Original Article

IN ONLINE SOCIAL NETWORK USING GREY WOLF AND DEEP LEARNING TECHNIQUE FOR INFLUENTIAL USER PREDICTION (IUP)

P. Jothi[1], R. Padmapriya[2]

[1]Research Scholar, School of Computer Studies, Rathnavel Subaramaniam College of Arts And Science, Coimbatore, Tamilnadu, India.
[2]Head of the Department, School of Computer Studies, Rathnavel Subaramaniam College of Arts And Science, Coimbatore, Tamilnadu, India.

DOI: 10.47750/pnr.2022.13.S10.062

Abstract

The far-reaching use of Online Social Networks (OSNs) and the often growing volume of knowledge provided by their members have motivated both corporate and scientific researchers to investigate how certain systems can be manipulated. According to recent findings, monitoring and evaluating the influence of OSN users has significant applications in the fields of health, economics, education, politics, entertainment, and other fields. The propagation model has an impact on a centrality measure's capacity to show a node's ability to disseminate influence. In certain modeling techniques, the centrality measures perform well on directed contacts. However, centrality measures not perform well for indirect contacts. To improve prediction performance, additional measures and combined centrality measures are proposed by employing linear combinations of measures is proposed in this article. The deep learning-based CNN algorithm is developed for Influential User Prediction (IUP). The relevant measures selected by Grey Wolf Optimization (GWO) are fed into Convolutional Neural Network (CNN) for training and trained model for IUP. GW algorithm initializes many grey wolves, finds the optimal measures and provides the best solutions (IUP) using CNN. The GW positions are updated to look for new solutions until a near-optimal solution is found. The proposed GW-CNN is compared with CNN, CPPNP, and TDSIP and proved GW-CNN provides the best results for IUP.

INTRODUCTION

In recent years, OSNs have established themselves as one of the most powerful and efficient platforms for sharing information, opinions, and ideas, as well as promoting events and organizations. Users of OSNs publish content and receive feedback (reviews, responses, and so on) from other users who either accept or reject the material. It is often the case that when clients publish their work as they draw the interest of a large number of other users [1]. The fact that their postings acquire a significant number of answers and opinions, or that they are republished on numerous occasions, allowing them to reach a significant population of additional people, suggests that they are garnering a lot of attention. Users who can catch the attention of a huge number of individuals are known as influencers. The problem of recognizing and predicting the individuals who are influential in an OSN is essential since it offers a broad number of Possibilities in a variety of sectors, including economics and politics. Measuring impact provides for the capture of real-world user attributes, which are important for both analyzing and explaining the system's development [2] delivering practical answers to significant reality challenges, including locating audiences and creativity, one has an impact on consumers' tastes and preferences [3], recognizing brand promoters [4], determining the value of users in microblogs or

games [5], recognizing travel bloggers who have an impact on tourism destinations [6], influencing political viewpoints [7], and having an impact on the social and healthcare areas of life [8]. The extensive usage of OSNs and the variety of user-generated content are the primary elements that draw scholars and businesses to the study of the influencer's phenomena. Many organizations, for example, rely on OSNs to spread information about their identity, products, and services.

In general, this commercial tactic is carried out by selecting a group of micro-activist who are relevant to a group and have the power to impact the greatest portion of prevailing individuals in terms of marketing. The development of a new measure that captures specific elements of influence is frequently the focus of analysis of influence between OSN users. In the scientific literature, there is a range of metrics for evaluating user influence [9].

However, acquiring a full view of the users' data (e.g., posts, images, comments, emotions, retweets, etc.) in numerous real OSNs, such as Facebook, turns out to be quite complicated because the data which can be accessed is prohibited by both the OSNs' API limits and the users' privacy policies. As a result, during the preceding decade, most of the research in the area of social impact concentrated on creating methods to estimate the influence of Twitter users [10, 11, 12]. Instead, research on the influence of Facebook has concentrated on IUP information such as posts [13] or images, influential pages, or prominent users of Facebook's services [14, 15].

The creation of user communities or social groupings is another primary impact of the OSN model. Communities are very common in today's OSNs, and most of them permit users to generate groups to make sharing information with other members of those groups easier. The flow of information generated by certain organizations may have a significant impact on acquiring the members' power [9]. The interactions between group members are particularly well represented using a temporal network, which encapsulates the system's efficient communication trends and user behavior. As a result, this research provides a methodology for determining the most prominent members of communities (groups) who might play a pivotal role in existing OSNs.

Most IUP methods used only standard measures called Analysis of Variance (ANOVA) [16]. These standard measures cannot find the indirect relationship among OSN users effectively. In this paper, new measures are identified by linearly combining available centrality measures. GW search method is proposed for selecting the most relevant features for OSN users based on the position updating process and fitness value. The Influence Maximization obtained from the MSE of CNN is used as a fitness function of the proposed GW-CNN algorithm. Thus, the IUP is proposed without increasing computational complexity.

The remaining portion of this work is structured as: Section 2 gives a review of the literature on modeling and predicting impact in OSNs. Section 3 provides a general overview of the proposed methodology, which includes data collection, transformation, centrality measures, training, and prediction. The experimental results are given in section 4. Finally, section 5 discusses the conclusions.

LITERATURE SURVEY

Gong et al. (2016) introduced a new memetic technique that encompasses social media societies for influence optimization issues. To strengthen the accuracy of this method, this algorithm performs two layers of optimization: population initialization and similarity-based search method. This model includes handling influence overlap, but it is incapable of handling the huge amount of data collected on time [17].

Wang et al. (2017) suggested a novel solution to the Distance Aware Influence Maximization (DAIM) challenge. The distance between a determined group of users and their location prompts is considered in this research, and the identification of an influential group of users differs according to the recommendations [18]. Index-based techniques like Maximum Influence In (out), Absorbance (MIA), and Reverse Influence Sampling (RIS) are utilized to accomplish this case. Minimal sampling is a type of unbiased sequencing that is utilized to manage the DAIM to significantly reduce the search spaces.

Tong et al. (2017) established a model that uses an adaptive seed selection strategy with a greedy algorithm to choose prominent users. Seed users in this approach fluctuate periodically depending on the type of social network. The system's extensibility is handled via greedy and adaptive approaches [19]. The heuristic greedy technique is not suited for establishing the dynamic standalone cascade in this scenario.

Rani et al. (2017) employed a Label Propagation Algorithm (LPA) with impact relevance. The LPA is a graph-based method with a semi-supervised learning technique that works quickly. In the lack of influence centrality, the LPA works poorly in some situations, resulting in either a massively huge community or no one community to cope with anyway [20]. LPA efficiency is enhanced using a hybrid method of influential centrality.

Talukder et al. (2019) developed a few threshold approximation methods for prominent or IUP using an influence weight and degree distribution. Appropriate threshold estimation for IUP is a challenging issue. Heuristic and sample-based threshold estimation were introduced. The suggested models apply to any influence-weight estimating technique because they merely provide more specific and shorter ranges of criteria rather than a vast number of parameters. However, the threshold estimation model selects a threshold value at random from a set of created thresholds within a certain range [21].

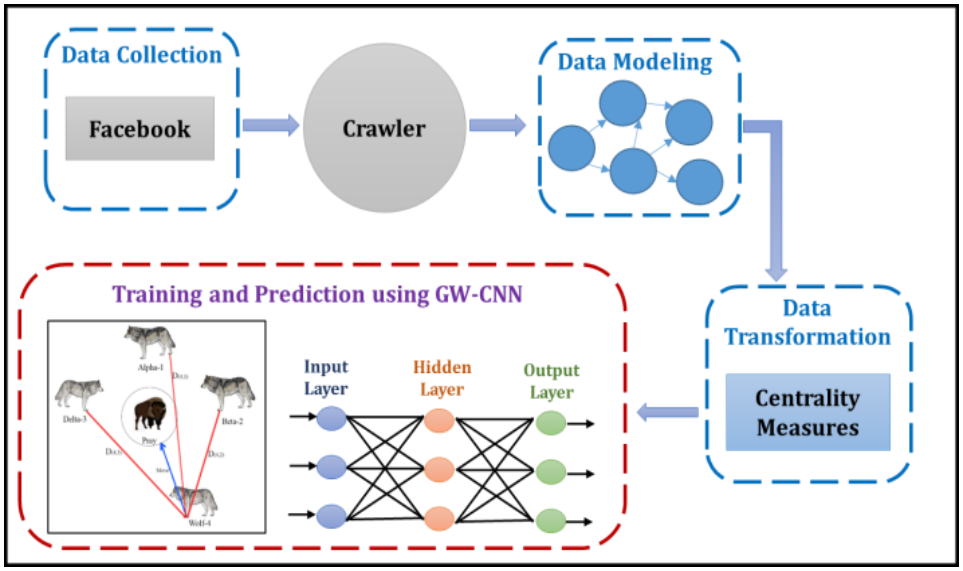
Zhao et al. (2020) proposed the k-shell method to consider multi attributes by using the sigmoid function and iterative process. The attribute values are updated in every iteration based on the location of users. The location updating found information of neighbor nodes which helps to rank users with neighbor property [22]. However, numerous sorts of investigations were carried out to prove the efficacy of the presented strategy. Rezaie et al. (2020) expanded the discovery of the dynamically varying influence of users using the time-sensitive ranking approach. This method used a wide range of features from users' activities and behaviors. The relative influence of each feature was also calculated and accounted for discovering influence [23]. Mao et al. (2021) offered a topological probable technique for forecasting prominent nodes in OSNs that takes into account both network topology and brand engagement characteristics. To overcome the limitations of previous techniques, the operation of extracting prominent portals from OSNs combines network model, brand awareness, and topological potential [24].

However, the solutions mentioned in this part gave only a small benefit in algorithm performance because variability among distinct qualities is not fully evaluated. The measurement models proposed in the literature are unable to fully exploit OSN datasets because only direct measures are used to label ground truth and for prediction. To address this issue, the new measures are formed by a linear combination of measures and the deep learning method is used for IUP in this paper.

PROPOSED METHODOLOGY

In this research work, the best-performing new centrality measures are considered. In addition to new centrality measures, linearly integrate two measurements and novel configurations of two negate measures; this results in 4 major merged coherence metrics.

FIGURE 1.Proposed architecture



Data Collection

The workflow is concerned with data collecting and guarantees that up-to-date information on people within a specific social group is acquired from the OSN. The target social group for influence prediction is chosen based on either target marketing methods or the domain of interest. The data can be collected through a crawler that investigates frequently a specific user by gathering information about items published by the members. The APIs can be used to construct applications that collect data from groups. However, Users of OSNs must authorize certain applications to acquire their data for a short time. The dataset is downloaded using a web crawler. The database contains four .csv documents.

The key postings are contained in post.csv, which may be useful for taking a fast glance at the page. Apostrophes and Commas have been substituted as {APOST} and {COMMA} in the msg field. It has the following eight characteristics: Group id (gid), Main Post id (pid), User Id (uid), Username, Timestamp, Shares, URL, Message Text, and Number of Likes comment.csv contains Facebook uploading remarks and contains eight properties such as gid, pid, Comment Id(cid), timestamp, uid, Name of user expressing, rid indicates Id of the person replying to initial post, and msg Txt. Likes and answers are included in like.csv. It contains the following information: gid, pid, cid, a response such as LIKE, ANGRY, etc., the id of the user replying (rid), and the name of the person responding. The participants of the organization are listed in member.csv. Most participants rarely or just infrequently publish or remark. To locate several records for a similar individual, the individual's name may not vary, but their account image does. When a person changes their profile image, Facebook assigns them a current id. It comprises the following information: gid, uid, the member's name, and their URL.

Data Modeling

A Group Interaction Graph (GIG) (V, E) is a directional multilayer graph in which V is a collection of vertices (or vertices) representing the groupings of members in the group. E is a set of edges that represent the instances that happened among the terminals in V, for example, $\vec{d}\,(v,w)$ e where $v,w \in V$. Comments on posts or answers to comments are examples of such events. Emotional responses are not considered occurrences in and of themselves, but rather as qualities of the events to which they refer. The label of the interaction is defined by the function $F: E \rightarrow \sum E$, and the list of possible labels is defined by $\sum E = \{Comment,\ Reply\}$.

In this scenario, the graph $\vec{G}$ is focused, on which source and target towards the interaction are determined by the sequence of nodes within every edge $\overrightarrow{d}\,(v,w)$. Each E $\vec{d}\,(v,w)$ is made up of the source vertex s (d) = v and

the destination vertex c (d) = w, sequentially. Graph G can contain several pathways among distinct apex, i.e., distinct Es $\overrightarrow{d_1}(v,w), \overrightarrow{d_2}(v,w), \ldots\ldots \overrightarrow{d_n}(v,w)$ in the graph $\vec{G}$ can all have the same source and target. Each E additionally includes information about the pertinent connection, including the type of response, the style of contact (opinion or response), and the period it was produced.

Data Transformation

The purpose of the information conversion step is to provide many key indicators that can be used to assess the consequence for group participants. A metric to explore, in addition, employs the above dynamic system design to calculate the relevance of users at different periods (Tp) t_1, t_2,...,t_s. To create a set of measurements that may be used to forecast the effect of the members.

Given a temporal network $\{\overrightarrow{T_{t_1}}, \overrightarrow{T_{t_2}}, \overrightarrow{T_{t_3}}, \ldots\ldots, \overrightarrow{T_{t_s}}\}$ of a group (where $t_1 < t_2 < t_3 < \cdots < t_s$) and a general GIG $\overrightarrow{G_{t_i}} = < V_{t_i}, E_{t_i} >$of the secular system (i.e., $t_i \in \{t_1, \ldots\ldots t_s\}$) representing the organization throughout at Tp $[t_i, t_i + \Delta)$, the time-aware centrality metrics task computes for each member$w \in V_{t_i}$ the metric vector $M_w^{t_i} = [m_1, m_2, \ldots\ldots m_n]$ where N is the dimension integer, and the vector variable m_j (for j=1,2,...,I) reflects the magnitude of the metric j in the Tp $(t_i, t_i + \Delta)$ under the prevalence.

Different prominence measures were utilized in this research, including reactions, republish, Edge Degree centrality (EDC) and Node Degree centrality (NDC), Interaction Rate (IR) and Activity Rate (AR), H-Index (HI), Betweenness Centrality (BC) and Closeness Centrality (CC) [21]. The additional measures like Page-Rank, Eigen Vector Centrality (EVC), In Degree (ID), Out–degree (OD), Strength Centrality (SC), Outbound Closeness (OC) and, Katz Centrality (KC) are considered as additional measures.

EDC. It estimates the connection quality that passes through a particular link. Regardless of the orientation of the Es, there are two separate variants of the EDC, since we assume a Generic GIG of the Dynamic Network (GGIG-DN) $\overrightarrow{G_{t_i}} = < V_{t_i}, E_{t_i} >$ of the dynamic network. The number of interactions between nodes v and w in the network is measured by the EDC e_v^+ . Similarly, the ID centrality e_v^- of a node v is defined as the count of distinct E in the network connecting a vertex w to a v. As a result, determining the size of the preceding groups produces the ID and OD of a node v.

$$e_v^+ = \{\vec{d}(v,w) \in E | v,w \in V_{t_i}|\} \quad (1)$$

$$e_v^- = \{\vec{d}(w,v) \in E | v,w \in V_{t_i}|\} \quad (2)$$

NDC. It calculates how many people engaged with a given user$v \in V_{t_i}$. In the instance of EDC, analyzing two alternate forms of the NDC indicates the orientation of the connections. The node OD centrality n_v^+ measures the amount of distinct clients who received one or more links from node v in the system. In a more formal sense, it examines the set of nodes linked by in going Es after v. Furthermore, the variety of different users that began one or more connections with the vertex v was characterized as the node ID centrality $n_v^-$of v. The sizes of the following sets determine the node ID and node OD of a node v given GGIG-DN as follows

$$n_v^+ = w \in V | \vec{d}(v,w) \in E_{t_i} \quad (3)$$

$$n_v^- = w \in V | \vec{d}(w,v) \in E_{t_i} \quad (4)$$

IR. It computes the mean amount of intersection that client v has had recently. Consider two distinct variants of the connection rate, one for each edge direction. The output IR i_v^+v+ is the proportion of different distinct IR initiated by node v(i.e.,e_v^+) to the diverse range of clients who gained such links (i.e.,n_v^+). The input interface rate $i_v^-$of node v is calculated as the proportion of the variety of diverse connections obtained by v(i.e.,e_v^-) to the number of distinct clients that initiated these interactions (i.e.,n_v^-). Given a GGIG-DN, the Output Interaction (Out-I)/Input Interaction (IN-I) percentage of a node v is estimated by the given equation as:

$$i_v^+ = \begin{cases} \frac{|e_v^+|}{n_v^+} \; if \; n_v^+ \neq \emptyset i_v^- = \begin{cases} \frac{|e_v^-|}{n_v^-} \; if \; n_v^- \neq \emptyset \\ 0 \end{cases} \\ 0 \end{cases} \quad (5)$$

It is important to note that clients who get no IN-I have an input IR of zero, whereas individuals who acquire just one IN-I for each ingoing neighbor in n v- have an input IR of 1.

AR. It determines the user's level of involvement through both Out-I and IN-I exchanges. The AR is the fraction of IN-I for node v. Given a GGIG-DN, the AR of a node v is determined as

$$a_v = \begin{cases} 0 \\ \frac{|e_v^-|}{|e_v^+|+|e_v^-|} otherwise \; if \; |e_v^+|+|e_v^-| = 0 \end{cases} \quad (6)$$

It's important to note that the user AR is between [0, 1]. To be more specific, if a user v has not performed any IN-I in the graph (i.e.,$e_v^- = 0$), the AR is 0, and if v has not undertaken any Out-I (i.e.,$e_v^+ = 0$), the AR is 0. When a client's ARis more than 0.5 and the customer has more IN-I than Out-I, he gets contacted. v is classified as a producer if it has more Out-I than IN-I or if its AR is less than 0.5.

HI. It is commonly used to assess a scientist's or scholar's output as well as the influence of his or her articles in terms of citations. An index of h indicates that the user has published h articles, each of which has been referenced at least h times in other works. The HI H(X) of a limited amount of reals X = (x_1,x_2, …..x_i) provides the greatest integer h such that (x_1,x_2, …..x_i) contains at least h elements, each of which is bigger than h. The H-index hv of a node v calculates, for every neighbor $w \in n_v^-$ the value of intended Es from w to s, i.e., $|\vec{d}(w,v)|$, given the collection n v- of nodes who have interacted with v in $\overrightarrow{G_{t_i}} = < V_{t_i}, E_{t_i} >$. Finally, the function H is given the finite series of interactions between the neighbors and the user v as an income variable.

$$h_v = H(|\vec{d}(s_1,v)|, |\vec{d}(s_2,v)|,\ldots, |\vec{d}(s_i,v)|) \quad s_1, s_1, \ldots\ldots, s_i \in n_v^- \quad (7)$$

CC. The sum of the lengths in shortest pathways between the node and all other nodes in the graph is used to compute a node's global importance in the network. The CC is determined using the GGIG-DN equations $\overrightarrow{G_{t_i}} = < V_{t_i}, E_{t_i} >$ and a node v ∈ V:

$$c_v = \sum_{w \in V_{t_i}} \frac{1}{d(w,s)} \quad (8)$$

where d is a variable that computes the range across w and s. The Dijkstra Shortest Route Algorithm (DSPM) is employed to determine values in a graph by taking the minimum line with the lowest expense. According to the standard concept of BC, all weights in the network must be equal to one and numerous Es among nodes must be eliminated. This idea was enhanced by employing a translation approach that computes the strength across two vertices as the inverse of the association value. Although v is near to all individuals of the unit, a node v with a high CC will participate with a large number of group participants frequently.

BC. The BC of a node determines its worth in terms of enabling information to propagate among the disconnected set of participants. The fraction of directed shortest routes via that specific node is used to compute the node's betweenness centrality which is provided as

$$bwc_v = \sum_{s \neq v \neq w} \frac{\Sigma_{sp}(v)}{\Sigma_{sp}} \quad (9)$$

Where $\sum sp$is represents the overall amount of simplest pathways among the consumers s and p, and $\sum sp\,(v)$ is the proportion of fewest pathways among s and p that traverse the node v. A user v with a high BC acts as a bridge between team participants who do not communicate with one another. The DSPM is used to find the proportion of shortest pathways connecting two nodes. The DSPM is used to compute distances in a system based on the shortest expense route.

The additional metrics are used for IUP are as follows

PR. The basic idea is that more important domains are more certain to receive more connections from other web pages, and it is used by search engines to determine the worth of online pages. The chance that a user would visit a specific site is represented by the PR, which is based on the chance of arbitrarily selecting a relationship from the present tab and the chance of going to a page chosen at arbitrary from the entire internet [25]. In the PR method, each vertex symbolizes a website, and the rating awarded to it may be regarded as the fraction of the moment spent by users contacting that vertex in a stochastic process (following outgoing Es from each vertex).PR modifies this stochastic procedure by adding the model as a chance, alpha, of passing to each vertex of the given Group Activity Graph. When alpha is 0, the EVC technique is employed; when alpha is 1, all vertices are assigned a similar value (1/ V_{t_i},). Because usual alpha levels are in the [0.1, 0.2] range, the variable alpha must be set to 0.15, however, it may be any integer from 0 to 1 inclusively.

EVC. It is a metric of a node's centrality that determines the node's neighbors. EVC of a node is influenced by the quantity and significance of its neighbors. It is usually calculated by adding the EVC of immediate neighbors, although it may also be approximated using iterative techniques based on random walks in specific cases [26].

$$x_v = \frac{1}{\lambda}\sum_{\mathrm{t}\in \mathrm{M(v)}} x_t = \sum_{\mathrm{t}\in \mathrm{G}} \mathrm{av,txt} \tag{10}$$

$$Ax = \lambda x \tag{11}$$

ID and OD. DC may be measured in two ways in a directed network: ID and OD [24]. The amount of connections directed to the node is counted as ID, while the number of connections directed to others is counted as OD. As a result, their equations are as follows:

$$\sum_{v\epsilon V} d_v^- = |A| \tag{12}$$

$$\sum_{v\epsilon V} d_v^+ = |A| \tag{13}$$

SC. SC is simply a weighted variant of DC [27]: rather than making use of the neighborhood density, the sum of all occurrences Es of u is added together.

$$\mathrm{S}_u = \sum_{j=1}^{\mathrm{N}} a_{ij} w_{ij} \tag{14}$$

OC. The total proximity among node v and the rest of the nodes, it can reach (outbound centrality) or all other nodes that can connect v (inbound centrality), and the different criteria of both sets [28]. The outbound connectivity set of v has a size of the following.

$$\vec{R}[v] = |\{u\ 2\ V \setminus \{v\} \mid v \rightsquigarrow u\}|, \tag{15}$$

Where v, u denotes that u may be reached from v. Similarly, v's inbound reachability set is rather large.

$$\overleftarrow{R}[v] = |\{u\ 2\ V \setminus \{v\} \mid u \rightsquigarrow v\}| \tag{16}$$

As a result, the total distance to the outward reachability set of v is defined as

$$\vec{S}[v] = \sum_{u|v \rightsquigarrow u} dvu \tag{17}$$

As well as the overall distance to v's inbound reachability set

$$\overleftarrow{S}[v] = \sum_{u|u \rightsquigarrow v} duv \tag{18}$$

The inverse of ratios is used to establish the outbound and inbound centralities. Closeness on the inside

$$\vec{S}[V]/\vec{R}[v] \text{ and} \overleftarrow{S}[v]/\overleftarrow{R}[v] \tag{19}$$

KC. KC is a metric of a node's relative importance in a network, considering both immediate and non-immediate adjacent nodes that are linked by direct neighboring nodes [29]. A node's KC is calculated as follows:

$$CKatz(vi) = \alpha \sum_{j=1}^{n} \text{Aj, iCKatz(vj)} + \beta \quad (20)$$

where α is a damping factor, which is generally lower than that of greatest eigenvalue, i.e. α<1/and β is a bias invariant, also known to be exogenous matrix, this is used to reduce zero centrality scores.

Based on the previous ideas for merging distinct metrics into novel versions, a linear combination of a local and a global centrality measure is presented to combine both viewpoints of an unit. This method unites a node's local and global perspectives. The developed aggregate centrality metrics offer a good forecasting validity for node influence propagation. In terms of correlation, monotonicity, and running duration, these metrics will be extension of the extant measure, greatly outperform the existing model.

Training and Prediction

The training and forecasting phase uses the converted measures gained from the Information Extraction step to estimate which groups will have the greatest IU and to quantify the degree of such impact.

Training

A training set is used, as is customary, to develop the forecasting model. The training set includes many S normalized samples $C_u^{t_1}, C_u^{t_2}, \ldots . C_u^{t_s}$, all of which are associated with a user u and are sorted by the time those happened. The series of samples of the user u on which the Forecasting work is done is denoted by the input variable s, and each converted observation $C_u^{t_i}$ corresponds to a genuine observation $M_u^{t_i}$. Depending on the duration of the temporal network's segment interval, each evaluation is conducted at a distinct time (hourly, daily, or weekly) in this framework.

For example, if the temporal network's granularity is 1 day, one approach is to provide a training set for user u that includes inferences from the past 7, 14, or 21 days. Instead, if the temporal network's granularity is one week, an acceptable training set may be the most recent 4, 8, or 12 weeks of data. To investigate the impact of distinct time resolutions on the temporal network used different granularity values for the temporal network in these experiments, such as daily or weekly. In addition, varying the training set sizes by evaluating a different subset of the available data in this experiment.

Each converted training set observation $C_u^{t_i} = [c_1, c_2, \ldots . c_{N'}]$ is connected with a particular participant u at a Tpt_i and consists of N′ variables indicating the modified metrics created by the Data Selection phase. To derive an individual impact score of u from the converted observation $C_u^{t_i} = [c_1, c_2, \ldots . d_{N'}]$, use the following technique to integrate the contributions of each component.

If the number of modules in the information N′ is identical to 1, the impact rating of the observation is simply equal to the score of each component. However, if the number of elements in the observation N′ is greater than 1 (i.e., N′ > 1), the influence value should be calculated by adding the component values. The corresponding influence score is derived in this case by simply putting the individual factor levels together. In formal terms, an individual's influence score $I_{t_i}(u)$ at a particular time ti equals $I_{t_i}(u) = \sum_{j=1}^{N'} C_u^{t_i}[j]$.

Prediction

Although several different prediction algorithms may be utilized to find the most IU, each with its own set of advantages and weaknesses, this work fully concentrates on the GWO-CNN in this paper since it is an effective prediction model that has been effectively employed in the area of OSNs. GW initializes many gray wolves, each wolf finds the best measures and gives solutions that are IUP using CNN. In CNN, each neuron works as a kernel in the convolutional layer, which is built up of a series of convolutional kernels. The convolution operation becomes a correlation function if the kernel is symmetric.

In the GWO algorithm, first, to accelerate the population as Xi = I = 1,2,....n population size and parameter a, R, and S. During the hunting process, the wolves change their locations depending on theα, β, γ. The mathematical model for hunting is as follows:

$$\overrightarrow{H_\alpha} = |S_1.\overrightarrow{X_\alpha} - \vec{X}| \quad (21)$$

$$\overrightarrow{H_\beta} = |S_2.\overrightarrow{X_\beta} - \vec{X}| \quad (22)$$

$$\overrightarrow{H_\gamma} = |S_3.\overrightarrow{X_\gamma} - \vec{X}| \quad (23)$$

Where$\overrightarrow{X_\alpha}, \overrightarrow{X_\beta}, \overrightarrow{X_\gamma}$ denotes the locations ofα, β, γ. S_1, S_2, and S_3. The various sets of random vectors are S1, S2, and S3, and $\overrightarrow{X}$ denote the recent solution point. When the distances have been defined using the following formula, the ultimate location matrices for the response will be determined as

$$\overrightarrow{P_t} = \overrightarrow{X_\alpha} - R_1.\overrightarrow{H_\alpha} \quad (24)$$

$$\overrightarrow{P_2} = \overrightarrow{X_\beta} - R_2.\overrightarrow{H_\alpha} \quad (25)$$

$$\overrightarrow{P_3} = \overrightarrow{X_\gamma} - R_3.\overrightarrow{H_\gamma} \quad (26)$$

$$\overrightarrow{X_t}(t+1) = \frac{\overrightarrow{P_t} + \overrightarrow{P_2} + \overrightarrow{P_3}}{3} \quad (27)$$

The random vectors are represented by R1, R2, and R3, respectively, and the total amount of rounds is denoted as t. Assume that the number of layers L in the result layer, and the number of terminals in the output nodes are denoted as NL. Furthermore,

t^{ip} is the input vector of the target and $\left[o_1^L, \ldots . o_{N_L}^L\right]'$ is the output vector. As a result, for the input ip in the output layer, the Mean Square Error (MSE) may be computed as follows: The MSE metric is used to analyze the efficacy of the performed CNN for each response in the search domain. To calculate the fitness value by using the formula

$$= MSE\left(t^{ip}, \left[o_1^L, \ldots . o_{N_L}^L\right]'\right) = \sum_{i=1}^{N_L} (o_i^L - t_i^{ip})^2 \quad (28)$$

The first stage is to produce a random collection of solutions. Each answer is paired with a wolf, which denotes its location. The wolf with the best fitness is known as the Alpha, while the wolves with the next and following highest fitness are known as the Beta and Delta wolves, accordingly. The surviving wolves are known as Omega wolves. To determine the location of the prey, the GW algorithm picks more essential measures for IU based on the locations of the Alpha, Beta, and Delta wolves (optimal solution). To put it another way, these three wolves calculate the position of the target, and the omega wolves alter their location based on these wolves' positions to come nearer to the predators.

ALGORITHM 1. Grey Wolf Optimization with Convolutional Neural Network (GW-CNN)

Input: Highest ranking of iterations and population size

Output: Predicted IU

Step 1: Designate the population of GWO X_i = (i=1,2,….n population size)

Step 2: Initialize parameters a, R and S

Step 3: For every response X_i in the GWO population do

Step 4: Assign feature subset configuration to each wolf

Step 5: Determine new attribute subsets

Step 6: Provide the classifier with the created attribute subset.

Assess the fitness of the feature subset.

$$fit_{ip} = MSE\left(t^{ip}, \left[o_1^L, \ldots . o_{N_L}^L\right]'\right) = \sum_{i=1}^{N_L} (o_i^L - t_i^{ip})^2$$

Step 7: End for

Step 8: Step 8: Choose a characteristic depending on its fitness value.

Step 9: Step 9: Make a conscious note of the most optimum feature subset.

Step 10: Assume the position α, β, γ and set the best solution $X_{\alpha,}$ X_{β}, X_{γ}

Step 11: While (i_n< n) do

Step 12: For every response Xi in the grey wolf population do

Step 13: Update the position of Xi

$$\overrightarrow{X_i}(t+1) = \frac{\overrightarrow{P_t} + \overrightarrow{P_2} + \overrightarrow{P_3}}{3}$$

Step 14: End for

Step 15: Update a using the following formula

$$a = 2\left(1 - \left(\frac{t-1}{t_{max}-1}\right)^2\right)$$

Step 16: Update $X_{\alpha,}$ X_{β}, X_{γ}

Step 17: Raise the number of iterations by one.

Step 18: Finish while

Step 19: Create a CNN model based on the solution variables X_{α}

Step 20: Predict the most IU in the test set using the CNN

Step 21: Stop the process

The proposed GW-CNN algorithm uses a grey wolf optimizer to optimize a one-dimensional Convolutional neural network. By using the hyperparameters optimized by the GW, the CNN is constructed, which involves the convolutional layers, pooling layer, fully connected followed by the softmax layer to forecast the most IU with the best solution and lowest error rates. GW is employed for selecting the most relevant influential measures and CNN is utilized in the prediction of random variables in test data. Due to finding the ideal choices of topologies, training CNNs is considered a challenging and demanding problem with an undefined search process. On the other hand, the GW algorithm's stability of local and global search stages is effective, which may be highly useful for handling difficult issues such as CNN training. As a result, substantial exploration of the GW algorithm is required to be extremely efficient as a CNN learner.

RESULT AND DISCUSSION

Dataset Description

Facebook has greatly expanded its functionality by allowing users to create numerous sorts of groups with diverse criteria. Users may establish three distinct sorts of groups, according to the Facebook Help Center.

Depending on the group's privacy settings, all Facebook groups need member permission, from either the group's administrator or a member of the group. Everyone can see who is a member of a public group and read the posts they make, and solely the present participants of the group may create a post.

An open group is open to anybody who wants to join. Closed groups have postings that can only be accessed and published by the active members of the group. A closed group can be requested by anyone. Finally, hidden groups can only be accessible if the group's administrator (or a member) has asked the user to join, and only the group's existing members can post and view the group's postings. A Facebook crawler program is used to get the interactions that happened in a set of specified Facebook groups to achieve a huge collection of diverse groups with varied features.

This work primarily focuses on public or closed groups because they may be searched by anybody or joined by becoming a member. Accessing secret groups, on the other hand, requires a request from an associate member. Facebook groups are classified in to different categories like Education, Politics, News, Entertainment, and Dalliance.

FIGURE 1.Influential users for 1 days and 1 week

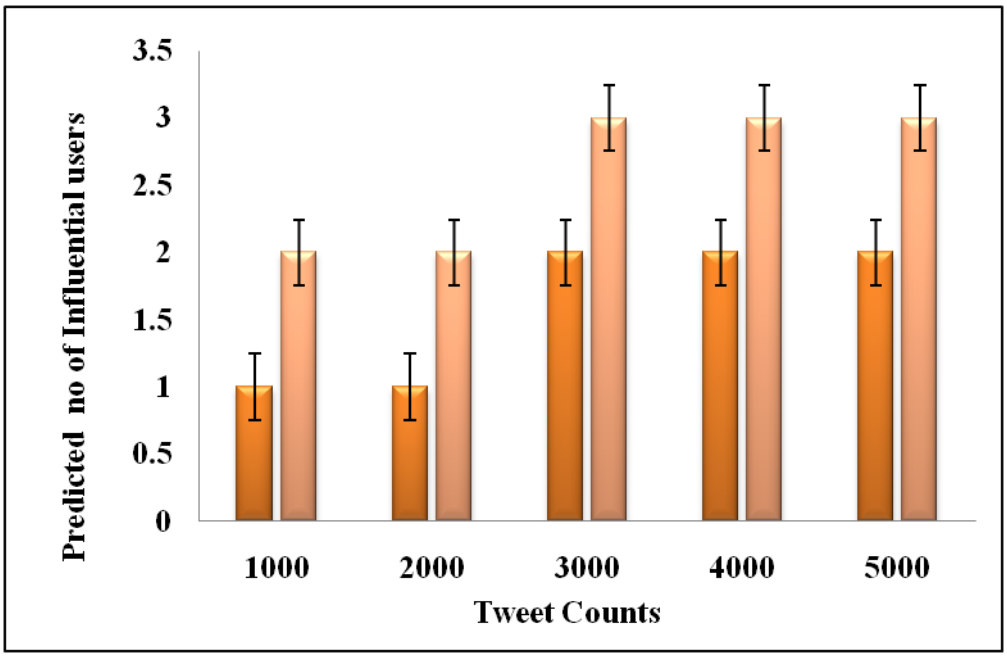


Figure 1 depicts the method's performance in identifying IU for various tweet populations. The IU for 1000 to 5000 tweet counts are depicted in the chart. In this case, the maximum number of correctly predicted influencers is equal to 3 for Δ = 1 day and to 5 for Δ = 1 week. Instead, the selections of 3 or more components do not improve the accuracy of the framework. Furthermore, it was possible to anticipate at least two of the significant individuals of around 80% of the groupings by utilizing the framework.

Predictive Performance

Detection performance is frequently measured using precision, recall, and accuracy. The formulas used to compute these measures are as follows:

$$Precision = \frac{TP}{TP+FP} \quad (23)$$

$$Recall = \frac{TP}{TP+FN} \quad (24)$$

$$Accuracy = \frac{TP+TN}{FN+TP+FP+FN} \quad (25)$$

The approaches' prediction efficacy is quantified using precision, recall, and accuracy measurements. The suggested GW-CNN, CNN, CPPNP (Combined Personalized Propagation of Neural Predictions) [29] and TDSIP [30] are compared (Time-Aware Domain-based Social Influence Prediction). Table 1 and Figure 2 present the findings of the training phase, whereas Table 2 and Figure 3 show the findings of the testing set. In all databases, the CNN, CPPNP, and TDSIP predictive models outperform the GW-CNN forecasting algorithm.

TABLE 1. Evaluation metrics for training set

Prediction Algorithm	Precision	Recall	Accuracy
GW- CNN	0.81	0.83	0.82
CNN	0.76	0.78	0.76
CPPNP	0.71	0.74	0.72
TDSIP	0.68	0.71	0.69

FIGURE 2.Evalutation metrics for training set

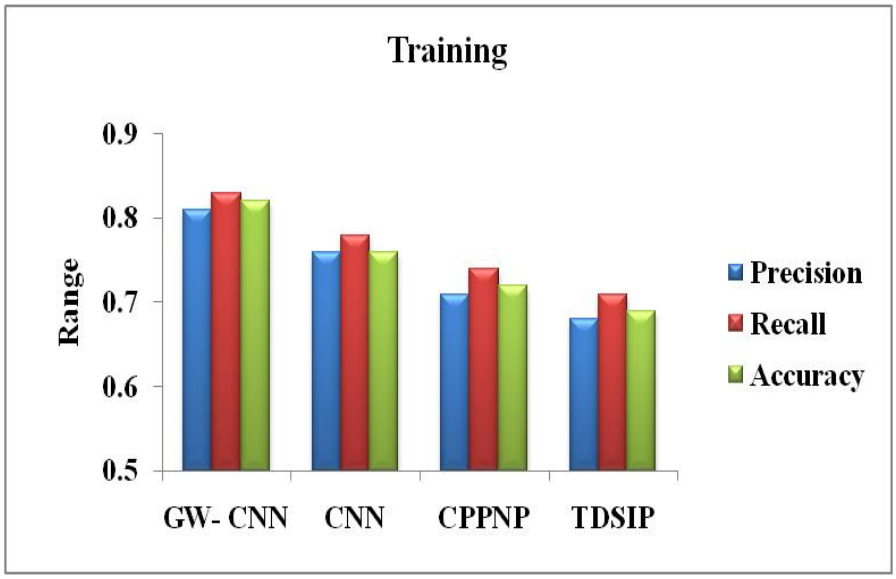


Figure 2 presents the findings of the different frameworks during training phase. It indicates that the precision of the proposed GW-CNN is 19.12% higher than the TDSIP, 14.08% higher than the CPPNP and 6.58% higher than the CNN [31]. The recall of the proposed GW-CNN is 16.9% higher than the TDSIP, 12.16% higher than the CPPNP and 6.41% higher than the CNN. Similarly, the accuracy of the proposed GW-CNN is 18.84% higher than the TDSIP, 10.81% higher than the CPPNP and 5.13% higher than the CNN [32]. Thus, it concludes that the presented GW-CNN achieves better efficiency during training phase than all other frameworks for predicting influential users [33].

TABLE 2. Evaluation metrics for testing set

Prediction Algorithm	Precision	Recall	Accuracy
GW- CNN	0.84	0.86	0.85
CNN	0.79	0.81	0.79
CPPNP	0.73	0.74	0.74
TDSIP	0.71	0.72	0.71

FIGURE 3.Evalutation metrics for testing set

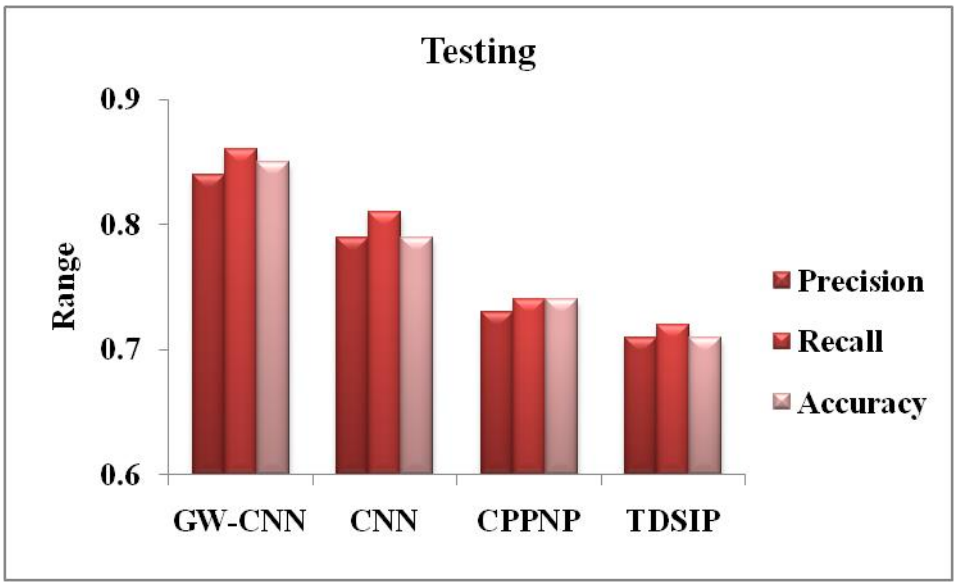


Figure 3 presents the findings of the different frameworks during testing phase. It indicates that the precision of the proposed GW-CNN is 18.31% higher than the TDSIP, 15.07% higher than the CPPNP and 6.33% higher than the CNN. The recall of the proposed GW-CNN is 19.44% higher than the TDSIP, 16.22% higher than the CPPNP and 6.17% higher than the CNN. Similarly, the accuracy of the proposed GW-CNN is 19.72% higher than the TDSIP, 14.86% higher than the CPPNP and 7.59% higher than the CNN. Thus, it concludes that the presented GW-CNN achieves better efficiency during testing phase than all other frameworks for predicting influential users.

CONCLUSION

In this paper, a CNN framework was presented, which solves the Influencer User Prediction (IUP) challenge. First, new centrality measures were computed and aggregated. Then, GW algorithm was applied to select more relevant measures, which were learned by the CNN to predict the influential users effectively. Moreover, this framework was tested in a Facebook scenario by analyzing the interactions of roughly 5000 individuals tweets representing a variety category (News, Education, Politics, Sport, and Entertainment,). This research covered some surprising data on the effect of members. The results of the investigations reveal that our methodology achieves 85% accuracy compared to the classical frameworks, which means this framework can forecast the members who would end up on the list of future prominent members for all the groups by using an additional group of centrality measures.

REFERENCES

1. J. Backaler, “Business to consumer (b2c) influencer marketing landscape”, in *Digital Influence* (Springer 2018), pp. 55–68.

2. Asha et al., Environmental Research **205**, 112560-74 (2022).

3. R. Pálovics and A. A Benczúr, Soc. Netw. Anal. Min. **5**, 1-12 (2015).

4. A. Majumdar, D. Saha, and P. Dasgupta, “An analytical method to identify social ambassadors for a mobile service provider's brand page on facebook”, in *Applications and Innovations in Mobile Computing* (IEEE 2015), pp. 117–123.

5. W. Ponchai, B. Watanapa, and K. Suriyathumrongkul, “Finding characteristics of influencer in social network using association rule mining”, in *Proceedings of the 10th International Conference on e-Business* (2015).

6. F. Magno and F. Cassia, Anatolia, 288-290 (2018).

7. B. E Weeks, A. Ardèvol-Abreu, and H. Gil de Zúñiga, Int. J. Public Opin. Res. **29**, 214-239 (2017).

8. J. Zhou, F. Liu, and H. Zhou, Perspect. Public Health, **138**, 173–179 (2018).

9. F. Riquelme and P. González-Cantergiani, Inf. Process. Manage. **52**, 949–975 (2016).

10. A. Deborah, A. Michela, and C. Anna, Heliyon **5**, 1-7 (2019).

11. Jain, Deepak Kumar, et al. IEEE Transactions on Industrial Informatics **18.7**, 4884-4892 (2021).

12. H. U. Khan, S. Nasir, K. Nasim, D. Shabbir and A. Mahmood, Expert Syst. Appl. **164**, 1-9 (2021).

13. C. C. Hsia and C. T. Li, C. T. J. Inf. Sci. Eng. **37**, 935-958 (2021).

14. S. Peng, Y. Zhou, L. Cao, S. Yu, J. Niu, and W. Jia, J. Netw. Comput. Appl. **106**, 17-32 (2018).

15. M. A. Al-Garadi, K. D. Varathan, S. D. Ravana, E. Ahmed, G. Mujtaba, M. U. S. Khan, and S. U. Khan, ACM Comput. Surv. **51**, 1-37 (2018).

16. M. Trusov, A. V. Bodapati, and R. E. Bucklin, J. Mark. Res. **47**, 643-658 (2010).

17. M. Gong, C. Song, C. Duan, L. Ma, and B. Shen, IEEE Comput. Intell. Mag. **11**, 22-33 (2016).

18. X. Wang, Y. Zhang, W. Zhang and X. Lin, IEEE Trans. Knowl. Data Eng. **29**, 599-612 (2017).

19. G. Tong, W. Wu, S. Tang, and D. Z. Du, IEEE/ACM Trans. Netw. **25**, 112-125 (2016).

20. S. Rani and M. Mehrotra, “Hybrid influential centrality based label propagation algorithm for community detection”, in *International Conference on Computing, Communication and Automation* (2017), pp. 11-16.

21. A. Talukder, M. G. R. Alam, N. H. Tran, D. Niyato, G. H. Park, and C. S. Hong, IEEE Access **7**, 105441-105461 (2019).

22. N. Zhao, J. Bao, and N. Chen, Complex. **2020**, 1-15 (2020).

23. B. Rezaie, M. Zahedi, and H. Mashayekhi, Knowl. Inf. Syst. **62**, 3481–3508 (2020).

24. Y. Mao, L. Zhou, and N. Xiong, IEEE Trans. Netw. Sci. Eng. **8**, 529-540 (2021).

25. A. De Salve, P. Mori, B. Guidi, L. Ricci and D. Di Pietro, ACM Trans. Knowl. Discov. Data **15**, 1–50 (2021).

26. Rajagopalan Arul, et al. Energies **15**, 9000-9024 (2022).

27. F. A. Parand, H. Rahimi, and M. Gorzin, Phys. A: Stat. Mech. Appl. **459**, 24-31 (2016).

28. M. de Laat, V. Lally, L. Lipponen, and R.J. Simons, Comput. Support. Learn. **2**, 87–103 (2007).

29. A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani, Proc. Natl. Acad. Sci. **101**, 3747-3752 (2004).

30. E. Cohen, D. Delling, T. Pajor, and R. F. Werneck, “Computing classic closeness centrality, at scale”, in *Proceedings of the second ACM Conference on Online Social Networks* (ACM, 2014), pp. 37–50.

31. Anupama et al., Computers, Materials and Continua **70**, 1297-1313 (2021).

32. A. Cuzzocrea, C. K. Leung, D. Deng, J. J. Mai, F. Jiang, and E. Fadda, Procedia Comput. Sci. **177**, 170-177 (2020).

33. B. Abu-Salih, K. Y. Chan, O. Al-Kadi, J. Big Data **7**, 1-37 (2020).