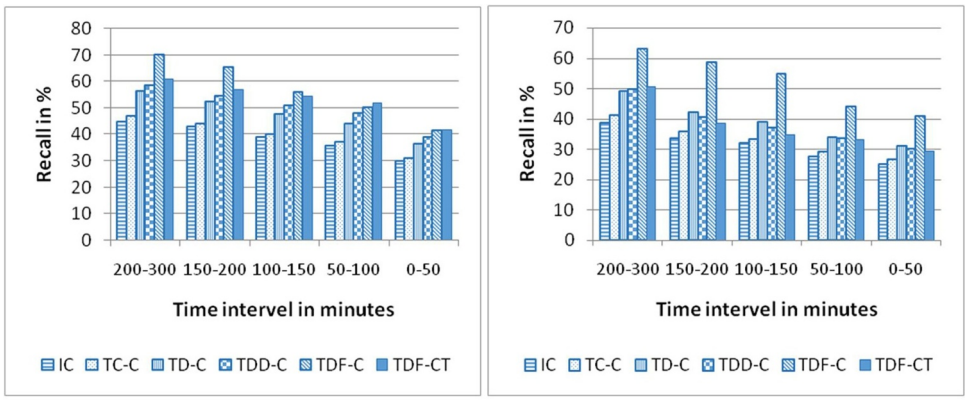

Fig. 12. Recall analysis of diffusion models.

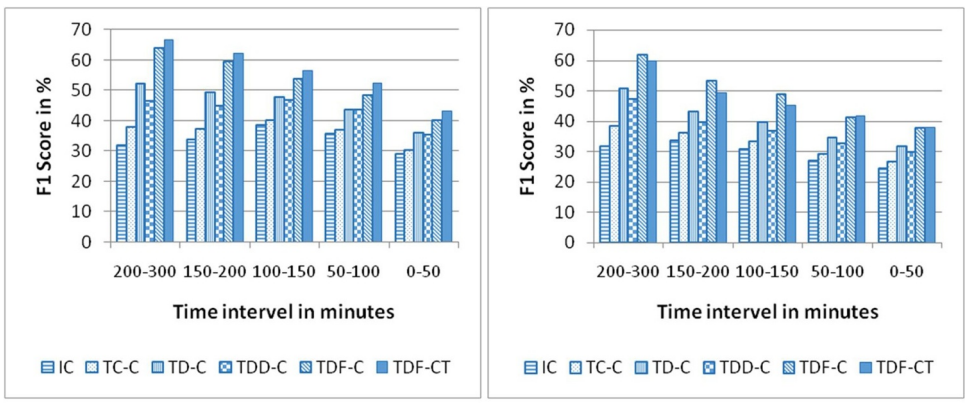

Fig. 13. F_1 Score analysis of diffusion models.

performed better than the other models and improved the accuracy up-to 34% and 39%, respectively. However, the performance of TDF-CT and TDF-C varies with respect to the time interval. It is observed from the accuracy results that in the first half of the time interval ($<=150$) TDF-C performed well. Whereas, in the latter half (>150), TDF-CT performed slightly better than the TDF-C. This behavior is common with respect to both datasets, as shown in Figs. 9 and 10. TDF-C and TDF-CT both performed well as compared to each other at different time intervals.

The datasets that are used in this paper are balanced. Therefore, accuracy is sufficient to analyze the performance of the models

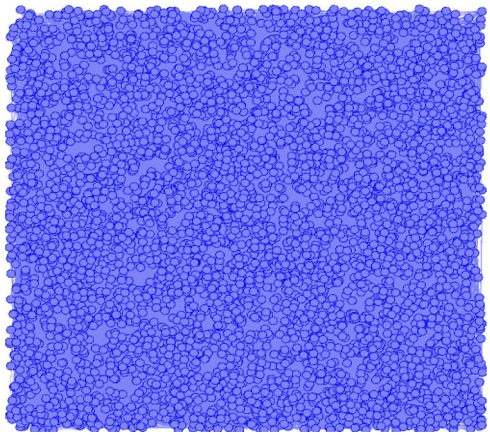

Fig. 14. Follower network of the “Darwin” Dataset.

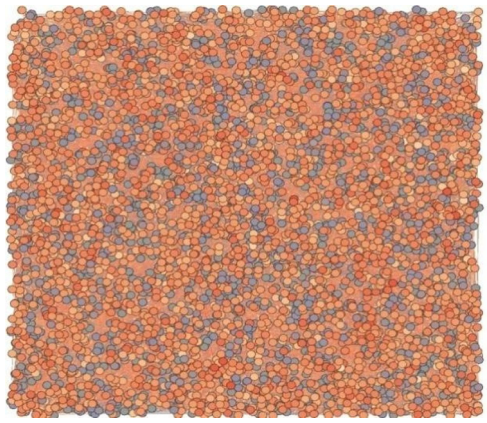

Fig. 15. Expected influence network of the “Darwin” Dataset.

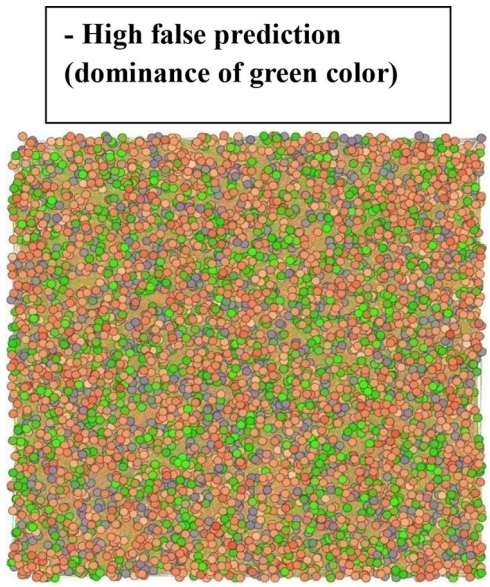

Fig. 16. Influenced network predicted by IC. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

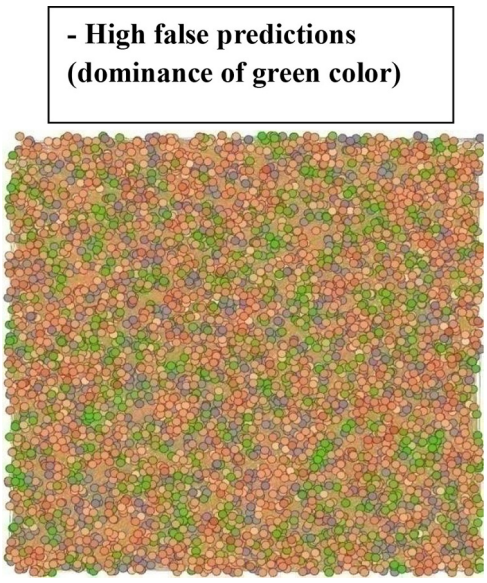

Fig. 17. Influenced network predicted by TC−C. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

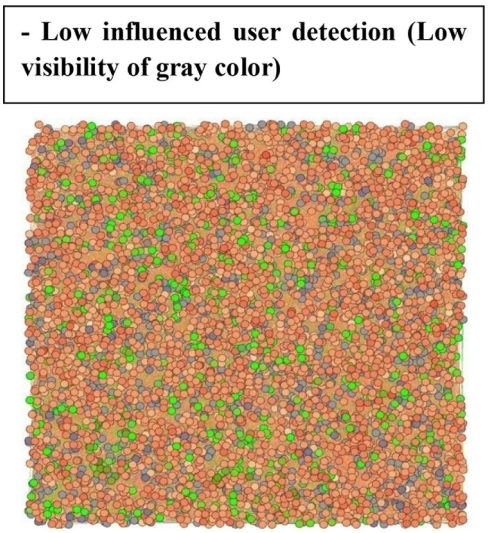

Fig. 18. Influenced network predicted by TD-C. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

on these datasets. Since TDF-C and TDF-CT both models demonstrate the good accuracy in different time intervals. It is difficult to select one of them as the single best-fitted model for diffusion.

TDF-C and TDF-CT both compute the link influence strength between users using the M-TAP method. The only difference between TDF-C and TDF-CT is the diffusion model. TDF-C estimates the diffusion through the IC model. In contrast, the TDF-CT estimates the diffusion through the Progressive Feedback Estimation (PFE) model, i.e., the fusion of LT and IC model that introduced the neutral state. Therefore, to further analyze the performance tradeoff between TDF-C and TDF-CT, the comparative analysis of the precision, recall, and F1 score is presented, as shown in Figs. 11–13, respectively.

First, the performance of TDF-C and TDF-CT is compared using the difference matrix of precision and recall. TDF-CT performed better than the TDF-C in terms of precision and improved precision by 11% and 17% with respect to the “Darwin” dataset and the “Melcup17” dataset, respectively. Therefore, TDF-CT is able to identify more influenced users correctly, out of the total number of influenced users predicted. From the results, it is seen that consideration of the neutral state limits the false prediction of the influence spread in diffusion, i.e., good precision. However, TDF-C performed better than the TDF-CT in terms of recall and improved the recall by 5% and 14% with respect to the “Darwin” dataset and the “Melcup17” dataset, respectively. Therefore, TDF-C is able to predict more number of influenced users out of the total number of influenced users in reality. From these results, it can be depicted that the

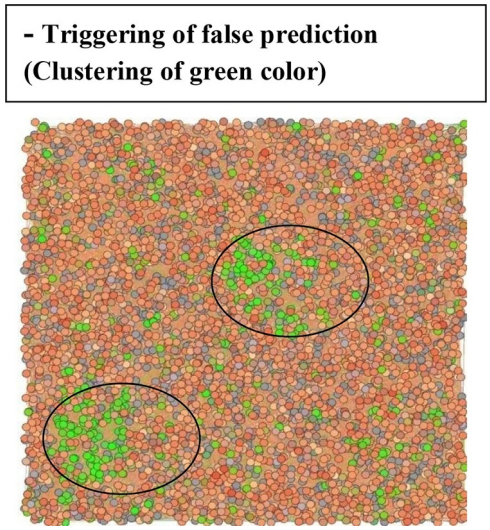

Fig. 19. Influenced network predicted by TDD-C. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

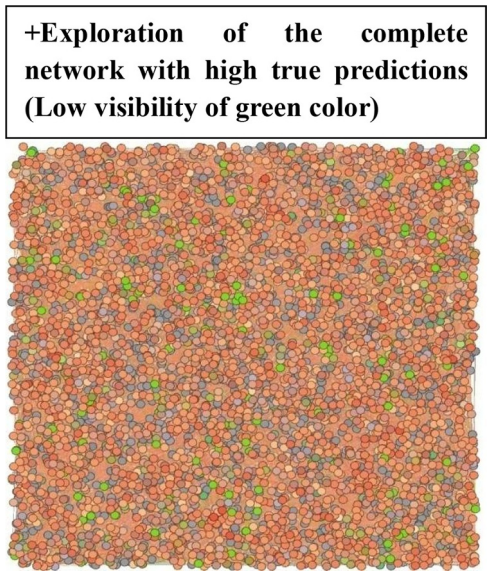

Fig. 20. Influenced network predicted by TDF-C. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

non-consideration of the neutral state increases the false positives and thus improves the chances to predict more influenced users of the network. Therefore, TDF-C and TDF-CT are both useful in different types of domains/applications. For example, if a person wants to advertise/promote their coaching class by giving them a free demo. TDF-CT will be more suitable as compared to TDF-C since the cost is also involved with demo classes. So, she will prefer to give a demo to those who are more likely to join the class. i.e., prediction of more number of influence students out of the total number of students attending the demo class. Whereas, TDF-C may fit appropriately for the applications where cost is not involved. Such as, somebody wants to promote their brand through social media advertisements. Therefore, the fitness of both models is application-specific.

The F_1 score is analyzed to optimize the effect of precision and recall, as shown in Fig. 13. From Fig. 13, it can be depicted that the performance of both models, i.e., TDF-C and TDF-CT, are very close to each other, and both are good for different application scenarios.

6.3.3. Predicting shape

The shape of the influence cascade discovers the various characteristics of the diffusion model, as discussed in Section 5 of this paper. Therefore, the performance of the models is also evaluated using the spread pattern, i.e., shape. Shape prediction helps to

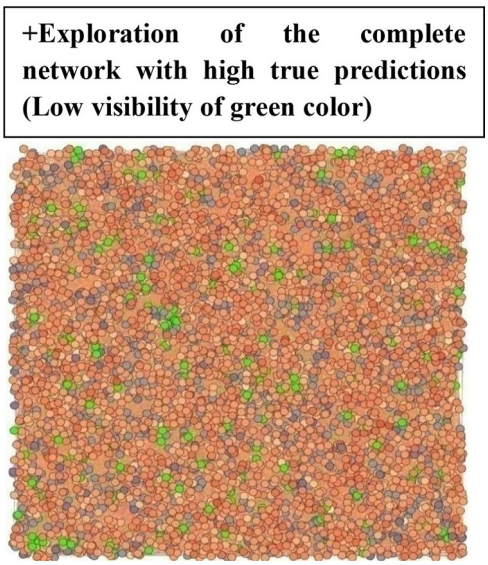

Fig. 21. Influenced network predicted by TDF-CT. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

analyze the false prediction spread and true prediction spread with respect to each diffusion model for the "Darwin" dataset. Fig. 14 shows the follower network G of the "Darwin" dataset. Where all users in the network are represented by the blue color.

Fig. 15 demonstrates the expected influence network of the "Darwin" dataset, where orange color shows the non-influenced users, and gray color shows the influenced users. Therefore, each user in the network is labeled as either influenced or non-influenced, i.e., gray color and orange color, respectively.

Similarly, Figs. 16–21 illustrated the influenced network predicted by IC, TC-C, TD-C, TDD-C, TDF-C, and TDF-CT, respectively. The color-coding in the predicted influence network is as follows:

- Gray color represents that the user is predicted as an influenced user and rightly labeled according to the expected influence network.
- Orange color represents that the user is predicted as a non-influenced user and rightly labeled according to the expected influence network.
- Green color shows the incorrect labeling of the users, i.e., not as per the expected influence network.

It is observed from the Figures that TDF-C and TDF-CT performed better than the other models in terms of shape as well. Following observations are derived from the experimental results, i.e., from Figs. 16–21:

- The number of false predictions by the IC model and TC-C model is very high, as shown in Figs. 16 and 17, respectively. However, the IC model and TC-C model both explore the network completely.
- The number of false predictions of TD-C model is low as compared to the IC model and TC-C model, as illustrated in Fig. 18. TD-C model is able to decrease the number of false labeling of the non-influenced user. However, the number of false labeling of the influenced users is still very high. Overall, the number of false predictions is high as compared to the proposed models TDF-C and TDF-CT.
- The number of false predictions of TDD-C model is also low as compared to the IC model and TC-C model, as illustrated by Fig. 19. TDD-C model is able to do more number of right predictions of both types of users, i.e., influenced and non-influenced as compared to TD-C model. TDD-C model shows the less number of false predictions. However, these fewer false predictions are observed in terms of false clusters. TDD-C's predicted influenced network contains green colored clusters, as shown in Fig. 19. Therefore, false prediction clusters can lead to more false predictions if the diffusion prediction process is not stopped at the right time. On the contrary, TDF-C and TDF-CT are able to predict more number of influenced users uniformly without forming any false prediction clusters.

Table X shows the results of the depth distribution of the correctly predicted influenced users. The depth of the influenced users in the network are analyzed with respect to five depth levels (<2, 2–5, 6–10, 11–15, >15). Each influenced users in the network is categorized in one of the five depth categories according to their depth level. The performance of each model is evaluated on the accuracy measure with respect to each depth level for both datasets to know the efficacy of each model for correct influence estimation at different depth levels.

Table X
Spread Shape Analysis Using Depth Distribution (in%).

	Depth Level	Diffusion Models						
		Actual	IC	TC−C	TD-C	TDD-C	TDF-C	TDF-CT
Darwin	< 2	79.6	23.3	32.6	34.1	46.3	69.8	63.3
	2–5	11.7	2.5	4.9	5.2	1.6	6.5	9.5
	6–10	6.3	1.6	3.2	2.7	1.8	3.5	5.5
	11–15	1.5	0.2	0.8	0.6	0.6	0.9	1.0
	> 15	0.8	0.1	0.1	0.1	0.2	0.4	0.4
MelCup17	< 2	72.4	33.8	37.5	39.3	44.3	58.6	54.6
	2–5	13.9	2.6	3.5	3.8	5.2	7.6	8.7
	6–10	5.6	1.6	2.2	2.4	2.1	3.1	3.9
	11–15	4.6	0.8	1.1	0.8	0.9	1.2	1.6
	> 15	3.4	0.2	0.5	0.4	0.5	1.3	1.5

Algorithm 1
M-TAP (G, F, A).

1. Compute the dominant_user function g using eq. (3)
2. Compute L_{ij} by eq. (5)
3. Set all $T_{ij} = 0$
4. Repeat until convergence
5. For all link e_{ij} in G
6. Determine the value of R_{ij} by eq. (6)
7. For all user 'i' in G
8. Determine the value of T_{jj} by eq. (7)
9. For all link e_{ij} in G
10. Determine the value of T_{ij} by eq. (8)
11. For all user 'i' in G
12. For all user $k \in NB(i) \cup \{i\}$
13. Determine the value of S_{ki} by eq. (9)
14. Return G^1 (V, E, S^0, S^1)

From the Table X, following observations are derived:

- Actual influence spread decreases with increase in the depth level. The same results are observed with respect to each model.
- IC and TC-C depict the lower performance despite having the high influence spread rate (as shown in Figs. 7 and 8). This implies the high false predictions (the same results are shown in Figs. 16 and 17).
- TD-C exhibits the lower performance along with low influence spread rate (as shown in Figs. 7 and 8), thereby indicating the low successful influence user detections (the same results are shown in Fig. 18).
- TDD-C shows good performance up to 2 depth-level post which the performance of the model is declined. This behavior reveals the triggering of the false results after the 2 depth level (the same results are shown in Fig. 19).
- TDF-C and TDF-CT both illustrate better and consistent accuracy in predictions at each depth level as compared to other models (the same results are observed from Figs. 20 and 21).

Overall, from experimental results, it is observed that TDF-C and TDF-CT performed better than the IC, TC-C, TD-C, and TDD-C in terms of each evaluative parameter.

7. Conclusion

This work explored the problem of diffusion estimation by addressing three challenges: estimation of link influence probability between a user-pair; time-based diffusion volume estimation; and time-based analysis to explore when a user is influenced. The work presented two diffusion models TDF-C and TDF-CT in which the diffusion process is divided into two components, i.e., link influence strength estimation, and diffusion spread estimation. TDF-C learns the link influence strength using the M-TAP algorithm that utilizes all four features of the social network, i.e., profile, structural, temporal, and interaction features using the action history log. TDF-CT is based on the hybridization of the independent cascade (IC) and Linear Threshold (LT) model. Both models are evaluated on two datasets over prediction size, accuracy, precision, recall, F1 score, and spread shape. Results illustrate that proposed models performed better than the Independent Cascade (IC), Time Constant Cascade Model (TC−C), Time Decay Cascade Model (TD-C), and Time-Depth Decay Cascade Model (TDD-CT) and improve the accuracy up to 39%. Both the proposed models are suitable for estimation of the diffusion in a wide variety of applications such as viral marketing, health-care, social psychology, drive-risk analysis.

Algorithm 2
PFE Model (G^1, A, 'i', 'x', $t = 0$).

1. Terminate ← False
2. D ← {i}
3. D_t ← D
4. State Allocation of each user using State() function in G^1
5. Movement Analysis of each user using M() function in G^1
6. While ~Terminate do
7. D_{t+1} ← NULL
8. For User 'i' ∈ ($V \cap D^C$)
9. if M(i, t) = = "IC" OR "Can't Say"
10. For j ∈ NBS(i) && State(j,t) != "neutral"
11. if $S^0_{ji} > \Theta_1$
12. 'j' gets enough influence probability to become "influenced"
13. if $S^1_{ji} > \Theta_1$
14. 'j' gets enough influence probability to become "non-influenced"
15. if M(i, t) = = "LT" OR "Can't Say"
16. For 'j'∈ NBS(i)
17. if $\sum_{k \in NBS(j)} (S^0_{kj}) > \Theta_2$:
18. 'j' gets enough influence probability to become "influenced"
19. if $\sum_{k \in NBS(j)} (S^1_{kj}) > \Theta_2$:
20. 'j' gets enough influence probability to become "non- influenced"
21. if 'j' gets enough influence probability to become "influenced"
22. $D_{t+1} \leftarrow D_{t+1} \cup \{j\}$
23. if 'j' gets enough influence probability to become "non-influenced"
24. $D_{t+1} \leftarrow D_{t+1} \cap \{j\}$
25. $D_t \leftarrow D_{t+1}$
26. $D \leftarrow D \cup D_t$
27. if D_t is empty:
28. Terminate ← True
29. else:
30. $t \leftarrow t + 1$
31. Goto, step 4.
32. Return DScore(i) = |D|

This work can be extended for the link prediction, which can be applied to many aspects of social network analysis, such as friend/page/group recommendations in social networks, prediction of potential links in biological protein networks, or the prediction of potential associations in collaborative networks. This work can also be extended to detect the source and spread patterns of the users in the social network such as rumor detection, voting behavior, and crime patterns. In the future, we intend to address the problem of heterogeneous features (such as images and video sharing) to predict activations.

CRediT authorship contribution statement

Sakshi Agarwal: Conceptualization, Methodology, Software, Data curation, Validation, Writing - original draft. **Shikha Mehta:** Conceptualization, Methodology, Writing - review & editing, Supervision.

Declaration of Competing Interest

None.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ipm.2020.102321.

References

Agarwal, S., & Mehta, S. (2018, August). Social influence maximization using genetic algorithm with dynamic probabilities. *Proceedings of the eleventh international conference on contemporary computing (IC3)* (pp. 1–6). .

Agarwal, S., & Mehta, S. (2019). Multi-perspective elicitation of influential parameters and measures in social network. *International Journal of Innovative Technology and Exploring Engineering, 8*(8), 2560–2571.

Aldous, K. K., An, J., & Jansen, B. J. (2019, November). Stylistic features usage: Similarities and differences using multiple social networks. *Proceedings of the international conference on social informatics* (pp. 309–318). .

Alkhodair, S. A., Ding, S. H., Fung, B. C., & Liu, J. (2020). Detecting breaking news rumors of emerging topics in social media. *Information Processing & Management, 57*(2), 102018.

Alp, Z. Z., & Öğüdücü, Ş. G. (2019). Influence factorization for identifying authorities in twitter. *Knowledge-Based Systems, 163*, 944–954.

Asim, Y., Malik, A. K., Raza, B., & Shahid, A. R. (2019). A trust model for analysis of trust, influence and their relationship in social network communities. *Telematics and Informatics, 36*, 94–116.

Baig, M. I., Shuib, L., & Yadegaridehkordi, E. (2019). Big data adoption: State of the art and research challenges. *Information Processing & Management, 56*(6), 102095.
Bozorgi, A., Haghighi, H., Zahedi, M. S., & Rezvani, M. (2016). INCIM: A community-based algorithm for influence maximization problem under the linear threshold model. *Information Processing & Management, 52*(6), 1188–1199.
Chen, W., Lakshmanan, L. V., & Castillo, C. (2013). Information and influence propagation in social networks. *Synthesis Lectures on Data Management, 5*(4), 1–177.
D'Angelo, G., Severini, L., & Velaj, Y. (2019). Recommending links through influence maximization. *Theoretical Computer Science, 764*, 30–41.
Fani, H., Jiang, E., Bagheri, E., Al-Obeidat, F., Du, W., & Kargar, M. (2020). User community detection via embedding of social network structure and temporal content. *Information Processing & Management, 57*(2), 102056.
Fibich, G. (2016). Bass-SIR model for diffusion of new products in social networks. *Physical Review E, 94*(3), 032305.
Gass, Robert H. (2015). Social Influence, Sociology of. *International Encyclopedia of the Social & Behavioral Sciences,* 348–354. https://doi.org/10.1016/b978-0-08-097086-8.32074-8 Available online.
Goyal, A., Bonchi, F., & Lakshmanan, L. V. (2010, February). Learning influence probabilities in social networks. *Proceedings of the third ACM international conference on web search and data mining* (pp. 241–250). .
G. Tong, "Adaptive influence maximization under general feedback models," arXiv preprint arXiv:1902.00192, 2019.
He, J. S., Han, M., Ji, S., Du, T., & Li, Z. (2019). Spreading social influence with both positive and negative opinions in online networks. *Big Data Mining and Analytics, 2*(2), 100–117.
Hoang, T. B. N., & Mothe, J. (2018). Predicting information diffusion on Twitter–Analysis of predictive features. *Journal of Computational Science, 28*, 257–264.
Iñiguez, G., Ruan, Z., Kaski, K., Kertész, J., & Karsai, M. (2018). Service adoption spreading in online social networks. In *Complex spreading phenomena in social systems*, 151–175.
Jendoubi, S., Martin, A., Liétard, L., Hadji, H. B., & Yaghlane, B. B. (2017). Two evidential data based models for influence maximization in twitter. *Knowledge-Based Systems, 121*, 58–70.
Kempe, D., Kleinberg, J., & Tardos, É. (2015). Maximizing the spread of influence through a social network. *Theory OF Computing, 11*(4), 105–147.
Kermack, W. O., & McKendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, 115*(772), 700–721.
Kuhnle, A., Alim, M. A., Li, X., Zhang, H., & Thai, M. T. (2018). Multiplex influence maximization in online social networks with heterogeneous diffusion models. *IEEE Transactions on Computational Social Systems, 5*(2), 418–429.
Liu, L., Qu, B., Chen, B., Hanjalic, A., & Wang, H. (2018). Modelling of information diffusion on social networks with applications to WeChat. *Physica A: Statistical Mechanics and its Applications, 496*, 318–329.
Liu, Y., Jin, X., & Shen, H. (2019). Towards early identification of online rumors based on long short-term memory networks. *Information Processing & Management, 56*(4), 1457–1467.
Mei, Y., Zhao, W., & Yang, J. (2017, May). Influence maximization on twitter: A mechanism for effective marketing campaign. *Proceedings of the IEEE International Conference on Communications* (pp. 1–6). .
Mei, Y., Zhong, Y., & Yang, J. (2015, March). Finding and analyzing principal features for measuring user influence on Twitter. *Proceedings of the IEEE first international conference on big data computing service and applications* (pp. 478–486). .
Najafabadi, M. K., Mohamed, A., & Onn, C. W. (2019). An impact of time and item influencer in collaborative filtering recommendations using graph-based model. *Information Processing & Management, 56*(3), 526–540.
Noekhah, S., Binti Salim, N., & Zakaria, N. H. (2020). Opinion spam detection: Using multi-iterative graph-based model. *Information Processing & Management, 57*(1), 102140.
Peng, S., Zhou, Y., Cao, L., Yu, S., Niu, J., & Jia, W. (2018). Influence analysis in social networks: A survey. *Journal of Network and Computer Applications, 106*, 17–32.
Ramos, G., Boratto, L., & Caleiro, C. (2019). On the negative impact of social influence in recommender systems: A study of bribery in collaborative hybrid algorithms. *Information Processing & Management*102058.
Saito, K., Nakano, R., & Kimura, M. (2008, September). Prediction of information diffusion probabilities for independent cascade model. *Proceedings of the international conference on knowledge-based and intelligent information and engineering systems* (pp. 67–75). .
Saxena, B., & Kumar, P. (2019). A node activity and connectivity-based model for influence maximization in social networks. *Social Network Analysis and Mining, 9*(1), 40.
Saxena, B., & Saxena, V. (2019). Hurst exponent based approach for influence maximization in social networks. *Journal of King Saud University-Computer and Information Sciences*. https://doi.org/10.1016/j.jksuci.2019.12.010 Available online.
Shin, J., Jian, L., Driscoll, K., & Bar, F. (2018). The diffusion of misinformation on social media: Temporal pattern, message, and source. *Computers in Human Behavior, 83*, 278–287.
Sun, J., & Tang, J. (2011). A survey of models and algorithms for social influence analysis. In *Social network data analytics*, 177–214.
Tang, J. (2017,). Computational Models for Social Network Analysis: A Brief Survey. *Proceedings of the 26th international conference on world wide web companion* (pp. 921–925). .
Tang, J., Sun, J., Wang, C., & Yang, Z. (2009, June). Social influence analysis in large-scale networks. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 807–816). .
Tong, G., Wu, W., Tang, S., & Du, D. Z. (2016). Adaptive influence maximization in dynamic social networks. *IEEE/ACM Transactions on Networking, 25*(1), 112–125.
Wen, Y. T., Peng, W. C., & Shuai, H. H. (2018, June). Maximizing social influence on target users. *Proceedings of the pacific-asia conference on knowledge discovery and data mining* (pp. 701–712). .
Xin, M., & Wu, L. (2020). Using multi-features to partition users for friends recommendation in location based social network. *Information Processing & Management, 57*(1), 102125.
Xuan, Q., Shu, X., Ruan, Z., Wang, J., Fu, C., & Chen, G. (2019). A self-learning information diffusion model for smart social networks. In Wu (Ed.). *Proceedings of the IEEE Transactions on Network Science and engineering*IEEEhttps://doi.org/10.1109/TNSE.2019.2935905.
Zhang, Z., Zhao, W., Yang, J., Paris, C., & Nepal, S. (2019, May). Learning influence probabilities and modelling influence diffusion in twitter. *Proceedings of the 2019 World Wide Web Conference Companion* (pp. 1087–1094). .
Zhu, W., Yang, W., Xuan, S., Man, D., Wang, W., & Du, X. (2019). Location-based seeds selection for influence blocking maximization in social networks. *IEEE Access, 7*, 27272–27287.