The Journal of Supercomputing (2022) 78:5450–5478
https://doi.org/10.1007/s11227-021-04079-7

Impact of second-order network motif on online social networks

Sankhamita Sinha[1] · **Subhayan Bhattacharya**[1] · **Sarbani Roy**[1]

Accepted: 5 September 2021 / Published online: 24 September 2021

Abstract

The behaviour of individual users in an online social network is a major contributing factor in determining the outcome of multiple network phenomenon. Group formation, growth of the network, information propagation, and rumour blocking are some of the many network behavioural traits that are influenced by the interaction patterns of the users in the network. Network motifs capture one such interaction pattern between users in online social networks (OSNs). For this work, four second-order (two-edged) network motifs have been considered, namely, message receiving pattern, message broadcasting pattern, message passing pattern, and reciprocal message pattern, to analyse user behaviour in online social networks. This work provides and utilizes a node interaction pattern-finding algorithm to identify the frequency of aforementioned second-order network motifs in six real-life online social networks (Facebook, GPlus, GNU, Twitter, Enron Email, and Wiki-vote). The frequency of network motifs participated in by a node is considered for the relative ranking of all nodes in the online social networks. The highest-rated nodes are considered seeds for information propagation. The performance of using network motifs for ranking nodes as seeds for information propagation is validated using statistical metrics Z-score, concentration, and significance profile and compared with baseline ranking methods in-degree centrality, out-degree centrality, closeness centrality, and PageRank. The comparative study shows the performance of centrality measures to be similar or better than second-order network motifs as seed nodes in information diffusion. The experimental results on finding frequencies and importance of different interaction patterns provide insights on the significance and representation of each such interaction pattern and how it varies from network to network.

Keywords Online social network · Network motifs · Second-order motif · Centrality measures · Information diffusion

✉ Sankhamita Sinha
sankhamita@gmail.com

Extended author information available on the last page of the article

1 Introduction

The OSNs such as Facebook, Twitter, YouTube, Instagram, WhatsApp, Snapchat, Google+, Quora, and LiveJournal are popular online communication platforms for the last few years, and their usages are increasing significantly [19]. Structurally, the OSNs are represented as a directed graph, where the users can be termed as nodes and the links are the different types of communications such as likes, replies to, and mentions established between nodes. Therefore, the OSN is an example of a complex network through which the users build social communication and interaction, passing message based on the Internet platform [12]. The pattern of interaction of the users implies the social relationships with other users in real-life connections. According to Rogers [31], communication is a process in which participants generate, transfer, and receive information with one another to reach a mutual understanding. Diffusion is also a kind of social change. It is defined as the process by which the structure and function of a social system are explained. To understand the behaviour of interaction patterns between the users in a network, a network motif plays an important role [27]. Many small sub-graphs of OSNs are significant in representing the fundamental topological communication patterns of the OSNs [24]. To uncover these structural interaction patterns, the network motif is an important tool. A network motif is a small sub-graph of a given input network that occurs in significantly higher frequencies than expected in random networks [25]. The two-edged motifs have been described in [40], where the authors have investigated how the two-edged motifs influence the synchrony in a neural network. The two-edged motifs can be used to distinguish the basic communication patterns of the users with immediate nodes. Some users act in OSNs only as listeners; some users participate to broadcast the messages. In OSNs, some users' interest is on both message passing and receiving and some users are close to each other so after getting the messages they reply. Using communication pattern findings, the overall behaviour of the information diffusion process in the network can be recognized [24]. To identify the message spreading patterns, basic communication pattern mining of a user is necessary. It tells how the users are participating to receive, broadcast, pass, and reciprocate the messages most efficiently in the network.

This paper focuses on the frequency computation of two-edged sub-graphs GS_i, i.e. the convergent, the divergent, the chain, and the reciprocal, where i is the index of the sub-graphs, which represent the basic communication patterns such as message receiving pattern, message broadcasting pattern, message passing pattern, and reciprocal message pattern between the users, respectively, in a social network $G(V, E)$. This paper also investigates the two-edged motifs GS_i, where $i = \{1, 2, 3, 4\}$ of a social network $G(V, E)$ using the statistical measures of network motif—Z-score, concentration, and significance profile. Furthermore, based on frequencies of the communication patterns the influence nodes have been selected as seed nodes. Then the popular diffusion models—forest fire (FF) model, independent cascade (IC) model, susceptible-infected-recovered (SIR) model—have been used to evaluate the activated nodes. The results have been compared with different centrality measures.

1.1 Motivation

In modern day life, OSNs have evolved as the most frequent medium of communication. All kinds of information, something as global as a natural disaster like Tsunami to something as simple as the last meal one had, all information and opinions are shared on OSNs. The multitude of patterns and dependencies present in OSNs, and are mathematically represented as graphs, play an important role in regulating the information propagation in a network. The velocity, volume, and direction of information flowing through a network are dependent on the structure and orientation of structural patterns in the network. One such pattern is the network motif. Network motif is a statistically significant sub-graph or pattern of a large communication network or graph [26]. Large-scale networks like biological networks, OSNs, electrical circuit networks, and so on can be represented as a graph, which includes a wide variety of network motifs. There are four second-order connection motifs (two-edged motifs) of reciprocal, convergent, divergent, and chain connections and [40] investigates how these network structures can influence the tendency for a neuronal network to synchronize, albeit, independent of the dynamical model for each neuron.

Different OSNs might have different dominating network motif, that is, the frequency of different network motifs might differ among different graphs. Based on the dominant network motif, the information propagation can be predicted for a network, or the flow of information can be regulated for beneficiary results. Rumours resulting in mass hysteria and panic can be stopped, whereas information of a natural disaster can be quickly propagated to benefit the masses.

1.2 Contribution

Identifying nodes of importance in a network is a research domain with multiple real-life applications. In that light, the contributions of this paper can be listed as

- A novel algorithm for generating random networks with similar degree distribution to a given network.
- Identify the second-order motifs in different real-life OSNs and compare their frequency using six real-life OSNs.
- Analyse the efficiency of such second-order motifs as seed nodes in information diffusion in OSNs using six real-life OSNs and four popular centrality measures.
- Compare the performance of second-order motifs with high centrality nodes as seed for information flow.

The identified seed nodes can be crucial in enhancing and manipulating the flow of information within a network. This can find many real-life applications such as directed marketing, spreading awareness on social reforms and healthcare issues through online networks, identifying potential flash mob initiators, and other such activities.

1.3 Organization

The rest of the paper is organized as follows. Section 2 reviews the literature based on network motifs to understand the patterns of complex networks, information diffusion, and random graph model. The required network properties—graph terminologies, measurement tools for network motif, degree centrality measures, information diffusion models, and communication pattern initialization based on second-order/two-edges sub-graphs—are defined in Sect. 3. In Sect. 4, the problem statement and proposed approach are discussed to identify the second-order motifs. Section 5 exhibits the experimental results, discussion, and analysis. In Sect. 6, relevant applications are discussed. Finally, Sect. 7 concludes the work with a discussion related to future scopes.

2 Related work

The concept of network motif as a simple building block of the complex network is introduced by Milo R et al. in [26]. They also define that the network motif is the patterns of interconnections occurring in complex networks at numbers that are significantly higher than those in randomized networks. Another paper by Milo et al. [25] present an approach for comparing network local structure, which is based on the SP of small sub-graphs in the network compared to randomized networks. Shen-Orr SS et al. [32] also define network motif as an interaction pattern that repeats in many different parts of a network at higher frequencies than those found in randomized networks. The network motifs with higher frequencies than expected at random networks suggest that they may have specific functions in the information processing performed by the network.

In very recent work [13], authors define a new method to explain a network motif using the graph compression technique. They explain a sub-graph M considered as a network motif if the probability of M in G is greater than the probability of M in a null model of G. In [29] and [39], the authors review the different tools for finding network motifs in a network. Several papers like [38] and [33] propose clustering techniques using network motifs. In these papers, the authors discuss the topological network motifs where patterns are similar but not necessarily identical, and they propose a statistical model for the occurrence of such motifs in a biological network, from which they derive a scoring function for statistical significance in [10]. Based on this scoring function, they introduce a search algorithm for topological motifs which is called graph alignment. In [32], the authors also focus on the gene network motif. In [27], the authors identify interaction pattern motifs using the coloured motif in an email network and measure the importance of nodes by degree prestige and degree centrality. Youngsoo et al. [20] concentrate on the user's communication pattern on a Mobile Social Network and explore the users' collective behaviour like chat, message, and group message.

In [36], the authors examine the records of user interactions to analyse interaction patterns across large user groups on Facebook. In [30], the authors explain the information diffusion process in the Twitter network based on the various measures

of users' activity, popularity, and influence. Another paper [23] focuses on the information diffusion model in OSN, and they categorize the models into explanatory models and predictive models. According to Milo et. al. [26] network motif theory, the real networks' frequencies of sub-graphs have been compared to suitably randomized networks' frequencies of sub-graphs and only select the patterns which appear in the real network at numbers significantly higher than in the randomized networks. Therefore, generating random networks is essential to determine the statistical significance of a sub-graph as a network motif. The undirected random network model with N nodes is proposed in [17] which creates every edge with probability $p \in (0, 1)$ independently of every other edge. Another very popular network model is the 'Barabasi–Albert'(BA) [7] model. It is an algorithm for generating random scale-free networks using a 'preferential attachment' mechanism which follows the power-law distribution:

$$p(k) \sim k^{-r} \tag{1}$$

Where $p(k)$ is the fraction of nodes in the network with degree k, r is a parameter usually between 2 and 3.

The authors propose a diffusion technique for tracking the rate with which information spread over underlying social interaction structure in the temporal domain and few social parameters in [21]. This work is motivated by the epidemic model and also proposes forward state transition and recoverable transition. This model supports predicting/forecasting of information diffusion in social media and infectious disease spreading in the community to find out the optimal value of the susceptible and infected number of people during the infection period. In [15], the authors investigate the information diffusion about the COVID-19 on several social media data sets. They have observed the spread of information using epidemic models and suggest that information spreading is driven by the interaction paradigm. This pattern is determined by the specific social media or/and by the specific interaction patterns of groups of users engaged with the topic. The authors establish the information diffusion model based on the FF model and shows that information spreading across online social networks depends upon user–followers relationships, the significance of the topic, and other features on Twitter network in [22]. In very recent work, [35] proposes the information diffusion model based on mean-field theory and compares it with the SIR model. They establish in their work that users' mobility increases the connections among users which affects the spreading of information diffusion. In [34], the authors work in maximizing influence diffusion in large-scale networks using heuristics on independent cascade models.

Unlike the existing methods above, this paper investigates the second-order motifs for each node using several statistical tools to find the communication patterns of the users. By considering the highest frequencies of the different second-order motifs of a node, the seed nodes have been selected to experiment with the information diffusion process using independent cascade, susceptible-infected-removed, and forest fire models. The result is also compared with the highest centrality nodes. This helps to understand the importance of basic communication patterns of the nodes in several OSNs. The key novelty of this work is that the findings of basic interaction

patterns (second-order motifs) of the users are based on the generation of a random graph with similar in-degree and out-degree sequences of the input graph and determining their effects on information diffusion in OSNs.

3 Network properties

3.1 Graph terminologies

3.1.1 Directed graphs and sub-graphs

A complex network, which is represented as a directed graph $G(V, E)$ composed by a finite non-empty set V of vertices or nodes connected by edges that belong to the set of edge E, has a direction associated with vertices. A directed sub-graph GS_i of a directed graph $G(V, E)$ is a graph whose vertex set belongs to V, that is $VS_i \subseteq V$, and whose edge set ES_i is a subset of the edge set E, that is $ES_i \subseteq E$ and the direction of edges in sub-graphs follow the same of $G(V, E)$.

3.1.2 Sub-graph frequency

The frequency of sub-graph GS_i is denoted by $F(GS_i)$, where $i = \{1, 2, 3, 4\}$, is the number of occurrences of patterns in graph $G(V, E)$. A motif is a pattern that is considered significant according to a particular frequency-based comparison.

3.1.3 In-degree and out-degree

For a directed graph $G(V, E)$ with edges E and vertices V, the out-degree of $v2$ refers to the number of edges incident from $v2$. That is, the number of edges directed away from the vertex $v2$. The in-degree of $v1$ refers to the number of edges incident to $v1$. That is, the number of edges directed towards the vertex $v1$.

3.1.4 Network motif

A network motif is a small sub-graph which appears recurrently in a complex network and satisfies the following conditions:

1. $Prob(F\_rand(GS) > F\_orig(GS)) \leq P$ (This is used to check the higher frequency in original network rather than random network with same degrees)
2. $F\_orig(GS) \geq U$ (It is used to check the minimum frequency occurrence in original network)
3. $F\_orig(GS) - F\_rand(GS) > D \times F\_rand(GS)$ (It checks the minimum deviation)

where $\{P, U, D, N\}$ is a set of parameters and Milo [25] considers $\{0.01, 4, 0.1, 1000\}$, which refers the number of similar random networks is 1000, a sub-graph is considered as a motif if the chance that it appears more often in a random network than in

the original network is less than 1%, the sub-graph is present at least 4 times in the original graph and the difference between its frequency in the original network and the average frequency in random networks is at least 10% of that average frequency in random network.

Here, P is probability threshold, U is uniqueness threshold, D is proportional threshold, N is the number of random networks, $F\_rand(GS)$ is the frequency of sub-graph in random graph, and $F\_orig(GS)$ is the frequency of sub-graph in original graph.

Table 1 refers to the calculated frequency of each second-order network motif in the original OSN graphs, and the mean frequency of the same in the random graphs. For the purpose of this work, the values of $\{P, U, D, N\}$ have been considered as $\{0.01, 4, 0.1, 100\}$. Based on these values, the network motifs that are expected to be not dominant in each of the real-life OSNs can be predicted. For the Facebook data set, the reciprocal motifs can be predicted to be non-dominant. Similarly, for Twitter,

Table 1 Values of the proposed network motif frequency of original graph and mean frequency of 100 random graphs with similar in-degree and out-degree sequences of original graph

Dataset	Interaction pattern	Original_Motif_frequency	Mean_Motif_Random_Frequency
Facebook	Receiving	2649368.0	2337039.16
	Broadcasting	3975462.0	1963597.76
	Message passing	2690019	2401185.64
	Reciprocal	0.0	451.54
Twitter-scrapped	Receiving	105264.0	245.28
	Broadcasting	4102.0	5.18
	Message passing	952	25.6
	Reciprocal	18	0.0
Email	Receiving	5483067.0	10792220.62
	Broadcasting	21123559.0	2428672.04
	Message passing	19000389	8892559.12
	Reciprocal	44620.0	3661.26
Gplus	Receiving	40652.0	10799406.26
	Broadcasting	14563326.0	9060.05
	Message passing	195592	48480.31
	Reciprocal	48.0	1.84
GNU	Receiving	153351.0	175955.94
	Broadcasting	185113.0	140042.62
	Message passing	180230.0	176648.72
	Reciprocal	0.0	10.2
Wiki-vote	Receiving	4285079.0	3622650.18
	Broadcasting	7062816.0	780579.74
	Message passing	4542805	1651593.4
	Reciprocal	2927.0	451.72

Email, GPlus, GNU data sets, the sets of non-dominant network motifs are (reciprocal), (receiving), (receiving, reciprocal), respectively. The Wiki-vote data set has no such apparently non-dominant network motif that can be predicted based on just the frequency. However, for the completeness of the comparative study, all the network motifs have been considered for all experiments hereinafter in this article.

3.2 Measures for network motif

3.2.1 Concentration

Concentration is the measure of a particular k size of network motif [37]. It is the ratio of the frequency of a particular size of sub-graph and the total frequency of all possible sub-graphs with the same size. Lets consider a sub-graph GS_i, the concentration of GS_i is defined as

$$C(GS_i) = \frac{F\_orig(GS_i)}{\sum_i F\_orig(GS_i)} \tag{2}$$

where $F\_orig(GS_i)$ is the frequency of the specific sub-graph and i is the index of all possible sets of size k sub-graph. So, the denominator represents the total number of all frequencies of the sub-graph of size k. Here, the size of the graph refers to the cardinality of its edge set. The concentration of a network motif refers to how frequent it is in the network compared to other sub-graphs of the same size [14]. Table 4 shows the percentage of the proposed network motifs. In Facebook, Email, GPlus, GNU, and Wiki-vote, the concentration of broadcasting motif is a higher percentage value that means the interaction pattern broadcasting is statistically significant than other motifs/interaction patterns like receiving, message passing, and reciprocal motifs. In Twitter-scrapped, the concentration value of the receiving interaction pattern is more significant as a motif rather than broadcasting and message passing.

3.2.2 Z-score

Z-score is another measurement tool of motif [37]. It is the ratio between the difference of the frequency of the sub-graph GS_i in the original network $G(V, E)$ and the arithmetic mean frequency of GS_i in n number of random networks and the standard division of frequency of GS_i of n number of random networks. The random networks will be followed the degree distribution of $G(V, E)$. The formulation of Z-score is defined as follows:

$$Zscore(GS_i) = \frac{F\_orig(G_i) - mean(F\_rand(GS_i))}{std(F\_rand(GS_i))} \tag{3}$$

where $F\_orig(GS_i)$ is the frequency of GS_i in original network, $mean(F\_rand(GS_i))$ is the mean frequency of GS_i and $std(F\_rand(GS_i))$ is the standard division of frequency of GS_i in n random networks. The Z-score is high if the sub-graph is over-represented and negative if it is under-represented and close to zero otherwise [16].

Therefore, the larger $Zscore(GS_i)$ value of GS_i means that GS_i is the more significant sub-graph as a network motif in graph $G(V, E)$. Table 3 shows the Z-score values of the proposed network motifs. In Facebook and Wiki-vote, the Z-score value of receiving, broadcasting, message passing motifs are positive and high which means the interaction pattern-receiving, broadcasting, message passing motifs are over-represented, whereas the motif/interaction pattern like reciprocal motif is under-represented in the original networks. In Twitter-scrapped, the reciprocal motif is not considered as a motif.

3.2.3 Significance profile

The significance profile $SP(GS_i)$ [37] is defined as a vector of Z-scores of a particular set of sub-graphs, which is normalized to length of 1.

$$SP(GS_i) = \frac{Zscore(GS_i)}{\sqrt{\sum_i Zscore(GS_i)^2}} \tag{4}$$

where i is the index of all possible sets of sub-graphs with the same size. It highlights the relative significance of sub-graphs, rather than the absolute significance when the Z-score value is higher in a large-scale network. The significance profile of a network is negative values, especially those close to -1, are associated with under-represented sub-graphs, while positive ones, especially those close to 1, allow to recognize the motifs [16]. Table 4 shows the significance profile values of the proposed network motifs. In Facebook, Email, GPlus, GNU, and Wiki-vote, the significance profile value of the broadcasting motif is positive and close to 1 that means the interaction pattern broadcasting is statistically significant than other motifs/interaction patterns like receiving, message passing, and reciprocal motifs. In Twitter-scrapped, the significance profile value of the receiving interaction pattern is more significant as a motif rather than broadcasting and message passing.

3.3 Different centrality measures

3.3.1 In-degree centrality

The in-degree centrality of a node is the in-degree of a node [8] in a directed graph G(V,E). It indicates the number of edges directed to the node $u \in V$.

$$C_i = Indegree(u) \tag{5}$$

3.3.2 Outdegree centrality

The outdegree centrality of a node is the outdegree of a node [8] in a directed graph G(V,E). It indicates the number of edges directed to others from the node $u \in V$.

$$C_o = Outdegree(u) \tag{6}$$

3.3.3 Closeness centrality

The closeness centrality [18] [7] of a node $u \in V$ is the average of the shortest path length from the node to every other nodes in the network. It indicates the closeness of a node to all other nodes $v \in V$ in the network.

$$C_c = \frac{1}{\sum_{v \in G} d(u, v)} \tag{7}$$

3.3.4 PageRank centrality

PageRank centrality is a ranking of the nodes in the graph G(V,E) based on the structure of the incoming links. The PageRank of node u is defined as following [28]:

$$C_{pr}(u) = \sum_{v \in B_u} \frac{C_{pr}(v)}{L(v)} \tag{8}$$

where PageRank value for a node u is dependent on the PageRank values for each node v contained in the set B_u (the set containing all nodes linking to node u), divided by the number $L(v)$ of outbound links from node v.

3.4 Information propagation model

3.4.1 Independent cascade model

The independent cascade (IC) [23] model explains the process of information diffusion in a network. In this model, nodes participate in two states: active (A) and inactive (I). The node is in A state when it receives the information being circulated in the network. The I state node does not receive the information. In each time step, the A node attempts to influence its neighbours with a diffusion probability value.

3.4.2 Susceptible-infected-recovered model

Susceptible-infected-recovered (SIR) [23] model is a stochastic process model. In this model, nodes participate in three states: susceptible (S), infected (I), recovered (R). S is the number of susceptible nodes. These nodes are not infected but could become infected. I is the number of infected nodes. These nodes can transmit information to the suspected nodes. R is the number of removed nodes. These nodes cannot become infected and cannot transmit the information.

In this process, at first, all nodes from network G are susceptible nodes except for a set of nodes that are initially infected. In each discrete time-step t, infected nodes

try to infect their susceptible neighbours with probability p. These infected nodes can also be recovered with probability q. All recovered nodes cannot be infected again.

3.4.3 Forest fire model

Forest fire (FF) model is a mathematical model that has been utilized in information propagation as well [9, 22]. The model simulates the pattern in which a fire spreads in a forest. The fire starts from the seed nodes and is spread through neighbouring nodes with some probability. In the initial condition, all but the seed nodes (trees) can be considered to be not on fire. The neighbours of tree are burning if at least one neighbour tree is burning. The trees which are not adjacent to burning trees can catch fire with a probability p. An empty space is filled with trees with probability f. A burning tree at time instance t is not on fire and cannot catch fire starting at time instance $t + 1$. The propagation continues as long as at least one tree is burning.

3.5 Communication pattern initialization based on second-order/two-edges sub-graphs

The OSN is a communication network where the structure of the network represents the significant pattern of the interactions between the users. All these interaction patterns can be visualized by the four second-order (two-edged) sub-graphs. The network motif represents the pattern of connection. The basic network motif is represented by the single edge with a pair of nodes. Similarly, the second-order network motif is represented by two-edges connecting distinct pairs of nodes [1]. There are four types of two-edges motifs: convergent, divergent, chain motifs, and reciprocal. In OSNs, some users actively participate to receive the message, some users participate to broadcast the message, some users involve with both message receiving and pass, and some users communicate reciprocally with other users. All these interaction patterns can be represented by the four second-order sub-graphs which are described as follows.

Message receiving pattern (convergent) Message receiving pattern is a triadic two edges sub-graph GS_1 of graph $G(V, E)$ if it consists of two directed edges (u, v) and (w, v) where $(u, v), (w, v) \in E$, this means that the interaction originates from u to v and w to v, where $u, v, w \in V$. This means that node v receives the message from nodes u and w, which is shown Fig. 1a.

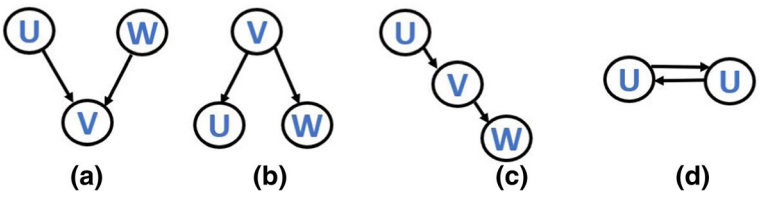


Fig. 1 **a** Convergent, **b** divergent, **c** chain motifs, **d** reciprocal [40]

Message broadcasting pattern (divergent) Message broadcasting pattern is a triadic two edges sub-graph GS_2 of graph $G(V, E)$ if it consists of two directed edges (v, u) and (v, w) where $(v, u), (v, w) \in E$, this means that the interaction originates from v to u and w where $u, v, w \in V$. This means that node v broadcasts the message to nodes u and w, which is shown Fig. 1b.

Message passing pattern (chain) Message passing pattern is a triadic two edges sub-graph GS_3 of graph $G(V, E)$ if it consists of two directed edges (u, v) and (v, w) where $(u, v), (v, w) \in E$, this means that the message flow in the graph originates from node u, it is first transferred to v, and then from v to w where $u, v, w \in V$. This means that u passes the message to v and then v passes the message to w, which is shown Fig. 1c.

Message reciprocal pattern (reciprocal) Message reciprocal pattern is a dyadic two edges sub-graph GS_4 of graph $G(V, E)$ if it consists of two directed edges (u, v) and (v, u) where $(u, v), (v, u) \in E$, this means that the message is flowing from u to v and v to u where $u, v \in V$. This means that u and v are interacting as reciprocal manner, which is shown Fig. 1d.

4 Problem statement and approach

4.1 Problem statement

Given a network $G(V, E)$, a set of second-order/two-edged sub-graphs or network motifs, namely, message receiving pattern, message broadcasting pattern, message passing pattern and message reciprocal pattern, represented by GS_i, where $i \in \{1, 2, 3, 4\}$ and a set of centrality measures C, the problem is to rank the nodes $v \in V$ based on C and GS, according to their suitability as sources for information propagation. The suitability is judged based on a set of metrics M and is verified using a set of information propagation methods IP.

Each node $v \in V$ has a score/value for each of the centrality measures $c \in C$ and motif in GS_i. The nodes can be ranked on this score/value, and the nodes with higher scores can be considered as good seeds.

4.2 Proposed approach

The proposed approach can be broadly broken down in the following steps -

- Count motif frequencies in original and random graphs
- Calculate statistical metrics for each motif for both the original as well as random graphs
- Use the statistical metrics as ranking mechanism for seed selection
- Use the seed nodes selected using network motif frequencies for information diffusion

– Compare the performance of these seed nodes with the seed nodes selected using popular centrality measures as ranking mechanism with respect to information diffusion.

The diffusion process is a special type of communication [31] among members of a network. Therefore, different communication patterns diffuse the information in a different way, and the aim of this study is to compare and contrast the difference in diffusion behaviour based on different communication patterns. Section 4.2.1 provides the algorithms utilized for counting the four different network motifs considered in this work. Section 4.2.2 provides the algorithms for generating random graphs with similar degree distribution as an input graph. The combination of all these algorithms provides the statistical metric-related empirical measures that allows relative ranking of nodes for suitability as seeds for information diffusion. Once the seed nodes have been identified, the top 1%, 2%, 3%, 4%, and 5% of the seed nodes are utilized to study the information diffusion in each of the real-life OSNs using three information propagation models, namely independent cascade, forest fire, and susceptible-infected-removed. Information diffusion is also studied by selection seed nodes based on centrality values, and the results are then compared.

4.2.1 Counting the frequencies of sub-graphs

The network motif detection technique is based on the contrasting of the occurrences of each sub-graphs between in the original network and in the randomized network which has the same nodes and degree sequences [26]. In this step, the frequencies of two-edged sub-graphs such as message receiving pattern, message broadcasting pattern, message passing pattern, and reciprocal message pattern between the users have been computed by the following algorithms:

Algorithm 1: Counting of Frequency -Receiving Motif

Input : $G(V, E)$ - Graph object
Output: $recv\_motif\_no\_total$ - Frequency of Receiving Motif Patterns,
$recv\_motif\_per\_node$ - Frequency of Receiving Motif Patterns per node

```
1  def FindReceivingMotif(G(V, E)):
2      recv_motif_per_node ← {}
3      recv_motif_no_total ← 0
4      forall vertex i ∈ V do
5          recv_motif_no ← 0
6          if Indegree(i) ≠ 0 then
7              v1 ← Indegree(i)
8              if v1 ≥ 1 then
9                  if v1 == 2 then
10                     recv_motif_no=1
11                     recv_motif_per_node[i] ← recv_motif_no
12                     recv_motif_no_total= recv_motif_no_total + recv_motif_no
13                 else
14                     recv_motif_no=(v1*(v1-1))/2
15                     recv_motif_per_node[i] ← recv_motif_no
16                     recv_motif_no_total= recv_motif_no_total + recv_motif_no
17                 end
18             end
19         end
20     end
21     return recv_motif_no_total, recv_motif_per_node
```

Algorithm 1 computes the frequency of message receiving sub-graphs for all nodes of the input graph $G(V, E)$. The value is stored in the variable *recv_motif _no_total*. The algorithm considers the in-degree of a vertex $i \in V$ at step 6, and if the value is not zero, stores it in variable $v1$ at step 7. If the value of $v1$ is 2, then the number of receiving motif is considered as 1 and the value of variables *recv_motif _per_node*[i] is set to 1 and *recv_motif _no_total* is increased by 1 accordingly at step 10 to step 12. If the value of $v1$ is not 2, the algorithm computes the number as receiving motif using the formula $v1 * (v1 - 1)/2$ and updates the value of variables *recv_motif _per_node*[i] and *recv_motif _no_total* accordingly at step 14 to step 16. The algorithm runs from step 5 to step 20 for all the vertex $i \in V$ and returns the frequency of message receiving sub-graphs for each vertex and the total message receiving sub-graphs of the input graph.

Algorithm 2: Counting of Frequency -Broadcasting Motif

Input : $G(V, E)$ - Graph object
Output: *broad_motif_no_total* - Frequency of Broadcasting Motif Patterns,
broad_motif_per_node - Frequency of Broadcasting Motif Patterns per node

```
1  def FindBoradcastingMotif(G(V, E)):
2      broad_motif_per_node ← {}
3      broad_motif_no_total ← 0
4      forall vertex i ∈ V do
5          borad_motif_no=0
6          if Outdegree(i) ≠0 then
7              v2 ← Outdegree(i)
8              if v2 ≥ 1 then
9                  if v2 == 2 then
10                     borad_motif_no=1
11                     broad_motif_per_node[i] ← broad_motif_no
12                     broad_motif_no_total= broad_motif_no_total + broad_motif_no
13                 else
14                     broad_motif_no=(v2*(v2-1))/2
15                     broad_motif_per_node[i] ← broad_motif_no
16                     broad_motif_no_total= broad_motif_no_total + broad_motif_no
17                 end
18             end
19         end
20     end
21     return broad_motif_no_total, broad_motif_per_node
```

Algorithm 2 computes the frequency of message broadcasting sub-graphs for all nodes of the input graph $G(V, E)$. The value is stored in the variable *broad_motif_no_total*. The algorithm considers the out-degree of a vertex $i \in V$ at step 6, and if the value is not zero, stores it in variable $v2$ at step 7. If the value of $v2$ is 2, then the number of broadcasting motif is considered as 1 and the value of variables *broad_motif_per_node*[i] is set to 1 and *broad_motif_no_total* is increased by 1 accordingly at step 10 to step 12. If the value of $v1$ is not 2, the algorithm computes the number as receiving motif using the formula $v2 * (v2 - 1)/2$ and updates the value of variables *broad_motif_per_node*[i] and *broad_motif_no_total* accordingly at step 14 to step 16. The algorithm runs from step 5 to step 20 for all the vertex $i \in V$ and returns the frequency of broadcasting sub-graphs for each vertex and the total message broadcasting sub-graphs of the input graph.

Algorithm 3: Counting of Frequency - Reciprocal Motif

Input : $G(V, E)$ - Graph object
Output: $reciprocal\_motif\_no\_total$ - Frequency of Reciprocal Motif Patterns,$reciprocal\_motif\_per\_node$ - Frequency of Broadcasting Motif Patterns per node

```
1  def FindReciprocalMotif(G(V, E)):
2      reciprocal_motif_per_node ← {}
3      reciprocal_motif_no_total ← 0
4      forall vertex i ∈ V do
5          if Indegree(i) ≠0 and Outdegree(i) ≠ 0 then
6              reciprocal_motif=0
7              forall x ∈ successors of i do
8                  if i ∈ successors of x then
9                      reciprocal_motif=+1
10                     reciprocal_motif_per_node[i]← reciprocal_motif
11                 end
12             end
13         end
14         reciprocal_motif_no_total=reciprocal_motif_no_total + reciprocal_motif
15     end
16     reciprocal_motif_total /= 2
17     return reciprocal_motif_no_total, reciprocal_motif_per_node
```

Algorithm 3 computes the frequency of message reciprocal sub-graphs for all nodes of the input graph *G*(*V*, *E*). The value is stored in the variable *reciprocal_motif _no_total*. The algorithm considers the out-degree and in-degree of a vertex $i \in V$ at step 5. If the values are not zero and if all successors of *i* get *i* as a successor, *reciprocal_motif* is increased by 1 at step 9. The value of variables *reciprocal_motif _per_node*[*i*] is set to *reciprocal_motif* at step 10. *reciprocal_motif _no_total* is increased by *reciprocal_motif* accordingly at step 4 to step 14. At step 16 *reciprocal_motif _no_total* is divided by 2 to avoid the double-counting and returns the frequency of reciprocal sub-graphs for each vertex and the total message reciprocal sub-graphs of the input graph.

Algorithm 4: Counting of Frequency - Message passing Motif

Input : $G(V, E)$ - Graph object
Output: $mesg\_pass\_motif\_no\_total$ - Frequency of Message_Passing Motif Patterns, $mesg\_pass\_motif\_per\_node$ - Frequency of Message_Passing Motif Patterns per node

```
1  def FindMessagePassingMotif(G(V, E)):
2      mesg_pass_motif_per_node ← {}
3      mesg_pass_motif_no_total ← 0
4      forall vertex i ∈ V do
5          motif_mesg_pass=0
6          if Indegree(i) ≠0 and Outdegree(i) ≠ 0 then
7              v3 ← Indegree(i)
8              v4 ← Outdegree(i)
9              motif_mesg_pass=v3 * v4
10             mesg_pass_motif_per_node[i]← motif_mesg_pass
11         end
12     end
13     mesg_pass_motif_no_total=mesg_pass_motif_no_total + motif_mesg_pass
14     return mesg_pass_motif_no_total, mesg_pass_motif_per_node
```

Algorithm 4 computes the frequency of message passing sub-graphs for all nodes of the input graph $G(V, E)$. The value is stored in the variable $mesg\_pass\_motif\_no\_total$. The algorithm considers the out-degree and in-degree of a vertex $i \in V$ at step 6, and the values are stored in $v3$ and $v4$. If the values are not zero, $motif\_mesg\_pass$ is computed by $v3 * v4$ at step 9 and stored in $mesg\_pass\_motif\_per\_node[i]$. At step 13 $mesg\_pass\_motif\_no\_total$ is updated and returns the frequency of message passing sub-graphs for each vertex and the total message passing sub-graphs of the input graph.

4.2.2 Generating randomized networks

This section focuses on generating sufficiently large N number of randomized networks and counting frequencies of initialized sub-graphs.

Algorithm 5 generates a random graph, which in-degree and out-degree sequences of nodes are similar with original graph and here, the input is the in-degree sequence $deg_{in}[]$ and the out-degree sequence $deg_{out}[]$ of the original graph $G(V, E)$. The two random vertices $v1$ and $v2$ are selected by Algorithm 6 from $deg_{in}[]$ and $deg_{out}[]$ sequences and returns random vertices $v1$ and $v2$ at Algorithm 5. If $v1$ and $v2$ are different then adding an edge between $v1$ and $v2$, the values of $deg_{in}[]$ and $deg_{out}[]$ are deducted by 1 by step 11 and step 12 accordingly. If $v1$ or $v2$ are not in $deg_{in}[]$ and $deg_{out}[]$ then removing $v1$ and $v2$ by step 13 to step 18. Then this algorithm returns the random graph $G_{rand}(V, E)$. Then, the frequencies $F\_rand(GS_i)$, where $i = \{1, 2, 3, 4\}$ of the sub-graphs for each randomized directed graph of input graph $G(V, E)$ has been computed using Algorithm $1 - 4$.

Algorithm 5: Random Network Generator with similar In-degree and Out-degree sequences of Input Graph

Input : $deg_{in}[], deg_{out}[]$ - In-degree and Out-degree sequences of G(V,E)
Output: $G_{rand}(V, E)$ - New Random Graph object

```
1  def FindRandomGraph(deg_in[],deg_out[]):
2      G_rand(V, E) ← New Graph Initialization
       forall vertex x ∈ V do
3          Node_in[] ← Indegree list with similar deg_in sequence
4          Node_out[] ← Outdegree list with similar deg_out sequence
5      end
6      while Node_in and Node_out do
7          while 1 do
8              v1, v2 ← FindRandomVertexPair(Node_in, Node_out)
9              if deg_in[v1] and deg_out[v2] then
10                 G_rand ← Add Edge From v1 to v2
11                 deg_in[v1] ← deg_in[v1] − 1
12                 deg_out[v2] ← deg_out[v2] − 1
13                 if not deg_in[v1] then
14                     Node_in ←Remove v1 From Node_in
15                 end
16                 if not deg_out[v2] then
17                     Node_out ←Remove v2 From Node_out
18                 end
19             end
20         end
21     end
22     return G_rand(V, E)
```

Algorithm 6 selects the two random vertices from $deg_{in}[]$ and $deg_{out}[]$ sequences of the original graph and stores at $v1$ and $v2$ at step 3 and step 4 accordingly. Then it checks the value of $v1$ and $v2$ at step 5. If they are not equal, then $v1$ and $v2$ are returned to Algorithm 5.

Algorithm 6: Vertex Pair Generator for Random Graph

Input : $Node_{in}, Node_{out}$-In-degree and Out-degree Sequences
Output: $v1, v2$- New Random Vertex Pair

```
1  def FindRandomVertexPair(Node_in, Node_out):
2      v1, v2 ← Random Veterx Pair
3      v1 ← Random Veterx Choosen From Node_in
4      v2 ← Random Veterx Choosen From Node_out
5      if v1≠v2 then
6          return v1, v2
7      end
```

4.2.3 Computation of mean and standard deviation of random networks' sub-graphs

Standard deviation refers to the amount of variability of the sub-graphs within a data set. The mean is the average number of sub-graphs in the data set. The Z-score indicates the number of standard deviations of a given data point lies above or below the mean [37]. In this step, the mean and standard deviation of the frequency of sub-graph of each type which has been generated from random networks are computed. The value of mean and standard deviation has been used in the next step to compute the Z-score of each sub-graph. Here 100 random networks and the frequencies of four types of sub-graphs have been considered which are defined in Sect. 3.5.

4.2.4 Computation of concentration, Z-score, and significance profile (SP) & finding the significant interaction pattern

In this section, the value of concentration, Z-score, and SP of each sub-graphs has been calculated using the Eqs. (1), (2) and (3). The result has been shown in Tables 3 and 4. The high concentration, Z-score, and SP value refer to the significant interaction pattern (network motif) which are considered as a building block of an original network. After that, finding the frequencies of sub-graphs or network motifs—message receiving pattern, message broadcasting pattern, message passing pattern, and message reciprocal pattern for each node to identify their individual involvement in the interaction network. Then the traditional ranking mechanism that is the highest valued node gets the highest rank has been used to measure the influences of the nodes in the mentioned interaction patterns. Then, 1%, 2%, 3%, 4%, and 5% highest ranking nodes are selected sequentially as the initial activated or seed nodes to examine how information propagates throughout the network using three popular information propagation algorithms-independent cascade, susceptible-infected-removed, and forest fire models. For comparison study, the closeness centrality, in-degree centrality, out-degree centrality, PageRank centrality have been measured for each node and similar ranking methods have been applied to select the initial seed nodes to apply in mentioned information propagation algorithms.

5 Result and analysis

5.1 Data set description

For this paper, six online social network data sets have been considered. The source of the data sets is described in Table 2. Most data sets considered for the purpose of this work are open-source and free-to-use data sets. The non-open

source data can be made available on request. The networks have been considered as directed unweighted graphs. The graphs are modelled as edge lists, and there are no self-loops or multiple parallel edges in any of the graphs.

5.2 Experimental set-up

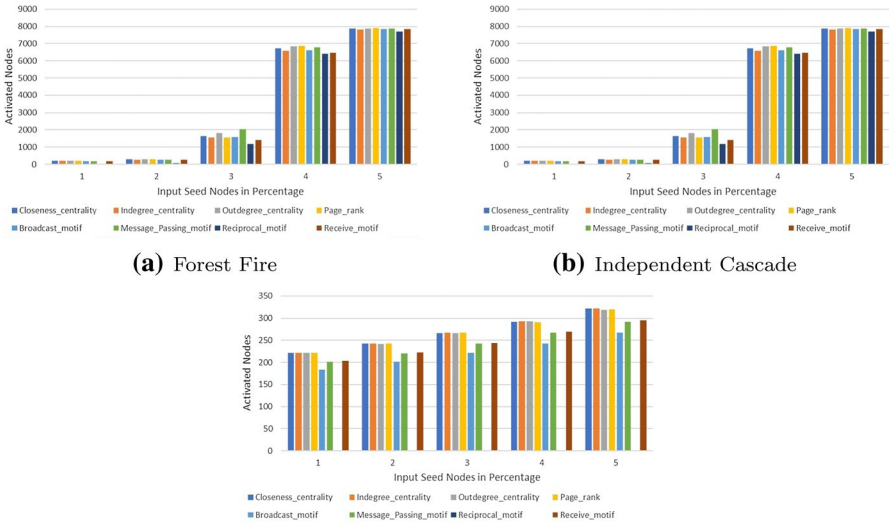
Section 4.2 provides the proposed approach, along with the algorithms used to generate random graphs and count the frequency of motifs, which are used for the purpose of this work. The stepwise breakdown of the experimental set-up is described as below -

- Generating the random graph with similar in-degree and out-degree sequences of the original graph using Algorithm 5 and Algorithm 6. An original graph is constructed using the edge lists of each of the six real-life OSNs considered. Hundred random graphs are generated for each input original graph in-degree and out-degree distribution.
- Count the frequency of each of the proposed network motifs for each of the original graphs using Algorithm 1 through 4.
- Count the frequency of each of the proposed network motifs for the random graphs corresponding to each of the original graphs using Algorithm 1 through 4. The mean and standard division of these frequencies are then calculated.
- The network motifs are validated using the validation parameters discussed in Sect. 3.1.3.
- Concentration, Z-score, and significance profile for each of the proposed network motifs for each of the original graphs are calculated to measure of their statistical significance as network motifs for a particular graph/network.
- The nodes in the original graph are ranked in descending order of the frequencies of the proposed network motifs for each of the original graphs.
- The top 1%, 2%, 3%, 4%, and 5% of the ranked nodes are considered as seed nodes for information propagation in original graphs using independent cascade, susceptible-infected-removed, and forest fire models.
- The propagation results are compared with the propagation depth of some of the baseline models for seed selection, namely, closeness centrality, in-degree centrality, out-degree centrality, and PageRank.

Since the information diffusion models considered are of probabilistic nature, and the same seed nodes in the same network can give varied results for the same diffusion model, all reported results are taken as an average of multiple runs. The experiments are conducted using on Anaconda Python 3.6 interpreter, with 8 GB RAM, Intel i5 8th Generation processor with frequency of 4 GHz. The experimental results are tabulated and represented as figures in the following section.

5.3 Result

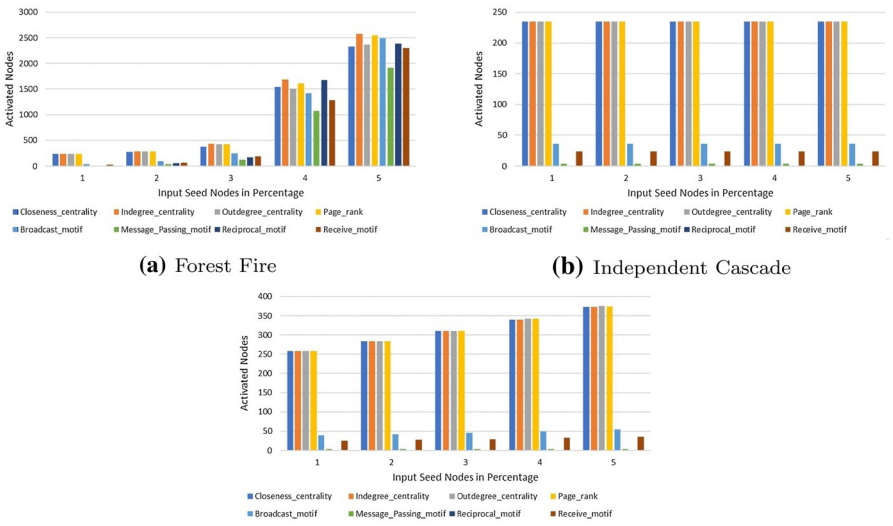
The proposed approach has been mentioned in the previous section and applied in several OSNs. The statistical measurements regarding network motifs are



(a) Forest Fire **(b)** Independent Cascade

(c) Susceptible Infected Removed

Fig. 2 Information propagation on Facebook graph

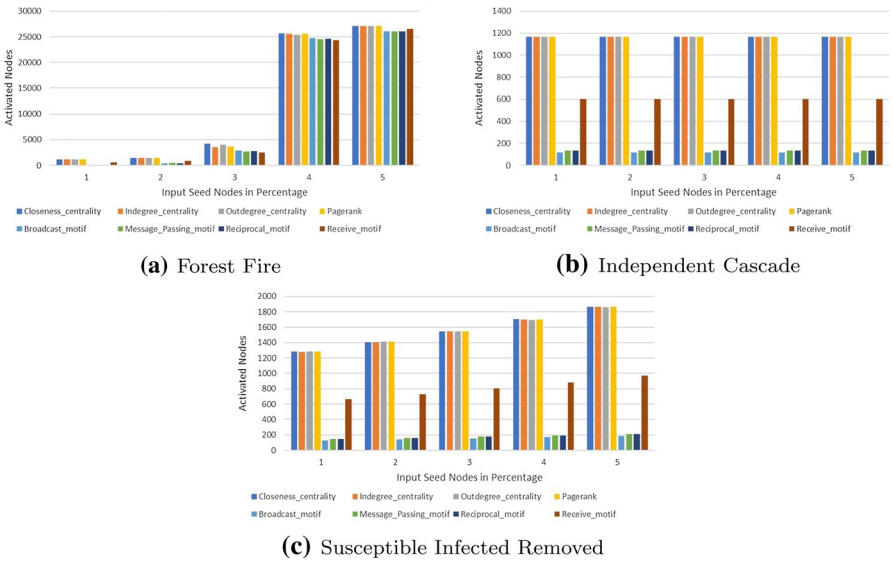


(a) Forest Fire **(b)** Independent Cascade

(c) Susceptible Infected Removed

Fig. 3 Information propagation on Twitter-scrapped graph

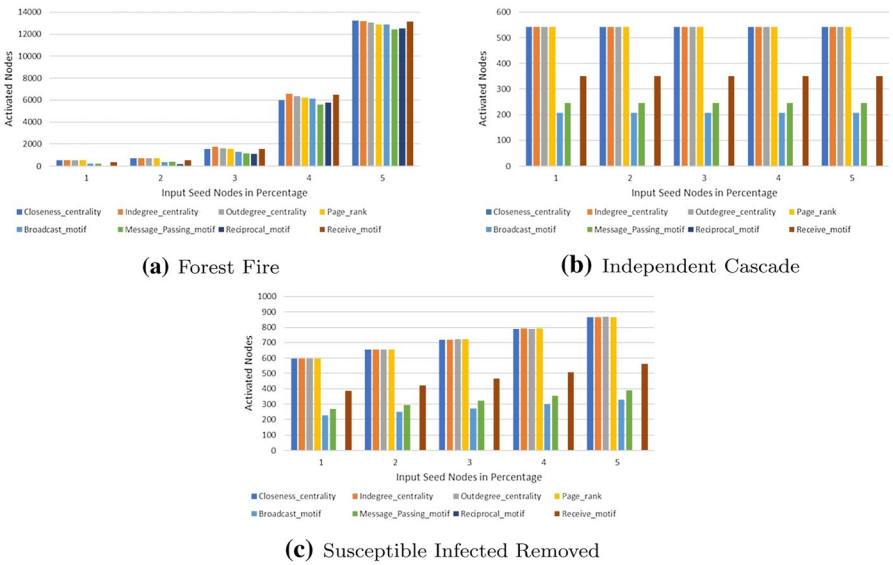
given in Tables 3 and 4. Here, six OSN data sets Table 2 have been examined. The mean and standard deviation of the frequencies of each sub-graphs of fifty random networks and Z-score of mentioned sub-graphs are represented in Table 3. Table 4 represents the values of concentration and SP of each type of



(a) Forest Fire **(b)** Independent Cascade

(c) Susceptible Infected Removed

Fig. 4 Information propagation on Email graph

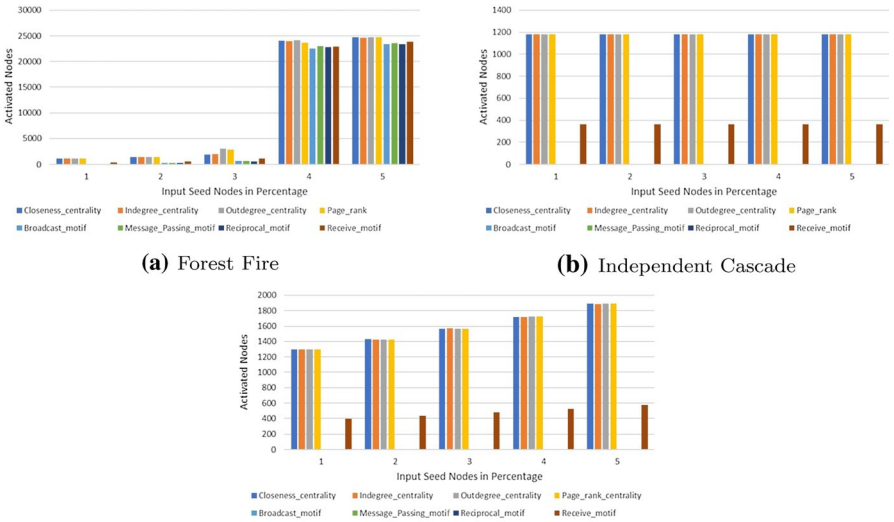


(a) Forest Fire **(b)** Independent Cascade

(c) Susceptible Infected Removed

Fig. 5 Information propagation on GNU graph

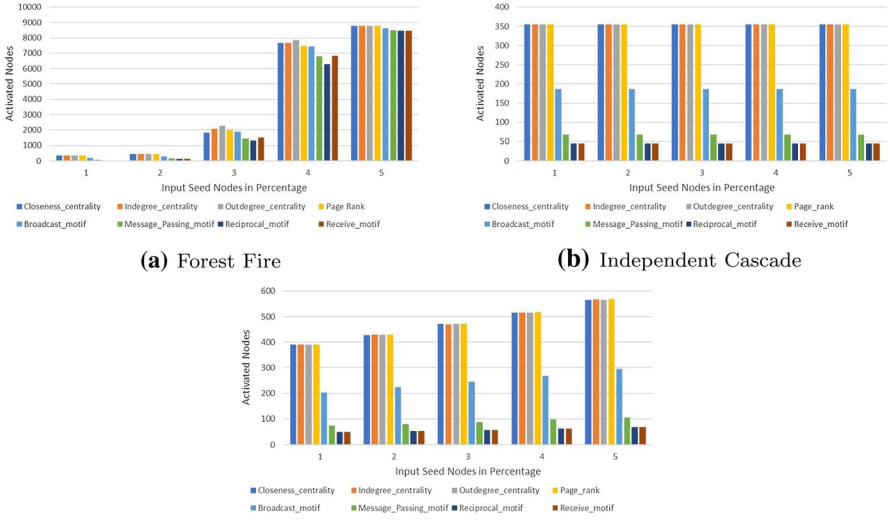
sub-graphs. The result of information propagation based on forest-fire model, information cascading model, and SIR model is shown in Figs. 2, 3, 4, 5, 6, and 7. For each data set, the initial seed nodes have been selected based on highest rank values of the closeness centrality, in-degree centrality, out-degree



(a) Forest Fire **(b)** Independent Cascade

(c) Susceptible Infected Removed

Fig. 6 Information propagation on GPlus graph



(a) Forest Fire **(b)** Independent Cascade

(c) Susceptible Infected Removed

Fig. 7 Information propagation on Wiki-vote graph

centrality, PageRank centrality, message receiving pattern, message broadcasting pattern, message passing pattern and message reciprocal pattern which are mentioned in output. In the output, it is shown that 1%, 2%, 3%, 4%, and 5% of

Table 2 Data set description

Data set name	No of nodes	No of edges	Source/references
Facebook	4039	88234	Open source [2]
Twitter-scrapped	4716	5000	Not open source
Email	23326	158726	Open source [3]
GPlus	23628	39242	Open source [4, 11]
GNU	10876	39994	Open source [5]
Wiki-vote	7115	103689	[6]

Table 3 Values of mean and standard deviation of sub-graphs in random networks and z-score of corresponding sub-graphs in original network

Dataset	Interaction pattern	Mean	Standard deviation	Z-score
Facebook	Receiving	2337039.16	7168.71	43.57
	Broadcasting	1963597.76	5087.281	395.47
	Message passing	2401185.64	4785.13	60.36
	Reciprocal	451.54	20.41	−22.12
Twitter-scrapped	Receiving	245.28	2.31	45354.41
	Broadcasting	5.18	2.36	1733.14
	Message passing	25.6	7.94	116.68
	Reciprocal	0.0	0.0	inf
Email	Receiving	10792220.62	24417.39	−217.43
	Broadcasting	2428672.04	6386.59	2927.21
	Message passing	8892559.12	24647.33	410.09
	Reciprocal	3661.26	5 1.68	792.56
GPlus	Receiving	10799406.26	36317.56	−296.24
	Broadcasting	9060.05	92.70	157008.54
	Message passing	48480.31	10664.07	13.80
	Reciprocal	1.84	1.22	37.67
GNU	Receiving	175955.94	348.24	−64.91
	Broadcasting	140042.62	301.37	149.55
	Message passing	176648.72	149.12	24.01
	Reciprocal	10.2	2.77	−3.68
Wiki-vote	Receiving	3622650.18	13073.79	50.67
	Broadcasting	780579.74	1621.52	3874.29
	Message passing	1651593.4	10999.78	262.84
	Reciprocal	451.72	19.57	126.51

highest nodes have been chosen sequentially to examine and analyse the impact of how the information propagates using the basic communication patterns and centrality measurements.

Table 4 Values of concentration and significance profile (SP) of sub-graphs

Data set	Interaction pattern	Concentration	SP
Facebook	Receiving	28.44%	0.1081
	Broadcasting	42.69%	0.9813
	Message passing	28.88%	0.1498
	Reciprocal	0.0%	−0.0549
Twitter-scrapped	Receiving	95.40%	0.9993
	Broadcasting	3.72%	0.0382
	Message passing	0.86%	0.0026
	Reciprocal	0.01%	NaN
Email	Receiving	12.01%	−0.0709
	Broadcasting	46.27%	0.9541
	Message passing	41.62%	0.1337
	Reciprocal	0.097%	0.2583
GPlus	Receiving	0.27%	−0.0019
	Broadcasting	98.40%	0.9999
	Message passing	1.32%	8.7862e-05
	Reciprocal	0.000324%	0.0002
GNU	Receiving	29.56%	−0.3938
	Broadcasting	35.69%	0.9073
	Message passing	34.75%	0.1457
	Reciprocal	0.0%	−0.0223
Wiki-vote	Receiving	26.96%	0.0130
	Broadcasting	44.44%	0.9971
	Message passing	28.58%	0.0676
	Reciprocal	0.02%	0.0326

5.4 Analysis

It can be seen that the concentration and significance profile (SP) values are different for each of the online social networks from Table 4. Reciprocal pattern has both low concentration and SP for all the networks. Thus, it is safe to say that relatively very few nodes take part in reciprocal interaction in the network. Broadcasting pattern is dominant in Facebook, GPlus, and Wiki-vote networks. Receiving pattern is highly dominant in the Twitter-scrapped network. For the Enron Email network, both broadcasting and message passing patterns are prominent and for GNU network all but reciprocal patterns are equally represented. The high concentration and SP of an interaction pattern imply that a high number of nodes take part in that pattern in the network. High broadcasting pattern signifies that the participating nodes pass out information to other nodes in the network. Thus, the senders work as source of information and they actively participate in generating and propagating information. Similarly, the receiving pattern signifies that the participating nodes accumulate information from other nodes in the network. Thus the receivers work as sinks of information. They do not actively propagate information, but they participate in

the flow. Message passing pattern signifies that the participating nodes neither act as source nor sink in the information flow, but work as facilitators, passing on received information. Reciprocal pattern signifies that the participating nodes act as both source and sink.

From Table 3, the Z-scores can be seen along with the mean and standard deviations of each of the interaction patterns for each of the networks. The Z-score implies how many standard deviations away from the mean is a value. Thus a very high positive or negative Z-score implies outliers or values which are over-represented where a low to moderate Z-score, both positive and negative, implies a closer to average value.

Combining the results from these two tables, it can be said that Facebook, GPlus, and Wiki-vote have nodes which can be used as seed nodes as they actively spread information. However, the reach of these broadcasting nodes might be limited as two-edge motifs have a reach of distance 1. For Twitter-scrapped network, most nodes play a passive role, where they receive information but do not propagate it forward. For GNU and Email network, all the interactions are evenly distributed.

Figure 2 presents the extent of information propagation on Facebook user interaction graph using forest fire, susceptible-infected-removed, and independent cascade models. The X-axis represents the number of seed nodes selected, and the Y-axis represents the number of infected nodes at the end of propagation. Figures 3, 4, 5, 6, and 7 are an equivalent representation on Twitter interaction graph, Enron Email graph, GNU graph, GPlus interaction graph and Wiki-vote graph, respectively.

In Figs. 2, 3, 4, 5, 6, and 7 it can be seen that the FF model is not spreading information more when low percentages like (1%, 2%, 3%) of seed nodes have been selected initially. The reciprocal pattern is also working significantly when (4%, 5%) of seed nodes have been selected through the reciprocal pattern has less Z-score value on Facebook. It can be also noticed that in the IC model and SIR model, the reciprocal motif is not propagating the information on Facebook and GNU networks. For each network, the total infected count is very high for the FF model. In Email and GPlus networks, the activated nodes are low for IC and SIR models. It can be observed that the FF model predicts very low spreaders count initially but later on, the count increases when the seed nodes' percentage is increased. In FF model, centrality measures and network motif patterns are spreading the information almost similar count, but in the IC model and SIR model, the centrality measures are spreading more than the different network motif patterns.

6 Applications

The basic human–interaction pattern analysis is very important in OSNs to find the influential nodes and to analyse the human activity patterns and information propagation, customer management in e-commerce, etc.

6.1 Influential nodes identification

The influential node identification is a hot topic in the social network. In social media, different interaction patterns can be observed for different groups of users, so using this basic interaction motif patterns the particular group of users can be categorized or identified. Generally, the nodes which are more involved with the broadcasting pattern, those nodes can be targeted nodes to publish a new product or brand or services in e-marketing. Hence, these are the influential nodes.

6.2 Human activity patterns and information propagation analysis

The OSN is a crucial platform for information propagation and viral marketing to political purposes. From the information propagation point of view, few nodes act as the good receiver and few nodes are good information spreader which can be identified by this methodology. The good information spreader nodes can be criminal for rumour spreading. From interaction patterns, users behaviours or activity patterns can be analysed by their profiles, which can be used to improve business and resource management in OSN. Even for rumour controlling and opinion monitoring, these interaction patterns can be used to analyse how particular users are propagating information.

6.3 Customer management in E-commerce site

E-commerce is a platform through which customers can electronically buy and sell products on online services or the Internet. The customers' activity can be analysed from the historical customer and product network. Using the interaction pattern analysis, the business organization can focus on the target nodes/customers for acquiring and retaining customers. This method can be applied to find profitable customers and popular product/services.

7 Conclusion

Earlier studies have mainly focused on network properties like degree distribution, clustering, density, shortest path, transitivity, and so on, and their effect on information propagation. Many studies have been conducted in evaluating the network behaviour as a whole from a structural perspective. In this approach, the focus is on the basic communication/ interaction patterns in OSN. The article focuses on finding the frequency of the second-order network motifs in real-life OSNs and random graphs with degree distribution similar to the real-life OSNs. The focus is also on comparing the performance of network motifs with some popular centrality measures with respect to information propagation, tested using three standard information propagation techniques: forest fire, independent cascade, and susceptible-infected-removed. The experimental results show that the performance of network motifs is

comparable to, but not out-performing, that of popular centrality measures in seed selection for information propagation. The experimental results also highlight that different network motifs are dominant for different OSNs, as well as for different propagation methods in the same OSN.

Future scope of research in related domains can include investigating the community detection in OSN based on these basic interaction patterns. Whether the nodes' basic interactions are affected by other factors such as a social event, age, gender, hometown, and profession can also be analysed. A lot of open problems such as how the basic interaction patterns affect the human behaviours, how the basic interactions patterns affect the velocity of information propagation, and other complex interaction motifs analysis can be studied in future as well.

References

1. https://mathinsight.org/generating_networks_second_order_motif_frequency
2. https://snap.stanford.edu/data/egonets-Facebook.html
3. https://snap.stanford.edu/data/email-Enron.html
4. http://konect.uni-koblenz.de/networks/ego-gplus
5. https://snap.stanford.edu/data/p2p-Gnutella04.html
6. https://snap.stanford.edu/data/wiki-Vote.html
7. Albert R, Barabási A-L (2002) Statistical mechanics of complex networks. Rev Modern Phys 74(1):47
8. Albert R, Jeong H, Barabási A-L (1999) Diameter of the world-wide web. Nature 401(6749):130–131
9. Bak P, Chen K, Tang C (1990) A forest-fire model and some thoughts on turbulence. Phys Lett A 147(5–6):297–300
10. Berg J, Lässig M (2004) Local graph alignment and motif search in biological networks. Proc National Acad Sci 101(41):14689–14694
11. Bhattacharya S, Sinha S, Roy S (2020) Impact of structural properties on network structure for online social networks. Procedia Comput Sci 167:1200–1209
12. Bhattacharya S, Sinha S, Roy S, Gupta A (2020) Towards finding the best-fit distribution for osn data. J Supercomput 76
13. Bloem P, de Rooij S (2020) Large-scale network motif analysis using compression. Data Mining Knowl Discov 34(5):1421–1453
14. Baur B, Quader S, Wong E, Huang C-H (2012) biological network motif detection: Principles and practice. Briefings Bioinf 13(2):202–15
15. Cinelli M, Quattrociocchi W, Galeazzi A, Valensise CM, Brugnoli E, Schmidt AL, Zola P, Zollo F, Scala A (2020) The covid-19 social media infodemic. Sci Rep, 10(1)
16. Ciriello G, Guerra C (2008) A review on models and algorithms for motif discovery in protein-protein interaction networks. Briefings Funct Genomics 7(2):147–156
17. Erdös P, Rényi A (1959) On random graphs i. Publicationes Mathematicae Debrecen 6:290–297
18. Freeman Linton C (1978) Centrality in social networks conceptual clarification. Soc Netw 1(3):215–239
19. Hruska J, Maresova P (2020) Use of social media platforms among adults in the united states-behavior on social media. Societies 10(1)
20. Kim Y (2013) The user's communication patterns on a mobile social network site. In: Proceedings of the 7th Workshop on Social Network Mining and Analysis, pp 1–6
21. Kumar P, Sinha A (2021) Information diffusion modeling and analysis for socially interacting networks. Soc Netw Anal Min 11(1):1–18
22. Kumar S, Saini M, Goel M, Panda BS (2021) Modeling information diffusion in online social networks using a modified forest-fire model. J Intell Inf Sys 56(2):355–377
23. Li M, Wang X, Gao K, Zhang S (2017) A survey on information diffusion in online social networks: models and methods. Information 8(4):118

24. Michienzi A, Guidi B, Ricci L, De Salve A (2021) Incremental communication patterns in online social groups. Knowl Inf Syst 63:06
25. Milo R, Itzkovitz S, Kashtan N, Levitt R, Shen-Orr S, Ayzenshtat I, Sheffer M, Alon U (2004) Superfamilies of evolved and designed networks. Science 303(5663):1538–1542
26. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U (2002) Network motifs: simple building blocks of complex networks. Science 298(5594):824–827
27. Musial K, Juszczyszyn K, Gabrys B, Kazienko P (2008) Patterns of interactions in complex social networks based on coloured motifs analysis. In: *International Conference on Neural Information Processing*, pp 607–614. Springer
28. Page L, Brin S, Motwani R, Winograd T (1999) The pagerank citation ranking: bringing order to the web. Technical report, Stanford InfoLab
29. Patra S, Mohapatra A (2020) Review of tools and algorithms for network motif discovery in biological networks. IET Syst Biol 14(4):171–189
30. Riquelme F, González-Cantergiani P (2016) Measuring user influence on twitter: a survey. Inf Process Manage 52(5):949–975
31. Rogers EM (2010) Diffusion of innovations. Simon and Schuster, New York
32. Shen-Orr SS, Milo R, Mangan S, Alon U (2002) Network motifs in the transcriptional regulation network of escherichia coli. Nature Genetics 31(1):64–68
33. Tsourakakis CE, Pachocki J, Mitzenmacher M (2017) Scalable motif-aware graph clustering. In: Proceedings of the 26th International Conference on World Wide Web, pp 1451–1460
34. Wang C, Chen W, Wang Y (2012) Scalable influence maximization for independent cascade model in large-scale social networks. Data Min Knowl Discov 25(3):545–576
35. Wang Y, Wang J, Wang H, Zhang R, Li M (2021) Users'mobility enhances information diffusion in online social networks. Inf Sci 546:329–348
36. Wilson C, Boe B, Sala A, Puttaswamy KPN, Zhao BY (2009) User interactions in social networks and their implications. In: Proceedings of the 4th ACM European conference on Computer systems, pp 205–218
37. Xia F, Wei H, Yu S, Zhang D, Xu B (2019) A survey of measures for network motifs. IEEE Access 7:106576–106587
38. Yin H, Benson AR, Leskovec J, Gleich DF (2017) Local higher-order graph clustering. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining
39. Yu S, Feng Y, Zhang D, Bedru HD, Bo X, Xia F (2020) Motif discovery in networks: a survey. Comput Sci Rev 37:100267
40. Zhao L, Beverlin BI, Netoff T, Nykamp DQ (2011) Synchronization from second order network connectivity statistics. Front Comput Neurosci 5:28

Authors and Affiliations

Sankhamita Sinha[1] · Subhayan Bhattacharya[1] · Sarbani Roy[1]

Subhayan Bhattacharya
b.subhayan@yahoo.com

Sarbani Roy
sarbani.roy@jadavpuruniversity.in

[1] Sankhamita Sinha, Meghnad Saha Institute of Technology, Kolkata, India