

Article

Community-Aware Evolution Similarity for Link Prediction in Dynamic Social Networks

Nazim Choudhury 

Department of Computer Science, University of Wisconsin, Green Bay, WI 54311, USA; choudhun@uwgb.edu

Abstract: The link prediction problem is a time-evolving model in network science that has simultaneously abetted myriad applications and experienced extensive methodological improvement. Inferring the possibility of emerging links in dynamic social networks, also known as the dynamic link prediction task, is complex and challenging. In contrast to the link prediction in cross-sectional networks, dynamic link prediction methods need to cater to the actor-level temporal changes and associated evolutionary information regarding their micro- (i.e., link formation/deletion) and mesoscale (i.e., community formation) network structure. With the advent of abundant community detection algorithms, the research community has examined community-aware link prediction strategies in static networks. However, the same task in dynamic networks where, apart from the actors and links among them, their community pattern is also dynamic, is yet to be explored. Evolutionary community-aware information, including the associated link structure and temporal neighborhood changes, can effectively be mined to build dynamic similarity metrics for dynamic link prediction. This study aims to develop and integrate such dynamic features with machine learning algorithms for link prediction tasks in dynamic social networks. It also compares the performances of these features against well-known similarity metrics (i.e., ResourceAllocation) for static networks and a time series-based link prediction strategy in dynamic networks. These proposed features achieved high-performance scores, representing them as prospective candidates for both dynamic link prediction tasks and modeling the network growth.

**Citation:** Choudhury, N.Community-Aware Evolution Similarity for Link Prediction in Dynamic Social Networks. *Mathematics* **2024**, *12*, 285. <https://doi.org/10.3390/math12020285>

Academic Editor: Pasquale De Meo

Received: 14 November 2023

Revised: 31 December 2023

Accepted: 5 January 2024

Published: 15 January 2024



Copyright: © 2024 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: social network analysis; dynamic networks; link prediction; community detection; actor dynamicity**MSC:** 91C03; 05C85

1. Introduction

Many real-world systems and complex phenomena can now be modeled as a network where actors represent the entities or individuals and links denote their relationships or inter-dependencies. Due to the ubiquity of such real-world network intrinsic applications in various disciplines, dynamic network data have recently become widely prevalent where network events are time-stamped. One of the inherent underlying structures of these networked systems is their evolution over time experiencing temporal changes in overall network dynamics. Understanding the mechanism by which these evolutions occur is yet to be congruously standardized. However, network science has offered various methods supporting the study and modeling of the network evolution process that governs their dynamics [1]. Among them, link prediction is the fundamental computational problem that models the underlying growth mechanism of evolving networks [2]. Due to its primacy in understanding the evolution of networks, the link prediction mechanism of complex social networks has attracted extensive research attention. Subsequently, a wide range of methodological improvements are also engendered to support this part of link analysis. Most of these methods attempt to estimate the possibility of the emergence of new links among non-connected network actors by leveraging topological properties; actor

and link attributes; local, global, or quasi-local network structures [3]; or probabilistic models [4]. The dependency on feature engineering [5] and failure to acknowledge the temporal changes that emanate in dynamic networks [6] are two major encumbrances of these methods. Furthermore, despite being called a time-evolving model, different link prediction strategies generally overlooked the evolutionary aspects of the network to take into account.

In an evolutionary (i.e., ‘longitudinal’, ‘temporal’, or ‘dynamic’) network, temporal patterns emerge through simultaneous arrivals and/or departure of actors including the creation and/or deletion of links among these actors. Actors (i.e., nodes) in dynamic social networks are subject to varying dynamicity regarding their network positions, neighborhoods, and communities formed within the temporal network snapshots. Temporal variations in different network activities (e.g., forming or severing links) result in temporal changes in actors’ structural positions and neighborhoods. Furthermore, these actor-oriented microscopic network changes may result in mesoscopic alterations of network structure (e.g., community). These facts led the scholars to take into account the evolutionary community related information in the dynamic network in the link prediction task.

Communities in social networks implicitly denote groups of actors with similar features or attributes or actors closely tied according to their roles, social interests, or collective behavior. As attributes, social patterns, roles, and interests of actors change over time, so do their network activities and association patterns. These result in fluctuations in both local and global network structures. Due to the evolutionary patterns of link structures of actors in dynamic networks, they eventually endure existing community membership or gain new membership to different communities. Consequently, communities of actors may shrink or increase in size or completely disappear, erode, or engender new ones over time. Therefore, it is believed that in evolving social networks, temporal microscale actor-level changes trigger mesoscopic or collective changes. By mining the similarity between actor-level temporal microscale (i.e., neighborhood changes) and mesoscale (i.e., community membership) fluctuations, it is possible to generate dynamic similarity metrics (i.e., dynamic features) for dynamic link prediction. Therefore, this study sought to develop such dynamic features by analyzing and mining temporal community-aware information, incident to actors, in dynamic networks. The contributions of this study are as follows: First, this study defines the rate of actor-level evolutionary changes regarding community memberships and associated neighborhood changes over time. Second, it computes dynamic features by mining community evolution representing the similarity between non-connected actors. To compute these features, it considers network structure, temporal information, and evolutionary community-aware information over several network snapshots. Third, this study conducted extensive experiments and evaluations of these features via supervised machine learning algorithms to measure their performance in dynamic link prediction tasks.

2. Related Work

The researchers explored historical or temporal information [7] in conjunction with a wide range of techniques to infer the possibility of future links among the network actors of dynamic networks—known as link prediction in dynamic (i.e., temporal) networks or dynamic (i.e., temporal) link prediction. Myriad methods were designed for this purpose that differ in the comprehension of the temporal nature of the networks and the definition of the network property to be preserved. The researchers used different centrality measures [8] as the influence factor of the network actors that support both the prediction of future associations among them and capturing their associated network structural changes in dynamic networks [9]. For example, Zhang et al. [10] used the eigenvector centrality measures for temporal link prediction that compared and contrasted the contributions of common neighbors in the emergence of future links. Chi et al. [11] categorized the actors in the dynamic network by considering their evolving influence strength in comparison to their neighbors and used this factor to compute the attraction force as the connection probability among them. Most of the primitive methods of link prediction used either

heuristics or different structural and network topological features to compute individual similarities as the connection probability between actors. The most comprehensive list of similarity indices based on both neighborhood topological and nodal attributes was presented by Bliss et al. [12], where the authors used the covariance matrix adaptation evolution strategy to compute the weight of individual similarity measures.

The time-varying nature of dynamic networks and the inter-dependency between the evolutionary patterns and link prediction in dynamic networks make the task of dynamic link prediction even more challenging. Therefore, researchers also delved into other techniques including dynamic latent space representation of actors and random walk-in temporal networks [13], probabilistic temporal measures [14], probabilistic generative models [15], matrix and tensor factorization [16], and deep learning methods [17]. Divakaran and Mohan [18] developed a taxonomy of dynamic link prediction methods based on various approaches and categorized them into five main classes including: (i) time series approaches, (ii) probabilistic approaches, (iii) matrix factorization, (iv) spectral clustering, and (v) deep learning methods. The frequently evolving structure of the network makes the time series approach a promising option for dynamic link prediction. Studies [19–21] in this approach deployed various time series of actor's centrality measures, network structural features, or various similarity indices between each node pair, along with forecasting models for predicting the future values of the features or indices as the connection probability between node pairs. For example, Wu et al. [22] considered the eigenvector-based nodal centrality in conjunction with a forecasting method (i.e., adaptive weighted moving) for link prediction in dynamic networks. A few techniques [23,24] of dynamic link prediction employed probabilistic models that deployed maximum likelihood approaches or probability distributions. This allowed them to consider variations and quantify the uncertainty around emerging links. As an effective tool for large-scale data processing and analysis, matrix factorization-based methods factorize (i.e., decompose) a matrix into its constituent factors to simplify complex operations. Dynamic link prediction methods based on matrix factorization [25,26] represent the network property in the form of a matrix (e.g., adjacency matrices) and factorize this matrix to form the features for performing the link prediction task. Finally, by exploring the properties of a graph via eigenvalues and eigenvectors of the adjacency matrix or Laplacian matrix associated with the graph, a few studies [27,28] of dynamic link prediction exploited the spectral graph theory. These techniques utilized a low-rank approximation approach that supports large-scale graphs where the matrix factorization approach does not fit well.

Nevertheless, the high representational ability [29] of a Deep Belief Network (DBN) built upon the Restricted Boltzmann Machine (RBM) allowed researchers to use deep learning approaches to solve the dynamic link predicting problems. A few earlier deep-learning-based dynamic link prediction models concentrated on modeling an RBM, which is a special case of Markov random field. The basic function of these models is to incorporate temporal and neighbor information to train an RBM over a sequence of observations of the dynamic network structures to compute the connection probability among neighbors. In one of the earliest models, Li et al. [7] proposed a generative model called the conditional temporal restricted Boltzmann machine(ctRBM) that can integrate the neighbors' influence and individual transition variance in a dynamic network with nonlinear transitional patterns to compute the connection probability between neighbors. Among recent techniques, Jinyin Chen and coauthors [30] developed a graph convolution network(GCN) embedded long-short-term memory(LSTM) deep-learning model known as GC-LSTM. It was capable of learning spatiotemporal features that extracted the network structural features from dynamic temporal network snapshots via graph convolution and learned temporal structure through LSTM. Their model can predict both the emerging links and obtain accurate predictions of the whole dynamic network evolution. Recently developed techniques such as network representation learning [31] and graph embedding techniques [32] promoted the representation of graphs in a low-dimensional vector space that not only preserved the network properties but also eased the strenuous feature engineering process. The

objective of the embedding-based dynamic link prediction approaches [33,34] is to predict the emerging links of a network from the low-dimensional embedding vectors.

Despite their improved performance in predicting emerging or hidden links, some of these aforementioned methods are subject to their inherent limitation. For example, probabilistic models require a prior definition of the distribution of link occurrences, which is strenuous for temporal networks. Some probability-driven models (e.g., exponential random graphs) are only suitable for small networks. Furthermore, matrix- or tensor-based methods are not feasible for real-time link prediction in large networks due to their inherent complexity regarding computational and processing time requirements [35]. Additionally, the data-hungry deep learning methods not only rely on large amounts of labeled data to train and optimize their models effectively but also employ an extra layer of complexity in the representation of data and the prediction process [36].

Researchers also exploited network community information in dynamic link prediction. Yuhang Zhu et al. [37] considered the concept of collective influence in percolation optimization theory—an effective attribute of nodes, community multi-feature fusion, and embedded representation to predict links in dynamic networks. Their method integrated collective influence, the community random-walk features, and the centrality features for dynamic link prediction. In a more recent study, Kumar et al. [38] proposed a dynamic link prediction algorithm that used the parameterized influence area of actors and their contribution to community partitions. Their method considered different features based on local, global, and quasi-local similarity and community information. By mining the temporal patterns of evolutionary changes associated with actors concerning their neighborhoods and communities in the dynamic networks, Choudhury and Uddin [39] developed dynamic similarity metrics (i.e., dynamic features) for supervised dynamic link prediction.

3. Evolutionary Community and Dynamic Similarity Metrics

In most social networks, there are parts where actors are more densely connected than the rest of the network. These condensed regions are known as clusters or communities and consist of actors with common properties, objectives, or goals. With the wide adoption of networks to understand the social interaction pattern, the term ‘community’ started representing closely knitted actors demonstrating certain common characteristic structural properties [40]. According to Santo Fortunato [41], global and local heterogeneous link distributions within networked systems spawn community structures within networks. In an evolving social network, interactions among its actors evolve dynamically over time, leading to a similar change pattern in their community patterns. The underlying reasons are divergent; for example, actors may change their roles, acquire new links, sever old ties with others, and new actors and links emerge. Simultaneously, in the course of network evolution, owing to various network events, actors may join or leave a community, resulting in shrinking or expanding the size of communities; merging, splitting, or diminishing the existing communities; or even engendering new ones. By considering these facts, this study first attempts to define the community-aware dynamicity experienced by actors in dynamic networks. It then derives dynamic features by mining the aspects of the actor-level evolutionary information that embodies the evolutionary changes in the actor’s community participation.

3.1. Community Dynamicity

Many real-world networks are longitudinal, involve dynamic processes, and evolve temporally. A dynamic network consists of a time series of network snapshots where each snapshot represents the corresponding network state at a particular timestamp. Each snapshot in this temporal sequence is known as a *short-interval network (SIN)*. The degree of temporal fluctuations, incident to actors in the dynamic networks, in regards to their link structures, neighborhoods, and network positions in every SIN, represents actor-level dynamicity [42,43]. The term ‘actor dynamicity’ refers to the variable involvement of indi-

vidual actors in dynamic social networks. In conjunction with the actor dynamicity, varying roles, and divergent network activities simultaneously trigger changes in social communities within these network snapshots. Communities may appear, disappear, merge, split, shrink, expand, or even sometimes remain unmodified without incurring any changes. In Figure 1A, this phenomenon is visualized with the help of two abstract SINs at two different timestamps (i.e., t_1, t_2) in a dynamic network metaphor. The sizes of actors in the network snapshots are proportionate to their degree of connections, and four actors are accompanied by their clustering coefficient values at two different timestamps. This figure demonstrates various aspects of actor-level temporal microscale changes resulting in community-aware mesoscale network alterations. For example, in this figure, at the time t_2 , actor a_4 changes its community as a result of its neighborhood changes. Likewise, the clustering tendency of actors changes as a result of altering link structures among neighborhoods; however, acquiring more neighborhoods does not implicitly extend cliquishness. It is also evident that varying neighborhood and actors' network positional changes simultaneously affect their clustering disposition.

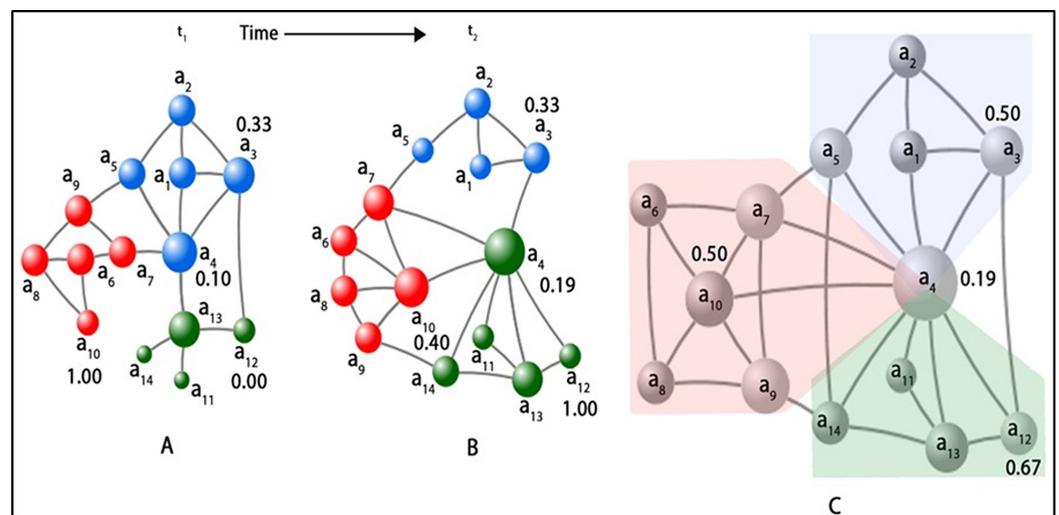


Figure 1. An abstract visualization of a dynamic network comprised of two short-interval networks (SIN) (A) G_1 at time t_1 and (B) G_2 at time t_2 and (C) an aggregation of G_1 and G_2 (i.e., $G_1 \cup G_2$). Each SIN has three communities represented by three different colors, and actors within these communities represent the color of the corresponding community. Actors a_3, a_4, a_{10} , and a_{12} are accompanied by their clustering coefficient values in G_1, G_2 and the aggregated network on the right.

Understanding the evolutionary patterns of network communities, actors, and their community participation may support researchers commending the underlying network evolution. Particularly, it can assist social scientists in comprehending the underlying growth pattern of social networks. Different types of evolutionary changes, evident from Figure 1, are triggered by temporal variations of different network activities performed by network actors, which were envisaged by Uddin et al. [42].

Subsequently, the authors developed two different types of actor-level dynamicity metrics (i.e., positional and participation) to quantify the temporal variations of actors regarding their network position and participation in dynamic networks. The authors also pointed out that according to social network topology, a dynamic network needs to be analyzed in regards to the temporal aggregation of links among its actors and simultaneously using both the static and dynamic topology of social network analysis. Embracing their perception of actor dynamicities, this study attempted to compute community-aware actor dynamicity. The term ‘community dynamicity’ in this study denotes the degree of evolutionary changes regarding an actor’s participation in communities or its clustering tendency in SINs over time against the temporally aggregated network in conjunction with the corresponding neighborhood changes over time. An example of an aggregated

network is portrayed in Figure 1B, where the network is the union of G_1 and G_2 . The rationale behind using the aggregated network is that the link prediction mechanism of network science predominantly deals with network growth, and in dynamic network analysis, links are aggregated by considering an aggregation window size to accumulate links temporally. In network theory, the actor’s *clustering coefficient* measures the degree to which actors in networks tend to cluster together. Since in social networks, actors tend to build friendships with other friends of their friends, this coefficient measures the extent to which one actor’s friends are also friends. Considering complex social networks, this measure is an important metric to characterize both global and local cliquishness of actors and the network in regards to the *triadic closure* mechanism that characterizes the network evolution. Triadic closure emerges when friends to a common friend become a friend as well, and this is a general phenomenon in social networks. This study considers the local clustering coefficient with a view to understanding the actor-level evolution instead of the network itself. The clustering coefficient of an actor v in a network snapshot G_t at the timestamp t is defined as:

$$CC_v(t) = \frac{2T_v}{D_v(t)[D_v(t) - 1]} \tag{1}$$

where T_v is the number of triangles an actor v is part of, and $D_v(t)$ is the degree of the actor v (i.e., the number of direct neighbors of actor v) in a network snapshot G_t at time t . Note that a triangle is a set of three actors where each actor has a link to the other two. In graph theory, it is sometimes referred to as a 3-clique. Subsequently, an actor’s community dynamicity using its clustering coefficient is measured in a SIN at timestamp t as follows:

$$\delta_v(t) = \left[\frac{|CC_v(t) - CC_v(t-1)|}{CC_v(T)} \right]^{e^{\left[\frac{2|N_v(t) \cap N_v(t-1)|}{D_v(t) + D_v(t-1)} \right]}} \tag{2}$$

where $CC_v(t)$ represents the local clustering co-efficient, and $N_v(t)$ denotes the neighborhood of actor v in SIN G_t at timestamp t , and $CC_v(T)$ denotes the local clustering co-efficient of the actor v in an aggregated network. The aggregated network is the union of two SINs at two adjacent timestamps (i.e., $G_t \cup G_{t-1}$). The numerator in the base part of Equation (2) represents the ratio of the rate of clustering coefficient changes in an actor in two adjacent SINs at timestamps t and $t - 1$. On the other hand, the denominator represents the clustering coefficient of that actor in an aggregated network consisting of SINs at those timestamps. The denominator basically normalizes the difference in the numerator by using the cliquishness of the actor by what it ought to achieve in a static network without losing any links. This score is further amplified by an exponent measuring the neighborhood achievement and retention score of that actor at two adjacent timestamps using the Sorensen index [44] of the actor’s neighborhood in G_{t-1} and G_t . For example, from Figure 1, the community dynamicity, as defined in Equation (2), of the actor a_{10} at timestamp $t = 2$ is calculated as $\left[\frac{(0.40-1.0)}{0.50} \right]^{e^{\left[\frac{2 \times 2}{2+5} \right]}} = 1.381$ in G_2 . Similarly, for actor a_4 , at timestamp $t = 2$, the community dynamicity value is measured as $\left[\frac{(0.19-0.10)}{0.19} \right]^{e^{\left[\frac{2 \times 4}{5+7} \right]}} = 0.233$. In this way, a time series of community dynamicity values for each actor considering SINs at each timestamp of a given dynamic network was built to develop dynamic features.

3.2. Time Series Forecasting

Researchers have used time series analyses and forecasting methods to model the changing behavior of network structure and predict its future values of topological alteration [19,20]. According to them, using time series to acquire historical information in relation to the topological changes of non-connected node pairs can increase the performance of time-series-based link prediction. In time series forecasting, past observations of a time variable can be analyzed to develop a model that describes the underlying re-

lationship, and extrapolation can be used to predict the future values of the variable. In this study, a univariate time series of actor-specific community dynamicity measures was considered to emulate the evolution of actors' positions or behaviors in evolving communities. A well-known forecasting model known as exponential smoothing [45] was used to predict the future values of community dynamicity values. In this method, forecasts are the weighted averages of historical observations and the weights of the observations decay exponentially with time. Single exponential smoothing (SES) with a weight of α is the simplest exponential smoothing method. The forecast equation can be defined as:

$$\hat{y}_t = \alpha y_{t-1} + (1 - \alpha)\hat{y}_{t-1} \quad (3)$$

where \hat{y}_t represents the forecasted value that depends on both the previous observations and previous forecasts. Linear exponential smoothing (LES) is a variation of this method that refines SES with a β component and considers any short trends in the series. The forecasting equation for LES can be described as:

$$\hat{y}_{t+h|t} = l_t + hb_t \quad (4)$$

where $l_t = \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1})$ and $b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1}$. Here, l_t is an estimate of the level of the series at time t , b_t denotes an estimate of the trend (i.e., the slope) of the series at time t , α is the smoothing parameter for the level, and β is the smoothing parameter for the trend in which $0 < \alpha, \beta \leq 1$. Notably, there are 15 variations of the exponential smoothing process (interested readers should refer to the work by Hyndman and Athanasopoulos [46] on forecasting methods and principles).

3.3. Dynamic Similarity Metrics

This subsection describes methodologies followed in this study to build dynamic similarity metrics or dynamic features used for dynamic link prediction. Three dynamic features were developed: first, by computing the temporal similarity; second, by measuring the correlation between temporal sequences of community-dynamicity values; and finally, considering temporal community-aware information in SInS over time incident to both actors of a non-connected actor pair.

3.3.1. Temporal Similarity of Community Dynamicity

In this approach, to define the similarity and/or proximity between actor pairs, this study compares the time series information comprised of the community dynamicity values, as calculated from the aforementioned section, incident to non-connected actor pairs. The temporal similarity or proximity between actor pairs was defined regarding the similarity of their community dynamicity values over time computed by the dynamic programming-based time series similarity approach. In time series analyses, dynamic time warping (DTW) provides intuitive distance measurements between temporal sequences by ignoring global and local deviations in the time dimension [47]. It measures the similarity between two time series by shrinking or expanding or simply "warping" the time axis of one (or both) sequences to achieve better alignment.

Considering two different time intervals (t_1, t') , (t_1, t'_1) , where $t_1 < t' < t'_1$ and a finite set of discrete time points within the range $[t_1 - t']$ as $T = t_1, (t_1 + \tau), (t_1 + 2\tau), \dots, (t_1 + n\tau), \dots, (t' - \tau), t'$, where τ denotes the temporal sampling interval, a dynamic social network $G_T = (V, E_T)$ consists of a set of uniquely labeled actors $V = \{v_1, v_2, v_3, \dots, v_n\}$ and $E_T = \{e_t(v_i, v_j, t) | v_i, v_j \in V; t \in T\}$, where t represents the timestamp of a link e between a pair of actors (v_i, v_j) in G_t . In addition, dynamic networks can be undirected, where $e = (v_i, v_j)$ and $e = (v_j, v_i)$ denote identical or directed links where these two links are not the same. Thus, the dynamic network is composed of an evolutionary sequence of network snapshots $G_T = \{G_{t_1}, G_{t_1+\tau}, G_{t_1+2\tau} \dots G_{t_1+n\tau} \dots G_{t'-\tau}, G_{t'}\}$ known as short-interval networks (SInS). Fluctuations in the total number of actors are taken into consideration across the time series of network snapshots. Any link may appear in multiple network

snapshots at different timestamp(s). Considering this temporal sequence of network snapshots $G_T = \{G_{t_1}, G_{t_1+\tau}, G_{t_1+2\tau} \dots G_{t_1+n\tau} \dots G_{t'-\tau}, G_{t'}\}$, for a given pair of actors (v_i, v_j) , dynamic link prediction attempts to predict the link probability between them during the interval (t', t'_1) in G_{T+1} by analyzing the link formation and the temporal information in G_T at timestamps $[t_1 - t']$ as $T = t_1, (t_1 + \tau), (t_1 + 2\tau), \dots, (t_1 + n\tau), \dots, (t' - \tau), t'$. Here, $G_T[t_1, t']$ and $G_T[t', t'_1]$ are considered as the networks in the training and testing phases, respectively. For each SIN at different discrete time intervals of G_T and for the network G_{T+1} in the test phase, this study built two time series of community dynamicity values incident to actor a and b that are $X^a = x_1, x_2, x_3, \dots, x_m, Y^a = x_1, x_2, x_3, \dots, x_m$. Here X^a and Y^b are time series of length $|m|$, and $|n|$ consists of community dynamicity measures for actors a and b using Equation (2) where $m, n \leq N$, and N is the total number of network snapshots in the training and the test phase. Note that X_m and Y_n are the forecasted community dynamicity values generated by the exponential smoothing process. A local cost/distance measure $d(x_i, y_j)$ was defined to compare two different points in X^a and Y^b . The goal of the DTW technique is to find an optimal alignment between X^a and Y^b with a minimum overall distance. The notion of this alignment depends on the definition of an (m, n) -warping path, which is a sequence $p = p_1, p_2, p_3, \dots, p_l$ with $p_l = (m_l, n_l) \in [1 : m][1 : n]$ for $l \in [1 : L]$ where L denotes the length of the warping path. The optimal warping path between X^a and Y^b is defined as a warping path p^* with the minimum distance among all possible warping paths. To accomplish that, it may encounter that a single point in one time series may be mapped to multiple points of the other. Figure 2 presents a visual presentation of the framework to generate the first dynamic similarity metric constructed in this study. In this figure, the solid green and red lines represent the community dynamicity values at each SIN during the training period, and the dotted lines represent the forecasted dynamicity values for the corresponding actor. The black arrow lines represent the mapping path utilized to measure the similarity between actor a and b using the DTW method. The similarity score, calculated by dynamic similarity metrics, is generated by the accumulated distance cost of this optimal mapping path. The temporal similarity between the time series of actors' community dynamicity values represents the similarity between actors in regard to their evolutionary community-aware information. Therefore, the value of the first dynamic similarity metric for actor pairs a and b considering community dynamicity values is defined as follows:

$$sim_1(a, b) = d_{p^*}(\delta_i^a, \delta_j^b) = \min[\sum_{l=1}^L d(\delta_{m_l}^a, \delta_{n_l}^b)] \tag{5}$$

where p is in (m, n) warping path

3.3.2. Correlation-Based Similarity

Correlation analysis is a statistical evaluation method that is used to quantify the strength and direction of the linear association between two variables. It is widely used in financial network analysis, asset allocation, portfolio optimization, and risk management [48]. This study applied correlation analysis to measure the affinities or similarities between actor pairs in the temporal sequences of dynamicity values in all SIN. The assumption here is that two actors are similar if they fluctuate in a similar fashion considering the community dynamicity measurement (i.e., dynamicity values of one actor increase or decrease with the other at the same time). If $\delta_a(t)$ and $\delta_b(t)$ denote the community dynamicity values of actor a and b at time t , then the evolution similarity between them is computed by using the Pearson correlation coefficient. Therefore, the second dynamic similarity metric to measure the similarity between actor pair a and b in this study is computed as follows:

$$sim_3(a, b) = \frac{\sum_t [(\delta^a(t) - \bar{\delta}^a)(\delta^b(t) - \bar{\delta}^b)]}{\sqrt{\sum_t (\delta^a(t) - \bar{\delta}^a)^2} \sqrt{\sum_t (\delta^b(t) - \bar{\delta}^b)^2}} \tag{6}$$

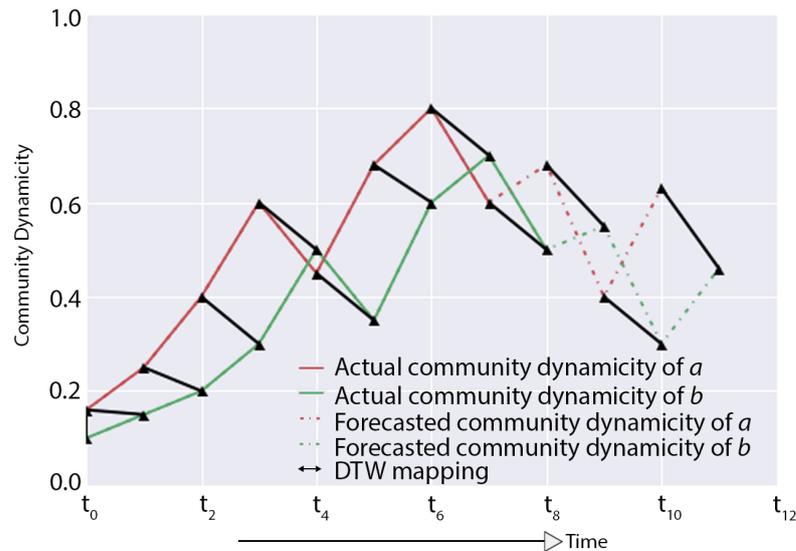


Figure 2. A visual representation of the framework to generate dynamic features by considering the temporal similarity of community dynamycity values. The solid green and red lines represent network measures (e.g., degree centrality) of actor a and b in short-interval networks during the training phase. The dotted lines represent the forecasted network measures during the test phase. The black lines represent the mapping path considering similar points of two time series using the dynamic time-warping technique.

3.3.3. Temporal Community-Aware Network Structure

In order to employ community-aware information in link prediction tasks, it is imperative to partition a network into communities. Most community-aware link prediction methods exploit an existing community detection algorithm to compute the similarity among actor pairs considering the community-oriented structural information. For example, the “InfoMap” [49] algorithm minimizes the length of random walks and is mostly used in information theory. Likewise, Valverde-Rebaza and Lopes [50] used the ‘Label Propagation’-based community detection method [51] to develop a similarity measure for the purpose of link prediction in static networks. Following them, this study used the Louvain algorithm [52] and the greedy agglomerative hierarchical community detection algorithm proposed by Newman [53] for community detection purposes. The former method has been successfully and widely used for detecting communities in many different types of large networks with millions of actors and links. As a greedy optimization method, the Louvain method optimizes the modularity by looking for smaller communities locally with optimized modularity (i.e., numerical index to evaluate partitions in a network) and aggregating actors belonging to the same community to build a network where individual communities act as an actor. The latter method of community detection follows a greedy approach to optimize and maximize the modularity and produces a tree-like dendrogram to present the hierarchical rendering of the network communities. This algorithm can efficiently cluster a large number of actors while generating the given number of communities and is also known well for its scaling capability. The third and final dynamic similarity metric in this study was computed using the community-aware information extracted from the communities detected by the aforementioned algorithms in each SIN of a given dynamic network. For each non-connected actor pair, in each SIN using the identified communities, the similarity between a pair of actors was computed depending on their community participation, including the structures of communities. Before delving into the actual similarity/proximity score between actor pairs, let us first define a few preliminary concepts and notations that are used in the following sections with the help of Figure 3.

Peripheral Actors: If an actor simultaneously belongs to more than one community or resides in one community but belongs to one end of a link where the other end belongs to another actor from a different community, that actor is considered a peripheral actor. Simi-

larly, if an actor is connected to another actor that has multiple community memberships, it is also considered a peripheral actor. For example, the green-colored actor a_4 in Figure 3 is a peripheral actor that has multiple community memberships. Similarly, the red-colored actors $a_3, a_5, a_7, a_8, a_9, a_{14},$ and a_{15} are considered peripheral actors for their respective communities since they are either part of links transcending more than one community or connected to an actor having multiple community memberships. If C_i and C_j are two communities in a SIN G_t and V_i and V_j denote the set of actors belonging to these two communities, then a peripheral actor is denoted by $v_t^{i,j}$. A set of peripheral actors between two communities C_i and C_j in a SIN G_t is denoted by $|V_t(i, j)|$, where $i \neq j$.

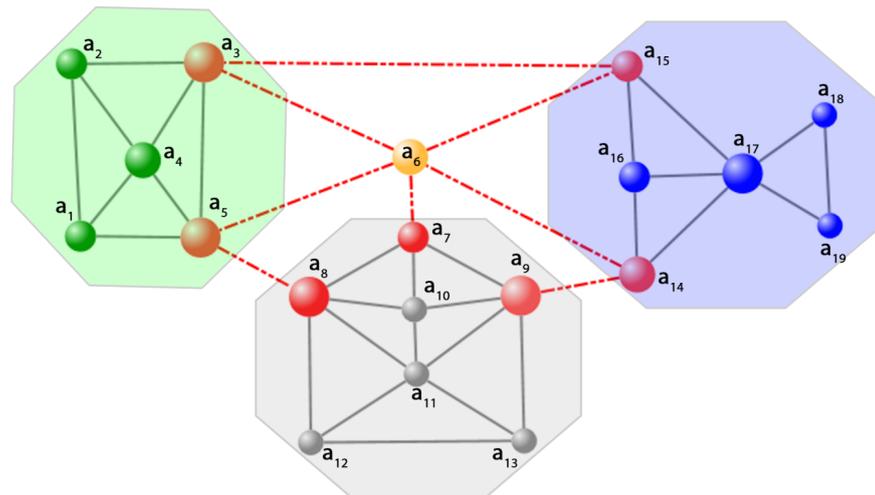


Figure 3. Community-aware network architecture supporting link prediction. The orange-colored actor a_6 is an actor with multiple community memberships. The red-colored actors (i.e., $a_3, a_5, a_7, a_8, a_9, a_{14}, a_{15}$) in each community represent the peripheral actors in each community and the red-colored dotted links denote the bilateral links bridging two communities. It is noteworthy that links connected to actor a_6 from every individual community are also considered bilateral links. a_1 – a_5 belong to the green community, a_7 – a_{13} belong to the grey community, and a_{14} – a_{19} belong to the blue community.

Bilateral Links: The number of links connecting two different communities. The actor on both ends of these links belongs to different communities. Similarly, in the presence of an actor with multiple community memberships, all links from a community connecting to that actor are also considered bilateral links. For example, in Figure 3, the red-colored dotted links (e.g., $(a_3, a_{15}), (a_5, a_8), (a_9, a_{14})$) are bilateral links as they are connecting two communities. Likewise, links, including $(a_5, a_6), (a_6, a_{15}), (a_6, a_{14}), (a_3, a_6)$, are considered bilateral links since these contain an actor with multiple memberships at one end of them. If C_i and C_j are two communities in a SIN G_t , then a bilateral link between these two communities is denoted by $e_t^{i,j}$ where $i \neq j$. A set of bilateral links between two communities C_i and C_j is denoted by $|E_t(i, j)|$.

Actor Connectivity: The actor connectivity between two actors a and b in a SIN G_t at timestamp t is the sum of the minimum number of actors and links that must be removed to disconnect all paths from actor a to b . If $E_t(a, b)$ denotes the set links and $V_t^c(a, b)$ denotes the set of actors of minimum cardinality such that when removed, they would sever off the connectivity between actor a and b , then the actor connectivity between these two actors is defined as:

$$\lambda_t(a, b) = |E_t(a, b)| + |V_t(a, b)| \tag{7}$$

A large value of $\lambda_t(a, b)$ denotes that there are many different alternative paths in a SIN G_t that are defined to maintain the connectivity between actor a and b .

To measure similarity/proximity between non-connected actor pairs using temporal community-aware network structural information using the aforementioned concepts, three different contexts were taken into consideration. First, *if both actors belong to the same community within a SIN*; if both actors belong to the same community within a SIN, then their similarity score for that SIN is strengthened by the rate of the clustering tendency of their common neighbors within the same community. However, the score is weakened by a dividing factor that represents the clustering tendency of the common neighbors residing in other communities different from the community where the corresponding actor pair belongs. The assumption here is that if more neighbors of the common neighbors, incident to a non-connected actor pair, perform triadic closure, then the possibility of that actor pair closing the triangle between them is amplified, and so is the probability of forming a link between them. Valverde-Rebaza and Lopes [50] exploited a similar concept where common neighbors within the same community strengthen twice more in the similarity/proximity score. Second, *if both actors in a pair reside in different communities within a SIN*; if both actors in a pair reside in different communities, then the similarity score between them is computed considering the number of peripheral actors, bilateral links, path length between actors, and their actor connectivity score. Finally, *if there is no path defined between a pair of actors residing in a different community within any SIN G_t* ; in that case, a score of zero is assigned to denote their proximity in that particular SIN.

If $C_i(t)$ denotes the i^{th} community, $\eta_t^i(a)$ denotes the neighborhood of actor a in a SIN G_t at timestamp t , and $P_t(a, b)$ denotes the geodesic path length between actor a and b , then by considering the aforementioned three different contexts, the final similarity metric using community-related and network structural information in every SIN is defined as follows:

$$sim_3(a, b) = \begin{cases} \sum_{t=1}^T \frac{\sum_{x \in \eta_t^i(a) \cap \eta_t^i(b)} CC_t(x)}{\sum_{j=1, j \neq i}^n \sum_{y \in \eta_t^j(a) \cup \eta_t^j(b)} CC_t(y)} & a, b \in C_i(t) \\ \sum_{t=1, i \neq j}^T |V_t(i, j)| + |E_t(i, j)| + \frac{\lambda_t(a, b)}{P_t(a, b)} & a \in C_i(t), b \in C_j(t), i \neq j \\ 0 & a \in C_i(t), b \in C_j(t), P_t(a, b) = \emptyset \end{cases} \quad (8)$$

Considering Equation (8), if two actors belong to the same community in a SIN G_t at timestamp t , then the similarity between them is increased by the increasing rate of the clustering tendency of the intra-community common neighbors incident to both actors but decreased by the clustering tendency of the inter-community neighbors in them who belong to other communities. The assumption here is if neighbors of the common neighbors, incident to the non-connected actor pair within the same community tend to close triangles, then the possibility of forming links between them is enhanced. Conversely, if they belong to different communities, then the similarity is calculated as the total of the number of peripheral actors, bilateral links, and actor connectivity score for the actor pair in conjunction with the inverse of the geodesic distance between both actors. The assumption here by considering the social network structure is that the peripheral actors are considered intercessors or negotiators between two distant actors, and bilateral links signify the common attributes or properties between communities. Furthermore, the higher the actor connectivity between non-connected actor pairs, the higher the probability of emerging links between them since there are more possible ways actors can reach each other. On the other hand, the connectivity score is undermined by the length of the geodesic distance between the corresponding actors. The rationale behind this part of the equation is that despite a higher connectivity score, if the corresponding actor resides in the furthest corner from each other, then the possibility of forming a link between them is reduced.

Since this study used two community detection methods, for the sake of simplicity, in the rest of the study, $sim_3^h(a, b)$ is used to denote the third dynamic similarity metric generated by considering the agglomerative hierarchical community detection method, and $sim_3^l(a, b)$ denotes the same metric that is generated by considering the Louvain community detection algorithm.

4. Experimental Settings

The dynamic features in this study, constructed above, were applied to five undirected dynamic network datasets in a supervised link prediction setup, and the performances were compared against a well-known topological similarity metric known as “ResourceAllocation”, which is widely used for link prediction purposes in cross-sectional networks. The prediction performance profiles of dynamic similarity metrics were also compared against a time-series-based dynamic link prediction approach [19], where a time series of a selected topological similarity metric (i.e., Jaccard coefficient) is constructed considering a series of SINs for a given network, and a time series forecasting method (i.e., ARIMA) is applied to predict the future values of that selected metric to train the classifier in supervised link prediction. Before delving into the supervised experimental setup, the dynamic network datasets, used in this study, are described below:

4.1. Network Datasets

The first four dynamic network datasets were selected from our previous study [54], where a novel method was proposed to determine the optimal sliding window size to sample a given dynamic network. The first undirected network dataset comes from a reality mining project at the Massachusetts Institute of Technology (MIT) in 2004, where the actors were tracked with the help of their personal smartphones to study interpersonal interaction. In this undirected network, an actor in the network represents a person, and a link indicates physical contact between two persons. The second dataset comes from internal email communications among employees of a mid-sized manufacturing company where actors represent employees and links mean individual emails between two employees. The next dataset contains undirected network data from a Facebook-like social network originating from an online community for students at the University of California, Irvine, where actors represent students within the community and a link represents that two students communicated via a message. The last undirected network dataset is a very small subset of the total “Facebook” friendship graph where an actor represents a Facebook user, and a link represents a friendship between two users. For the sake of brevity, this study names these four networks as G_{MIT} , G_{Email} , G_{UCI} , and G_{FF} to denote the network originated from MIT reality project, a small manufacturing company, University of California Irvine and real Facebook Friendships, respectively, in the rest of the study. In these network datasets, links are date stamped with individual dates, and the smallest temporal granularity of these networks is a day. In addition to these four network datasets, this study also considered one collaboration network of authors of scientific papers in “arXiv” in the high energy physics—theory (Hep-th) section. A link between two authors in these networks represents a joint publication where both authors have co-authored. Similar to the other four, this study uses G_{th} in the rest of the study to denote this network. Table 1 sets out the basic statistics of these network datasets.

Table 1. Basic Statistics of network datasets used in this study. The training duration represents the interval used to generate temporal short-interval networks, and the sampling interval denotes the sliding window sizes used to sample dynamic networks. SINs represent the number of short-interval networks or network snapshots generated using the corresponding window size.

| Dataset | Actors | Links | Training Duration | | Test Duration | | Window Size | # SINs |
|-------------|--------|-----------|-------------------|----------|---------------|----------|-------------|--------|
| | | | dd/mm/yy | dd/mm/yy | dd/mm/yy | dd/mm/yy | | |
| | | | Start | End | Start | End | | |
| G_{MIT} | 96 | 1,086,404 | 14/09/04 | 31/01/05 | 01/02/05 | 05/05/05 | Monthly | 5 |
| G_{Email} | 167 | 82,927 | 02/01/10 | 31/07/10 | 01/08/10 | 30/09/10 | Monthly | 8 |
| G_{UCI} | 1899 | 61,734 | 24/03/04 | 31/05/04 | 01/06/04 | 26/10/04 | Daily | 45 |
| G_{FF} | 11,715 | 42,698 | 01/01/07 | 31/03/07 | 01/04/07 | 30/04/07 | Daily | 90 |
| G_{th} | 6798 | 290,597 | 01/10/93 | 31/12/98 | 01/01/99 | 10/12/99 | Yearly | 6 |

4.2. Dynamic Networks

A dynamic network evolves over time among a set of actors where the network activities (i.e., link formation or deletion) have a temporal pattern. Thus, a dynamic network consists of a temporal sequence or time series of smaller network snapshots. As mentioned earlier, these snapshots are known as short-interval networks (SIN). One important aspect of dynamic network design is the selection of sliding window size (i.e., the amount of time lapse between the aggregations of links) used to sample large cross-sectional networks with timestamped links to generate SINS. From the dynamic network perspective, the order and frequency of actors and the temporal duration of active links can directly affect the associated network properties and dynamics. The selection of aggregation granularity to bin micro-scale network activities can also have a large impact on the expected outcome of the dynamic network study since it may lead to under- or oversampling of the network activities. In this study's perspective, since the dynamic similarity metrics are constructed by considering the evolutionary aspects, incident to individual actors, measured in each SIN of a time series of SINS, therefore, choosing the optimal window size can effectively regulate the generated metric values. Furthermore, the sampling resolution used to accumulate microscale network data can induce the mesoscale network properties of SIN(s). Considering the aforementioned aspects, and since the first four datasets are selected from the same study that deals with the time scale detection problem in dynamic networks, this study attempted to adhere to the same window size determined for each of these four datasets in [54]. The contenders are daily window sizes for G_{UCI} and G_{FF} and monthly window sizes for G_{MIT} and G_{Email} . On the other hand, since the temporal granularity of the co-authorship network G_{th} is a year, this study sampled dynamic networks considering yearly duration as the window size. In Table 1, this study also provides the number of SINS generated for the corresponding network datasets when the whole network was sampled by considering the selected time scale.

4.3. Supervised Link Prediction

The primary objective of the link prediction mechanism is to analyze the network structure and actors' attributes in the training phase $[t_1, t']$ to predict the possibility of future links in the test phase $[t', t_2]$. From a dynamic network perspective, the network in the training phase $G_T[t_1, t']$ is sampled using an aggregation granularity (i.e., sliding window size) to generate evolutionary network snapshots (i.e., SIN). As mentioned earlier, to split the network G_T and generate time series of SINS, this study used the optimal window size as described in Table 1. Supervised methods for link prediction problems need to predict emerging links by successfully discriminating positive and negatively labeled links within a classification dataset. Hence, supervised link prediction is considered a binary classification task by learning positive and negative instances with the help of interesting features describing each instance. In a supervised link prediction setup, this study built classification datasets consisting of positive and negative instances where each instance is a non-connected actor pair from the network in the test phase $G_{T+1}[t_1, t']$. Instances had positive labels if they appeared during the test phase and negative labels, providing that they were absent in both the training and test phases. This study considered a workload ratio of positive vs. negative instances as 1:5 for G_{MIT} , G_{Email} , G_{UCI} , and G_{FF} . Thus, the number of negatively labeled links is five times higher than the positively labeled ones in each classification dataset. However, in the case of the co-authorship network G_{th} , the workload ratio is 1:2. For the sake of simplicity in the link prediction problem, loops (i.e., links where source and destination are the same actors) were ignored, and unique links in G_{T+1} were considered (i.e., links not present in G_T) as positive instances. Choosing the appropriate feature set to describe instances in the classification dataset and to train classifiers is one of the most important tasks in supervised link prediction. In each classification dataset of this study, both positively and negatively labeled actor-pair instances were described using features computed using Equations (5), (6), and (8), described in Section 2. This study constructed a single classification dataset consisting

of instances, and dynamic features describing those instances, depending on the selected optimal window size for each network.

4.4. The Classifiers

In regards to classifiers, this study used simple logistic regression, random forest, and bagging algorithms. Logistic regression is a statistical method that uses the logistic function (sigmoid function) to build classification models for data that are linearly separable. It describes the relationship between one dependent variable with one or more independent variables where the dependent variable is either dichotomous or categorical and the independent variables can be nominal, ordinal, or interval type. The sigmoid function maps the predicted values to probabilities ranging from 0 to 1. This study considered the well-known machine learning library WEKA [55] for logistic regression with the default parameters. The WEKA workbench supports classes for building and using multinomial logistic regression models with ridge estimators [56] to improve the parameter estimates and to diminish the error made by further predictions. For the rest of the two classifier algorithms, this study again relied on the same workbench. For random forest, the two most important parameters that the workbench used are the maximum depth of the tree and the number of features to be used. The former was set to 0, which denotes the unlimited number of trees and the latter used $\lfloor \log_2(n) + 1 \rfloor$ formula to automatically calculate the number of features (i.e., predictors) to be considered, where n denotes the number of predictors. The bagging algorithm is an ensemble-based method that works as an ensemble meta-estimator fitting base classifier(s) on random subsets of the original dataset. It finally aggregates the predictions made by the base classifiers via voting or averaging to generate the final prediction. This classifier can reduce the variance of a black-box estimator (e.g., a decision tree) through the introduction of randomization and then make an ensemble out of it. This study considered the decision tree as the only base classifier with unlimited tree depth. The rest of the parameters for all three classifiers in the workbench were left to the default values. This study considered a k -fold cross-validation with $k = 10$ to estimate the skill of the machine learning models built via three classifiers on unseen data. This means that the training dataset was split into $k = 10$ smaller sets followed by a classifier training via $k - 1$ of the folds as training data. The resulting model performance is then validated on the remaining fold of the data.

4.5. Performance Evaluation

As mentioned earlier, this study utilized dynamic similarity metrics as dynamic features, as constructed in Section 2, to describe both positively and negatively labeled instances (i.e., non-connected actor pairs or links) in the classification datasets. Dynamic feature values were normalized such that the distribution has zero mean and one standard deviation. For the validation purpose, this study considered a 10-fold cross-validation and the mean scores to determine the accuracy of the results, AUCROC (“Area Under the Curve” of the “Receiver Operating Characteristic” curve), and AUCPR (“Area Under the Curve” of the “Precision and recall” curve). While the AUCROC measure is the de facto standard for measuring supervised learning-based classification, AUCPR is reported for a more differentiated view in regard to the learning task in the imbalanced dataset. Despite its criticism [57], AUCROC is a popular metric (after accuracy) used in binary classification. Accuracy only classifies the class label as right or wrong; however, AUCROC quantifies the uncertainty associated with classifiers by introducing a probability value. As an important traditional measure, the AUCROC score is interpreted as the probability that a randomly chosen missing link (i.e., link to be predicted) in the test phase that belongs to G_{T+1} is given a higher probability score than a randomly chosen non-existent link that is absent both in the training G_T and test network G_{T+1} . The formula to calculate AUCROC is defined as $AUCROC = \frac{n' + 0.5n''}{n}$, where n denotes the number of independent comparisons, n' denotes the times where when a missing link in the test network has been given a higher score, and n'' denotes the times when a non-existent link has been given a higher score. The AUCROC

curve demonstrates how the number of correctly classified positive examples varies with the number of incorrectly classified negative examples and shows an overly optimistic view of an algorithm's performance, whereas, the area under the precision–recall (P-R) curve (i.e., AUCPR) often serves as summary statistics while comparing the performances of several different algorithms. The minimum value of AUCPR can be determined as $AUCPR_{min} = 1 + \frac{(1-\phi)\ln(1-\phi)}{\phi}$ with skew $\phi = \frac{\text{positivesamples}}{n}$, where n = total number of samples in the classification dataset [21]. According to this equation, considering the ratio of positive and negative samples as 1:5 (i.e., the ratio of positive and negative samples is 1:5 in this study) in the classification datasets of G_{MIT} , G_{Email} , G_{UCI} , and G_{FF} with the value of the skew $\phi = 0.167$, the minimum value of AUCPR in these datasets should be 0.09. However, for the co-authorship network (i.e., G_{th}), since the skew $\phi = 0.33$ (i.e., ratio 1:2), the minimum value of AUCPR should be 0.189.

For comparison's sake, this study compared the performances of dynamic features with a well-known metric, "ResourceAllocation" [58], which is widely used for link prediction purposes in static networks and has demonstrated improved performance. The current study also implemented the link prediction strategy in dynamic networks proposed by Soares and Prudêncio [19], where the authors built time series of traditional topological metrics (e.g., Jaccard coefficient) for non-connected actor pairs for each SIN in the training phase and used a time series forecasting method (e.g., ARIMA) to predict the final score of the topological metrics and used those forecasted values to train the classifier. Different variations of this method are also extensively followed by other authors to support link prediction in dynamic networks [20,59]. For the sake of brevity, the rest of the study used $sim_{RA}(a, b)$ and $sim_{Soares}(a, b)$ to denote the values computed for the positively and negatively labeled actor pairs considering the "ResourceAllocation" metric and dynamic link prediction strategy proposed by Soares and Prudêncio. It is noteworthy that to compute sim_{Soares} , the current study considered the well-known "Jaccard Coefficient" measure as the topological similarity metric and used the ARIMA forecasting method to predict the future values of the selected metric incident to actor pairs. This study used the *Relative Performance Index* (RPI) to investigate how good the underlying link prediction approach is compared with others across various research datasets. In doing so, the lowest-performing approach for a given dataset is considered the baseline. They proposed this RPI variant to compare various K-nearest neighbors algorithms for disease risk prediction. The following equation describes this measure for a given link prediction approach:

$$RPI = \sum_{i=1}^d \frac{a_i - a_i^*}{d}$$

where a_i^* is the minimum accuracy/AUCROC/AUCPR value among all link prediction approaches for dataset i , a_i is the accuracy/AUCROC/AUCPR value for the link prediction approach under consideration for dataset i , and d is the number of datasets considered in this study. This study then adds the RPI scores for all machine learning methods (RF, bagging, and LR). This allows for the comparison of different link prediction approaches across various datasets using different performance measures. A higher RPI value indicates the prediction superiority (higher prediction performance) of the link prediction approach under consideration and vice versa. In (Table 1), the summary of six dynamic similarity metrics/dynamic features constructed in this study to measure the similarity/proximity between non-connected actor pairs is presented.

5. Results

Table 2 sets out the performance scores of three different classifiers in classifying positively and negatively labeled links using the dynamic features $sim_d(a, b)$, a topological similarity metric $sim_{RA}(a, b)$ known as "ResourceAllocation", and a time series forecasting-based metric $sim_{Soares}(a, b)$ for link prediction in dynamic networks. Note that $sim_d(a, b)$ denotes all dynamic features (i.e., $sim_1(a, b)$, $sim_2(a, b)$, $sim_3^h(a, b)$, and $sim_3^l(a, b)$). The metric $sim_{RA}(a, b)$ was computed by considering an aggregated network consisting of all SINs in the training phase of each network dataset. On the other hand, to compute $sim_{Soares}(a, b)$, as mentioned earlier, for each pair of actors in the classification dataset, the Jaccard coefficient was calculated for each SIN to build a time series of topological similarity metrics,

followed by the ARIMA forecasting method to predict future values of this co-efficient. These forecasted values were fed into the classifiers for training purposes. The classifiers' performances are demonstrated by considering three different performance metrics (i.e., accuracy, AUCROC, AUCPR), as described before. In regards to the accuracy score, this study observed that both linear and ensemble-based classifiers performed reasonably well using the dynamic similarity metrics/dynamic features constructed in this study compared to the other two. In each row of Table 2, for each dataset (G_{Email} , G_{FF} , G_{MIT} , G_{UCI} , and G_{th}), the highest score for three different evaluation metrics (i.e., accuracy, AUCROC, and AUCPR), the top-performing metric (i.e., $sim_d(a, b)$, $sim_{RA}(a, b)$, and $sim_{Soares}(a, b)$) are presented in bold-faced numbers. For example, in G_{Email} , for the bagging classifier, $sim_{Soares}(a, b)$ was the highest performer in accuracy score (i.e., 77.26), $sim_d(a, b)$ was the highest performer in AUCROC (i.e., 0.617), and $sim_{RA}(a, b)$ was the best performer in terms of AUCPR (i.e., 0.26). Alternatively, according to each classifier, irrespective of the datasets, the highest score for each evaluation metric category is colored red, and the second highest is colored green. For example, considering the same bagging classifier, the highest (i.e., 93.24) and the second-highest accuracy scores (i.e., 90.82) were recorded in the co-authorship dataset G_{th} . However, the highest and the second-highest AUCROC scores for the same classifier (bagging) were recorded in the co-authorship dataset G_{th} and the Facebook dataset G_{FF} , respectively (i.e., 0.876, and 0.655). Conversely, the highest and the second-highest AUCPR scores for the same classifier (bagging) were recorded in the Facebook dataset alone G_{FF} .

Table 2. Classification performances by three classifiers in classification datasets of five network datasets. sim_d represents the classification performance demonstrated by the dynamic features (i.e., $sim_1(a, b)$, $sim_2(a, b)$, $sim_3^h(a, b)$, and $sim_3^l(a, b)$), sim_{RA} denotes the performance of the "ResourceAllocation" topological metric in a cross-sectional network consisting of the aggregation of all SINs during the training phase, and finally, sim_{Soares} denotes the scores developed by following a time-series-forecasting-based dynamic link prediction method. The highest and the second-highest scores in each evaluation metric category in all datasets are colored red and green, respectively.

| Dataset | Random Forest | | | | | | | | |
|-------------|---------------------|--------------|----------------|--------------|--------------|----------------|-------------|-------------|----------------|
| | Accuracy | | | AUCROC | | | AUCPR | | |
| | sim_d | sim_{RA} | sim_{Soares} | sim_d | sim_{RA} | sim_{Soares} | sim_d | sim_{RA} | sim_{Soares} |
| G_{Email} | 79.19 | 77.92 | 67.99 | 0.706 | 0.678 | 0.552 | 0.33 | 0.33 | 0.23 |
| G_{FF} | 88.13 | 81.51 | 75.56 | 0.550 | 0.655 | 0.511 | 0.62 | 0.61 | 0.58 |
| G_{MIT} | 72.60 | 66.85 | 70.13 | 0.571 | 0.645 | 0.541 | 0.33 | 0.39 | 0.29 |
| G_{UCI} | 84.64 | 83.59 | 84.63 | 0.734 | 0.569 | 0.501 | 0.26 | 0.20 | 0.18 |
| G_{th} | 91.46 | 90.72 | 90.67 | 0.885 | 0.617 | 0.603 | 0.51 | 0.27 | 0.29 |
| | Bagging | | | | | | | | |
| G_{Email} | 75.32 | 72.86 | 77.26 | 0.617 | 0.608 | 0.576 | 0.24 | 0.26 | 0.22 |
| G_{FF} | 77.15 | 81.23 | 75.61 | 0.541 | 0.655 | 0.509 | 0.61 | 0.66 | 0.56 |
| G_{MIT} | 69.52 | 56.00 | 61.94 | 0.590 | 0.583 | 0.487 | 0.40 | 0.37 | 0.25 |
| G_{UCI} | 82.90 | 82.59 | 84.47 | 0.579 | 0.484 | 0.498 | 0.24 | 0.16 | 0.18 |
| G_{th} | 93.24 | 90.82 | 90.63 | 0.876 | 0.587 | 0.557 | 0.60 | 0.30 | 0.28 |
| | Logistic Regression | | | | | | | | |
| G_{Email} | 78.23 | 78.55 | 78.12 | 0.663 | 0.721 | 0.577 | 0.31 | 0.40 | 0.20 |
| G_{FF} | 77.00 | 75.82 | 75.54 | 0.549 | 0.655 | 0.516 | 0.62 | 0.67 | 0.58 |
| G_{MIT} | 70.58 | 72.21 | 71.12 | 0.529 | 0.621 | 0.527 | 0.40 | 0.42 | 0.34 |
| G_{UCI} | 84.52 | 84.59 | 84.64 | 0.620 | 0.503 | 0.562 | 0.23 | 0.19 | 0.20 |
| G_{th} | 91.11 | 90.48 | 90.52 | 0.852 | 0.618 | 0.601 | 0.39 | 0.25 | 0.26 |

By considering the dynamic features, and the accuracy scores, the highest accuracy-based performance was achieved in the co-authorship dataset G_{th} using the bagging

classifier, and the lowest performance was recorded in G_{MIT} considering the linear classifier logistic regression. Considering the AUCROC scores, the highest performance was also achieved in the G_{th} dataset using the random forest classifier, whereas the lowest was logged again in G_{MIT} by considering the linear classifier logistic regression. Considering the lowest AUCPR score as defined earlier, most of the classifiers demonstrated optimal performances exceeding the minimum values calculated earlier (i.e., 0.189 for G_{th} and 0.167 for the rest of the datasets); however, the highest value was recorded in the G_{FF} dataset using both the ensemble classifier random forest and the linear classifier logistic regression. Conversely, the lowest was recorded in G_{UCI} considering both the ensemble classifiers. Regarding different classifiers, using the random forest classifier, the dynamic similarity metrics outperformed both $sim_{Soares}(a, b)$ and $sim_{RA}(a, b)$ in all datasets in regards to the accuracy scores. However, considering the AUCPR and AUCROC, it is four out of five (i.e., $sim_{RA}(a, b)$ exceeded $sim_d(a, b)$ in G_{MIT}). On the other hand, considering the bagging algorithm, $sim_d(a, b)$ was outperformed by the other metrics in three out of five datasets in regards to the accuracy score, two out of five in regards to the AUCPR, and only in G_{FF} considering the AUCROC score. Overall, $sim_d(a, b)$ showed superior performance compared with the other two approaches considered in this study. However, in a few cases, $sim_{RA}(a, b)$ and $sim_{Soares}(a, b)$ outperformed $sim_d(a, b)$. Therefore, a further comparison of these three approaches is needed.

Table 3 serves this purpose by taking RPI to compare these link prediction approaches across three performance measures. According to this table, the proposed dynamic attribute-based link prediction approach outperformed the other two considered in this study. In summary, our proposed approach for dynamic link prediction in complex networks showed superior performance by a large margin compared with two other existing well-known approaches across three performance measures (i.e., accuracy, AUCROC, and AUCPR).

Table 3. Comparison of the relation performance index score for different dynamic link prediction approaches considering the three performance measures and five datasets used in this research. The higher the score (red colored), the better the performance.

| | $sim_d(a, b)$ | $sim_{RA}(a, b)$ | $sim_{Soares}(a, b)$ |
|----------|---------------|------------------|----------------------|
| Accuracy | 54.00 | 24.15 | 17.24 |
| AUCROC | 1.82 | 1.15 | 0.073 |
| AUCPR | 1.51 | 0.90 | 0.060 |

Considering the accuracy scores, the worst performance, although the extent was insignificant, was noticed in the case of logistic regression where $sim_d(a, b)$ was overtaken by the other metrics in three datasets out of five. With regard to the AUCROC and AUCPR, a similar performance was observed by the linear classifier. From the aforementioned performance observations, it is evident that the dynamic features constructed in this study undoubtedly outperformed the existing metrics, used in link prediction, in most cases by considering the ensemble classifiers. In the case of the linear classification, the rivalry between three different features (i.e., $sim_d(a, b)$, $sim_{Soares}(a, b)$, $sim_{RA}(a, b)$) were competitive. Nevertheless, the performance demonstrated by the logistic regression algorithm is better than a random classifier and justifies the fact that the dynamic features of this study can be effective in predicting emerging links in dynamic networks, even by considering a simple linear classifier. In the case of the ensemble-based classifiers, bagging, where a decision tree was used as a base classifier, is susceptible to overfitting and computationally expensive, as it considers all the available features to split a node in decision trees. Conversely, the random forest, a special case of bagging, randomly considers only a subset of the best features of those available. Therefore, it performed superior to bagging in some cases. In the co-authorship networks, considering the dynamic features, the bagging algorithm observed better performance by considering all three performance metrics.

In G_{th} , considering all three performance metrics and all three classifiers, improved performances were demonstrated, as in Figure 4. This study presents the ROC and P-R curves of the other four datasets to portray a comparable picture of the three classifiers' performances. It is noteworthy that in P-R plots, curves tend to lie in the bottom left corner of the graph. The closer a curve is to the diagonal line, the higher the classifier's performance in classification. Conversely, in ROC plots, curves tend to lie in the top-left region of the plots. The higher the curve is from the diagonal line, the better the predictor's performance. Apart from G_{FF} , considering both the P-R and ROC curves in the other datasets, logistic regression was found to compete with the random forest algorithm for superiority, whereas bagging was found comparably to be the least-performing one in most cases. In G_{FF} , considering the ROC plots, it is observed that all classifiers tend to perform similarly and closer to a random classifier, which can achieve a maximum AUCROC score of 0.50. However, the best performance was observed in the P-R curve (i.e., closer to the diagonal line), which established the fact found in Table 2 in regards to the AUCPR score. Further study can reveal the underlying reason behind the classification performance differences, demonstrated by different classifiers; however, from the aforementioned classification performances, demonstrated in the table and figure, it can be concluded that the dynamic similarity metrics constructed in this study can be successfully employed to predict future links in dynamic networks.

After the performance measurement of the dynamic features, at this stage, this study attempted to determine the relative importance of four different dynamic features (i.e., $sim_1(a, b)$, $sim_2(a, b)$, $sim_3^h(a, b)$, and $sim_3^l(a, b)$) to assess their relative competency in dynamic link prediction tasks in all five datasets. For this purpose, this study took advantage of two different algorithms (i.e., information gain and chi-square evaluation) provided in the WEKA machine learning software (<https://sourceforge.net/projects/weka/files/>). Table 4 provides a comparable picture of these features regarding their rank of importance obtained by these algorithms. The ranks of the features are assigned in decreasing order, with one denoting the highest ranking. Information gain and chi-square evaluator algorithms evaluate the worthiness of a feature by calculating the information gain and chi-square statistics for the class variables. On the other hand, the last two columns denote the rank of a feature regarding the support vector machine (SVM) and random forest classifier. Finally, all the ranks for the four algorithms were aggregated to generate the final rank. From this table, it is observable that $sim_2(a, b)$, which represents the dynamic similarity metric, constructed by considering the correlation between time series of actor-level community dynamicity values, became the most prominent feature in two datasets (i.e., G_{MIT} , G_{Email}), and the second-best in the co-authorship datasets G_{th} . On the other hand, $sim_1(a, b)$, the dynamic similarity metric constructed by considering the temporal similarity of community dynamicity values of actor pairs using the DTW method, became the leading feature in G_{th} , G_{UCI} , and G_{FF} . Among the dynamic features, generated by considering the temporal community-aware network structures and with the help of two different community detection algorithms, the feature constructed by considering the Louvain algorithm was found to be more effective than the other.

To answer the second research issue of this study, as described in the introduction section, distributions of dynamic feature values are represented in Figure 5. For each network dataset, the first and second-best-performing features were selected from Table 4; for example, $sim_2(a, b)$ and $sim_3^l(a, b)$ for the G_{Email} dataset. This study observed from this figure that it is not always obvious whether either dissimilar or distant actors in regards to their dynamic feature values (i.e., lower values) or similar and closer actors (i.e., higher feature values) participate in emerging links considering any particular feature. For example, in G_{UCI} , considering $sim_1(a, b)$ (i.e., the temporal similarity of community dynamicity values computed by the DTW method), the non-participating actor pairs (i.e., negatively labeled links in the classification datasets) had higher values than the actors' genuinely formed links in the test phase. This signifies that actors with dissimilar temporal evolution had a higher possibility of forming emerging links. Conversely, in the same

dataset, considering $sim_3^l(a, b)$ (i.e., temporal community-aware network-structure-based features constructed by considering the Louvain community detection method), the picture is the opposite. In this case, the positively labeled links (i.e., actors participating in emerging links) had higher values. On the other hand, in the case of G_{th} , from the distribution of $sim_1(a, b)$, this study observed that actors having similar temporal evolution had a higher possibility of forming links.

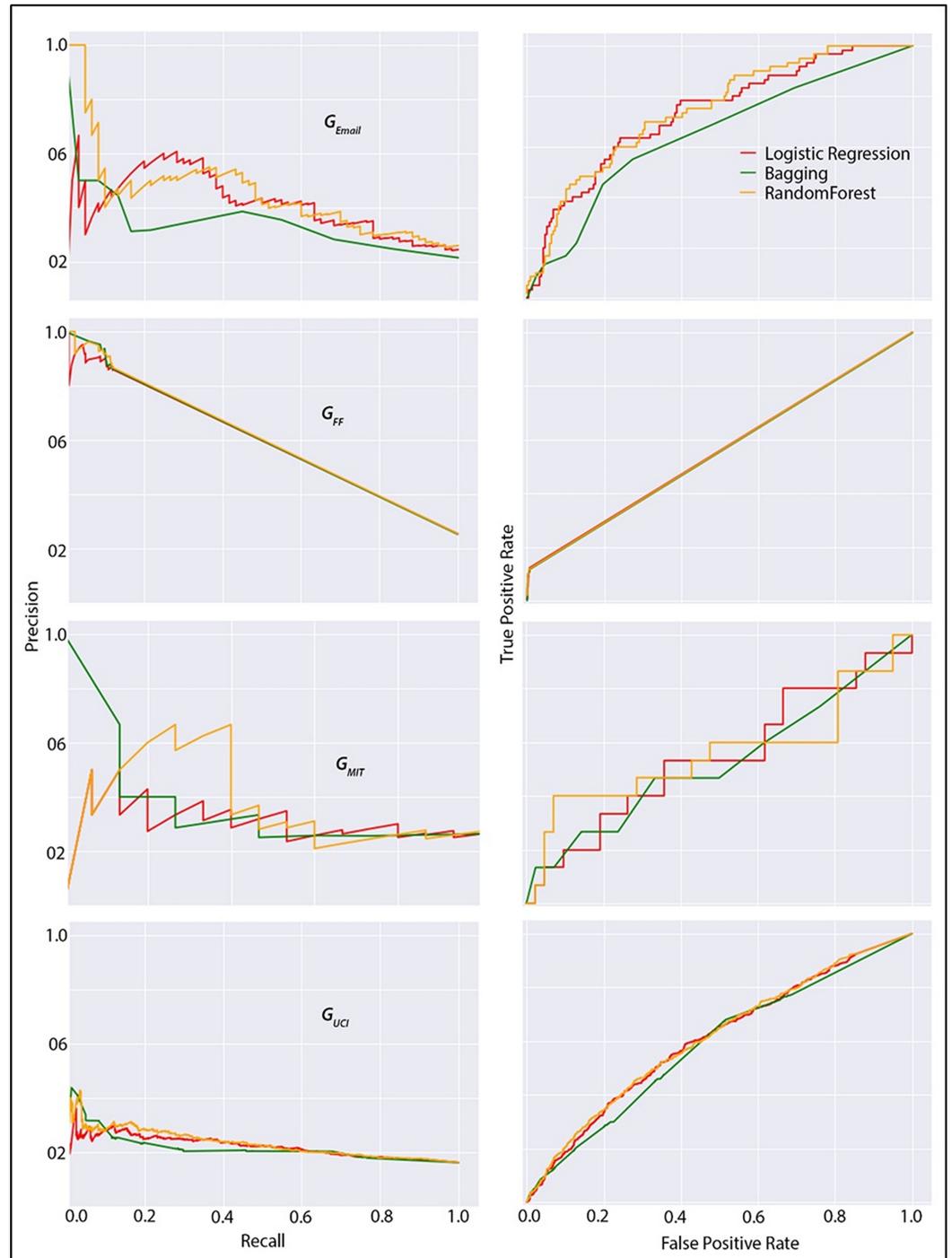


Figure 4. A Visual representation of P-R (i.e., precision–recall) (left column) and ROC curves (right column) for four network datasets considering three classifiers.

Table 4. The rank of different dynamic features constructed in this study using different algorithms. Ranks are in decreasing order with number one denoting the highest ranking. The “Total” column represents the aggregation of all ranking scores to generate the final ranking. The red-colored row denotes the highest ranked (most important) feature in each dataset.

| Feature Name | Information Gain | Chi-Square Evaluation | SVM Evaluator | Random Forest Evaluator | Total |
|--|------------------|-----------------------|---------------|-------------------------|-------|
| <i>G_{Email}</i> | | | | | |
| <i>sim</i> ₁ (<i>a, b</i>) | 2 | 2 | 3 | 4 | 11 |
| <i>sim</i> ₂ (<i>a, b</i>) | 1 | 1 | 1 | 1 | 4 |
| <i>sim</i> ₃ ^h (<i>a, b</i>) | 4 | 4 | 4 | 2 | 14 |
| <i>sim</i> ₃ ^l (<i>a, b</i>) | 3 | 3 | 2 | 3 | 11 |
| <i>G_{FF}</i> | | | | | |
| <i>sim</i> ₁ (<i>a, b</i>) | 1 | 1 | 1 | 1 | 4 |
| <i>sim</i> ₂ (<i>a, b</i>) | 4 | 4 | 4 | 4 | 16 |
| <i>sim</i> ₃ ^h (<i>a, b</i>) | 2 | 2 | 3 | 3 | 10 |
| <i>sim</i> ₃ ^l (<i>a, b</i>) | 3 | 3 | 2 | 2 | 10 |
| <i>G_{MIT}</i> | | | | | |
| <i>sim</i> ₁ (<i>a, b</i>) | 2 | 2 | 4 | 4 | 12 |
| <i>sim</i> ₂ (<i>a, b</i>) | 1 | 1 | 2 | 3 | 7 |
| <i>sim</i> ₃ ^h (<i>a, b</i>) | 4 | 4 | 3 | 2 | 13 |
| <i>sim</i> ₃ ^l (<i>a, b</i>) | 3 | 3 | 1 | 1 | 8 |
| <i>G_{UCI}</i> | | | | | |
| <i>sim</i> ₁ (<i>a, b</i>) | 1 | 1 | 3 | 1 | 6 |
| <i>sim</i> ₂ (<i>a, b</i>) | 4 | 4 | 4 | 4 | 16 |
| <i>sim</i> ₃ ^h (<i>a, b</i>) | 3 | 3 | 2 | 3 | 11 |
| <i>sim</i> ₃ ^l (<i>a, b</i>) | 2 | 2 | 1 | 2 | 7 |
| <i>G_{th}</i> | | | | | |
| <i>sim</i> ₁ (<i>a, b</i>) | 1 | 1 | 4 | 1 | 7 |
| <i>sim</i> ₂ (<i>a, b</i>) | 2 | 2 | 3 | 2 | 9 |
| <i>sim</i> ₃ ^h (<i>a, b</i>) | 4 | 3 | 2 | 4 | 13 |
| <i>sim</i> ₃ ^l (<i>a, b</i>) | 3 | 4 | 1 | 3 | 11 |

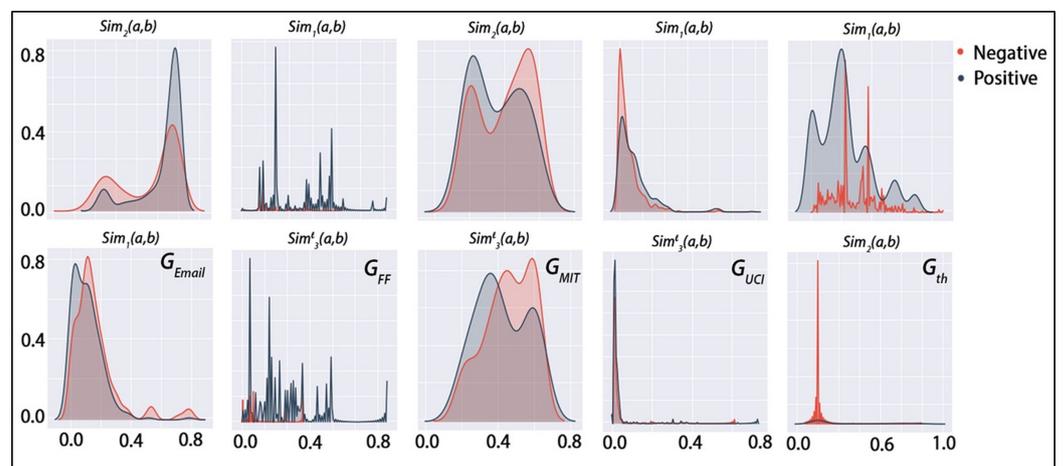


Figure 5. Distribution of dynamic feature values in each network dataset for both positively and negatively labeled links in the classification dataset. Each column represents a network dataset where the top plot visualizes the distribution of the most important feature and the bottom plot visualizes the second-best performing dynamic feature in the corresponding dataset. The red label denotes the classification instances with negative labels and the blue represents the positively labeled ones.

6. Discussion and Conclusions

The link prediction problem in social networks has gained considerable interest from various domains, and consequently, divergent prediction strategies, metrics and methodologies have emerged in aiding this problem of network science. The ineptness of these strategies in accommodating the associated dynamicity and evolutionary information in

dynamic networks has led them to be incompetent in dynamic link prediction, despite their compliance with the performance expectations. Therefore, the “time” component needs to be integrated as a parameter to the dynamic link prediction problem to better approximate the temporal network evolution. Consequently, researchers complied with this requirement and applied both time series analyses and evolutionary aspects (e.g., temporal link decay, duration of link activeness) in the link prediction task in dynamic networks. Although the topological information and actor attributes are predominantly the principal sources of information used in the prediction problem, however, due to the modular structure of social networks, community information can also be effectively exploited for this purpose. The dominant rationales behind this are: (i) community structure manifests the information about actors with similar behavior that can be conducive to predicting their future interaction [41] and (ii) the high and low condensation of links among actors can be an effective prophecy towards emerging links [60]. Furthermore, incorporating community-related structural information can drastically improve the accuracy of link prediction [61]. Therefore, scholars tend to acknowledge that emerging links can be predicted by mining the evolutionary information extracted from the network snapshots over time, in association with dynamic network topology, evolutionary mesoscale network structure, and temporal actor-level neighborhood changes. Motivated by the aforementioned phenomenon, this study attempted to propose a novel solution to the problem of dynamic link prediction by defining dynamic similarity metrics using the dynamic community-aware information extracted both at the local (i.e., actor-level) and global (i.e., network) level. In addressing the problem of dynamic link prediction, this study first defined an actor-level measure to render the temporal community-aware evolution, known as community dynamicity. It also considered the rate of changes concerning the actor’s cliquishness, community participation, and associated neighborhood changes. These attributes were later used to develop evolutionary features. Since these features were constructed by considering the temporal evolution experienced by actors, it is noteworthy that one of the important aspects of dynamic network analysis is to define the optimal time scale to sample the network to generate time series of network snapshots (i.e., SIN). For this purpose, we selected a method from the literature. Once the optimal temporal window size was defined, three different dynamic features were constructed: first, by measuring the temporal similarity of both temporal sequences of community dynamicity values, incident to a pair of non-connected actors, with the help of the DTW method; and second, by computing the correlation between both sequences. Finally, with the help of two different existing community detection algorithms, by integrating evolutionary community-aware topologies in conjunction with both inter- and intra-community network structures. In a supervised link prediction setup, these features were applied to five different undirected social networks of different sizes and domains. Two ensemble-based classifiers and one linear classifier were used to measure the performance of the dynamic features. Needless to mention here, since time series analysis is well-adopted in dynamic link prediction tasks, this study used a well-defined time series forecasting method, known as exponential smoothing, to predict the future values of actor-level community dynamicity. However, unlike other dynamic link prediction strategies, instead of using these predicted values to train the classifiers, it computed the similarity between two time series with the help of DTW and Pearson correlation measures and used these similarity measures to train the classifiers. By considering the performance metrics, this study observed that these features could be indulged for dynamic link prediction purposes and can effectively support modeling the network growth. The performance of dynamic features was also compared with a traditional topological metric (i.e., ResourceAllocation), which is widely used for link prediction purposes in cross-sectional networks. We also considered a time-series-based dynamic link prediction strategy as a baseline method. In both cases, this study observed that dynamic features, constructed by leveraging the evolutionary community-aware aspect of actors, performed not only as outstanding as the existing ones but also, surprisingly in most cases, outweighed them to demonstrate superior prediction performance. This study can further

be extended in different ways. For example, instead of the temporal clustering tendency of actors, other network structures or topology (e.g., assortativity) can be exploited, including other time series forecasting methods (e.g., ARIMA) instead of exponential smoothing, and other similarity measures (e.g., Euclidean, Manhattan) can be employed to measure the similarity between temporal information. In the case of the third metric, other community detection algorithms (e.g., edge betweenness) can be used to enhance the prediction performance. Finally, like many other applications of link prediction problems, this study can be valuable to help define new dynamic similarity metrics for dynamic link predictions in networks that inherently evolve over time, including terrorist networks, online social networks (e.g., Twitter), scholarly and knowledge networks (e.g., keyword network), and collaborative filtering to model the consumers' buying behavior.

Funding: This research received no external funding.

Data Availability Statement: The network data used in this study was collected from online sources (<https://networkrepository.com/>, accessed on 14 November 2023).

Acknowledgments: The author of this article is grateful to Shahadat Uddin from the University of Sydney for his expert feedback on the result/evaluation section.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Opsahl, T.; Hogan, B. Growth mechanisms in continuously-observed networks: Communication in a Facebook-like Community. *arXiv* **2010**, arXiv:1010.2141.
- Liben-Nowell, D.; Kleinberg, J. The link prediction problem for social networks. In Proceedings of the Twelfth International Conference on Information and Knowledge Management, New Orleans, LA, USA, 3–8 November 2003; pp. 556–559.
- Lü, L.; Zhou, T. Link prediction in complex networks: A survey. *Phys. A Stat. Mech. Its Appl.* **2011**, *390*, 1150–1170. [\[CrossRef\]](#)
- Chen, Y.; Chen, K.J.; Li, Y. A link prediction method that can learn from network dynamics. In Proceedings of the 2014 IEEE International Conference on Data Mining Workshop, Shenzhen, China, 14 December 2014; pp. 549–553.
- Li, T.; Wang, J.; Tu, M.; Zhang, Y.; Yan, Y. Enhancing link prediction using gradient boosting features. In *Intelligent Computing Theories and Application; ICIC 2016. Lecture Notes in Computer Science*; Huang, D.S., Jo, K.H., Eds.; Springer: Cham, Switzerland, 2016; Volume 9772_7 [\[CrossRef\]](#)
- Tylenda, T.; Angelova, R.; Bedathur, S. Towards time-aware link prediction in evolving social networks. In Proceedings of the 3rd Workshop on Social Network Mining and Analysis, Paris, France, 28 June 2009; pp. 1–10.
- Li, X.; Du, N.; Li, H.; Li, K.; Gao, J.; Zhang, A. A deep learning approach to link prediction in dynamic networks. In Proceedings of the 2014 SIAM International Conference on Data Mining, Philadelphia, PA, USA, 24–26 April 2014; SIAM: Philadelphia, PA, USA, 2014; pp. 289–297.
- Choudhury, N.; Uddin, S. Time-aware link prediction to explore network effects on temporal knowledge evolution. *Scientometrics* **2016**, *108*, 745–776. [\[CrossRef\]](#)
- Choudhury, N.; Uddin, S. Mining actor-level structural and neighborhood evolution for link prediction in dynamic networks. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Sydney, NSW, Australia, 31 July–3 August 2017; pp. 721–728.
- Zhang, T.; Zhang, K.; Lv, L.; Li, X. Temporal link prediction using node centrality and time series. *Int. J. Fut. Comput. Commun.* **2020**, *9*, 62–65. [\[CrossRef\]](#)
- Chi, K.; Yin, G.; Dong, Y.; Dong, H. Link prediction in dynamic networks based on the attraction force between nodes. *Knowl.-Based Syst.* **2019**, *181*, 104792. [\[CrossRef\]](#)
- Bliss, C.A.; Frank, M.R.; Danforth, C.M.; Dodds, P.S. An evolutionary algorithm approach to link prediction in dynamic social networks. *J. Comput. Sci.* **2014**, *5*, 750–764. [\[CrossRef\]](#)
- Ahmed, N.M.; Chen, L. An efficient algorithm for link prediction in temporal uncertain social networks. *Inf. Sci.* **2016**, *331*, 120–136. [\[CrossRef\]](#)
- Jaya Lakshmi, T.; Durga Bhavani, S. Temporal probabilistic measure for link prediction in collaborative networks. *Appl. Intell.* **2017**, *47*, 83–95. [\[CrossRef\]](#)
- Safdari, H.; Contisciani, M.; De Bacco, C. Reciprocity, community detection, and link prediction in dynamic networks. *J. Phys. Complex.* **2022**, *3*, 015010. [\[CrossRef\]](#)
- Ma, X.; Sun, P.; Qin, G. Nonnegative matrix factorization algorithms for link prediction in temporal networks using graph communicability. *Pattern Recognit.* **2017**, *71*, 361–374. [\[CrossRef\]](#)
- Chen, J.; Lin, X.; Jia, C.; Li, Y.; Wu, Y.; Zheng, H.; Liu, Y. Generative dynamic link prediction. *Chaos Interdiscip. J. Nonlinear Sci.* **2019**, *29*, 123111. [\[CrossRef\]](#) [\[PubMed\]](#)

18. Divakaran, A.; Mohan, A. Temporal link prediction: A survey. *New Gener. Comput.* **2020**, *38*, 213–258. [[CrossRef](#)]
19. da Silva Soares, P.R.; Prudêncio, R.B.C. Time series based link prediction. In Proceedings of the 2012 International Joint Conference on Neural Networks (IJCNN), Brisbane, QLD, Australia, 10–15 June 2012; pp. 1–7.
20. Güneş, İ.; Gündüz-Öğüdücü, Ş.; Çataltepe, Z. Link prediction using time series of neighborhood-based node similarity scores. *Data Min. Knowl. Discov.* **2016**, *30*, 147–180. [[CrossRef](#)]
21. Choudhury, N.; Uddin, S. Evolution similarity for dynamic link prediction in longitudinal networks. In Proceedings of the Complex Networks VIII: Proceedings of the 8th Conference on Complex Networks CompleNet, Dubrovnik, Croatia, 21–24 March 2017; Springer: Cham, Switzerland, 2017; pp. 109–118.
22. Wu, X.; Wu, J.; Li, Y.; Zhang, Q. Link prediction of time-evolving network based on node ranking. *Knowl.-Based Syst.* **2020**, *195*, 105740. [[CrossRef](#)]
23. Sajadmanesh, S.; Zhang, J.; Rabiee, H.R. NPGLM: A Non-Parametric Method for Temporal Link Prediction. *arXiv* **2017**, arXiv:1706.06783.
24. Wang, T.; He, X.S.; Zhou, M.Y.; Fu, Z.Q. Link prediction in evolving networks based on popularity of nodes. *Sci. Rep.* **2017**, *7*, 7147. [[CrossRef](#)]
25. Lei, K.; Qin, M.; Bai, B.; Zhang, G. Adaptive multiple non-negative matrix factorization for temporal link prediction in dynamic networks. In Proceedings of the 2018 Workshop on Network Meets AI & ML, Budapest, Hungary, 24 August 2018; pp. 28–34.
26. Ma, X.; Sun, P.; Wang, Y. Graph regularized nonnegative matrix factorization for temporal link prediction in dynamic networks. *Phys. A Stat. Mech. Its Appl.* **2018**, *496*, 121–136. [[CrossRef](#)]
27. Fang, C.; Kohram, M.; Ralescu, A.L. Spectral regression with low-rank approximation for dynamic graph link prediction. *IEEE Intell. Syst.* **2011**, *26*, 48. [[CrossRef](#)]
28. Wu, T.; Chang, C.S.; Liao, W. Tracking network evolution and their applications in structural network analysis. *IEEE Trans. Netw. Sci. Eng.* **2018**, *6*, 562–575. [[CrossRef](#)]
29. Liu, F.; Liu, B.; Sun, C.; Liu, M.; Wang, X. Deep learning approaches for link prediction in social network services. In Proceedings of the Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Republic of Korea, 3–7 November 2013; Springer: Berlin/Heidelberg, Germany, 2013; pp. 425–432, Part II 20.
30. Chen, J.; Wang, X.; Xu, X. GC-LSTM: Graph convolution embedded LSTM for dynamic network link prediction. *Appl. Intell.* **2022**, *52*, 7513–7528. [[CrossRef](#)]
31. Yang, C.; Liu, Z.; Zhao, D.; Sun, M.; Chang, E.Y. Network representation learning with rich text information. In Proceedings of the IJCAI, Buenos Aires, Argentina, 25–31 July 2015; Volume 2015, pp. 2111–2117.
32. Cai, H.; Zheng, V.W.; Chang, K.C.C. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 1616–1637. [[CrossRef](#)]
33. Taheri, A.; Berger-Wolf, T. Predictive temporal embedding of dynamic graphs. In Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Vancouver, BC, Canada, 27–30 August 2019; pp. 57–64.
34. Goyal, P.; Chhetri, S.R.; Canedo, A. dyngraph2vec: Capturing network dynamics using dynamic graph representation learning. *Knowl.-Based Syst.* **2020**, *187*, 104816. [[CrossRef](#)]
35. Ibrahim, N.M.A.; Chen, L. Link prediction in dynamic social networks by integrating different types of information. *Appl. Intell.* **2015**, *42*, 738–750. [[CrossRef](#)]
36. Chiu, C.; Zhan, J. Deep learning for link prediction in dynamic networks using weak estimators. *IEEE Access* **2018**, *6*, 35937–35945. [[CrossRef](#)]
37. Zhu, Y.; Liu, S.; Li, Y.; Li, H. TLP-CCC: Temporal link prediction based on collective community and centrality feature fusion. *Entropy* **2022**, *24*, 296. [[CrossRef](#)]
38. Kumar, M.; Mishra, S.; Singh, S.S.; Biswas, B. Community Enhanced Link Prediction in Dynamic Networks. *ACM Trans. Web* **2023**.r [[CrossRef](#)]
39. Choudhury, N.; Uddin, S. Evolutionary community mining for link prediction in dynamic networks. In Proceedings of the Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications), Lyon, France, 29 November–1 December 2017; Springer: Cham, Switzerland, 2018; pp. 127–138.
40. Papadopoulos, S.; Kompatsiaris, Y.; Vakali, A.; Spyridonos, P. Community detection in social media: Performance and application considerations. *Data Min. Knowl. Discov.* **2012**, *24*, 515–554. [[CrossRef](#)]
41. Fortunato, S. Community detection in graphs. *Phys. Rep.* **2010**, *486*, 75–174. [[CrossRef](#)]
42. Uddin, S.; Khan, A.; Piraveenan, M. A set of measures to quantify the dynamicity of longitudinal social networks. *Complexity* **2016**, *21*, 309–320. [[CrossRef](#)]
43. Choudhury, N.; Uddin, S. Evolutionary Features for Dynamic Link Prediction in Social Networks. *Appl. Sci.* **2023**, *13*, 2913. [[CrossRef](#)]
44. Sorensen, T.A. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons. *Biol. Skar.* **1948**, *5*, 1–34.
45. De Gooijer, J.G.; Hyndman, R.J. 25 years of time series forecasting. *Int. J. Forecast.* **2006**, *22*, 443–473. [[CrossRef](#)]

46. Hyndman, R.J.; Athanasopoulos, G. *Forecasting: Principles and Practice*; OTexts: Melbourne, Australia, 2018.
47. Salvador, S.; Chan, P. Toward accurate dynamic time warping in linear time and space. *Intell. Data Anal.* **2007**, *11*, 561–580. [[CrossRef](#)]
48. Chi, K.T.; Liu, J.; Lau, F.C. A network perspective of the stock market. *J. Empir. Financ.* **2010**, *17*, 659–667.
49. Rosvall, M.; Bergstrom, C.T. Maps of random walks on complex networks reveal community structure. *Proc. Natl. Acad. Sci. USA* **2008**, *105*, 1118–1123. [[CrossRef](#)]
50. Valverde-Rebaza, J.C.; de Andrade Lopes, A. Link prediction in complex networks based on cluster information. In Proceedings of the Advances in Artificial Intelligence-SBIA 2012: 21th Brazilian Symposium on Artificial Intelligence, Curitiba, Brazil, 20–25 October 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 92–101.
51. Raghavan, U.N.; Albert, R.; Kumara, S. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E* **2007**, *76*, 036106. [[CrossRef](#)]
52. Blondel, V.D.; Guillaume, J.L.; Lambiotte, R.; Lefebvre, E. Fast unfolding of communities in large networks. *J. Stat. Mech. Theory Exp.* **2008**, *2008*, P10008. [[CrossRef](#)]
53. Newman, M.E. Fast algorithm for detecting community structure in networks. *Phys. Rev. E* **2004**, *69*, 066133. [[CrossRef](#)]
54. Uddin, S.; Choudhury, N.; Farhad, S.M.; Rahman, M.T. The optimal window size for analysing longitudinal networks. *Sci. Rep.* **2017**, *7*, 13389. [[CrossRef](#)]
55. Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I.H. The WEKA data mining software: An update. *ACM SIGKDD Explor. Newsl.* **2009**, *11*, 10–18. [[CrossRef](#)]
56. Cessie, S.I.; Houwelingen, J.V. Ridge estimators in logistic regression. *J. R. Stat. Soc. Ser. C Appl. Stat.* **1992**, *41*, 191–201. [[CrossRef](#)]
57. Yang, Y.; Lichtenwalter, R.N.; Chawla, N.V. Evaluating link prediction methods. *Knowl. Inf. Syst.* **2015**, *45*, 751–782. [[CrossRef](#)]
58. Zhou, T.; Lü, L.; Zhang, Y.C. Predicting missing links via local information. *Eur. Phys. J. B* **2009**, *71*, 623–630. [[CrossRef](#)]
59. Rossetti, G.; Guidotti, R.; Miliou, I.; Pedreschi, D.; Giannotti, F. A supervised approach for intra-/inter-community interaction prediction in dynamic social networks. *Soc. Netw. Anal. Min.* **2016**, *6*, 86. [[CrossRef](#)]
60. Valverde-Rebaza, J.; de Andrade Lopes, A. Exploiting behaviors of communities of twitter users for link prediction. *Soc. Netw. Anal. Min.* **2013**, *3*, 1063–1074. [[CrossRef](#)]
61. Feng, X.; Zhao, J.; Xu, K. Link prediction in complex networks: A clustering perspective. *Eur. Phys. J. B* **2012**, *85*, 3. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.