Interpretable Approach to Predicting Dynamics in Temporal Weighted Networks

Master's Thesis in Computer Science (AI Track)

Department of Intelligent Systems
Multimedia Computing Group
Delft University of Technology

Author:
Andrejs Karklins
Student Number: 5110238

Keywords:
Weighted Link Prediction, Temporal Networks, Machine Learning,
Interpretable Models, Contact Networks

March 18, 2025

Abstract

Accurately predicting future interactions in temporal networks is essential for diverse applications, including disease modeling [7] and transparent policy-making under regulatory requirements [4]. While much research has focused on topological or temporal link prediction, relatively few studies address *temporal weighted* link prediction, where both the *presence* of future links and their *intensity* are critical. This paper develops an *interpretable* approach for forecasting weighted links over time, using contact networks that are aggregated into temporal weighted snapshots. By adapting the Self-Driven (SD) and Self-Cross-Driven (SCD) models introduced in Zou et al. (2023) [21], we design prediction methods that incorporate memory decay while preserving explanatory clarity—avoiding the "black-box" nature common to deep learning.

We benchmark four methods - Baseline, SD, SCD, and an extended SCD* - across a range of physical (face-to-face) and virtual (online communication) contact networks, each exhibiting unique structural and dynamic properties when aggregated into temporal weighted networks. Our results show that the SCD model achieves a 32.86% reduction in Mean Squared Error (MSE) over the Baseline, while SD yields a 19.31% improvement. Moving from SD to SCD confers an additional 15.87% decrease in MSE, underscoring the benefits of incorporating both temporal and structural information. Although SCD* introduces further complexity, it did not show any consistent improvements in predictive performance across datasets.

Additional evaluations using the Area Under the Precision-Recall Curve (AUPRC) highlight dataset-specific variability in capturing active links. Correlation analysis reveals that MSE scales with average link weight distributions, whereas AUPRC correlates strongly with the proportion of active links per network snapshot. These findings emphasize that incorporating decay and structural context, in an interpretable manner, significantly enhances predictive accuracy, although parameter tuning remains crucial for different network topologies and interaction patterns.

Acknowledgements

I would like to express my gratitude to my supervisor, **Dr. Huijuan Wang**, for her unwavering support, patience, and insightful guidance throughout the course of this research. Her expertise in Multimedia Computing and her thoughtful mentorship have not only shaped the direction of this thesis but also fostered my personal and academic growth.

I am likewise thankful to the entire Multimedia Computing Group at TU Delft for creating a collaborative and intellectually stimulating environment that encouraged exploration and learning. The rigorous coursework in the Computer Science (AI Track) provided a solid theoretical and practical foundation, directly informing the research I present here. Finally, I would like to extend my appreciation to TU Delft for granting access to its high-performance computing cluster, which was indispensable for conducting the data-intensive experiments central to my findings.

Contents

1 Introduction

Networks are a fundamental tool for representing complex relationships in both natural and engineered systems, showing how different parts interact across various fields. A basic network has nodes (the elements in the system) and links (the connections between these elements). Contact networks, which are the focus of this research, specifically represent human interactions over time, where nodes correspond to individuals, and links denote their interactions. These networks could be physical, capturing face-to-face interactions in settings like workplaces and schools, or virtual, representing online communications.

Understanding the evolution of contact networks over time is pivotal for accurately forecasting future interactions. During the COVID-19 pandemic, for instance, timely and precise estimates of human contact patterns were instrumental in anticipating outbreaks. Prior studies in temporal network analysis [7] highlight the necessity of incorporating both structural (i.e., which nodes connect) and dynamic (i.e., how often they connect) aspects when modeling phenomena like contagion spread.

Aggregating contact data at a coarser time scale yields a temporal weighted network whose link weights reflect the cumulative number of contacts within each chosen interval. This approach is especially advantageous when fine-grained timestamps are unavailable or impractical, and it can simplify longer-term predictions—which are inherently more challenging [21]. By varying the aggregation window (Δt), researchers can systematically explore how different temporal resolutions affect network properties, leading to more extensive information.

This thesis focuses on designing and evaluating interpretable temporal weighted link prediction model. There were plentiful of researches that focused on network topology or temporal link prediction problems, however, temporal weighted link prediction problem remains relatively unexplored. This research seeks to address this gap by exploring interpretable models that coul predict the weight of the link. Importantly, this work prioritizes transparency over "black-box" methods such as deep learning, which, though powerful, often lack explainability. This is essential for decision-making in areas such as government, finance, and healthcare. Recent regulations, such as the European Union's General Data Protection Regulation (GDPR), emphasize the need for explainable AI, discouraging the use of automated, opaque decision-making systems [4].

To achieve these goals, this study builds upon Self-Driven (SD) and Self-Cross-Driven (SCD) models [21]. These models were designed to predict the temporal network one time step ahead based on the network observed in the past. Both models allow for explainability on how interactions between different types of node pairs in the past contribute to future contacts of a target node pair. In this research SD and SCD models will be adapted for temporal weighted network prediction problem. The methodology also includes a thorough evaluation of each model by comparing its predicted contact patterns to actual observed interactions. This comparison helps determine how well the model can estimate future link weights. Additionally, both models performance will be evaluated against a baseline prediction approach that assumes future interactions will be identical to those observed in the previous time step.

This research uses a variety of publicly available datasets, including physical and virtual contact networks. These datasets, ranging from face-to-face interactions at conferences to online communication patterns among university students, provide a diverse foundation for testing model performance in short-term temporal link weight predictions. The use of multiple datasets also ensures that the models developed can generalize across different types of interaction data, further validating their real-world applicability.

1.1 Problem Definition

The objective of this research is to develop interpretable models that accurately predict weighted link activities in temporal networks. A contact network can be measured/aggregated at discrete times Δt and can be represented as a sequence of network snapshots, $G = \{G_1, G_2, ..., G_T\}_{\Delta t}$ where T is the number of snapshots. $G_t = (V; E_t)$ is the snapshot at time step, t with V and E_t being the set of nodes and contacts that happened in that time window, respectively. It is assumed that the topology of the time aggregated network G^* is known, which contains the same set of nodes V as all snapshots in the network and the set of links E^* between nodes that had at least one contact in the temporal network. Using historical link activities within a prior observation window $[p, p+L]$, where $p \in [1, T-L-1]$, this

research aims to predict the links' weights at time $p + L + 1$, capturing both the network's temporal dynamics and changes in interaction intensity while remaining limited to the observation window.

1.2 Research Questions

1. How can an interpretable network-based algorithm be developed to predict a temporal weighted network in the future?

2. How do network properties influence the prediction quality of network-based models?

2 Background Research

This section reviews the foundational concepts of temporal networks, highlighting existing methodologies in temporal link prediction, including both unweighted and weighted prediction techniques, and outlines current challenges, establishing the context necessary for understanding how this research seeks to advance temporal weighted link prediction by addressing critical gaps related to scalability, interpretability, and accuracy.

Over the past few decades, network theory has evolved significantly, moving beyond static structures to capture the dynamic properties of real-world systems. This shift reflects the understanding that many natural and engineered systems are rarely static; rather, they change and evolve over time, requiring models that can capture these changes.

Temporal networks capture the evolution of interactions over time, providing a more comprehensive view of relational dynamics than static networks, which only depict connections at a single point. In temporal networks, links between nodes may appear, disappear, or change in intensity at each time step, making them indispensable for a wide range of applications in social network analysis, epidemiology, and communication systems [7, 8]. Typically, these networks are represented as a collection of nodes and links, each labeled with the time or time interval at which the corresponding interaction occurs.

In the context of temporal networks, contact networks represent sequences of real-world human interactions documented over time. These networks are particularly valuable for studying the spread of diseases, as they reveal how face-to-face contacts evolve and thus provide insight into transmission pathways [10, 16]. Typically, contact networks are recorded as lists of recurrent time-stamped interactions between individuals. In this research, both physical and virtual contact networks are considered, reflecting a wide range of interaction patterns and intensities in diverse settings.

Weighted temporal networks are type of temporal networks that additionaly incorporate the strength or intensity of connections, such as the volume of bank transactions or the frequency of contact events within an aggregated timeframe [11, 3]. In these studies, weighted temporal networks are obtained by aggregating contact data into predefined time windows; the link weight then represents the number of interactions or accumulated duration of interactions between individuals over each aggregation window.

2.1 Existing Unweighted Temporal Link Prediction Algorithms

Temporal link prediction involves forecasting the activity of links in a temporal network based on history of link interactions. Several algorithms have been developed for this purpose, taking advantage of various methodologies from machine learning and network theory. One notable approach is the use of Graph Convolutional Networks (GCNs), which have been adapted for temporal networks to capture the evolving structure of the network [20, 12]. In addition, generative adversarial networks (GANs) have been used to model the distribution of network links over time, providing a probabilistic framework for link prediction [12].

Despite advances made with GCNs and GANs, these models often suffer from high computational complexity and limited interpretability of the underlying mechanisms. GCNs, while powerful in capturing structural changes in the network, tend to be computationally intensive due to the need to process large adjacency matrices and learn complex graph structures. Similarly, GANs, although useful in modeling probabilistic distributions, require substantial computational resources for training adversarial models and can be challenging to interpret.

In contrast, more interpretable approaches, such as those discussed by Zou et al. (2022) [22], focus on leveraging network memory properties to forecast future link activity. Zou et al. (2022) introduce

interpretable prediction models that combine statistical methods with network-based features. These models predict future interactions while maintaining low computational overhead and providing interpretation for the predicted outcomes. Specifically, their approach predicts each target link in the next network snapshot by integrating the past state of the link along with its neighboring links and assuming the natural memory decay. This method has demonstrated effectiveness in achieving accurate link predictions without sacrificing interpretability, providing a straightforward understanding of how past interactions influence future connections [22].

Building on these findings, Zou et al. (2023) [21] further develop the concept by introducing two specific models: the Self-Driven (SD) model and the Self- and Cross-Driven (SCD) model. The SD model focuses solely on the past activity states of a link, with a stronger influence from more recent states, capturing the intrinsic memory of temporal networks. The SCD model extends this by also considering the influence of neighboring links. This model differentiates between different types of neighboring links, such as those forming a triangle with the target link, and the distinct neighbors, then uses these distinctions to refine predictions. The SCD model has been shown to outperform the SD model in several real-world networks, particularly in networks with strong memory effects [21].

This research will be built upon the SD and SCD models described in Zou et al. (2023) [21] as a basis, adapting them to solve the problem of weighted temporal link prediction. This adaptation will involve integrating the weight of links into the existing framework, allowing for a more nuanced understanding of link dynamics that accounts for the intensity and frequency of interactions. As also an attempt to refine SCD model will be made to further improve the overall performance of this approach on temporal weighted prediction task.

2.2 Existing Weighted Temporal Link Prediction Algorithms

The field of weight prediction in temporal networks is relatively new and emerging. Traditional link prediction focuses on whether a connection will occur, while weight prediction aims to quantify the strength or frequency of these connections. This area of study is motivated by the need to understand the intensity and quality of interactions over time, which are critical in applications ranging from social network analysis to biological systems. Accurate weight predictions can reveal patterns in communication frequency, transaction volumes, or interaction strengths, providing deeper insights into the nature of relationships within the network.

Graph Neural Network (GNN)-based methods are widely used for dynamic link and weight prediction, effectively capturing structural and temporal information through techniques like Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs). GCNs have been particularly valuable in processing neighborhood information to learn robust feature representations, as seen in their application to anomaly detection and node classification in Ethereum networks [19]. GATs, on the other hand, employ attention mechanisms to weigh neighboring nodes selectively, further enhancing prediction accuracy in dynamic contexts [18]. Despite their effectiveness, GNN-based methods are computationally intensive and may be challenging to interpret.

Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, have also been applied to weight prediction in temporal networks. LSTMs are designed to capture long-term dependencies in sequential data, making them suitable for predicting link weights based on prior interactions. Greff et al. (2017) thoroughly explored the design space of LSTM architectures, showing their effectiveness in retaining essential information over time in dynamic networks [5]. This architecture, often combined with self-attention mechanisms, can enhance accuracy by emphasizing critical temporal dependencies, although these models are computationally demanding and may be prone to overfitting.

Finally, *tensor factorization* offers a scalable alternative for weight evolution prediction. By representing network data as a three-dimensional tensor (dimensions for sender, receiver, and time), tensor factorization techniques decompose this tensor into lower-dimensional matrices. This method is particularly promising in fields like financial transactions, where temporal and relational patterns are integral. For example, Charlier et al. applied tensor-based methods to model smart contract activities in Ethereum, highlighting the scalability and predictive accuracy of this approach [1]. Although tensor factorization methods are generally more interpretable, they may require extensive parameter tuning for optimal results.Moreover, a common drawback is that they model the temporal dynamics in the target CTN primarily from a numerical perspective, often lacking richer domain-driven modeling considerations that could capture deeper structural and causal mechanisms behind observed link

evolutions.

2.3 Current Challenges

Despite advancements in temporal weighted network analysis and link prediction, several challenges persist. A primary issue is the scalability of algorithms to process large and complex networks. Many existing methods face computational constraints, struggling to handle vast amounts of temporal data, which limits their applicability to real-world networks where interaction volumes are high. Furthermore, real-world networks are often characterized by noisy and incomplete data, presenting a need for more robust techniques that can accurately predict link dynamics despite such irregularities [6].

Another critical challenge is enhancing the interpretability of complex models. While models like Graph Convolutional Networks (GCNs) and Generative Adversarial Networks (GANs) deliver strong predictive capabilities, they often function as black boxes, obscuring the reasoning behind their predictions. Increasing the transparency of these models, or developing inherently interpretable alternatives, is essential for their adoption in practical applications [17]. Interpretability is particularly important in fields where the implications of predictions are significant, such as healthcare, finance, and governance, where decisions based on opaque models may lead to ethical and legal challenges [17, 14].

This paper aims to address these shortcomings by proposing a more interpretable, network-based algorithm that incorporates both relational and temporal insights, thus bridging the gap between expressive power and explanatory depth. By carefully integrating domain-driven modeling assumptions, we strive to mitigate the purely numerical limitations of earlier tensor-based methods while retaining the scalability and predictive strengths vital for real-world applications.

3 Data Analysis

In the following sections, we describe the datasets used, present details on how they were pre-processed, and examine key properties of the aggregated temporal networks. We then explore how these contact networks retain *memory* over time, providing a foundation for subsequent prediction tasks.

Real-world contact networks often have varying levels of temporal resolution due to technological constraints, privacy considerations, or simply the impracticality of tracking every interaction at fine-grained time scales. As a result, one of the key motivations for *aggregating* contact networks in this study is to emulate realistic data scenarios, where timestamps may be coarse or irregular. Additionally, making longer-range predictions based on high-frequency data can be overly complex and computationally expensive, a challenge highlighted by recent work on short-long term memory based predictions [21]. By choosing different temporal aggregation periods, we balance the need for accurate predictive modeling with tractable data processing. A further advantage of experimenting with different Δt values is the ability to capture a range of structural and temporal patterns within the same network, enabling a more exhaustive and robust investigation of network dynamics.

3.1 Original data

All datasets for this research were selected from publicly available. The approach will be tested on physical and virtual contact networks: CollegeMsg[1], Social Evolution[2], Hypertext 2009[3], SFHH Conference[4].

3.2 Data pre-processing

Data pre-processing is a crucial stage that ensures the integrity and quality of our datasets before engaging in more advanced analyses. In our approach, each dataset was first mapped to a time series plot with timestamps on the x-axis and the number of interactions on the y-axis, as illustrated in Table 1. This visual representation plays a key role in identifying any abnormalities or inconsistencies in the raw data.

[1] https://snap.stanford.edu/data/CollegeMsg.html
[2] http://realitycommons.media.mit.edu/socialevolution.html
[3] http://konect.cc/networks/sociopatterns-hypertext/
[4] http://www.sociopatterns.org/datasets/sfhh-conference-data-set/

Network	N	C	M
CollegeMsg	1,899	59,835	22876
Social Evolution	75	10636	3221
Hypertext 2009	113	20,818	4642
SFHH Conference	403	70261	12063

Table 1: The number of nodes $N = |V|$, number of contacts (C), node pairs that have contact(s) (M)

3.2.1 Physical contact networks

Both physical contact networks were published on SocioPatterns website [5] and are similar in the way data was collected.

Hypertext 2009.
This dataset was collected during ACM Hypertext 2009 conference[6]. Conference visitors were wearing physical sensors that were registering face-to-face proximity contacts over 2.5 days. The dataset was collected consistently with intervals of 20 seconds, thus no additional pre-processing was needed to perform data aggregation.

SFHH Conference.
This data was recorded at 2009 SFHH conference in Nice, France. The dataset contains only 2 days, featuring 405 participants and their interactions using physical Radio Frequency Identification (RFID) that registered close face-to-face contacts. The dataset was consistent and no additional pre-processing was required.

3.2.2 Virtual contact networks

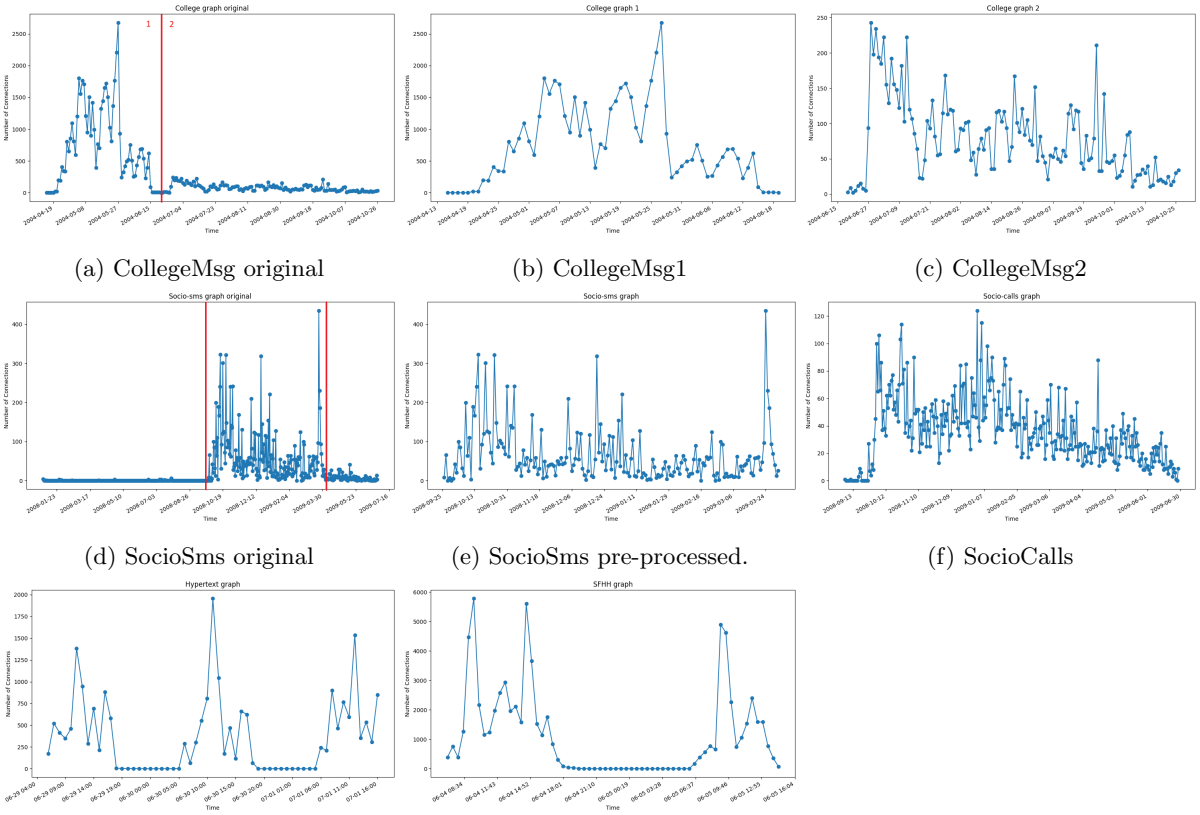
CollegeMsg.
This dataset contains private messages sent in an online social network at the University of California, Irvine. Upon closer investigation of the dataset, it was clear that 2 periods were visible in the dataset with an inactivity period in the middle. Since 2 of those activity periods were different in magnitude of interactions, it was decided to separate the CollegeMsg dataset into two named CollegeMsg1 and CollegeMsg2 which is visible in Figure 1a, with a split at 2004-06-20 00:00:00

Social Evolution.
This dataset was collected during an experiment at the university, deploying a smartphone application within a tight undergraduate student community. The application was recording calls and messages between students over 1 academic year. The dataset also contained monthly self-report surveys related to their health habits, diet, and exercise, weight changes, and political opinions during the presidential election campaign [13]. For this research we have focused only on SMS and Calls collected data, thus naming the datasets Socio-sms and Socio-calls respectively. The Socio-sms dataset had a visible difference in number of interactions made, thus it was decided to take only the period where 90% of interactions took place from 2008-09-01 00:00:00 to 2009-04-01 00:00:00, visible on Figure 1d.

3.3 Data Aggregation

Real-world contact networks frequently exhibit varying levels of temporal granularity, arising from factors such as device sampling rates, privacy regulations, or the sheer impracticality of capturing every micro-interaction. In this study, we therefore aggregate contacts into coarser time intervals to balance the need for meaningful temporal structure with computational tractability. By reducing the granularity, edge weights in each aggregated snapshot represent the frequency of interactions over a chosen interval (Δt). This approach not only helps to smooth out noisy or irregular data but also allows

[5] http://www.sociopatterns.org/datasets/
[6] http://www.ht2009.org/



(a) CollegeMsg original (b) CollegeMsg1 (c) CollegeMsg2

(d) SocioSms original (e) SocioSms pre-processed. (f) SocioCalls

(g) Hypertext 2009 (h) SFHH

Figure 1: Datasets contact activity over time

us to handle different predictive horizons without exploding the complexity, a challenge highlighted in recent work on short–long-term memory-based predictions [21].

A key motivation for experimenting with multiple aggregation periods is to mimic more realistic data scenarios. For instance, high-frequency data may exist in principle, but practical considerations—like privacy or data storage constraints—often necessitate coarser temporal snapshots. Moreover, using different Δt values can reveal a range of structural and temporal patterns within the same dataset, enabling a more robust exploration of how network dynamics unfold across multiple scales.

Finally, it was decided to employ different sets of aggregation times for physical and virtual networks, given their distinct durations, densities, and usage contexts. By tailoring Δt to each domain, we reduce the risk of either oversimplifying complex interactions or overburdening the model with excessively granular data, thereby ensuring consistent and comparable analyses across heterogeneous network types.

Specifically, physical contact networks, which involve direct face-to-face interactions, are aggregated at intervals of **10 minutes**, **30 minutes**, and **1 hour**. These finer temporal resolutions allow for capturing rapid fluctuations and short-term interactions typical of environments such as conferences and exhibitions, where proximity sensors (e.g., RFID badges) record frequent interactions.

Conversely, virtual contact networks, representing digital communications such as emails, phone calls, and social media interactions, are aggregated using longer intervals of **1 hour**, **1 day**, and **3 days**. These extended aggregation windows reflect the intermittent and prolonged nature of virtual interactions, where communication events are less frequent but sustained over time.

By applying different aggregation intervals, networks with distinct temporal and structural properties are produced, providing a robust framework for an extensive analysis of model performance. This approach allows us to assess how varying temporal granularity impacts the predictive capabilities of the models and ensures their applicability across diverse real-world scenarios.

The aggregation process is formalized by the *Aggregate Temporal Contact Network* algorithm (Algorithm 1), which is used to transform the raw temporal contact data into a temporal weighted network.

The algorithm takes the contact network G and the aggregation interval Δt as inputs and outputs a aggregated temporal weighted network $G^* = (V, E^*)$ and matrix W where columns represent set of links E^* from resulting aggregated network and rows represent aggregated time steps (T). Each separate link can be defined as $x_i(t)$, where $t \in [1, T]$ is a timestamp and $i \in E^*$ the index of a connection.

Algorithm 1 Aggregate Temporal Contact Network

Require: Temproal contact network G, that consists of contacts with timestamps (u, v, t), aggregation interval Δt

Initialize *time_dict* as an empty dictionary to store contact counts for each time window
Identify the minimum timestamp *min_timestamp* from all edge timestamps
for all $(u, v, t) \in G$ **do**
    Compute the new aggregated time index for the timestamp as $t = \frac{t^* - min\_t^*}{\Delta t}$
    if t is not in *time_dict* **then**
        Initialize *time_dict*$[t]$ as a zero vector of length equal to the number of edges
    end if
    Increment the contact count for edge (u, v) in *time_dict*$[t]$
end for
Initialize an empty matrix W with dimensions $(T \times \mathrm{M})$, where T is number of timestamps in new aggregated network.
for each time window t in sorted order **do**
    Assign the corresponding vector from *time_dict*$[t]$ to the row in W for that time window
end for
return Aggregated weighted network matrix W, Aggregated Network G^*

3.3.1 Omission of Empty Time Windows

During the aggregation process, time intervals (columns) with zero contacts are removed, so that the final matrix consists solely of non-zero rows, centering the analysis on periods with actual contact events. This approach reduces the matrix size, thereby minimizing extraneous computations while retaining the key temporal patterns of the network.

Discarding these empty windows yields a more concise representation of temporal activity, enabling efficient, targeted analyses of physical and virtual contact networks. Moreover, excluding intervals without contacts prevents over-expansion of the dataset, ensuring that the aggregation process emphasizes meaningful interaction dynamics rather than periods of inactivity. As a result, the final dataset remains informative.

3.4 Aggregated dataset properties

Table 2: Aggregated Dataset Properties for Physical and Virtual Networks

Dataset	Δt	T	$\mu_{\lvert E_t^+\rvert,\%}$	$\sigma_{\lvert E_t^+\rvert,\%}$	μ_{E_t}	$\mu_{H_t,\%}$	μ_{cc}
Physical Datasets							
Hypertext graph	10m	223	45.36	25.91	4.21	94.9	0.6
	30m	81	62.0	29.41	11.58	94.08	0.56
	1h	43	70.91	31.72	21.82	94.47	0.55
SFHH graph	10m	128	41.93	26.14	7.7	94.92	0.36
	30m	44	54.02	29.55	22.41	94.24	0.33
	1h	22	66.92	29.79	44.82	93.87	0.32
Virtual Datasets							
College graph 1	1h	1249	11.56	9.14	0.37	96.3	0.08
	1d	63	53.69	24.75	7.29	94.25	0.11
	3d	22	66.38	28.74	20.89	94.56	0.11
College graph 2	1h	2072	5.29	4.28	0.11	98.41	0.04
	1d	129	32.58	13.48	1.81	93.55	0.05
	3d	43	52.1	13.75	5.44	91.34	0.06
Socio-calls graph	1h	2825	8.02	5.19	0.18	98.67	0.33
	1d	279	39.79	15.53	1.84	93.44	0.33
	3d	96	61.96	20.34	5.34	91.66	0.36
Socio-sms graph	1h	1893	9.27	6.38	0.52	97.69	0.1
	1d	186	25.1	12.66	5.26	88.15	0.16
	3d	63	43.06	12.58	15.52	85.38	0.2

G - Network name.
Δt - Aggregation interval (in seconds).
T - Total number of snapshots in the aggregated temporal network.
$\mu_{|E_t^+|,\%}$ - Average percentage of active links per snapshot, relative to the M.
$\sigma_{|E_t^+|,\%}$ - Standard deviation of the percentage of active links per snapshot, relative to the M.
μ_{E_t} - Average link weight per snapshot.
$\mu_{H_t,\%}$ - Average *weighted interaction entropy* (3.4.1), indicating the diversity of interaction weights across snapshots.
μ_{cc} - Average *weighted clustering coefficient* (3.4.2), reflecting the strength of triadic connectivity in the network per snapshot.

3.4.1 Normalized Weighted Interaction Entropy

The *normalized weighted interaction entropy* measures the diversity of interaction weights across the edges in a temporal network, normalized to account for the total number of edges. For each snapshot G_t, the entropy is computed using the formula:

$$H_t^{'} = -\sum_{i=1}^{M} p_i(t) \log(p_i(t)), \tag{1}$$

where $p_i(t) = \frac{x_i(t)}{\sum_{j=1}^{M} x_j(t)}$ is the proportion of the weight of the given link to the total weight of the snapshot t. To ensure interpretability and comparability, the entropy is normalized by dividing by the maximum possible entropy, given by $\log(|E_t^+|)$, where $|E_t^+|$ is the total number of active links in the snapshot:

$$H_t = \frac{H_t^{'}}{|E_t^+|}. \tag{2}$$

The normalized entropy H_t lies within the range $[0, 1]$, where:

- $H_{\text{normalized}} = 1$: Indicates a perfectly uniform distribution of weights across all edges (maximum diversity).
- $H_{\text{normalized}} = 0$: Indicates that all interaction weight is concentrated on a single edge (no diversity).

3.4.2 Weighted Clustering Coefficient

The *weighted clustering coefficient* [15] quantifies the extent to which nodes in a weighted network tend to form closed triangles, considering the strength of connections. Unlike the standard clustering coefficient, which only accounts for the presence or absence of edges, the weighted version incorporates edge weights, providing a richer understanding of local connectivity patterns.

The weighted clustering coefficient for a node i is defined as:

$$C_i^w = \frac{1}{s_i(k_i - 1)} \sum_{j,k} (x_{ij} + x_{ik}) \cdot x_{jk} \tag{3}$$

where:

- $s_i = \sum_j x_{ij}$ is the strength of node i, defined as the sum of the weights of edges connected to i.
- k_i is the degree of node i, representing the number of neighbors.
- x_{ij} is the weight of the edge between nodes i and j.
- The summation $\sum_{j,k}$ is over all pairs of neighbors j and k of node i, and x_{jk} is the weight of the edge between neighbors j and k, if it exists.

The coefficient C_i^w measures the relative intensity of triangular connections involving node i, normalized by the strength and degree of the node.

The overall *weighted clustering coefficient* [15] of the network is the average of C_i^w over all nodes:

$$C^w = \frac{1}{N} \sum_i C_i^w, \tag{4}$$

where N is the total number of nodes in the network.

The weighted clustering coefficient reflects the local structure and the strength of the interaction in a weighted network. A high value of C^w indicates that nodes are embedded in tightly connected communities with strong interactions, while a low value suggests that local structures are sparse or weakly interconnected.

In the context of temporal or aggregated networks:

- A high C^w highlights regions where interactions are not only frequent but also strong, emphasizing the role of key hubs or communities.
- A low C^w may indicate weak or transient connections, typical of networks with loosely connected components or sporadic interactions.

3.5 Autocorrelation

Autocorrelation measures the correlation between a network's structure at one time point and its structure at future time points. It helps assess the persistence of interactions over time and evaluate the chosen temporal aggregation interval, Δt. The granularity of the temporal changes in the network is directly influenced by Δt, and autocorrelation reveals whether the newly created network has memory.

For two vectors X and Y (e.g., weights for an edge at different time points), the Pearson correlation coefficient is given by:

$$\rho(X,Y) = \frac{\sum_{k=1}^{n}(x_k - \bar{X})(y_k - \bar{Y})}{\sqrt{\sum_{k=1}^{n}(x_k - \bar{X})^2 \sum_{k=1}^{n}(y_k - \bar{Y})^2}} \tag{5}$$

where $\bar{X}$ and $\bar{Y}$ are the means of X and Y, respectively. This concept can be extended to all links by averaging the Pearson coefficients across all links to obtain the autocorrelation function $AC(\Delta)$ at lag Δ:

$$AC(\Delta) = \frac{1}{M}\sum_{i=1}^{M}\rho(G_t, G_{t+\Delta}) \tag{6}$$

where M is total number of links from E^*.

As illustrated in Figures 2 and 3, both physical and virtual datasets exhibit distinct autocorrelation decay patterns depending on the chosen aggregation interval Δt. In physical datasets such as Hypertext and SFHH, when aggregated at $\Delta t = 10m$, autocorrelation declines sharply after the first lag, indicating highly transient interactions. Notably, the $10m$ interval consistently yields the smallest memory decay.

In contrast, virtual datasets like CollegeMsg and Socio-calls demonstrate a steeper drop in autocorrelation when aggregated at $\Delta t = 1h$ compared to longer intervals. For all virtual datasets, choosing $\Delta t = 1d$ or $3d$ preserves memory more effectively, while the fine-grained $1h$ interval often leads to faster autocorrelation decay, likely because these datasets are relatively shallow.

Overall, these findings confirm that both physical and virtual weighted temporal networks retain meaningful memory in their interactions—an essential characteristic for the feasibility of memory-based prediction models.

3.6 Weighted Jaccard Similarity

The weighted Jaccard similarity index [9], or Ruzicka similarity, compares sets of edge weights across different time intervals, taking into account both the presence and intensity of interactions. This measure is critical for understanding how the chosen temporal aggregation interval Δt reflects the dynamics of network interactions, particularly with respect to memory retention.

The weighted Jaccard similarity between two weighted network snapshots G^t and $G^{t+\Delta}$ is defined as:

$$J_w(G_t, G_{t+\Delta}) = \frac{\sum_{i=1}^{M}\min(x_i(t), x_i(t+\Delta))}{\sum_{i=1}^{M}\max(x_i(t), x_i(t+\Delta))} \tag{7}$$

where $x_i(t)$ and $x_i(t+\Delta)$ represent the weights of edge i at times t and $t+\Delta$, respectively.

A high-weighted Jaccard similarity indicates that interaction patterns, including their intensities, remain stable over the considered time interval, implying strong memory retention. Conversely, low similarity suggests a significant shift in interaction patterns, leading to weaker memory retention.

Figures 2 and 3 show that the decay in weighted Jaccard similarity closely mirrors the trends observed in autocorrelation.

In physical datasets like Hypertext and SFHH, with $\Delta t = 10m$, the sharp drop in similarity highlights indicates frequent shifts in both the presence and strength of connections, reinforcing the short-term volatility observed in autocorrelation.

For virtual datasets, such as CollegeMsg and Socio-sms, we can observe a more gradual decay in weighted Jaccard similarity with larger aggregation intervals (e.g., $\Delta t = 1d$ or $3d$), which aligns with the slower memory decay observed in autocorrelation, suggesting that these networks have a longer memory.

Overall, the Jaccard similarity decay, similarly to autocorrelation analysis, supports that networks retain memory in their interactions, and it would be feasible to predict future weights relying on the past.

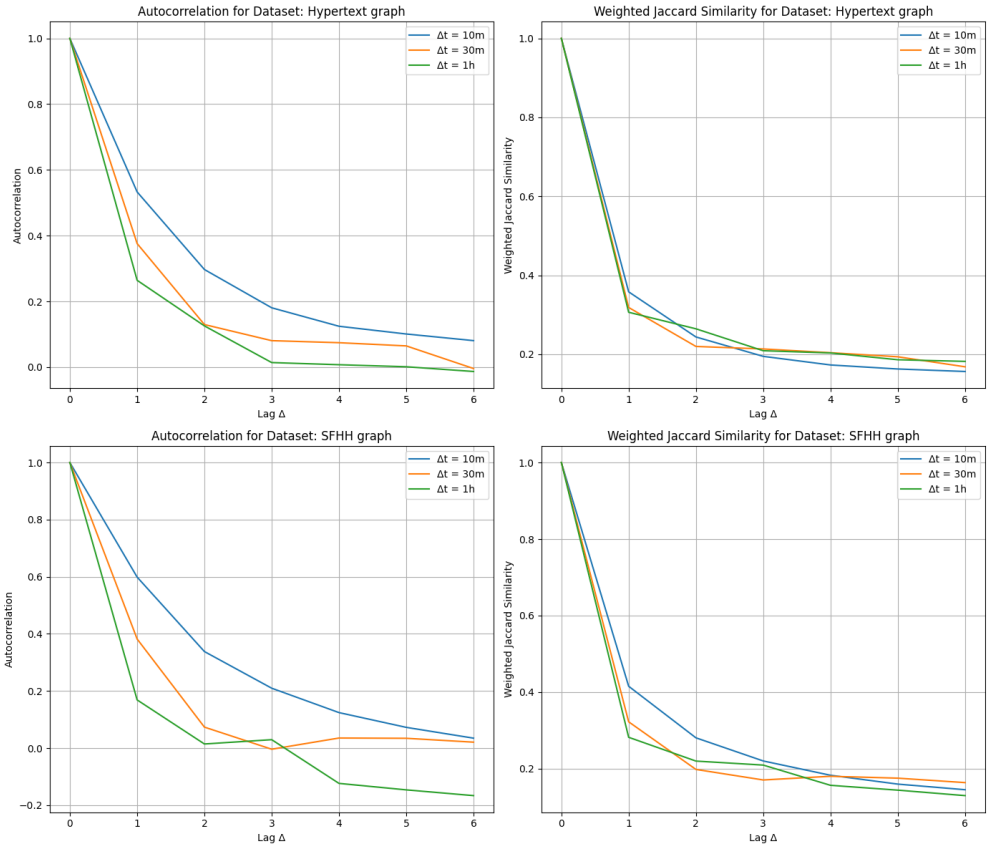


Figure 2: Autocorrelation and Weighted Jaccard Similarity on Physical datasets

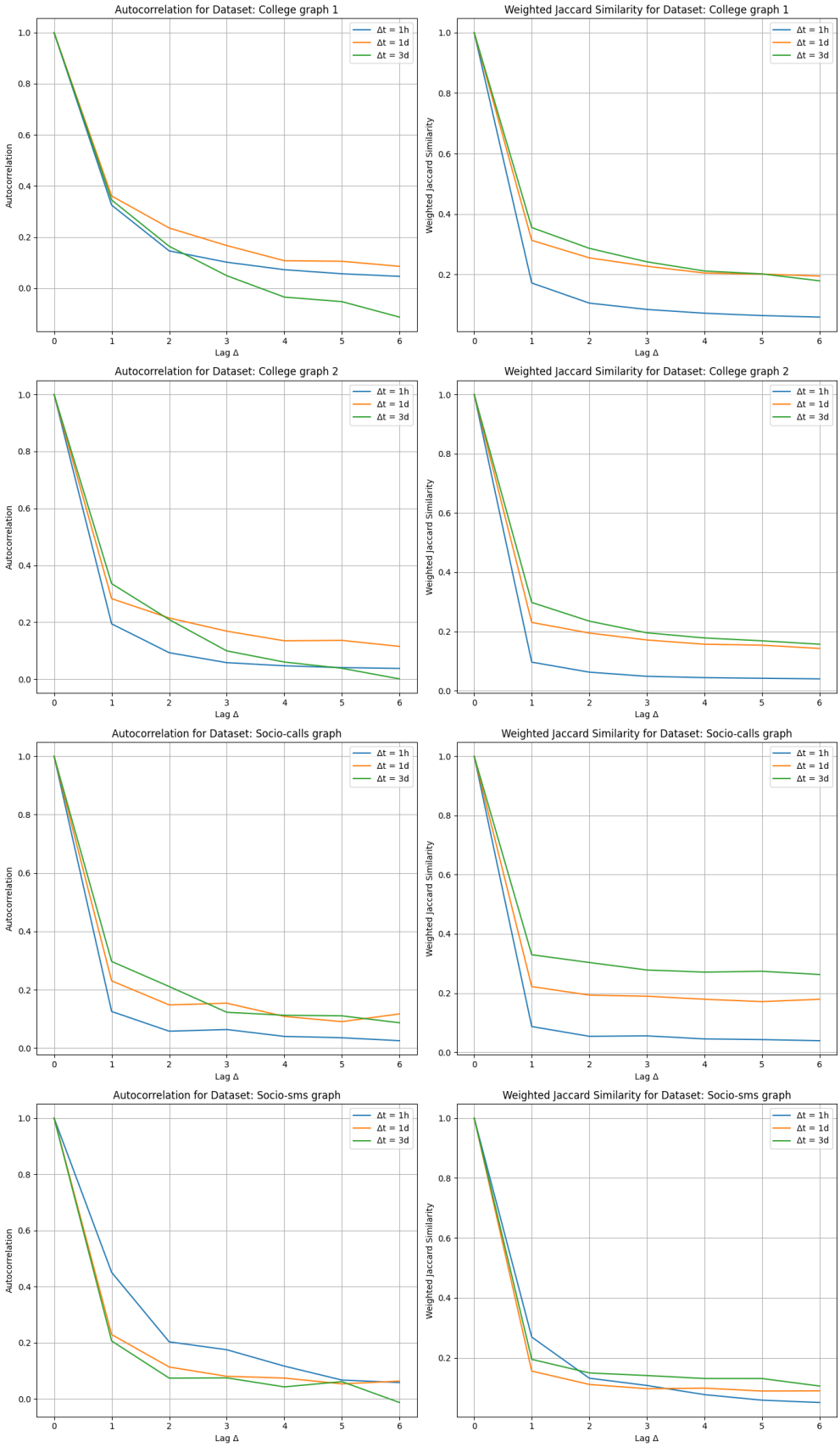


Figure 3: Autocorrelation and Weighted Jaccard Similarity on Virtual datasets

4 Methodology

Predicting weighted link activity in temporal networks requires a careful balance between *interpretability*, *computational feasibility*, and *predictive performance*. To address these requirements, we adapt the self-driven (SD) and self- and cross-driven (SCD) models introduced in [22, 21] to the task of link weight prediction, leveraging their demonstrated capacity to capture memory effects in a range of real-world networks. These models focus on low complexity and high interpretability, making them especially suitable for large-scale temporal networks with strong autocorrelation or limited computational resources.

In addition to the SD and SCD models, a simple *baseline model* is employed to gauge the performance improvements offered by more sophisticated approaches. Moreover, we propose a *modified* SCD variant (SCD*) that refines how neighboring link influences are incorporated, offering a potential advantage in networks where different sets of neighbors exert distinct impacts.

Lastly, to assess the predictive accuracy and robustness of each model, we introduce a set of evaluation metrics, which are described in detail in a later section. Together, these models and metrics form an interpretable, computationally efficient framework designed to handle weighted link prediction tasks under diverse temporal and structural conditions.

4.1 Models

The SD and SCD models were chosen based on a balance between prediction accuracy, computational efficiency, and interpretability. Both models demonstrated strong performance in predicting the dynamics of the temporal network in a variety of real-world datasets [21].

Low Complexity: The SD and SCD models offered significantly lower computational costs compared to more complex machine learning techniques such as graph neural networks (GNNs). This efficiency arose from the use of simple and interpretable parameters (decay factor τ and beta coefficients) rather than extensive optimization processes. In long-term and large-scale network predictions, this low complexity was crucial to maintain computational efficiency without sacrificing predictive accuracy.

High Interpretability: The interpretability of the SD and SCD models differentiates them from black-box machine learning models. The SD model focused on the memory of the past activities of each link, while the SCD model incorporated contributions from neighboring links, providing valuable information on how historical interactions and neighborhood dynamics affected future link activities.

Good Performance: Both models outperformed traditional methods, such as common neighbor and Lasso regression, particularly in networks with significant memory effects. By accounting for temporal correlations and neighboring link activities, the SD and SCD models captured both short-term and long-term dynamics effectively [21].

4.1.1 Baseline Model

The baseline model provided a simple benchmark, assuming the weight of a link at the next time step would remain unchanged from the last observed weight. This assumption reflects a "persistence" approach, often effective in settings where temporal changes are minimal or exhibit strong autocorrelation. The predicted weight of the link i at time step $t+1$, $x_i(t+1)$, was:

$$x_i(t+1) = x_i(t) \tag{8}$$

Despite its simplicity, the baseline model serves as an important point of comparison, helping to evaluate the incremental value provided by more complex models such as the SD and SCD.

4.1.2 Self-Driven Model (SD)

The Self-Driven (SD) model was designed to predict the weights of links by capturing the memory effect in temporal networks. The core motivation for using the SD model was its simplicity and ability to represent how recent interactions carried more influence than older ones, making it particularly well-suited for dynamic networks where temporal decay played a crucial role. Importantly, the SD model serves as the foundation for the more comprehensive Self- and Cross-Driven (SCD) model, which extends its capabilities by incorporating neighborhood effects.

The SD model operated under the assumption that the recent activity of a link was a good predictor of its future activity, but past interactions lost relevance as time passed. In the original research, the SD model was used to perform binary classification to determine whether a link was active or not. However, in this research, the model is adapted to predict link weights, requiring a normalization of historical contributions. The weight of a link i at time step $t+1$ is computed as a weighted mean of its past activities, where the decay factor $e^{-\tau(t-k)}$ acts as a weight:

$$x_i^{SD}(t+1) = \frac{\sum_{k=t-L+1}^{t} e^{-\tau(t-k)} x_i(k)}{\sum_{k=t-L+1}^{t} e^{-\tau(t-k)}} \tag{9}$$

Here, the decay factor τ controls the rate at which past interactions lose influence, allowing the model to flexibly adjust the importance of historical data. The observation window L limits the number of past time steps considered, focusing the prediction on the most recent interactions.

Normalization is necessary in the SD model to ensure that the influence of historical link activities is properly balanced when predicting future link weights. Without normalization, the decay factor $e^{-\tau(t-k)}$, which assigns decreasing importance to older interactions, could distort the contributions of past activities, especially when the decay rate (τ) is high or when the observation window (L) is large. This ensures the predictions remain consistent and interpretable across different settings, where decay rates or observation windows may vary.

4.1.3 Self- and Cross-Driven Model (SCD)

The Self- and Cross-Driven (SCD) model extends the Self-Driven (SD) model by incorporating influences from neighboring links to enhance link prediction accuracy in temporal networks, as introduced and detailed in [21]. The SCD model builds upon the SD model, using it as a foundational component to represent the self-driven contribution of a link's past activities. In addition to the self-driven dynamics, the SCD model integrates cross-driven contributions from neighboring links, making it particularly effective in networks exhibiting clustering and triadic closure properties. This adaptation enables the SCD model to capture the interconnected nature of link activities, where both the link's history and the dynamics of its neighbors play crucial roles.

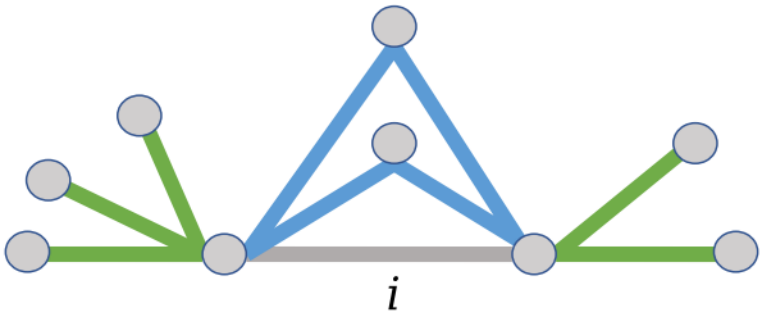


Figure 4: SCD Model visualization: The target link itself, links forming a triangle with the target link, and other neighboring links are colored in gray, blue, and green, respectively. [21]

The predicted weight of link i at time $t+1$ is defined by:

$$x_i^{SCD}(t+1) = \beta_0 + \beta_1 x_i^{SD}(t+1) + \beta_2 u_i(t+1) + \beta_3 g_i(t+1) \tag{10}$$

In this formula:

- $x_i^{SD}(t+1)$ represents the target link's own past activity, consistent with the SD model defined earlier. This ensures that the SCD model inherits the memory effects captured in the SD model.
- $u_i(t+1)$ represents the average of the geometric means of the SD components of neighboring links that form a triangle with link i, capturing the common neighbor effect. For example, if links w and v form a triangle with link i, the geometric mean for these links is calculated as $\sqrt{x_w^{SD}(t+1) \cdot x_v^{SD}(t+1)}$. The function $u_i(t+1)$ is the average of all such geometric means for link i.

- $g_i(t+1)$ represents the average SD component of all distinct neighboring links, capturing broader network connectivity and reflecting the overall influence of the neighborhood on the target link.

As in the original research [21], the parameters β_0, β_1, β_2, and β_3 are trained using Lasso regression, where β_0 serves as the intercept term. Lasso regression is particularly well-suited for this application due to its ability to impose regularization, which helps prevent overfitting by shrinking less important coefficients to zero. This ensures a more robust and interpretable model, with the parameter values reflecting the relative contributions of the self-driven and cross-driven components to the predicted link weights.

4.1.4 Modified Self- and Cross-Driven Model (SCD*)

The Modified Self- and Cross-Driven (SCD*) model is a potential refinement of the original SCD model, aimed at exploring an alternative approach to calculating the $g_i(t+1)$ component, which captures the influence of distinct neighboring links. This suggestion seeks to better represent the diverse impacts of different neighboring link types in temporal weighted networks, particularly in scenarios where the influence of neighbors varies significantly across the network. While promising, this modification is a proposed extension and not a definitive improvement over the SCD model.

Unlike the SCD model, which treats all distinct neighbors equally, the SCD* model suggests categorizing these neighbors into two groups based on their connection to each of the two nodes of the target link. This separation aligns with certain network scenarios, such as social networks, where distinct "friend" groups associated with each node can influence link dynamics differently. If the average of one group is zero, the geometric mean of both groups will also be zero, highlighting a mutual dependency in their influence on the target link.

The predicted weight of link i at time $t+1$ in the SCD* model follows the structure of the SCD model, with a modified calculation for the $g_i(t+1)$ component represented as $g_i^*(t+1)$:

$$x_i^{SCD*}(t+1) = \beta_0 + \beta_1 x_i^{SD}(t+1) + \beta_2 u_i(t+1) + \beta_3 g_i^*(t+1) \tag{11}$$

In this equation:

- $x_i^{SD}(t+1)$ represents the SD component, capturing the target link's own past activities with time decay, as defined in the original SCD model.
- $u_i(t+1)$ reflects the average of the geometric means of the SD components of neighboring links that form triangles with link i, maintaining the common neighbor effect as in the SCD model.
- $g_i^*(t+1)$ incorporates the proposed refinement, where distinct neighbors are separated into two groups based on their connection to each of the two nodes defining link i. The calculation takes the geometric mean of the averages of the two groups, offering a potentially more nuanced representation of how each node's neighbors uniquely contribute to the overall link dynamics.

This modification aims to improve the specificity with which the SCD* model captures network dynamics, particularly in networks with heterogeneous neighbor influence patterns. By emphasizing the unique contributions of neighbors tied to each node individually, the SCD* model might offer better predictive accuracy in such contexts. However, further experimentation is required to validate the practical benefits of this modification compared to the original SCD model, particularly in terms of scalability, interpretability, and generalizability across different network types.

4.2 Model Evaluation

In this study, a sliding window approach was used to calculate the performance metrics for the Baseline, SD, SCD and SCD* models. This approach ensures that the models are evaluated over multiple time steps, allowing for a robust assessment of their predictive capabilities. The sliding window technique is particularly effective for temporal networks, where the goal is to predict future link activity based on past interactions.

4.2.1 Sliding Window

Dataset					
t1	t2	Predict	t4	t5	t6
t1	t2	t3	Predict	t5	t6
t1	t2	t3	t4	Predict	t6
t1	t2	t3	t4	t5	Predict

Figure 5: Sliding Window Approach with $L = 2$

The sliding window approach operates as follows:

1. A fixed **observation window** L is defined, representing the number of past time steps considered by the model to make a prediction. The model uses the interactions within this window to predict the link activities for the next time step, $L + 1$.

2. Once the prediction for $L + 1$ is made, the observation window shifts forward by one time step. The oldest time step is discarded, and the next time step is added to the window. This shifting mechanism allows the model to continuously update its predictions for subsequent time steps.

3. This process repeats iteratively across the dataset, resulting in a total of $T - L - 1$ predictions, where T is the total number of time steps in the dataset. By evaluating predictions over the entire time span of the dataset, the sliding window approach provides a comprehensive measure of the model's performance.

4. To manage computational efficiency and ensure uniform evaluation, the number of predictions was capped at 300. If $T - L - 1$ exceeded this threshold, predictions were sampled at evenly spaced intervals throughout the dataset, maintaining consistent coverage.

4.2.2 Mean Squared Error (MSE)

MSE is a key metric in this study and will be used as the target for optimization. It measures the average squared difference between the predicted and actual link weights and is defined as:

$$\text{MSE} = \frac{1}{M} \sum_{i=1}^{M} \left(x_i(t+1) - \hat{x}_i(t+1) \right)^2 \tag{12}$$

Where $x_i(t+1)$ represents the true weight of link i at time $t+1$, $\hat{x}_i(t+1)$ is the predicted weight, and M is the number of links. MSE is particularly useful for penalizing larger errors due to the squared term, making it sensitive to outliers and extreme deviations in predictions. However, MSE alone may not fully capture other aspects of model performance, such as robustness to outliers or prediction accuracy on imbalanced datasets.

4.2.3 Area Under the Precision-Recall Curve (AUPRC)

The Area Under the Precision-Recall Curve (AUPRC) [2] is a critical metric for evaluating model performance in imbalanced datasets, particularly for distinguishing active and inactive links in temporal networks. Unlike Mean Squared Error (MSE), which focuses on the accuracy of predicted link weights, AUPRC emphasizes the ability of the model to correctly identify active links.

Data analysis revealed that some resulting temporal networks had an average percentage of active links ranging from only 5% to 15% (Table 2). Such sparsity highlights the limitations of traditional metrics like accuracy, which can be artificially inflated by models that trivially predict most links as inactive. In contrast, AUPRC directly evaluates the model's ability to detect the rare active links while avoiding false positives, providing a robust measure of performance in highly imbalanced settings.

To compute the AUPRC, continuous link weight predictions are converted into binary classifications. A link is considered *active* if its predicted weight exceeds 1.0 and *inactive* otherwise. This binary classification enables the calculation of precision and recall, defined as:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \tag{13}$$

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}} \tag{14}$$

Precision quantifies the proportion of correctly identified active links among all links predicted as active, while recall measures the proportion of correctly identified active links among all actual active links. Varying the decision threshold (e.g., 0.1, 0.2, ..., 0.9) generates a Precision-Recall curve, showing the trade-off between precision and recall across thresholds. The AUPRC is computed as the area under this curve:

$$\text{AUPRC} = \int_0^1 \text{Precision}(r)\, dr, \tag{15}$$

where $\text{Precision}(r)$ represents precision as a function of recall r. This integral is approximated numerically using the discrete points from the Precision-Recall curve.

AUPRC not only addresses class imbalance but also reflects topological differences between predicted and true networks. Active links define the network's structure, influencing connectivity and dynamics, such as information or disease spread. By emphasizing accurate identification of active links, AUPRC ensures the predicted network aligns more closely with the true topology, reducing errors that could distort clustering, pathways, or connectivity. Complementing MSE, AUPRC evaluates both numerical accuracy and structural fidelity, which is crucial in sparse temporal networks where active links are pivotal.

4.3 Parameters Selection

In this section, we discuss the motivation behind choosing key model parameters—namely the decay factor τ, the observation window L, and the β coefficients in the SCD and SCD* models—and describe the strategies used to systematically determine their values. The proper selection of these parameters ensures balanced contributions from recent and historical data, more robust predictive accuracy, and a clearer interpretation of the underlying processes governing weighted link predictions.

4.3.1 Selection of Decay Factor τ

The decay factor τ governed how quickly the influence of past activities diminished over time in the SD and SCD models. Selecting an appropriate value for τ was critical for balancing the contributions of recent and historical link activities, thereby optimizing prediction accuracy and effectively capturing the temporal dynamics of the network.

In [21], it was argued that τ could be chosen arbitrarily within the range $[0.5, 5]$ without the need for dynamic learning, which has lower computational complexity and better prediction accuracy than learning τ over time. Building on this insight, the grid search in this study tested an expanded set of candidate values to ensure broader coverage of potential decay rates:

$$\tau \in \{0.1, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 2, 3, 5\}.$$

This range spans from very slow memory decay (low τ), where historical interactions retain influence for an extended period, to rapid memory decay (high τ), where only the most recent activities significantly impact predictions. The predefined range allowed for systematic tuning of τ without resorting to computationally expensive dynamic learning. This approach ensured robust and interpretable analysis of the network's temporal behavior while leveraging the benefits of low computational complexity and reliable prediction accuracy identified in the original research.

4.3.2 Selection of Observation Window L

The observation window L controlled how many past time steps (snapshots) were used to predict future link weights. While the decay factor τ naturally reduces the influence of older interactions through an exponential decay, it often retains a thin tail of influence from distant past interactions. Limiting the observation window L provides an additional mechanism for restricting the model's memory, explicitly cutting off any contributions beyond a fixed number of recent time steps.

Moreover, limiting L aligns with real-world scenarios where only a fixed window of recent data is available due to constraints such as storage, computational resources, or real-time processing requirements. By simulating these practical limitations, the observation window L introduces an additional layer of realism to the model's design.

In this study, the observation window L was systematically varied across a set of predefined values:

$$L \in \{1, 3, 5, 10, 20\}.$$

These values cover a range of memory depths, from very short-term memory ($L = 1$) to longer-term memory ($L = 20$). Smaller values of L could fit better for networks with highly dynamic interactions, where recent activity dominates the temporal dynamics, while larger values are better suited for networks with slower memory decay.

4.3.3 Selection of β Parameters

The β parameters in the SCD and SCD* models were optimized using Lasso regression, with the goal of minimizing the mean squared error (MSE) between the predicted and actual link weights. Lasso regression was chosen for its ability to introduce regularization, which shrinks less significant coefficients towards zero, effectively removing them if they do not contribute meaningfully to the prediction.

The β parameters capture the contributions of various components influencing the predicted link weights:

- **Self-driven (SD) prediction** (β_1): Representing a link's own historical activity, as computed by the SD model. This term accounts for the memory effect of the target link.
- **Common neighbors** (β_2): Reflecting the shared influence from mutual connections (triadic closure), calculated as the geometric mean of neighboring links forming triangles with the target link.
- **Distinct neighbors** (β_3): Capturing the impact from unique neighboring links that are not directly connected to each other, providing a broader view of network connectivity beyond common neighbors.

An intercept term, β_0, was also included in the model to account for baseline activity levels. The intercept ensures that the model captures inherent biases or average behaviors in the network that are independent of the predictive components.

By minimizing the MSE through Lasso regression, the model achieves an optimal balance between predictive accuracy and complexity. The regularization introduced by Lasso ensures that only the most meaningful predictors contribute to the final model, improving its robustness and scalability to large temporal networks. This approach ensures that the β parameters are fine-tuned to effectively reflect the contributions of the self-driven, common neighbor, and distinct neighbor effects, while maintaining computational efficiency.

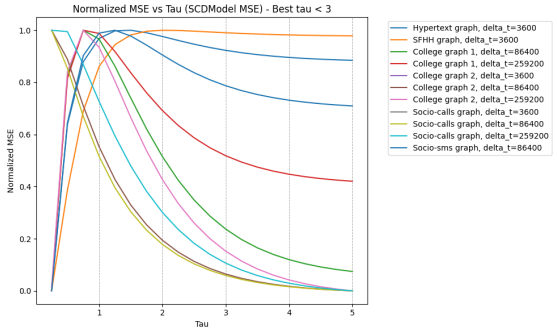
5 Results

In this section, we present a comprehensive evaluation of the proposed models - Baseline, SD, SCD, and SCD* across multiple temporal weighted networks. The performance assessment focuses primarily on numerical accuracy, measured by the Mean Squared Error (MSE), and the ability to distinguish active from inactive links, gauged by the Area Under the Precision-Recall Curve (AUPRC). We further analyze the influence of two central parameters, the decay factor τ and the observation window L, highlighting how each impacts model performance in diverse network settings. Finally, we incorporate

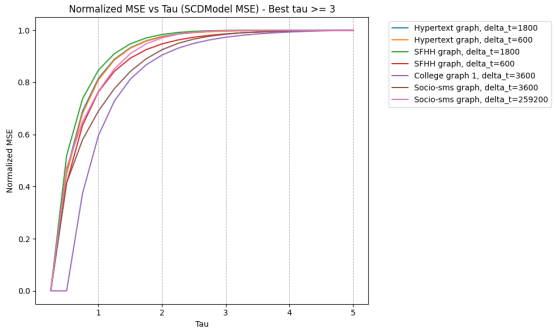
a correlation analysis to explore whether certain network properties such as clustering, link density, or weighted entropy have a discernible effect on the models' predictive capabilities and parameter choices. The subsequent subsections detail the key findings and insights gleaned from these experiments.

5.1 Decay Factor τ Analysis

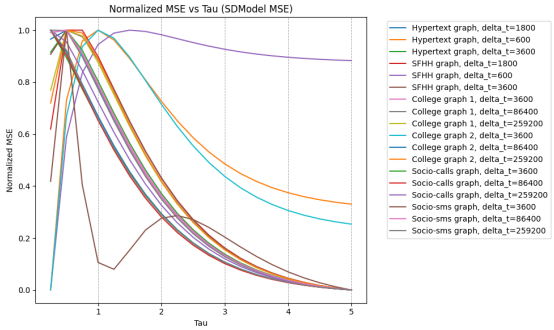
The impact of the decay factor τ on model performance was examined by varying τ within the range [0.25, 5.0] in increments of 0.25, while keeping the observation window fixed at $L = 10$. Figure 6 presents the relationship between τ and the Mean Squared Error (MSE) for the SD, SCD, and SCD* models. To facilitate a consistent comparison across datasets and models, the MSE values were normalized to a scale between 0 and 1, where 1 represents the best possible performance (lowest MSE), and 0 represents the worst (highest MSE).



(a) SCD Model where optimal MSE with $\tau < 3$



(b) SCD Model where optimal MSE with $\tau >= 3$



(c) SD Model

Figure 6: τ to MSE analysis with L=10

The results indicate that the SD model consistently achieved optimal performance when τ was within the range of 0.25 to 1.5 across all datasets. This suggests that the SD model benefits from a relatively slower memory decay, where past interactions retain a significant influence on future link predictions. In contrast, the SCD model exhibited more varied behavior depending on the the dataset.

Upon further analysis, datasets were grouped based on their optimal τ values into two categories: those with $\tau < 3$ and those with $\tau \geq 3$. Visual categorization, illustrated in Figure 6b and Figure 6a, highlights distinct behavioral patterns across datasets, suggesting fundamental differences in temporal dependencies and activity levels.

To examine the relationship between dataset properties and optimal τ values, five key features were analyzed: average clustering coefficient, average and standard deviation of the percentage of links per snapshot, average weighted interaction entropy, and average edge weight. Pearson[7] and Spearman[8] correlations were used to assess linear and rank-based dependencies.

The analysis did not reveal any definitive correlations between the examined properties and the optimal τ categories, indicating the complexity of the relationship between temporal network char-

[7] https://en.wikipedia.org/wiki/Pearson_correlation_coefficient
[8] https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient

acteristics and decay factor selection. This suggests that more datasets are required, or additional features may be required to better capture the underlying dynamics.

5.2 Grid search results

The performance of the models was compared across several datasets, focusing on their ability to predict link weights in temporal networks. Table 3 summarizes the MSE for each model. Lower MSE values indicate better performance in predicting link weights, while higher AUPRC suggests better prediction of network topology. The results selected in bold represent best MSE/AUPRC result for each dataset across different models.

Table 3: Comparison of Prediction Results Across Different Datasets and Models

Dataset	Δt	**Baseline**		**SD Model**		**SCD Model**		**SCD* Model**	
		MSE	**AUPRC**	**MSE**	**AUPRC**	**MSE**	**AUPRC**	**MSE**	**AUPRC**
Physical Datasets									
Hypertext graph	10m	56.52	**0.78**	54.0	0.77	**45.85**	0.73	45.88	0.73
	30m	409.12	**0.84**	339.07	0.82	**304.11**	0.8	**304.11**	0.8
	1h	1372.53	**0.89**	1070.57	0.85	**958.55**	0.84	**958.55**	0.84
SFHH graph	10m	134.62	**0.84**	128.7	0.82	111.97	0.71	**111.81**	0.71
	30m	1011.45	**0.76**	931.93	0.72	**756.14**	0.72	759.15	0.72
	1h	4337.83	0.73	3182.34	**0.74**	1347.1	0.73	**1337.06**	0.73
Virtual Datasets									
College graph 1	1h	4.73	**0.44**	3.95	0.43	**3.31**	**0.44**	**3.31**	**0.44**
	1d	305.46	**0.86**	236.31	0.85	212.78	0.8	**212.16**	0.8
	3d	83.74	0.64	83.93	0.65	**11.42**	**0.7**	11.73	0.69
College graph 2	1h	0.6	0.27	**0.53**	0.32	0.67	**0.53**	0.67	**0.53**
	1d	61.73	0.7	36.54	0.7	**33.26**	**0.71**	**33.26**	**0.71**
	3d	141.22	0.8	106.91	**0.81**	**100.4**	0.75	**100.4**	0.75
Socio-calls graph	1h	1.1	0.3	0.72	0.26	**0.64**	**0.54**	**0.64**	**0.54**
	1d	19.09	0.7	12.44	0.72	**11.77**	**0.73**	**11.77**	**0.73**
	3d	60.62	0.86	42.9	**0.87**	**41.14**	0.83	**41.14**	0.83
Socio-calls graph	1h	1.1	0.3	0.72	0.26	**0.64**	**0.54**	**0.64**	**0.54**
	1d	19.09	0.7	12.44	0.72	**11.77**	**0.73**	**11.77**	**0.73**
	3d	60.62	0.86	42.9	**0.87**	**41.14**	0.83	**41.14**	0.83

Overall, the results demonstrate a consistent trend where the SCD model significantly outperforms the Baseline across all datasets, achieving an average improvement of 32.86% in MSE. The SD model also shows a notable improvement of 19.31% over the Baseline, highlighting the impact of incorporating temporal decay. Furthermore, transitioning from the SD to SCD model yields an additional 15.87% reduction in MSE, underscoring the advantage of integrating structural and dynamic components in link prediction.

While MSE provides insights into numerical accuracy, the Area Under the Precision-Recall Curve (AUPRC) was used as a supplementary metric to evaluate the models' ability to distinguish active and inactive links within the network. Despite the improvements in MSE, AUPRC results exhibit dataset-specific variations. In some datasets, the Baseline model achieves higher AUPRC scores, while in others, SD and SCD models demonstrate slight improvements. This suggests that although the SCD model excels in predicting link weights, its ability to capture network topology varies across different temporal characteristics. During the correlation analysis, it was found that datasets with higher average edge activity per snapshot are often having higher AUPRC score, as can be seen in

Section 5.3.

Interestingly, the SCD* model did not show significant improvements over the SCD model, indicating that the added complexity does not necessarily translate to better predictive performance. This suggests that balancing model complexity with dataset-specific characteristics is crucial to achieving optimal results without overfitting.

Table 4: Comparison of Best Parameters Found Across Different Datasets for SD, SCD and SCD* Models

Dataset	Δt	**SD Model**		**SCD Model**					**SCD* Model**				
		τ	**L**	τ	**L**	β_1	β_2	β_3	τ	**L**	β_1	β_2	β_3
Physical Datasets													
Hypertext graph	10m	1.0	5	1.5	1	0.58	0.0	0.05	1.5	1	0.6	0.0	0.02
	30m	0.5	20	2.0	20	0.5	0.03	0.0	2.0	20	0.5	0.03	0.0
	1h	0.5	10	1.5	10	0.46	0.0	0.0	1.5	10	0.46	0.0	0.0
SFHH graph	10m	1.5	20	5.0	20	0.51	0.09	0.18	5.0	20	0.55	0.0	0.27
	30m	1.0	20	5.0	20	0.29	0.0	0.3	5.0	20	0.41	0.0	0.2
	1h	0.5	20	5.0	20	0.07	0.09	0.2	5.0	20	0.08	0.0	0.39
Virtual Datasets													
College graph 1	1h	0.5	5	5.0	3	0.15	0.0	0.0	5.0	3	0.15	0.0	0.0
	1d	0.5	20	1.0	20	0.43	0.1	0.09	1.0	20	0.43	0.07	0.2
	3d	5.0	20	5.0	20	0.17	0.0	0.17	5.0	20	0.29	0.0	0.0
College graph 2	1h	1.0	10	1.5	10	0.0	0.0	0.0	1.5	10	0.0	0.0	0.0
	1d	0.5	20	0.5	20	0.51	0.0	0.0	0.5	20	0.51	0.0	0.0
	3d	0.5	10	1.0	10	0.7	0.02	0.0	1.0	10	0.7	0.02	0.0
Socio-calls graph	1h	0.5	5	0.5	5	0.0	0.0	0.0	0.5	5	0.0	0.0	0.0
	1d	0.5	20	0.5	20	0.55	0.0	0.0	0.5	20	0.55	0.0	0.0
	3d	0.5	20	0.5	20	0.78	0.01	0.0	0.5	20	0.78	0.01	0.0
Socio-sms graph	1h	1.0	20	5.0	20	0.41	0.0	0.0	5.0	20	0.41	0.0	0.0
	1d	0.5	20	1.0	20	0.49	0.11	0.04	1.0	20	0.5	0.13	0.02
	3d	0.5	10	3.0	10	0.43	0.05	0.01	5.0	10	0.43	0.01	0.05
Mean		0.92	15.28	2.47	14.94	0.39	0.03	0.06	2.58	14.94	0.41	0.02	0.06

The grid search results highlight consistent patterns in the optimal parameter configurations for the SD, SCD, and SCD* models. Most datasets favored maximizing the observation window length L, except in cases with extreme aggregation densities, such as densely or sparsely aggregated virtual networks (e.g., College graph at 1-hour and 3-day intervals) and the Hypertext graph at 10-minute intervals, where shorter windows performed better.

The decay factor τ for the SD model was consistently optimal in the range of 0.5 to 1 across nearly all datasets, indicating a preference for shorter memory spans. In contrast, the SCD and SCD* models occasionally benefited from higher values, such as $\tau = 5.0$, which was discussed in Section 5.1.

For the SCD* model, the distinct neighbor weighting parameter (β_3) showed mixed trends—sometimes exceeding and sometimes falling below values observed in the original SCD model. However, on average, its contribution remained comparable, indicating no substantial deviation in performance due to this modification.

These findings underscore the importance of dataset-specific parameter tuning, as interaction frequency and aggregation levels significantly influence optimal values. The results also suggest that while SD benefits from long-term dependencies, SCD and SCD* models might sometimes leverage short-term temporal dynamics to enhance predictive accuracy.

5.3 Correlation Analysis

In order to investigate potential factors influencing model performance, an initial correlation analysis was conducted by comparing the Mean Squared Error (MSE) results with the squared average weight of links in the network. This comparison aimed to determine whether a linear relationship exists between these two variables.

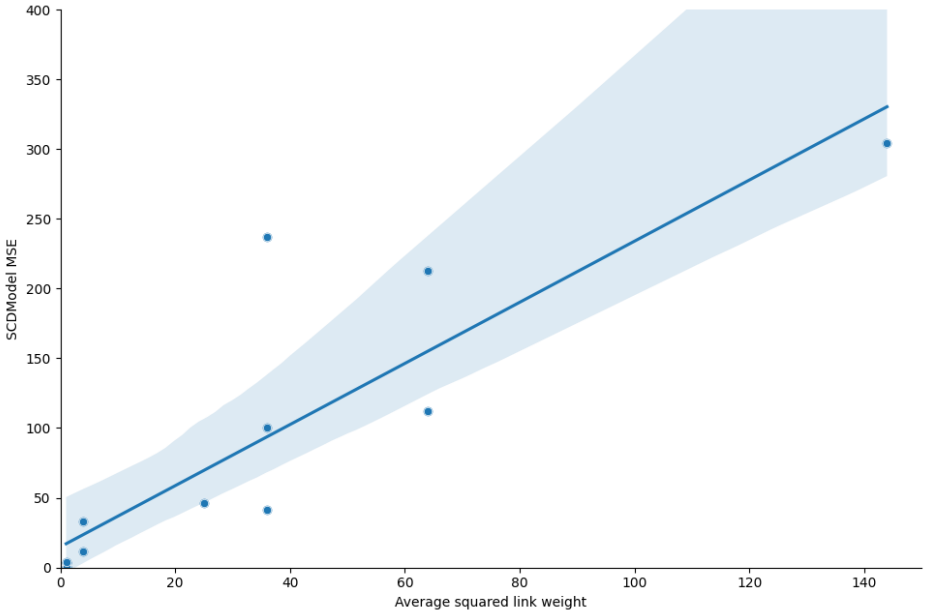


Figure 7: Squared Average Link Weight vs SCD Model MSE plot.

As shown in Figure 7, the scatter plot suggests a linear correlation between the squared average link weight and the SCD model's MSE. This relationship can be attributed to the fact that both MSE and squared link weight share the same scale, meaning that networks with higher link weights tend to produce larger errors in absolute terms. Consequently, we will normalize MSE by squared link weight (Figure 8) before doing the correlation analysis with other network properties.

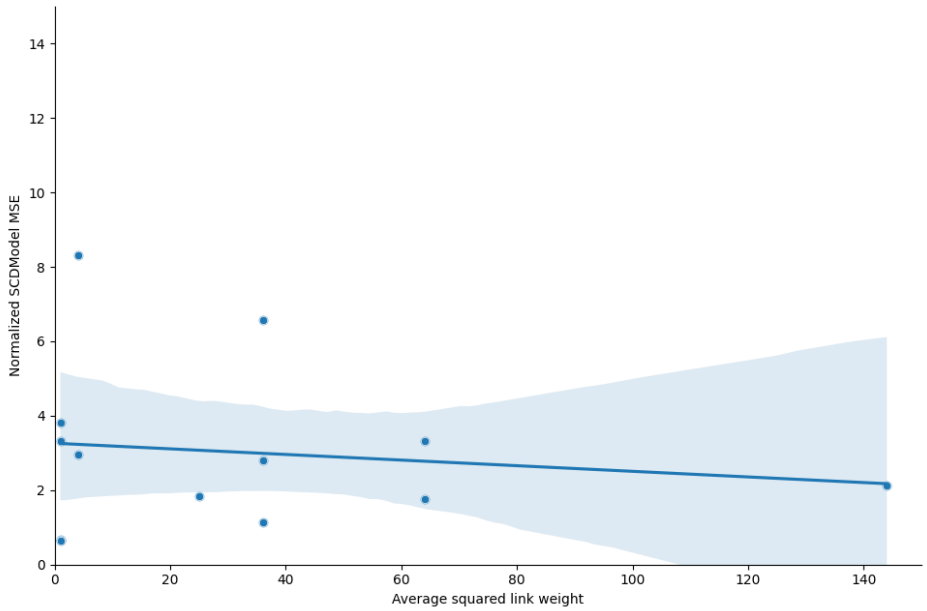


Figure 8: Squared Average Link Weight vs Normalized SCD Model MSE plot.

After MSE was properly normalized, the Spearman correlation analysis was performed to find relationships between network properties and model performance. (Figure 9)

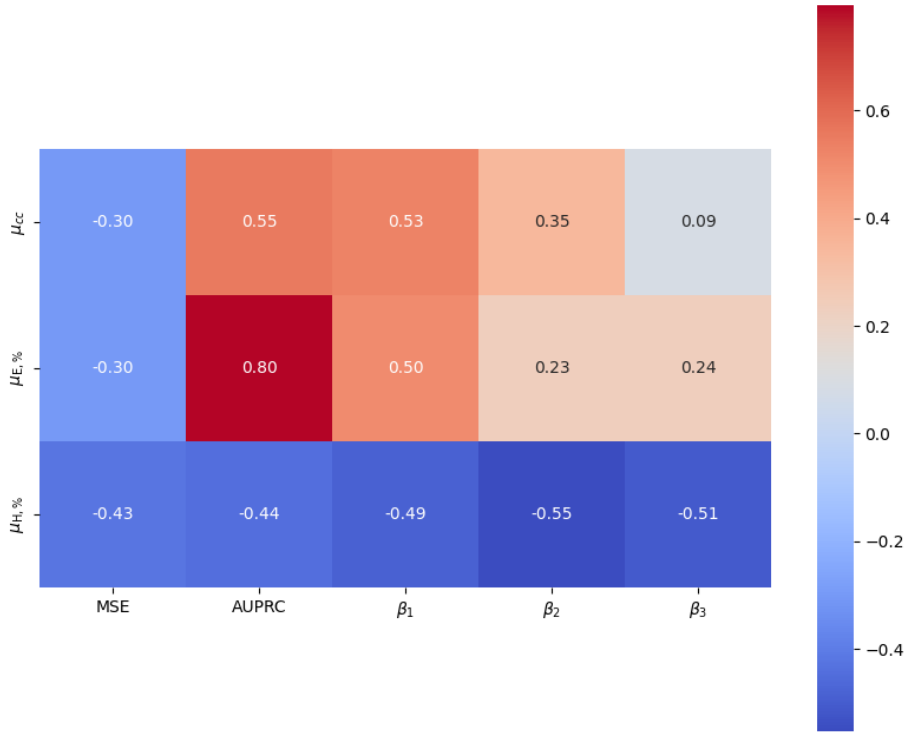


Figure 9: Spearman correlation heatmap of dataset properties to SCD model results

The analysis revealed no strong correlations between network properties and MSE scores. In contrast, AUPRC was found to exhibit a strong positive correlation (0.80) with the average percentage of active links. This suggests that datasets with a higher proportion of active links increases the potential of SCD model to distinguish between links that would have at least one contact or no contacts at all at next timestamp, resulting in better topological predictions. Additionally, a slight positive correlation was observed between AUPRC and the average clustering coefficient (0.55), implying that in networks with higher clustering, link formation patterns may be more predictable, thus enhancing the model's ability to identify active links.

Further investigation into the relationship between network properties and the optimal β parameters of the SCD model revealed minimal correlations. The strongest observed trend was a moderate positive correlation (0.53) between the average clustering coefficient and the SD component β_1, indicating that more clustered networks may benefit from a stronger temporal decay effect. Additionally, a slight negative correlation of -0.55 was found between average weighted entropy and the common (β_2) and distinct (β_3) neighbor components, suggesting that in networks with higher entropy, the importance of neighbors in predicting link formation slightly diminishes.

These observations highlight the complexity of the relationship between network features and model performance. While certain properties such as link activity and clustering exhibit clear trends, others, like entropy and weight distributions, present more nuanced effects that require further exploration. The findings suggest that predictive performance is influenced by a combination of factors, reinforcing the need for adaptive approaches that dynamically adjust model parameters based on the specific characteristics of each dataset.

6 Conclusions

This thesis has extended established Self-Driven (SD) and Self- and Cross-Driven (SCD) models from binary to weighted temporal link prediction, aiming to preserve the dual goals of interpretability and computational scalability. By focusing on the *intensity* of link interactions, in addition to their presence or absence, our work offers a more nuanced view of how network weights change over time. Our principal objectives were twofold: *(1)* to develop an interpretable, memory-based prediction algorithm that makes use of the underlying structure of temporal networks, and *(2)* to investigate how different network properties affect model performance. Below, we describe our findings with respect to these goals.

Research Question 1: Developing an Interpretable Algorithm for Weighted Temporal Link Prediction

We showed that by incorporating exponentially decaying memory of past link activity (SD) and neighborhood-driven interactions (SCD), it is possible to model and forecast future link weights in a transparent manner. In particular:

- **Baseline *vs.* SD.** SD model, which focuses purely on a link's own decaying memory, outperformed a persistence-based baseline in terms of Mean Squared Error (MSE). This highlights the potential in models based on the natural memory decay in the networks.
- **SCD Enhancements.** By capturing local connectivity patterns (e.g., shared neighbors or distinct neighbors), the SCD model achieved further improvements in predictive accuracy. Incorporating memory decay of neighboring links have led to a significant MSE score improvement over the baseline and SD models.
- **SCD* Variation.** A refined neighbor-grouping approach (SCD*) did not yield a systematic improvement, indicating that the benefits of additional structural granularity depend heavily on dataset-specific factors. However, this direction remains a promising avenue for networks where distinct neighbor influences vary significantly across the two nodes of a link.

These findings confirm that interpretable models, built upon simple but flexible notions of temporal decay and network topology, can serve as practical tools for weighted link prediction in temporal networks.

Research Question 2: Network Properties Influencing Model Performance

Our analysis further revealed that certain properties of the network can guide model parameter choices and provide insight into predictive limitations:

- **Temporal Decay (τ).** While SD consistently performed best with moderate decay factors (τ in the range of 0.25-1.5), the more comprehensive SCD/SCD* frameworks occasionally benefited from larger values, especially in datasets with extreme aggregation densities.
- **Observation Windows (L).** Most datasets favored larger time windows, reflecting the value of broader historical context and long-tail impact of memory decay. Yet some cases with extreme densities — very high or very low link activity — suggested that shorter windows (or specifically chosen parameters) can yield better results, illustrating the importance of data-specific tuning.
- **Impact on Topology Prediction (AUPRC).** We observed that numerical improvements (lower MSE) did not always translate into higher Area Under the Precision-Recall Curve (AUPRC). Networks with a higher proportion of active links or stronger local clustering often benefitted more in terms of accurately identifying which edges remain active.
- **Correlations with Network Properties.** Although no single feature (e.g., clustering, entropy, or average weight) unilaterally explained optimal decay or model structure, certain moderate trends emerged. More strongly clustered networks tended to leverage the self-driven component (β_1) more, while higher entropy networks slightly reduced the importance of neighboring connections (β_2, β_3).

Overall, these findings emphasize the complexity of network evolution and point toward the benefits of adaptive strategies - for instance, data-driven selection of decay factors, observation windows, or neighbor-grouping mechanisms. In many cases, the interplay between link-level memory and local structural cues proved essential.

Summary

By adapting and systematically evaluating SD and SCD models for temporal weighted link prediction, this thesis demonstrated the importance of combining decaying historical patterns and local network structure to forecast future link intensity. While not universally optimal, these approaches offer a compelling balance between computational simplicity, clarity of interpretation, and predictive capability. Ultimately, our results indicate that network memory and structural properties both play a decisive role in shaping link dynamics, reinforcing the value of interpretable, adaptive modeling approaches in the evolving field of temporal network analysis.

7 Future Work

While the proposed methodology successfully integrates temporal decay and neighborhood influences for interpretable link weight prediction, several possibilities remain open for further investigation. In particular, certain assumptions and design choices in our approach introduce limitations that may constrain its applicability in more complex or large-scale contexts. Below we highlight key directions for future work, emphasizing both methodological enhancements and the practical challenges associated with them:

1. **Adaptive Decay Factors.** The current framework employs a single global decay factor τ, chosen from a fixed set of candidate values. While this simplifies parameter tuning and preserves interpretability, it can mask node- or link-specific differences in how past interactions influence future behavior. Future research might explore *heterogeneous* or *adaptive* decay mechanisms, where τ depends on local properties (e.g., node degrees, clustering coefficients, or volatility of interaction frequency). Such flexibility could improve prediction accuracy in systems with highly diverse activity patterns, but care must be taken to maintain interpretability and avoid overfitting.

2. **Refining Neighbor Contributions and the SCD* Approach.** Although SCD* showed only marginal improvements in our experiments, its idea of grouping neighbors by their connection to each node underscores how local structure can vary greatly even within a single network. Further work could systematically evaluate which grouping schemes best capture different types of neighbor influence, potentially drawing on domain knowledge or community-structure analyses. In both weighted and unweighted temporal link prediction settings, advanced grouping strategies (e.g., separating neighbors by link-level attributes, node roles, or multi-layer affiliations) may offer new insights, particularly if domain constraints or highly varied neighbor behaviors are present. Such refinements must be balanced, however, to avoid inflated model complexity or reduced transparency.

3. **Beyond One-Step-Ahead Predictions.** Our methodology focuses on predicting link weights one step into the future, aligning with short-term forecasting in dynamic networks. In many applications (epidemiology, supply-chain management, etc.), multi-step forecasts are more valuable. Adapting SD, SCD, and SCD* for longer predictive horizons would involve examining how errors propagate over multiple time steps and whether the simplistic exponential decay remains appropriate when data must be extrapolated further. Incorporating rolling or iterative retraining, as well as designing novel regularization schemes, might be necessary to maintain interpretability and accuracy over extended time frames.

4. **Incorporating Domain-Specific Constraints.** Our approach assumes that all link weights are interchangeable, relying mainly on historical activities and network structure. Certain domains such as finance, biology, or social policy may impose known constraints (e.g., capacity limits, regulatory policies, or intrinsic node features) that directly shape link intensities. Embedding these domain-specific constraints or features within the model could significantly improve

realism, though the added complexity might reduce the straightforward interpretability currently offered by exponential decay and neighbor-based terms.

Despite these limitations, the proposed SD, SCD, and SCD* models remain appealing for short-term temporal forecasting in networks characterized by strong memory effects. Their inherently interpretable parameters - decay factors and neighbor coefficients - allow domain experts to not only predict future link weights but also *explain* how historical interactions and local structures drive these predictions. By addressing the above methodological frontiers, future work can expand the capabilities and applicability of these models, ensuring they remain both practical and transparent for increasingly diverse and demanding real-world scenarios.

Bibliography

[1] Jeremy Charlier, Radu State, and Jean Hilger. *Modeling Smart Contracts Activities: A Tensor Based Approach*. May 2019. DOI: 10.48550/arXiv.1905.09868.

[2] Jesse Davis and Mark Goadrich. "The relationship between Precision-Recall and ROC curves". In: *Proceedings of the 23rd International Conference on Machine Learning*. ICML '06. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 2006, pp. 233–240. ISBN: 1595933832. DOI: 10.1145/1143844.1143874. URL: https://doi-org.tudelft.idm.oclc.org/10.1145/1143844.1143874.

[3] Valeria Gelardi et al. "From temporal network data to the dynamics of social relationships". In: *Proceedings of the Royal Society B: Biological Sciences* 288.1959 (Sept. 2021). ISSN: 1471-2954. DOI: 10.1098/rspb.2021.1164. URL: http://dx.doi.org/10.1098/rspb.2021.1164.

[4] Bryce Goodman and Seth Flaxman. "European Union Regulations on Algorithmic Decision Making and a "Right to Explanation"". In: *AI Magazine* 38.3 (Sept. 2017), pp. 50–57. ISSN: 2371-9621. DOI: 10.1609/aimag.v38i3.2741. URL: http://dx.doi.org/10.1609/aimag.v38i3.2741.

[5] Klaus Greff et al. "LSTM: A Search Space Odyssey". In: *IEEE Transactions on Neural Networks and Learning Systems* 28.10 (Oct. 2017), pp. 2222–2232. ISSN: 2162-2388. DOI: 10.1109/tnnls.2016.2582924. URL: http://dx.doi.org/10.1109/TNNLS.2016.2582924.

[6] Roger Guimerà and Marta Sales-Pardo. "Missing and spurious interactions and the reconstruction of complex networks". In: *Proceedings of the National Academy of Sciences* 106.52 (Dec. 2009), pp. 22073–22078. ISSN: 1091-6490. DOI: 10.1073/pnas.0908366106. URL: http://dx.doi.org/10.1073/pnas.0908366106.

[7] Petter Holme and Jari Saramäki. "Temporal networks". In: *Physics Reports* 519.3 (Oct. 2012), pp. 97–125. ISSN: 0370-1573. DOI: 10.1016/j.physrep.2012.03.001. URL: http://dx.doi.org/10.1016/j.physrep.2012.03.001.

[8] Petter Holme and Jari Saramäki. *Temporal Networks as a Modeling Framework*. 2021. arXiv: 2103.13586 [physics.soc-ph].

[9] Sergey Ioffe. "Improved consistent sampling, weighted minhash and l1 sketching". In: *2010 IEEE international conference on data mining*. IEEE. 2010, pp. 246–255.

[10] Lorenzo Isella et al. "What's in a crowd? Analysis of face-to-face behavioral networks". In: *Journal of Theoretical Biology* 271.1 (Feb. 2011), pp. 166–180. ISSN: 0022-5193. DOI: 10.1016/j.jtbi.2010.11.033. URL: http://dx.doi.org/10.1016/j.jtbi.2010.11.033.

[11] Andreas Koher et al. "Contact-Based Model for Epidemic Spreading on Temporal Networks". In: *Physical Review X* 9.3 (Aug. 2019). ISSN: 2160-3308. DOI: 10.1103/physrevx.9.031017. URL: http://dx.doi.org/10.1103/PhysRevX.9.031017.

[12] Kai Lei et al. "GCN-GAN: A Non-linear Temporal Link Prediction Model for Weighted Dynamic Networks". In: *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. 2019, pp. 388–396. DOI: 10.1109/INFOCOM.2019.8737631.

[13] Anmol Madan et al. "Sensing the "Health State" of a Community". In: *IEEE Pervasive Computing* 11.4 (Oct. 2012), pp. 36–45. DOI: 10.1109/mprv.2011.79.

[14] Christoph Molnar. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. 2nd ed. 2022. URL: https://christophm.github.io/interpretable-ml-book.

[15] Tore Opsahl and Pietro Panzarasa. "Clustering in weighted networks". In: *Social networks* 31.2 (2009), pp. 155–163.

[16] André Panisson et al. "Fingerprinting Temporal Networks of Close-Range Human Proximity". In: *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. 2013, pp. 261–266. DOI: 10.1109/PerComW.2013.6529492.

[17] Cynthia Rudin. *Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead*. 2019. arXiv: 1811.10154 [stat.ML].

[18] Petar Veličković et al. *Graph Attention Networks*. 2018. arXiv: 1710.10903 [stat.ML]. URL: https://arxiv.org/abs/1710.10903.

[19] Xuhong Wang et al. "One-class graph neural networks for anomaly detection in attributed networks". In: *Neural Computing and Applications* 33.18 (Mar. 2021), pp. 12073–12085. ISSN: 1433-3058. DOI: 10.1007/s00521-021-05924-9. URL: http://dx.doi.org/10.1007/s00521-021-05924-9.

[20] Muhan Zhang and Yixin Chen. *Link Prediction Based on Graph Neural Networks*. 2018. arXiv: 1802.09691 [cs.LG].

[21] Li Zou, Alberto Ceria, and Huijuan Wang. "Short-and long-term temporal network prediction based on network memory". In: *Applied Network Science* 8.1 (2023), p. 76.

[22] Li Zou et al. "Temporal Network Prediction and Interpretation". In: *IEEE Transactions on Network Science and Engineering* 9.3 (2022), pp. 1215–1224. DOI: 10.1109/TNSE.2021.3138643.