Information Cascade Popularity Prediction via Probabilistic Diffusion

Zhangtao Cheng, Fan Zhou, *Member, IEEE*, Xovee Xu, *Graduate Student Member, IEEE*, Kunpeng Zhang, Goce Trajcevski, *Member, IEEE*, Ting Zhong, and Philip S. Yu, *Life Fellow, IEEE*

***Abstract*—Information cascade popularity prediction is an important problem in social network content diffusion analysis. Various facets have been investigated (e.g., diffusion structures and patterns, user influence) and, recently, deep learning models based on sequential architecture and graph neural network (GNN) have been leveraged. However, despite the improvements attained in predicting the future popularity, these methodologies fail to capture two essential aspects inherent to information diffusion: (1) the temporal irregularity of cascade event – i.e., users' re-tweetings at random and non-periodic time instants; and (2) the inherent uncertainty of the information diffusion. To address these challenges, in this work, we present CasDO – a novel framework for information cascade popularity prediction with probabilistic diffusion models and neural ordinary differential equations (ODEs). We devise a temporal ODE network to generalize the discrete state transitions in RNNs to continuous-time dynamics. CasDO introduces a probabilistic diffusion model to consider the uncertainties in information diffusion by injecting noises in the forwarding process and reconstructing cascade embedding in the reversing process. Extensive experiments that we conducted on three large-scale datasets demonstrate the advantages of the CasDO model over baselines.**

***Index Terms*—Information cascade, neural ordinary equations, popularity prediction, probabilistic diffusion.**

I. Introduction

THE booming use of social media platforms such as Twitter, Sina Weibo, Facebook, etc., has positioned the online content generated and disseminated by the users to become one of the main sources of information to guide many of the individuals' everyday decisions [1], [2]. The dynamics of users activities facilitate the fast propagation of information and, in turn, bring the important aspect of information cascades [1], [3]. A similar phenomenon has been identified in other (non-social media) settings: paper citations [4], blogging space [5], and article sharing [6]. Understanding information cascades has significant economic and societal impacts – and one of the typical tasks, which has attracted great attention in the both academia and industry, is the prediction of the size of potentially affected users after a certain time [7]. For example, predicting the number of affected cases and deaths in a region during the COVID-19 pandemic [8] based on users tweets, is critical for policymakers to plan subsequent actions and resources allocation. However, the process of information diffusion has a high level of stochasticity, which makes the prediction task a challenging one.

Over the years, many research efforts have been devoted to information cascade popularity prediction [1]. The works have analyzed the patterns of information diffusion and proposed various popularity prediction approaches. In a broad sense, these approaches can be categorized into three groups. (1) *Diffusion model-based approaches:* these researches focus on modeling the intensity functions of the arrival for incoming messages to study the propagation process [9]. (2) *Feature model-based approaches:* most of the earlier studies [7] try to exploit useful hand-crafted feature-sets from cascade items, e.g., time-series, diffusion structure, and contents. (3) *Deep learning-based approaches:* these methods have achieved great successes for many applications and have been extensively applied in information dissemination learning and cascade popularity prediction [3], [10], [11]. Existing works utilize various deep neural network models towards capturing the temporal and sequential processes of information diffusion using recurrent neural networks (RNNs) [12], learning structured representations with network embedding models [3] or graph neural networks (GNNs) [13], and predicting individual user activity (e.g., forwarding or not) [14].

Challenges: Despite the success of existing deep learning-based methods, some challenges remain unaddressed and potential improvements are still possible. The real-world information diffusion processes are often irregularly sampled (i.e., the time series of different users activities are non-uniform) and exhibit noise as well as uncertainty – aspects which have not been properly modeled in the previous works. Specifically, following the observations along these lines which motivate our work: (1)

Received 3 January 2023; revised 28 May 2024; accepted 10 September 2024. Date of publication 19 September 2024; date of current version 13 November 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62176043 and Grant 62072077 and in part by the NSF under Grant III-2106758 and Grant POSE-2346158.D. Recommended for acceptance by G. Wang. *(Corresponding author: Fan Zhou.)*

Zhangtao Cheng and Xovee Xu are with the University of Electronic Science and Technology of China, Chengdu 610054, China (e-mail: zhangtao.cheng@outlook.com; xovee@live.com).

Fan Zhou is with the University of Electronic Science and Technology of China, Chengdu 610054, China, and also with the Kash Institute of Electronics and Information Industry, Kashgar 844000, China (e-mail: fan.zhou@uestc.edu.cn).

Kunpeng Zhang is with the Department of Decision, Operations & Information Technologies, University of Maryland, College Park, MD 20742 USA (e-mail: kpzhang@umd.edu).

Goce Trajcevski is with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: gocet25@iastate.edu).

Ting Zhong is with the University of Electronic Science and Technology of China, Chengdu 610054, China (e-mail: zhongting@uestc.edu.cn).

Philip S. Yu is with the Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607 USA (e-mail: psyu@uic.edu).

Digital Object Identifier 10.1109/TKDE.2024.3465241

Irregularly-sampled information diffusion process. Since people have their personal preferences and timetable in real life, they may browse tweets/microblogs and retweet them at different times. This is also the case for other types of networks, such as paper citations. The time intervals between adjacent events (e.g., retweeting and citation) are irregular. RNNs such as LSTM and GRU are the dominant models for capturing the temporal patterns of the information diffusion, which, however, are initially designed for regularly-sampled sequences and cannot reflect the influence of irregular events due to the underlying inflexible iterative structure [15]. (2) *Information diffusion evolution uncertainty.* Generally, information cascade prediction aims to predict future popularity by peeking into the early stage of a cascade's evolving process. However, the observed sharing/retweeting behaviors would inevitably introduce noises, biases, and uncertainties, which have not been well considered in existing methods. For example, many fake followers and spam accounts exist in social platforms, affecting other users' behaviors and, consequently, the diffusion process while introducing propagation uncertainty.

Nevertheless, it is non-trivial to model the irregularly-sampled diffusion process and incorporate the uncertainty in diffusion evolution, and there still exist two major limits: (**L1**) We argue that effective exploration of this uncertainty can be realized by estimating it from the perspective of underlying distributions. However, learning the underlying distribution for the structure of information diffusion while incorporating uncertainties from historical user retweets poses a challenge for existing generative models (i.e., GAN [16] and VAE [17]). On the one hand, although current generative models can generate the distribution in a sequence-to-sequence manner [18], [19], they still face instability during optimization [16] and posterior collapse issues [17]. On the other hand, no existing methods are able to capture uncertainties during information diffusion process from different perspectives within a unified framework. **(L2)** Previous studies [11], [20] typically assume discrete and equal time intervals for user re-sharing behaviors. However, this approach primarily focuses on the order or position of users within the information diffusion process, thereby restricting their ability to effectively capture and express temporal information. While some works [21] also notice the significance of time span, their models either struggle to capture time differences between past interactions or lack the ability to generalize across various time differences. Hence, the modeling of irregularly-sampled diffusion processes remains under-explored in the field of information cascade popularity prediction.

Present work: To address the mentioned challenges, we present a novel information **Cas**cade popularity prediction model based on probabilistic **D**iffusion model [22] and neural **O**rdinary differential equations [23] (**CasDO**). Specifically, the diffusion model's process of injecting noise into the target and then reconstructing the original distribution through denoising allows the model to better capture uncertainties by predicting noise. We devise a novel temporal neural ODE network (T-ODE) that generalizes discrete state transitions to continuous-time dynamics of the information cascade. It allows us to better match the real information propagation, obeying an ODE between successive observations to possess continuous hidden states. Once a new event occurs, the state will be updated by a gating mechanism, which jointly considers the new input and the temporal interval. In addition, CasDO can capture the uncertainties associated with information cascades by integrating the probabilistic diffusion models and the latent ODEs. In particular, we train CasDO by injecting noises into the stochastic hidden layers with a regularizer encouraging the injection process. Subsequently, the diffusion models are used to reconstruct the structural embeddings of cascades by approximating the scores of the posterior distribution transformed from the prior by conditioning the implicit feature of cascade data. The latent ODE defines a generative process over time based on the deterministic evolution of an initial latent state. Compared with previous probabilistic cascade learning models [11], [24], [25] that simply model the latent distribution of information cascade with a variational autoencoder or normalizing flow, CasDO exploits the reverse diffusion process to learn the noise-perturbed distribution and therefore can model the more complex cascade networks. Besides, our CasDO is capable of denoising the information diffusion stochasticity and, more importantly, simulates the uncertainty of real-world information propagation. Our main contributions can be summarized as follows:

- We present a Temporal ODE-based approach for modeling the irregular-sampled events in information cascades. It captures the continuous-time dynamics in a principled way compared to previous RNN-based counterparts.
- We devise a novel framework entitled CasDO to capture the uncertainties in information cascades, which not only models the probabilities of sharing behavior among nodes but also preserves the uncertainty of information diffusion and cascade growth.
- We conduct extensive experiments on real-world large-scale information cascade datasets and demonstrate that CasDO can improve the prediction performance compared to existing state-of-the-art models while explaining its behavior.

The reminder of this paper is organized as follows. We review related literature in the next section and then introduce necessary background in Section III. In Section IV, we present the details of the proposed CasDO model. The experimental evaluations including performance comparison, ablation study, parameter analysis and model interpretability are presented in Section VI. Finally, we conclude this work and point out our the potential future work in Section VII.

II. Related Work

In this section, we review prior works that are most relevant to our paper. Our task focuses on information cascade popularity prediction, which involves the techniques of diffusion probabilistic models and ordinary differential equation-based recurrent neural networks.

A. Information Cascade Prediction

Modeling information cascades and predicting information diffusion in social and academic networks have been well-studied in recent years [1], [26]. Both macroscopic and microscopic approaches have been proposed for information

cascade learning on practical downstream prediction tasks such as popularity prediction [3], rumor detection [27], and user activation prediction [21].

Existing methods analyze the diffusion of information from a variety of aspects, including the underlying diffusion structure (e.g., the topology of users in a social network), content features of information sources (e.g., texts, topics, and images of posts and news articles), temporal characteristics of time-series (e.g., the arrival time of retweets and citations). Various techniques have been utilized to extract useful cascade features [7], to model the diffusion mechanisms and protocols [28], and to learn expressive cascade representations [24], [29].

Early efforts in information cascade modeling mainly fall into the engineering of cascade features. Researchers have explored various kinds of features to inspect whether they can inform cascade propagation in the future and, if so, to what extent one can predict cascade sizes [30]. For example, the authors in [7] group five representative feature-sets retrieved from cascade observation in the early stage for cascade prediction (e.g., predicting future cascade size and structure), including content features, original poster features, structural features, and temporal features. After experimenting on 150K Facebook photo share cascades, the author found that all feature-sets are indicative, among which, temporal and structural features performed the best, e.g., the time elapsed between the post and reshares, the number of users who saw the post, and the number of friends/subscribers/fans of the poster.

Predictability of information cascades and the problem formulations in different social networks, as well as potential explanations for why certain cascades grow faster, have been discussed in [30]. The authors analyzed two predicting strategies, including a priori prediction and *peaking* prediction, and selected a wide range of features covering structure, time, early resharers, and similarity between users. Experiments conducted on Last.fm, Flickr, and Twitter using logistic regression show the following findings: temporal features not only perform well (beating all other features combined) but also generalize across domains. However, the opinions on the roles of non-temporal features are inconsistent. e.g., structural features in subgraphs can foresee either higher or lower popularity depending on specific social networks [30], while local structures perform well on social networking services but worse on academic networks [11]. In addition, content features were generally considered as poor predictors that are neither generalizable to different platforms nor relevant to the final popularity [1]. Nevertheless, contents of cascades were also reported to be useful for popularity prediction in [31].

In addition to various features crafted from information cascades, researchers also proposed numerous generative models to capture the temporal dynamics of information diffusion. Notably, the family of self-exciting Hawkes processes [32] has been extensively studied in the literature, which describes the diffusion process as an event sequence and the intensity function of the process is conditioned on all past events (a.k.a. the "rich-get-richer" phenomenon). Various kinds of stochastic point processes have been proposed to simulate the information diffusion mechanism, which, more or less, incorporated different endogenous and exogenous factors to model the intensity and decay functions [26]. For example, the reinforced Poisson process was proposed in [4] to track the citation dynamics of a cascade, and the self-exciting point process was studied in [28], [32] for predicting cascade popularity in microblogging networks.

More and more researchers are exploring the potential of deep learning for improving the performance of information cascade prediction. Deep neural networks – which can learn expressive representations from large-scale data – were proven to succeed in learning and predicting in domains such as images, voice, and language processing. Recurrent neural networks (RNNs), graph neural networks (GNNs) and Transformers were utilized to model the temporal and structural characteristics of cascades, respectively. For example, CasCN [13], and VaCas [24] combine the information cascade graph and temporal sequence for prediction; CasFlow [11] models the local cascade graph in the context of global social network; Coupled GNNs [33] was proposed to capture the cascading effect; CCGL [34] proposed to pre-train cascade graphs to improve the generalization capability of prediction; to name a few. Readers are referred to [1], [26] for comprehensive reviews on the recent advances of deep learning-based information cascade prediction.

B. Diffusion Probabilistic Models

Generative models, e.g., normalizing flows (NF) [35], variational autoencoders (VAE) [36], and generative adversarial networks (GAN) [37], are able to generate high-quality images [38], image-to-image translations [39] among many others. Diffusion probabilistic model (DPM) [22] enables both flexible and tractable modeling of complex data by Markov chain that consists of transitions from well-known simple distributions (e.g., Gaussian) to actual data distribution, and recently has shown excellent results on modeling high-dimensional data distributions (e.g., texts and images). DPMs use a forward diffusion process (inspired by non-equilibrium statistical physics [40]) to gradually convert the data distribution into tractable distribution by adding noises. Then, a reverse diffusion process is defined to recover the data in a generative way. Denoising diffusion implicit model (DDIM) [41] was presented to speed up the sampling process of the Markov chain, which generalized DPM by a class of non-Markovian processes without changing the training objective. Researchers also explored various network architecture [42] and stochastic differential equations (SDEs) [43] to improve the quality of generated samples. Recently, DPMs have achieved comparable or even better performance compared to GANs and VAEs in a range of applications such as audio synthesis [44], and time-series imputation [45].

C. Neural Ordinary Differential Equations

Ordinary differential equations (ODEs) are recently connected to deep neural networks [23] by parameterizing the derivative of hidden states with neural networks instead of the discrete sequence of hidden layers used in traditional models such as ResNet and RNNs, and have advantages in trading-off between numerical precision and computation, while significantly saving memory cost.

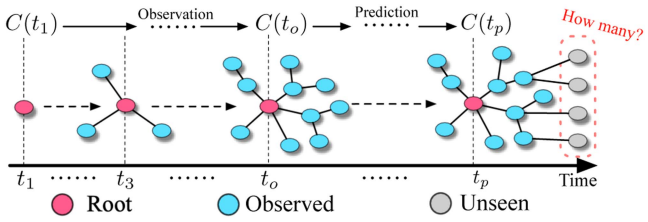


Fig. 1. Evolving cascade graph for popularity prediction.

TABLE I
MATHEMATICAL SYMBOLS

Symbol	Description
$\mathbf{A}$	Adjacency matrix.
$\mathbf{D}$	Diagonal degree matrix.
$\mathbf{E}_c$	Node embeddings in cascade graph $\mathcal{G}_c$.
$\mathcal{E}_c$	Edges in cascade graph $\mathcal{G}_c$.
$\mathcal{V}_c$	Nodes in cascade graph.
$\mathcal{G}_c$	Information cascade graph.
$\mathcal{G}_g$	Global graph.
$C(t)$	Information cascade observed at time t.
$P(t_p)$	Popularity at prediction time t_p.
Φ	spectral graph wavelets network.
$\mathbf{Z}$	the output of T-ODE.
r	r-th round of the forward/reverse process in diffusion models.
$\boldsymbol{\epsilon}$	Gaussian noise.
$\boldsymbol{\epsilon}_\theta$	The network predicting the $\boldsymbol{\epsilon}$.
d	Dimension of the hidden states in RNNs.

Subsequent works designed new variants of neural ODEs from different perspectives and applied them on real-world applications such as time series data modeling [15], and generative models [46]. For example, in [15] the authors proposed to model irregular-sampled time-series data by generalizing the RNNs to have continuous-time states.

III. PRELIMINARIES

In this section, we provide the necessary background and formally define the studied problem. Information cascades can be seen as a sequence of actions that disseminate the information (e.g., a tweet in microblogging network or a paper in academic network) to a large body of audience. Take Twitter platform as an illustrative example (throughout the paper), the tweets posted by users can be seen as information items, which can be retweeted (reshared) by other users. We consider the retweeting action as the diffusion process and the retweet sequence as an information cascade. Following previous works [3], [13], [24], [29], in this paper, we formulate the information cascade as a diffusion tree, as shown in Fig. 1. Our goal is to predict the popularity of an information cascade in the future (*Definition 3.3*) by observing its early evolution (*Definition 3.1*) and the underlying diffusion network (*Definition 3.2*). Table I summarizes the frequently used notations used in this paper.

Definition 1 (Information Cascade Graph): Given an information item and all its diffusion, an information cascade graph is defined as $\mathcal{G}_c = \{\mathcal{V}_c, \mathcal{E}_c\}$, where $\mathcal{V}_c = \{u_1, u_2, u_3, \dots\}$ is the set of users participated in propagating the information, and $\mathcal{E} = \{e_1, e_2, e_3, \dots\}$ is the set of edges between users – each edge $e_{i,j} = (u_i, u_j, t_k)$ denotes that user u_i forwarding user u_j at time t_k.

Definition 2 (Global Graph): The global graph $\mathcal{G}_g = \{\mathcal{V}_g, \mathcal{E}_g\}$ represents a social/citation network. The nodes are users/authors and the edges are relationships between them, e.g., follower/followee relationship.

Definition 3 (Information Cascade Popularity Prediction): Given an information cascade graph snapshot $\mathcal{G}_c(t_o)$ observed at time t_o, and the underlying global graph $\mathcal{G}_g$, we aim to train a model that can predict the future popularity $P(t_p) = |\mathcal{V}_c(t_p)|$ (the number of retweets/citations) at a prediction time t_p ($t_p \gg t_o$).

IV. METHODOLOGY

We now present the architecture of our proposed CasDO model which not only captures the structural and temporal characteristics of information cascades but also models the irregularity of time-series and the diffusion uncertainty in continuous-time – discussed in detail in respective subsections. As shown in Fig. 2, CasDO consists of four components:

1) *Hierarchical structure learning:* To learn both local and global graph structures efficiently and effectively, we use spectral graph wavelets from signal processing [47] to produce the structural-equivalent embeddings of nodes in the cascade graph. Besides, we employ the sparse matrix factorization [48] to obtain users' global embeddings that implicitly capture the latent relationships between users, e.g., user interests and node proximity.
2) *Irregular temporal diffusion modeling:* The retweeting behavior of a tweet can occur at any time after the posting time and need not follow a strict periodicity. This, in turn, leads to irregular time series in information cascades. Traditional time-series modeling methods, such as RNN-based ones, are not a good fit for the irregularly-sampled time-series data [15]. We design a module named T-ODE that integrates neural ordinary differential equations (ODEs) with RNNs to build continuous-time hidden dynamics to handle the arbitrary time gaps between each retweet and the temporal dependencies between retweets in a cascade are simultaneously modeled.
3) *Diffusion uncertainty modeling:* Deterministic models cannot capture the inherent uncertainties in information diffusion, which may cause performance degradation. CasDO incorporates a probabilistic temporal model, which considers the uncertainty from cascade graph structures and the cascade growth through diffusion models and ODE-based variational inference framework [23] to enhance its robustness. The whole model is trained with noises injected in the stochastic hidden layers, with a regularizer encouraging this injection process.
4) *Prediction network:* With the hierarchical cascade structure embeddings, we feed them into T-ODE for irregular time-series modeling and use temporal-structural diffusion probabilistic models for uncertainty handling. Afterward, combined with irregular time-series modeling and structure-evolution uncertainty modeling, the learned information cascade representations are fed into

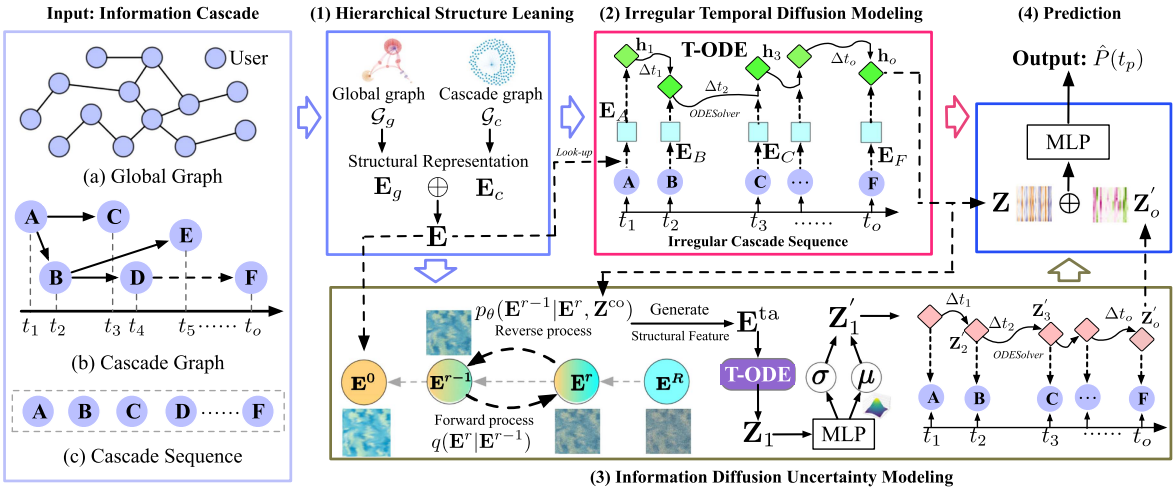


Fig. 2. Overall framework of CasDO. (1) Local and global structure feature extraction; (2) Irregular time-series learning by temporal ODE with continuous-time hidden dynamics; (3) Evolution uncertainty learning by diffusion probabilistic models with noise injection; and (4) Final popularity prediction network.

multi-layer perceptrons (MLPs) to predict the final popularity of cascades.

A. Hierarchical Structure Learning

For a given information cascade graph and its underlying global social network, we first learn their structural embeddings that can better represent the roles of users in the diffusion process. These embeddings are later used for modeling irregular temporal diffusion and uncertainties. We note that many structural learning methods [49], [50], [51], [52], [53] can be used here to extract meaningful representations of information cascade graphs and global graph, e.g., traditional methods like DeepWalk, LINE, and node2vec; GNN-based models such as GCN, GAT, and GraphSage; as well as heterogeneous and dynamical methods [54], e.g., EvolveGCN, HetGNN, and metapath2vec.

Following previous works [11], [24], we adopt structural equivalent local embeddings for cascade graph learning, and structure proximity embeddings for large-scale global graph. For information cascade graph, we use techniques from graph signal processing [55] and graph Laplacian to calculate the spectral graph wavelets with heat kernel function on the spectrum. For global graph with millions of users, generating node embeddings is challenging due to the high computational overhead. Thus, we use sparse matrix factorization [48] to learn the structural properties of nodes in global graph.

1) Learning Structural-Equivalent Local Embeddings for Cascade Graphs: Given an information cascade graph $\mathcal{G}_c(t_o)$ observed at time t_o, we have its weighted adjacency matrix $\mathbf{A}_c$ and diagonal degree matrix $\mathbf{D}_c$. Then an unnormalized graph Laplacian $\mathbf{L}_c = \mathbf{D}_c - \mathbf{A}_c = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ can be used to calculate the spectral graph wavelets Φ with heat kernel function on the spectrum [55]. Each column vector of $\Phi \in \mathbb{R}^{|\mathcal{V}_c|\times|\mathcal{V}_c|}$ is the wavelets for a node in cascade graph. In order to solve the graph mapping problem (i.e., solve the "isomorphism" problem between two nodes' neighbors), the wavelet coefficients are processed as a probability distribution. Then the empirical characteristic functions are utilized to obtain the final structural-equivalent node embeddings $\mathbf{E}_c$. The Chebyshev polynomials are used to calculate the wavelet coefficients. The overall complexity for cascade graph structure learning is linear in the number of edges in the graph.

2) Learning Structure-Proximities for Nodes in Large-Scale Global Graph: The nodes in a global graph possess certain properties distinct from the nodes in a cascade graph. Consider a global graph in which there are millions of nodes. On one hand, embedding such a large graph is highly challenging due to the intensive computational overhead (or, even impossible) – e.g., the factorization-based model requires $O(|\mathcal{V}_g|^3)$ time complexity. On the other hand, in a global graph, we want to learn different information from the graph structure, e.g., close nodes (in terms of distance) should have similar structure embeddings. Thus, we use sparse matrix factorization to learn the structural properties of the global graph, which is efficient and scalable to graphs with millions of nodes and edges.

Specifically, given a global graph $\mathcal{G}_g = \{\mathcal{V}_g, \mathcal{E}_g\}$ (e.g., a Twitter follower/followee graph or an academic collaboration graph), $\mathbf{A}_g$ and $\mathbf{D}_g$ are the weighted adjacency matrix and diagonal degree matrix, respectively. A sparse randomized truncated singular value decomposition is used to approximate the factorization of the proximity matrix. Considering the long-tailed distribution and sparsity of nodes in social networks, the expensive computation for a large-sized matrix becomes feasible and efficient, with tolerable information loss. Now we have the node embeddings $\mathbf{E}_g$ in the global graph for later training. Learning both local and global structures of information cascades is proved to be effective across different domains [1], and we also adopt this setting from [11] in CasDO. For each user u_i in cascade graph, we concatenate its global embedding $\mathbf{E}_{g,u_i}$ with cascade embedding $\mathbf{E}_{c,u_i}$, denoted as $\mathbf{E}_{u_i}$. Then we feed the cascade sequence of user embeddings $\{\mathbf{E}_{u_i}|u_i \in \mathcal{V}_c\}$ to the T-ODE for irregular time-series modeling, and feed the user embedding matrix $\mathbf{E} = [\mathbf{E}_{u_1}, \mathbf{E}_{u_2} \ldots]^T \in \mathbb{R}^{|\mathcal{V}_c|\times|\mathbf{E}_u|}$ to the diffusion probabilistic model for the uncertainty modeling from structure to evolution.

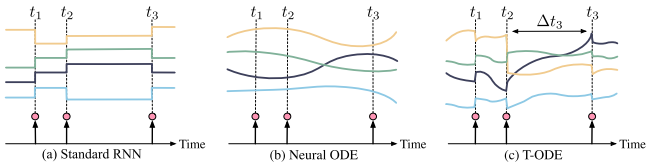


Fig. 3. Evolution of the hidden state in a time series. Colored horizontal lines denote different dimensions of the hidden state. (a) The hidden state of standard RNNs only updates at new observations. (b) In Neural ODEs [23], the state obeys an ODE all the time but is only determined by the initial state. (c) T-ODE has continuous states which obey an ODE between successive points and can be updated at any observation.

B. Irregular Temporal Diffusion Modeling

Due to the randomness of human reaction behavior, retweeting actions can occur at any time, which leads to irregular time series in information cascades. The time intervals between real-world cascade events can be tens of seconds or even several days. As depicted in Fig. 3, using standard RNNs in this setting is ineffective because that RNNs can only update their states upon the occurrence of a new point, which, are more suitable for learning regularly-sampled time-series data (e.g., electricity forecasting). To better match real cascade events, a new method is needed to inherently consider the underlying continuous-time dynamics of information cascades. Inspired by neural ordinary differential equations (ODEs) [23], we propose a module named as temporal-ODE, which generalizes state transitions in RNNs to continuous-time dynamics defined by ODEs while considering such temporal interval information among cascade events by a gating mechanism.

First, following [56], we employ an LSTM cell before performing the ODE solver to avoid the vanishing or exploding of gradients. Subsequently, we use a numerical ODE solver – the Euler method [23] in our implementation – to evaluate the hidden states between successive observations and update the hidden states using a GRU cell at each observation. Besides, we propose a temporal gating mechanism T-Gate to merge the latent representation from the first two steps to learn the continuous-time dynamics. We use the gating mechanism T-Gate to handle the irregular sampling problem caused by the time-varying from seconds to hours between successive cascade events that make the ODE solver challenging to evaluate continuous hidden dynamics.

In Fig. 2, we depict the detailed structure of T-ODE in one-step. Within the previous hidden state pair $(\mathbf{c}_{i-1}, \mathbf{h}_{i-1})$ at time t_{i-1} and user embedding $\mathbf{E}_{u_i}$, we first feed them into the LSTM cell to generate a new hidden state $(\mathbf{c}_i, \mathbf{h}'_i)$:

$$(\mathbf{c}_i, \mathbf{h}'_i) = \text{LSTMCell}\left(\theta_l, (\mathbf{c}_{i-1}, \mathbf{h}_{i-1}), \mathbf{E}_{u_i}\right), \tag{1}$$

where $\mathbf{c}_{i-1}$ is the memory cell and $\mathbf{h}_{i-1}$ is the output state. θ_l denote the learnable parameters in LSTM Cell.

Then, we feed $\mathbf{h}_{i-1}$ into the ODE solver based on the Euler method to obtain the ODE hidden state $\mathbf{z}_i$ at each step. This operation is to construct the continuous-time dynamics of the hidden states between irregular time intervals in all consecutive (t_{i-1}, t_i) pairs. To construct the true user u_i's representation at time t_i, we take two states of LSTM cell – output state $\mathbf{h}'_i$ and hidden state $\mathbf{z}_i$ – as the input for latent representation learning of u_i, which outputs $\mathbf{h}''_i$. The above process is summarized as follow:

$$\mathbf{z}_i = \text{ODESolver}\left(f_\omega, \mathbf{h}_{i-1}, \mathbf{h}'_i, (t_{i-1}, t_i)\right), \tag{2}$$

$$\mathbf{h}''_i = \text{GRUCell}\left(\theta_g, \mathbf{h}'_i, \mathbf{z}_i\right), \tag{3}$$

where $\mathbf{z}_i \in \mathbb{R}^d$ is the solution at t_i to an ODE started from time t_{i-1}; $\mathbf{h}''_i$ is the updated hidden state; θ_g denotes the learnable parameters in GRU Cell. The neural ODEs [23] consider the parameter updating in neural networks as the process of solving ODEs and the discrete layers of neural networks (e.g., the hidden states of RNNs) can be regarded as an Euler discretization of a differential equation from the perspective of numerical methods:

$$\frac{d\mathbf{h}(t)}{dt} = f_\omega(\mathbf{h}(t), t), \quad \text{where} \quad \mathbf{h}(t) = \mathbf{h}_t, \tag{4}$$

$$\mathbf{h}\left(t_2\right) = \mathbf{h}\left(t_1\right) + \int_{t_1}^{t_2} f_\omega(\mathbf{h}(t), t)dt, \tag{5}$$

where neural network is parameterized by f_ω specifying the continuous dynamics of the hidden states. The parameter update process of neural ODE blocks can be regarded as Solving ODEs with numerical methods such as Euler, Runge-Kutta, and the adjoint method. Given the latent states $\mathbf{h}'_i$ and $\mathbf{h}''_i$, we update $\mathbf{h}_i$ using the gating mechanism:

$$\mathbf{h}_i = \boldsymbol{\nu}_i \odot \mathbf{h}''_i + (\mathbf{1} - \boldsymbol{\nu}_i) \odot \mathbf{h}'_i, \tag{6}$$

where the temporal gate $\boldsymbol{\nu}_i = e^{-(\Delta t_i)} \in \mathbb{R}^d$ helps the model determine how much of the state is solved by ODE that needs to be passed to the future. Finally, we compute the output states $\{\mathbf{o}_1 \ldots \mathbf{o}_n\}$ via a fully-connected layer for downstream tasks, where $\mathbf{o} \in \mathbb{R}^d$ and n denotes the number of users $|\mathcal{V}_c|$ in the early evolution. Generally, we use the final output state $\mathbf{o}_n$ as the cascade latent representation $\mathbf{Z}$. The overall training process of T-ODE is formalized in Algorithm 1.

Algorithm 1: Training of the T-ODE Model.

Input: $n = |\mathcal{V}_c|$, LSTMCell weight θ_l, GRUCell weight θ_g, neural ODE blocks weight f_ω, output weight and bias $\mathbf{W_o}, \mathbf{b_o}$, $\{\mathbf{h}_{i-1}, t_i, t_{i-1}, \Delta t_i\}_{i=1\cdots n}$;
Output: The output state $\{\mathbf{o}_{i=1\cdots n}\}$;
$\mathbf{h}_0 = \mathbf{0}$; ▷ Initialize hidden state.
$\mathbf{c}_0 = 0$; ▷ Initialize memory cell.
for all i in $\{1 \ldots n\}$ **do**
  Calculate cascade hidden state $(\mathbf{c}_i, \mathbf{h}'_i) = \text{LSTMCell}(\theta_l, (\mathbf{c}_{i-1}, \mathbf{h}_{i-1}), \mathbf{E}_{u_i})$;
  Obtain $\mathbf{z}_i = \text{ODESolver}(f_\omega, \mathbf{h}_{i-1}, \mathbf{h}'_i, (t_{i-1}, t_i))$;
  Calculate continuous-time dynamics of the hidden states $\mathbf{h}''_i = \text{GRUCell}(\theta_g, \mathbf{h}'_i, \mathbf{z}_i)$;
  Update cascade hidden state $\mathbf{h}_i = \text{T-Gate}(\mathbf{h}'_i, \mathbf{h}''_i, \Delta t_i)$;
  Obtain output state $\mathbf{o}_i = \mathbf{h}_i\mathbf{W_o} + \mathbf{b_o}$
end for

C. Diffusion Uncertainty Modeling

The uncertainty estimation on the prediction of information cascades is important for information diffusion learning in large-scale social networks. Considering and quantifying the uncertainty-level dynamics of information items exposed to potential adopters in the neighborhood are desired in real-world applications. For example, in [7] the authors showed that the popularity of information cascades is unpredictable to some extent due to the uncertainty during diffusion. In recent years, researchers made initial efforts to exploit uncertainty in the information cascade. The self-exciting Hawkes point process is used in [57] to model the uncertainty of information cascade popularity prediction and improve the generalization performance. In [11], [24], [25], variational inference and normalizing flows are used to model the diffusion uncertainty within the rich family of posterior distributions.

The information diffusion uncertainty exists in not only the evolution of the cascade (temporal dependence of cascaded graphs) but also the spatial correlations among users (spatial structure of cascaded graphs). In CasDO, we integrate the probabilistic diffusion model and latent ODEs to model the information diffusion uncertainty from the perspective of the spatio-temporal latent variables. We propose conditional score-based diffusion models, which is used to reconstruct the structural embeddings of cascades by approximating the scores of the posterior distribution obtained from the prior by conditioning the implicit features of cascade data. The conditional probabilistic diffusion models are explicitly trained for structural generation and can exploit useful correlations between observed structural features. The latent ODE defines a generative process over time based on the deterministic evolution of an initial latent state.

First, we exploit the diffusion model [22], [58] to capture the uncertainties of the structural embeddings in a generative probabilistic way. The diffusion models consist of a *forward* process and a *reverse* process. The forward process transforms a complex data distribution into a simple distribution (e.g., Gaussian). The reverse process then transforms the simple noisy distribution into the target (cascade representation) distribution through a diffusion chain. In order to exploit useful information in structural embeddings and better generate the samples meeting the target distribution, we design a conditional score-based diffusion model that aims to learn the conditional distribution of cascade structural embeddings.

The conditional score-based diffusion model generates target cascade structural embeddings $\mathbf{E}_0^{\text{ta}}$ by exploiting conditional observations $\mathbf{Z}^{\text{co}}$ which represent the cascade implicit features $\mathbf{Z}$. The goal of the probabilistic generation is to estimate the true conditional data distribution $q(\mathbf{E}_0^{\text{ta}}|\mathbf{Z}^{\text{co}})$ with a model distribution $p_\theta(\mathbf{E}_0^{\text{ta}}|\mathbf{Z}^{\text{co}})$. Given an information cascade graph $\mathcal{G}_c$, a global graph $\mathcal{G}_g$ and its structural embedding $\mathbf{E}$ learned in Section IV-A, we denote the start data distribution as $q(\mathbf{E}_0)$ and let $\mathbf{E}_0 = \mathbf{E}$. The forward process gradually converts the prior $q(\mathbf{E}_0)$ into a tractable distribution by adding noise to the data. Each time-step $r \in \{0, 1, 2, \ldots, R\}$ of the forward process is defined as a Gaussian transition:

$$q\left(\mathbf{E}_{1:R}|\mathbf{E}_0\right) := \prod_{r=1}^{R} q\left(\mathbf{E}_r|\mathbf{E}_{r-1}\right), \tag{7}$$

$$q\left(\mathbf{E}_r|\mathbf{E}_{r-1}\right) := \mathcal{N}\left(\mathbf{E}_r; \sqrt{1-\beta_r}\mathbf{E}_{r-1}, \beta_r\mathbf{I}\right), \tag{8}$$

where β_r is fixed as a constant or scheduled as variances learned by reparameterization [36] to control the procedure of adding Gaussian noise to the data. We focus on the conditional diffusion model with reverse process and aim to model the conditional distribution $p(\mathbf{E}_{r-1}^{\text{ta}}|\mathbf{E}_r^{\text{ta}}, \mathbf{Z}^{\text{co}})$ without approximations. Specifically, we define a conditional denoising function $\boldsymbol{\epsilon}_\theta : (\mathbf{E}^{\text{ta}}|\mathbf{Z}^{\text{co}}) \to \mathbf{E}^{\text{ta}}$, which takes the cascade temporal latent representation $\mathbf{Z}^{\text{co}}$ learned from T-ODE running forward as inputs:

$$p_\theta\left(\mathbf{E}_{0:R}^{\text{ta}}|\mathbf{Z}^{\text{co}}\right) := p\left(\mathbf{E}_R^{\text{ta}}\right) \prod_{r=1}^{R} p_\theta\left(\mathbf{E}_{r-1}^{\text{ta}}|\mathbf{E}_r^{\text{ta}}, \mathbf{Z}^{\text{co}}\right), \tag{9}$$

$$p_\theta\left(\mathbf{E}_{r-1}^{\text{ta}}|\mathbf{E}_r^{\text{ta}}, \mathbf{Z}^{\text{co}}\right) := \mathcal{N}(\mathbf{E}_{r-1}^{\text{ta}}; \boldsymbol{\mu}_\theta\left(\mathbf{E}_r^{\text{ta}}, r|\mathbf{Z}^{\text{co}}\right), \Sigma_\theta\left(\mathbf{E}_r^{\text{ta}}, r|\mathbf{Z}^{\text{co}}\right). \tag{10}$$

When sampling $\mathbf{E}_r$ at round r, we let $\alpha_r := 1 - \beta_r$ and $\bar{\alpha}_r := \prod_{k=1}^{r} \alpha_k$, and the distribution $q(\mathbf{E}_r|\mathbf{E}_0)$ can be expressed as

$$q\left(\mathbf{E}_r|\mathbf{E}_0\right) = \mathcal{N}\left(\mathbf{E}_r; \sqrt{\bar{\alpha}_r}\mathbf{E}_0, (1-\bar{\alpha}_r)\mathbf{I}\right). \tag{11}$$

The goal of $p_\theta(\mathbf{E}_{r-1}^{\text{ta}}|\mathbf{E}_r^{\text{ta}}, \mathbf{Z}^{\text{co}})$ is to eliminate the Gaussian noise added in the diffusion process. The parameters θ are learned to fit the data distribution $q(\mathbf{E}_0)$ by minimizing the negative log likelihood via a variatioanal bound:

$$\begin{aligned}\mathbb{E}\left[\log p_\theta\left(\mathbf{E}_0^{\text{ta}}\right)\right] &\geq \mathbb{E}_q\left[\log \frac{p_\theta(\mathbf{E}_{0:R}^{\text{ta}}|\mathbf{Z}^{\text{co}})}{q(\mathbf{E}_{1:R}|\mathbf{E}_0)}\right] \\ &= \mathbb{E}_q\left[\log p_\theta(\mathbf{E}_R^{\text{ta}}) + \sum_{r\geq 1} \log \frac{p_\theta(\mathbf{E}_{r-1}^{\text{ta}}|\mathbf{E}_r^{\text{ta}}, \mathbf{Z}^{\text{co}})}{q(\mathbf{E}_r|\mathbf{E}_{r-1})}\right] =: \mathcal{L},\end{aligned} \tag{12}$$

which can be simplified and efficiently trained with stochastic gradient descent [58] by rewriting (12) as

$$\mathcal{L}(\theta) := \mathbb{E}_{r,\mathbf{E}_0,\boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_r}\mathbf{E}_0 + \sqrt{1-\bar{\alpha}_r}\boldsymbol{\epsilon}, r|\mathbf{Z}^{\text{co}}\right)\right\|^2\right], \tag{13}$$

where noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\boldsymbol{\epsilon}_\theta$ is a network predicting the $\boldsymbol{\epsilon}$. An illustration of the network $\boldsymbol{\epsilon}_\theta$ is shown in Fig. 4. The sampling process is defined as:

$$\mathbf{E}_{r-1}^{\text{ta}} = \frac{1}{\sqrt{\alpha_r}}\left(\mathbf{E}_r^{\text{ta}} - \frac{\beta_r}{\sqrt{1-\bar{\alpha}_r}}\boldsymbol{\epsilon}_\theta\left(\mathbf{E}_r^{\text{ta}}, r|\mathbf{Z}^{\text{co}}\right)\right) + \sigma_r\bar{\mathbf{z}}, \tag{14}$$

where $\mathbf{E}_R^{\text{ta}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\sigma_r^2 = \tilde{\beta} = \frac{1-\bar{\alpha}_{r-1}}{1-\bar{\alpha}_r}\beta_r$, and $\bar{\mathbf{z}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $r > 1$; otherwise $\bar{\mathbf{z}} = \mathbf{0}$.

After training, we wish to exploit more useful structural information between the cascade graph and the global graph and better model the spatial structural uncertainty. As in training, we do not incorporate the loss function of the conditional

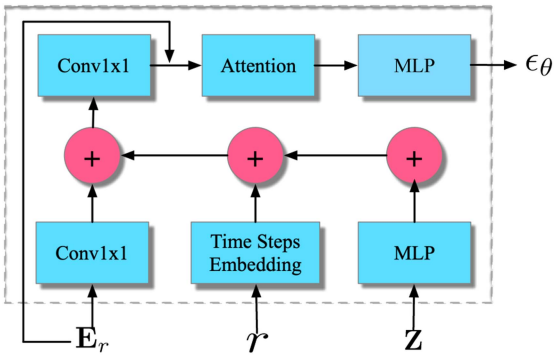


Fig. 4. Implementation of network ϵ_θ in diffusion models.

Algorithm 2: Training of the Diffusion Model.

Input: distribution of training data $q(\mathbf{E}_0)$, the number of iteration r, the sequence of noise levels β_r, cascade temporal representation $\mathbf{Z}^{\text{co}}$ and the number of iterations N_{iter}.
Output: Trained denoising function ϵ_θ.
1: **for all** $i = 1$ **to** N_{iter} **do**
2: Initialize time step $r \sim \text{Uniform}(\{1, \ldots, R\})$;
3: $\mathbf{E}_0 \sim q(\mathbf{E}_0)$;
4: Initialize noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
5: Calculate noisy targets $\mathbf{E}_r = \sqrt{\alpha_t}\mathbf{E}_0 + (1 - \alpha_t)\boldsymbol{\epsilon}$;
6: Compute gradient $\bigtriangledown_\theta ||(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{E}_r, r\,|\mathbf{Z}^{\text{co}}))||_2^2$ via (13);
7: **end for**

Algorithm 3: Sampling in Probabilistic Diffusion.

Input: noisy $\mathbf{E}_R^{\text{ta}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, the number of iteration r, the sequence of noise levels β_r, trained denoising function $\boldsymbol{\epsilon}_\theta$ and state $\mathbf{Z}^{\text{co}}$.
Output: Sampling $\mathbf{E}_0^{\text{ta}}$;
1: Initialize noisy targets $\mathbf{E}_R^{\text{ta}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
2: **for all** $r = R$ **to** 1 **do**
3: **if** $r > 0$ **then**
4: Initialize noise $\bar{\mathbf{z}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
5: **else**
6: Initialize noise $\bar{\mathbf{z}} = 0$;
7: **end if**
8: Sampling $\mathbf{E}_{r-1}^{\text{ta}} = \frac{1}{\sqrt{\alpha_r}}(\mathbf{E}_r^{\text{ta}} - \frac{\beta_r}{\sqrt{1-\bar{\alpha}_r}}\boldsymbol{\epsilon}_\theta(\mathbf{E}_{(r)}^{\text{ta}}, r|\mathbf{Z}^{\text{co}})) + \sigma_r\bar{\mathbf{z}}$ via (14);
9: **end for**

score-based diffusion model into the final objective of CasDO but optimize the diffusion model to capture the implicit feature of the user embedding $\mathbf{E}$. As in sampling procedure, the process of sampling trajectories from the "warm-up" state $\mathbf{Z}^{\text{co}}$ can be repeated R times to obtain empirical quantiles of the spatial structure uncertainty. By sampling procedure, we obtain a sample $\mathbf{E}^{\text{ta}}$ used for modeling evolution uncertainty introduced below.

Next, we design a latent ODE network based on the VAE framework, which copes with randomness and uncertainty and leverages them to model the latent variable's evolutionary uncertainty. We use the T-ODE and the diffusion models as the encoder and then utilize the ODE's extrapolation to simulate the complete evolution dynamics with uncertainty. In addition to user embedding $\mathbf{E}^{\text{ta}}$ generated from the conditional diffusion models, we also use the backward T-ODE to obtain the initial cascade state $\mathbf{Z}_1^{\text{ta}}$. Let $p_\vartheta = (\mathbf{Z}_1^{\text{ta}}|\mathbf{E}^{\text{ta}})$ be the true posterior distribution. We follow VAEs to approximate $p_\vartheta = (\mathbf{Z}_1^{\text{ta}}|\mathbf{E}^{\text{ta}})$ with $q_\phi = (\mathbf{Z}_1^{\text{ta}}|\mathbf{E}^{\text{ta}})$ using a neural network, where ϕ is the parameter set of q. We first derive the mean μ and variance σ of the initial cascade state $\mathbf{Z}_1^{\text{ta}}$ from the output of T-ODE($\mathbf{E}^{\text{ta}}$) using linear transformations. According to the reparameterization trick [36], we have $\mathbf{Z}_1' = \mu_{\mathbf{Z}_1^{\text{ta}}} + \sigma_{\mathbf{Z}_1^{\text{ta}}} * \varsigma$ where ς are samples from standard Gaussian. This process is formalized by:

$$\mu_{\mathbf{Z}_1^{\text{ta}}}, \sigma_{\mathbf{Z}_1^{\text{ta}}} = g\left(\text{T-ODE}\left(\mathbf{E}^{\text{ta}}\right)\right), \quad \mathbf{Z}_1' = \mu_{\mathbf{Z}_1^{\text{ta}}} + \sigma_{\mathbf{Z}_1^{\text{ta}}} * \varsigma, \tag{15}$$

where g is a neural network translating the final hidden state of the T-ODE encoder into the mean and variance of $\mathbf{Z}_1^{\text{ta}}$. To get the approximate posterior at time t_1, we run T-ODE encoder backwards-in-time from t_n to t_1.

Moreover, we use the ODE solver to evolve it in the probabilistic space, which can be formulated by:

$$\mathbf{Z}_o' = \mathbf{Z}_1' + \int_{t_1}^{t_n} f_\xi\left(\mathbf{Z}_t', t\right) dt, \tag{16}$$

where $f_\xi(\cdot)$ is the ODE block that estimates the derivative of $\mathbf{Z}(t)$, i.e., $d\mathbf{Z}_t/dt$. In this way, we produce a continuous evolutionary trajectory where each point denotes the latent variables following the posterior distribution $p(\mathbf{Z}_t'|\mathbf{Z}_1', \ldots, \mathbf{Z}_{t-1}')$. Besides, we can extrapolate $\mathbf{Z}_o'$ while explicitly considering the irregular interval evolution uncertainty in the information diffusion process.

Finally, we train the encoder and decoder in the latent ODE network jointly by maximizing the evidence lower bound (ELBO):

$$\begin{aligned}
&\log p_\vartheta\left(\mathbf{E}^{\text{ta}}\right) = \log \int p_\vartheta\left(\mathbf{E}^{\text{ta}} \mid \mathbf{Z}_1^{\text{ta}}\right) p\left(\mathbf{Z}_1^{\text{ta}}\right) \mathrm{d}\mathbf{Z}_1^{\text{ta}} = \\
&\mathbb{E}_{q_\phi(\mathbf{Z}_1^{\text{ta}}|\mathbf{E}^{\text{ta}})} \log\left[\frac{p_\vartheta\left(\mathbf{E}^{\text{ta}}, \mathbf{Z}_1^{\text{ta}}\right)}{q_\phi\left(\mathbf{Z}_1^{\text{ta}} \mid \mathbf{E}^{\text{ta}}\right)}\right] \\
&+ \mathbb{D}_{\text{KL}}\left(q_\phi\left(\mathbf{Z}_1^{\text{ta}} \mid \mathbf{E}^{\text{ta}}\right) \| p_\theta\left(\mathbf{Z}_1^{\text{ta}} \mid \mathbf{E}^{\text{ta}}\right)\right) \\
&\geq \mathbb{E}_{q_\phi(\mathbf{Z}_1^{\text{ta}}|\mathbf{E}^{\text{ta}})}\left[\log p_\theta\left(\mathbf{E}^{\text{ta}}, \mathbf{Z}_1^{\text{ta}}\right) - \log q_\phi\left(\mathbf{Z}_1^{\text{ta}} \mid \mathbf{E}^{\text{ta}}\right)\right] \\
&\triangleq \text{ELBO}\left(\mathbf{E}^{\text{ta}}\right).
\end{aligned} \tag{17}$$

The training and sampling processes of the diffusion model are provided in Algorithms 2 and 3.

D. Prediction

The last component of CasDO is multi-layer perceptrons (MLPs) with one final output unit. We concatenate $\mathbf{Z}$ from the T-ODE (based on the original user embedding $\mathbf{E}$) and $\mathbf{Z}_o'$ from the latent ODEs (based on the generated user embedding $\mathbf{E}^{\text{ta}}$),

and then feed them into MLP to predict cascades' popularity:

$$\hat{P}\left(t_p\right) = \mathrm{MLPs}\left(\mathrm{Concat}\left(\mathbf{Z}, \mathbf{Z}_o'\right)\right). \tag{18}$$

During training, we use the mean square logarithmic error (MSLE) as the objective and combine the MSLE and ELBO to train CasDO. The final loss function is defined as:

$$\mathcal{L} = \frac{1}{N}\sum_{k=1}^{N}\left(\log \hat{P}_k\left(t_p\right) - \log P_k\left(t_p\right)\right)^2 - \lambda\,\mathrm{ELBO}\left(\mathbf{E}_k^{\mathrm{ta}}\right), \tag{19}$$

where N is the total number of cascades, $P_k(t_p)$ is the ground truth, i.e., the number of user who retweets the cascade, $\hat{P}_k(t_p)$ is the predicted popularity for cascade C_k, and $\mathrm{ELBO}(\mathbf{E}_k^{\mathrm{ta}})$ is the ELBO that needs to be maximized given by (17).

V. Model Complexity

Since online social networks contain millions of nodes and edges, it is important to model structure and time dependencies among information cascade graphs for popularity prediction. CasDO models both and, compared to conventional graph cascade models especially those GNN-based models, uses graph wavelets and sparse matrix factorization to mitigate the time complexities for efficiently modeling node embedding from the cascade graph and global graph.

- *Complexity of computing embeddings of nodes from cascade graph and global graph:* the spectral graph wavelets are computed by Chebyshev polynomials, the time complexity is $O(h|\mathcal{E}_c|)$ by computing embeddings of nodes from cascade graph. Since the computation of truncated singular value decomposition is $O(d_g^2|\mathcal{V}_g|)$, the overall complexity of sparse matrix factorization is $O(d_g^2|\mathcal{V}_g| + |\mathcal{E}_g|)$. Here d_g is the dimension of nodes in global graph.
- *Complexity of diffusion models:* in the forward process, the diffusion models utilize reparameterization tricks to calculate noisy target $\mathbf{E}^{(r)}$, which only requires linear time complexity. During training of diffusion models, the denoising function ϵ_θ is only related to $\mathbf{E}^{(r)}$, the dimensions of $\mathbf{Z}$, and the time steps R. Therefore, the time complexity is $O(R \cdot d \cdot \mathbf{E}^{(r)} + R \cdot d \cdot |\mathbf{Z}|)$ in the whole forward process. Note that the time complexity of sampling process is the same as the forward process.
- *Complexity of computing latent feature for irregularly sampled data:* one computational difficulty that arises from irregularly sampled data $\mathcal{V}_c$ is that observation times can be different for each time-series in a particular mini-batch. To solve all ODEs in a mini-batch in-sync, we should output the solution of the combined ODE at the union of all time points in the batch. Taking the union of time points does not substantially affect the running time of the ODE solver, as the adaptive time stepping in ODE solvers is not sensitive to the number of time points $(t_1 \ldots t_n)$.
  Instead, it depends on the length of the time interval $[t_1, t_n]$ and the complexity of the dynamics. Thus, our T-ODE has a similar asymptotic time complexity to standard RNN models, which is determined by the input dimensions of latent variables.

TABLE II
Statistics of the Datasets

Dataset	Twitter	Weibo	APS
# Cascade	88,440	119,313	207,685
# Nodes in $\mathcal{G}_g$	490,474	6,738,040	616,316
# Edges in $\mathcal{G}_g$	1,903,230	15,249,636	3,304,400
Average popularity	124	240	51
Average sequence length	2.196	2.237	3.999
Number of cascades in two different observation settings			
Train (1d/ 0.5h/ 3y)	9,639	21,463	18,511
Val (1d/ 0.5h/ 3y)	2,066	4,599	3,967
Test (1d/ 0.5h/ 3y)	2,065	4,599	3,967
Train (2d/ 1h/ 5y)	12,739	29,908	32,102
Val (2d/ 1h/ 5y)	2,730	6,409	6,879
Test (2d/ 1h/ 5y)	2,730	6,409	6,879

VI. Experiments

We now describe datasets and baselines, and report the results of experimental evaluation, including prediction intepretations and ablation studies. For reproducibility, the code is available at https://github.com/CZ-TAO12/CasDO.

A. Experimental Settings

1) Datasets: We select three public large-scale information cascade datasets. They include two social networks (Twitter and Weibo) and one academic citation network (APS). The first scenario is used for predicting the number of retweets of posts in social networks, and the second one to forecast the citation count of academic papers. Both types of datasets provide real-world information propagation for comparing the performance between CasDO and the baselines. In addition, using two scenarios could verify the generalizability of the proposed model and avoid the risk of limiting the model to a specific type of application. We randomly split each dataset into a training set (70%), a validation set (15%), and a test set (15%). Statistics of datasets are shown in Table II.

- The *Twitter* hashtag cascade dataset is collected by [59], which contains public English written tweets published between Mar 24 and Apr 25, 2012. A Hashtag and its adopters form an information cascade. The global graph is built by hashtag adopters and their relationships, including follower/followee, retweet and mention interactions. We set observation time t_o to 1 d or 2 days, and the prediction time t_p to 32 days.
- *Sina Weibo* is the most popular microblogging platform in China. A Weibo retweet cascade is composed of the original tweet and its retweets. This dataset is introduced by [29], which collected all original tweets posted on Jun 1, 2016, and tracks their retweets the same day. All user retweeting relationships form the Weibo global graph. We set the observation time t_o to 0.5 h or 1 h, and the prediction time t_p to 24 hours.
- *APS* is a citation dataset released by American Physical Society (https://journals.aps.org/datasets). It contains 616,316 scientific papers published by 17 APS journals. Every paper in the APS dataset and its citations form a

citation cascade. The global graph in APS is defined as the author interaction graph. We set the observation time t_o to 3 years or 5 years, and the prediction time t_p to 20 years.

Following [11], we filter out the cascades with observation size $|C(t_o)| < 10$, and for observed cascade size $|C(t_o)| > 100$, we only select the first 100 adopters for training. Twitter hashtags are tracked at least 15 days during the observation window. Following [29], we only consider Weibo tweets published between 8 AM and 6 PM, ensuring at least 6 hours for retweets growth. For the APS dataset, we select papers published between 1893 and 1997, giving each article ≥ 20 years to gain citations.

2) Baselines: We compare our model with 11 information cascade learning methods (indicated in boldface) from three categories:

(1) Feature-based methods are widely used for information cascade prediction. Features extracted from cascades are fed into various machine learning models for prediction. Previous studies [7] show that structural features and temporal features are informative for information cascade prediction, e.g., in [60] the authors used observed popularity $P(t_o)$ to predict $P(t_p)$. We denote this method as **Feature-*P***. In addition, we extract all features mentioned in [11] and feed them into a linear regression model and a two-layer MLP model, and denote these two methods as **Feature-Linear** and **Feature-Deep**, respectively.

(2) Statistical-based models: [61] builds a time-series model to predict information popularity, denoted as **TimeSeries**. We also include **DeepHawkes** model [29], which integrates the high prediction power of end-to-end deep neural networks into the interpretable factors of point process for popularity prediction, which considers three main aspects of Hawkes process, i.e., the influence of users, self-exciting mechanism, and time decay.

(3) Deep learning-based: Existing deep learning-based models utilize recurrent neural networks (RNNs) [62] to model the temporal information automatically from cascade itself. In addition, some works also consider the cascade structure that depends on the underlying social network. They leveraged graph neural networks (GNNs) or attention mechanisms to learn social correlations for information cascade popularity prediction.

- **CasCN** [13] combines recurrent neural network and graph convolution network to exploit both temporal and structural information for cascade prediction. Specifically, it samples sub-cascade graphs and uses LSTM to capture the evolving process.
- **CoupledGNN** [33] leverages two coupled GNNs to capture the interplay between node activation states and the spread of influence and stacks graph neural network layers to capture the cascading effect.
- **FOREST** [10] combines reinforcement learning and RNNs to handle the multi-scale cascade prediction problem.
- **DyDiff-VAE** [25] learns user interest evolution using GRU and estimates the propagation likelihood with a dual attentive variational autoencoders.
- **LatentODE** [15] generalizes discrete RNNs to continuous-time hidden dynamics defined by ODEs. It considers the latent representation a time-series variable in RNNs, and therefore is capable of handling arbitrary time gaps between observations.
- **CasFlow** [11] learns both local and global structures in information cascades and leverages variational autoencoders and normalizing flows to enhance the learned cascade representations.
- **CTCP** [63] designs a graph learning framework for cascade popularity prediction. It considers different cascades via a universal sequence of user-cascade and user-user interactions, as well as then extracts the dynamic states of users and cascades learned from graph sequences.
- **CasTformer** [20] incorporates a global spatio-temporal positional encoding and relative relationship bias matrices into the self-attention architecture, which enables this model to extract diverse cascade relationships for the popularity prediction.
- **MINDS** [64] constructs sequential hypergraphs to capture dynamic interactions among various cascades. It combines adversarial training and orthogonality constraints to address feature redundancy between shared and task-specific features for predicting popularity.

3) Parameter Settings: All models are tuned to the best performance on the validation set, with early stopping (patience is 10 epochs). The batch size is set to 64. The weight hyperparameter λ is selected from $\{0.001, 0.01, 0.1, 0.5, 1\}$ to trade-off between the prediction loss and uncertainty modeling during training. We select the dimensions of latent states and cascade embeddings from 8 to 128. In the prediction network, we use three layers of MLPs and ReLU activation. All deep learning models, including CasDO, are optimized by Adam [65] with a learning rate of 5×10^{-4} on the training set. In the diffusion models, the diffusion steps are set to 8 using a linear variance schedule starting from $\beta_1 = 1 \times 10^{-4}$ and $\beta_R = 0.02$. The architecture for learning $\boldsymbol{\epsilon}_\theta$ is a conditional 1-dim dilated ConvNets with residual connections adapted from DDIM [41].

4) Evaluation Metrics: Following the previous studies [11], [13], [29], we use mean square log-transformed error (MSLE) and mean absolute percentage error (MAPE) to evaluate the performance of prediction:

$$\text{MSLE} = \frac{1}{N}\sum_{i=0}^{N-1}\left(\log_2 \hat{P}_i - \log_2 P_i\right)^2, \tag{20}$$

$$\text{MAPE} = \frac{1}{N}\sum_{i=0}^{N-1}\frac{\left|\log_2 \hat{P}_i - \log_2 P_i\right|}{\log_2 P_i}, \tag{21}$$

where N is the total number of cascades and $P_i = P_i(t_p) - P_i(t_o)$ is the incremental cascade size.

B. Performance Comparison

The performance of baselines and CasDO on three datasets are summarized in Table III, from which we have the following observations:

TABLE III
PERFORMANCE COMPARISONS ON THREE DATASETS OVER TWO DIFFERENT OBSERVATION PERIODS

Model	Twitter				Weibo				APS			
	1 Day		2 Days		0.5 Hour		1 Hour		3 Years		5 Years	
	MSLE	MAPE	MSLE	MAPE	MSLE	MAPE	MSLE	MAPE	MSLE	MAPE	MSLE	MAPE
Feature-P	14.792	0.961	13.515	0.983	4.455	0.391	4.001	0.398	2.382	0.316	2.348	0.352
Feature-Linear	9.326	0.520	6.758	0.459	2.959	0.258	2.640	0.271	1.852	0.272	1.728	0.291
Feature-Deep	7.438	0.485	6.357	0.500	2.715	0.228	2.546	0.272	1.844	0.270	1.666	0.282
TimeSeries	8.214	0.547	6.023	0.445	3.119	0.277	2.693	0.268	1.867	0.271	1.735	0.291
DeepHawkes	7.216	0.587	5.788	0.536	2.891	0.268	2.796	0.282	1.573	0.271	1.324	0.335
CasCN	7.183	0.547	5.561	0.525	2.804	0.254	2.732	0.273	1.562	0.268	1.421	0.265
CoupledGNN	7.216	0.562	5.578	0.525	2.845	0.255	2.766	0.281	1.673	0.275	1.395	0.259
FOREST	7.256	0.556	5.558	0.493	2.755	0.249	2.753	0.279	1.603	0.265	1.404	0.265
LatentODE	7.112	0.475	5.232	0.377	2.429	0.228	2.234	0.245	1.367	0.227	1.373	0.251
DyDiff-VAE	7.012	0.463	5.205	0.372	2.416	0.220	2.227	0.253	1.372	0.232	1.362	0.247
CasFlow	6.955	0.456	5.144	0.362	2.404	0.211	2.281	0.242	1.364	0.224	1.352	0.246
CTCP	6.625	0.421	4.937	0.353	2.546	0.236	2.399	0.245	1.401	0.238	1.539	0.266
CasSampling	6.837	0.437	5.072	0.360	2.519	0.239	2.325	0.240	1.499	0.252	1.536	0.271
MINDS	6.514	0.431	4.757	0.339	2.589	0.232	2.478	0.236	1.425	0.241	1.418	0.236
CasDO-LocalStruct	7.054	0.425	5.112	0.364	2.641	0.226	2.423	0.248	1.768	0.257	1.648	0.275
CasDO-GlobalStruct	11.532	0.724	9.619	0.651	2.914	0.254	2.783	0.292	1.378	0.231	1.536	0.256
CasDO-TODE	6.812	0.436	5.029	0.354	2.432	0.215	2.195	0.217	1.121	0.208	1.218	0.229
CasDO-Sum	6.559	0.434	4.744	0.345	2.409	0.214	2.241	0.231	1.135	0.219	1.262	0.239
CasDO-Pooling	6.485	0.425	4.664	0.331	2.369	0.212	2.221	0.221	1.105	0.215	1.252	0.231
CasDO-MLP	6.529	0.428	4.684	0.339	2.389	0.214	2.238	0.225	1.155	0.229	1.261	0.237
CasDO-Diffusion	6.521	0.416	4.702	0.335	2.356	0.213	2.185	0.218	1.105	0.213	1.122	0.231
CasDO-ODEs	6.642	0.425	4.704	0.336	2.328	0.208	2.235	0.225	1.119	0.216	1.204	0.236
CasDO	**6.376**	**0.376**	**4.511**	**0.311**	**2.308**	**0.201**	**2.178**	**0.214**	**1.093**	**0.207**	**1.011**	**0.229**

A paired t-test was performed for statistical significance of the results ($p < 0.005$).

(O1): The proposed CasDO outperforms all previous models, in terms of both MSLE and MAPE, for both application scenarios (social and academic information diffusion) on cascade popularity prediction, demonstrating the benefits of capturing the uncertainties between observations in continuous-time settings and exploiting probabilistic diffusion to resist noisy observation in modeling and learning the information propagation.

(O2): The performance gap between Feature-Deep and Feature-Linear is quite small, suggesting the importance of features in cascade prediction, if we have a set of representative features. Notably, in some cases, feature-based methods and diffusion model-based methods outperform some deep learning models. However, their performance heavily depends on handcrafted features that are difficult to select for different scenarios in practice.

(O3): DeepHawkes relies on the capability of time-series modeling along the diffusion route to predict the information popularity, without considering the topology information of cascades. As previously observed [11], [13], it often overrates the cascade size due to its rudimentary self-excitation mechanism [66]. Interestingly, LatentODE achieves better performance than TimeSeries and DeepHawkes, verifying our motivation of modeling irregularly sampled information diffusion, rather than treating the irregular events as fixed time intervals widely adopted in RNN-based models.

(O4): Deterministic GNN-based approaches, such as CasCN, CoupleGNN, CTCP, CasSampling and MINDS, exploit the structural and temporal factors for cascade prediction showing relatively well prediction performance. However, these methods focus solely on learning local or global structures and ignore the interactions between users in both graphs. More importantly, they fail to account for diffusion uncertainty. DyDiff-VAE and CasFlow, in contrast, explicitly utilize variational inference for modeling the latent structure of information diffusion and therefore perform the best among baseline approaches. Nevertheless, their models still sample cascades in fixed time intervals, which limits the performance of learning temporal variances and decaying factors.

C. Ablation Study

We conduct ablation studies to investigate the effects of different parts in CasDO, and consider the following variants of CasDO.

– *CasDO-LocalStruct* and *CasDO-GlobalStruct* remove the global and the local cascade node embedding $\mathbf{E}_c$ and $\mathbf{E}_g$, respectively.

– *CasDO-Sum*, *CasDO-Pooling* and *CasDO-MLP* utilize different strategies to process the global node embedding $\mathbf{E}_c$ and local node embedding $\mathbf{E}_g$. Specifically, CasDO-Sum utilizes a sum operation, CasDO-Pooling employs average pooling, and CasDO-MLP applies a linear layer.

– *CasDO-TODE* only uses T-ODE to output $\mathbf{Z}$ for modeling cascades and predicting the cascade popularity, i.e., without using the structure and evolution uncertainty representation learning.

– *CasDO-Diffusion* and *CasDO-ODE* – the former only considers the structural uncertainty representation learning, and the latter only utilizes evolution's uncertainty representation

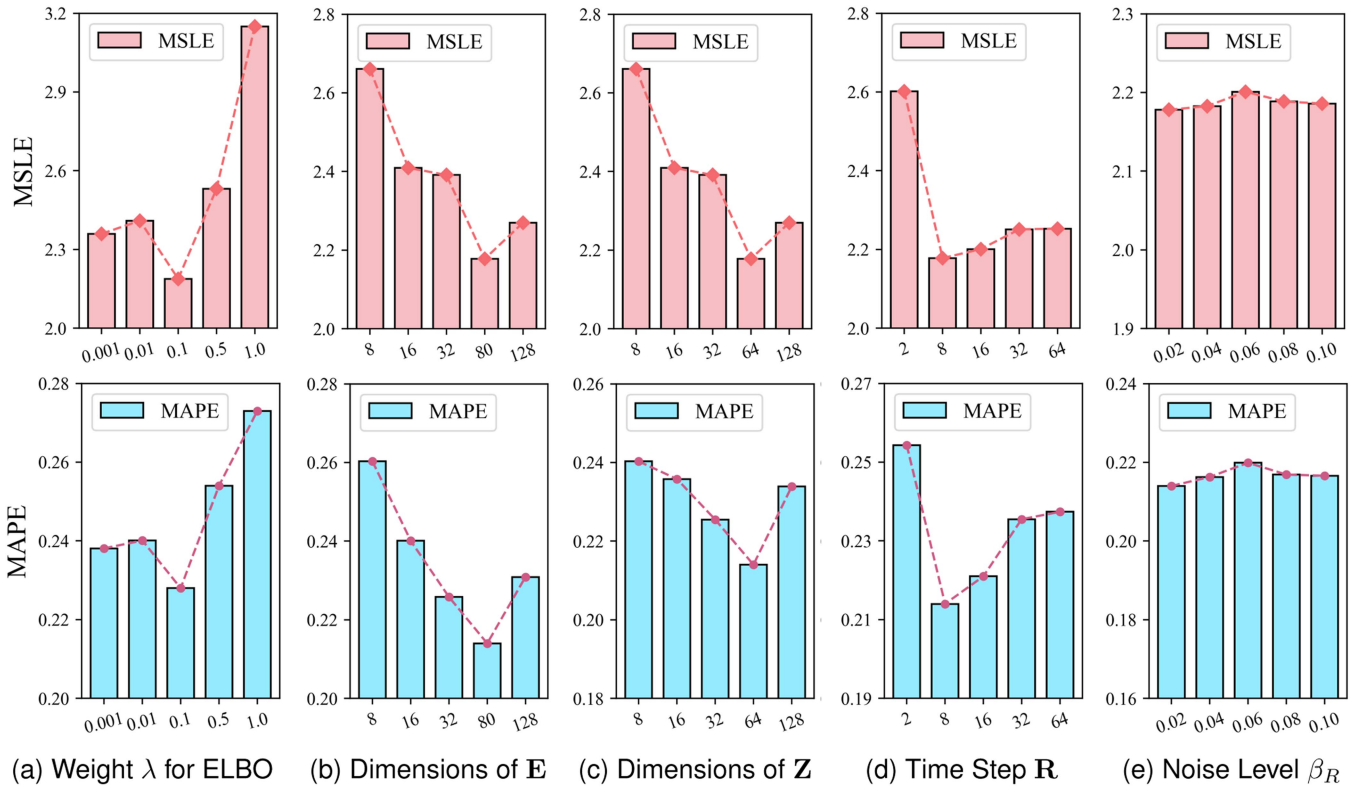


Fig. 5. Impact of four important hyper-parameters of CasDO on Weibo dataset (observation time is 1 h), measured by MAPE and MSLE.

learning. Both of the two variants use T-ODE to model temporal characteristics for information cascade popularity prediction.

The lower part of Table III reports the performance comparisons among CasDO and its variants: (1) CasDO-LocalStruct shows considerably better performance than CasDO-GlobalStruct on Twitter and Weibo, but the global structure is a more reliable predictor for APS. It indicates that specific local structural features are beneficial to social cascade prediction. Apparently, the combination of local and global graphs indeed improves the performance. (2) In the investigation of fusing the global and local cascade structural representations, we have devised several variant models to analyze the influence of different fusion strategies, including utilizing a sum operation, average pooling, and a linear layer. As shown in Table III, we observe a marginal decline in prediction performance when compared to the original method of concatenation. This discovery indicates that our CasDO model can be effectively substituted with alternative representation fusion techniques. Furthermore, it validates the scalability of our CasDO. (3) CasDO-Diffusion and CasDO-ODEs consistently achieve better performance than CasDO-TODE across all datasets. This result indicates that our T-ODE approach can generalize state transitions to continuous-time dynamics, which is sensitive to the initial state $\mathbf{h}_0$. In other words, T-ODE alone cannot adequately model cascade uncertainty, and therefore results in degenerated performance. (4) CasDO-Diffusion performs slightly better than CasDO-ODEs, suggesting that modeling the structural uncertainty is more important than modeling the evolutionary uncertainty, although the two types of probabilistic diffusion complement each other.

D. Analysis of Parameter Impact in CasDO

We now report the impact of several important parameters of CasDO: the weight for latent ODEs losses, the dimensions of cascade embedding, the dimensions of latent factors, and the time steps of probabilistic diffusion. We only report the results on Weibo dataset (with 1 h observation time) due to the same trends observed on other datasets. Detailed results are explained as follows:

1) *Weight λ for ELBO:* Recall that we gave a weight λ to the losses of latent ODEs (cf. Eq (19)), which trades off between supervised learning w.r.t. cascade popularity and uncertainty learning during information propagation. We can see that a smaller λ is desired for CasDO, which implies the information prediction task is still heavily dependent on labeled training for future prediction. It also raises an interesting question, i.e., how to adaptively set the uncertainty learning factor, which is left as our future work.
2) *Dimensions of latent factor and cascade embedding:* We change the dimensions of latent factors and cascade embedding from 8 to 128. As shown in Fig. 5(b) and (c), the best performance of CasDO is achieved when the dimensions of latent factor is 64 and the dimension of cascade embedding is 80.
3) *Time steps of diffusion model:* We varied the time steps in the diffusion models from 1 to 64 and plot the results in Fig. 5(d). We can observe that the best performance was obtained when $R = 8$, which means a few steps of

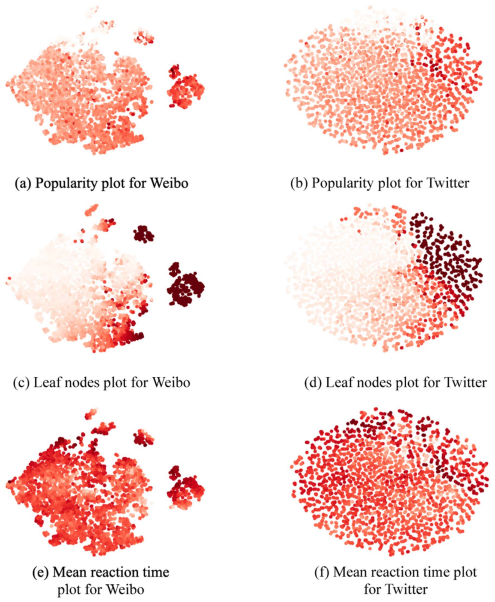


Fig. 6. Visualization of learned latent representations on Weibo and Twitter datasets (observation time = 0.5h/1d) using t-SNE. Each point is a sample from test cascades. The darker the point, the larger the value of the corresponding feature.

diffusion is enough for CasDO to capture the structural uncertainty.

4) *Noise level β_R of diffusion model:* We experimented with varying the noise level β_R in the diffusion models, ranging from 0.02 to 0.1. The corresponding results are plotted in Fig. 5(e). Notably, we observed a minor fluctuation in the model performance of CasDO as the value of β_R increased. This finding indicates that our model demonstrates low sensitivity to the hyper-parameter β_R. After evaluating the model's performance on the MSLE and MAPE metrics, we determined that setting β_R to 0.02 provides a balanced model performance.

E. Learning Interpretability

In this section, we investigate and interpret the performance of CasDO on information cascade representation learning. To do this, we select a few hand-crafted features related to the information cascade that characterize either network or time properties in the information diffusion process. In three respective rows of Fig. 6, we plot the learned latent space of cascades for CasDO, including the Weibo and Twitter dataset, using t-SNE [67]. In each subfigure, each point in the plot represents a cascade in the test set – cascades with similar latent vectors are close in the plot. Then, to connect the hand-crafted features with the learned cascade representations, we color each point in the plot by the values of each feature (e.g., number of leaf nodes), implying a connection between the learned representation and that network property. Furthermore, we color the layout by the ground-truth labels (i.e., cascade popularity) to study whether CasDO can distinguish between distinct information cascades. Note that the darker the point, the large the value of the corresponding feature to the cascade, e.g., popularity (1st row), number of leaf nodes (2nd row), and mean reaction time (3rd row).

As demonstrated in Fig. 6(a) and (b), it shows the latent representations from the last MLP layer of CasDO and the color denotes the cascade popularity. In Fig. 6(a) and (b), we can observe that the visualizations show clear clustering effects. It suggests that CasDO could distinguish between distinct information cascades. For example, the colors of the right points are darker than those of the left or top points. This result shows a smooth representation-to-2D projection and meets the long-tailed effect in the information diffusion process [68] that infrequent cascades (e.g., outbreak tweets) are few. This means that CasDO mappings the cascades' latent representations to corresponding labels (i.e., popularity).

Fig. 6(c) and (f) depict the latent representations from the last MLP layer of CasDO. We can observe that CasDO produces smooth representation-to-2D projections in respect of both structural and temporal patterns. The visualizations show clear clustering effects compared to the ground truth popularity: with larger number of leaf nodes and longer mean reaction time, the popularity of cascade tends to be larger, e.g., the points clustered to the left have fewer leaf nodes, while graphs on the right have the most in Fig. 6(c) and (d). It implies that a Cascade graph with a larger number of leaf nodes promotes the growth of its ego-net. Indeed, when we compare the color scheme of Fig. 6(c) and (d) with Fig. 6(a) and (b), we can see that the number of leaf nodes in an information cascade is indeed positively correlated with its growth. Furthermore, the visualization of mean reaction time in Fig. 6(e) and (f) have similar color distributions with the true popularity size Fig. 6(a) and (b). Note that in CasDO we did not explicitly use these features for training, but the model itself learns meaningful and explainable semantics of the corresponding features correlated to the future popularity. This result further demonstrates the benefits of incorporating the diffusion model and T-ODE to capture the diffusion uncertainty between observations when modeling the information propagation.

VII. Conclusion

We presented CasDO, a novel probabilistic framework for end-to-end modeling and prediction of information cascade growth. It does not rely heavily on feature engineering and can be easily generalized, enabling the information cascade popularity prediction by exploiting both structural and temporal information. CasDO leverages continuous-time evolutionary variational information diffusion model to handle the irregular-sampled cascade events and exploit the uncertainties at both node level and cascade level with diffusion model and latent ODEs. Our experiments conducted on three real-world datasets demonstrated the superior performance of CasDO over state-of-the-art baselines. In our future work, we plan to extend CasDO with location-awareness and enable the incorporation of urban geospatial features in the prediction of cascades evolution.

References

[1] F. Zhou, X. Xu, G. Trajcevski, and K. Zhang, "A survey of information cascade analysis: Models, predictions, and recent advances," *ACM Comput. Surv.s*, vol. 54, no. 2, pp. 1–36, 2021.

[2] S. Ranganath, S. Wang, X. Hu, J. Tang, and H. Liu, "Facilitating time critical information seeking in social media," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 10, pp. 2197–2209, Oct. 2017.

[3] C. Li, J. Ma, X. Guo, and Q. Mei, "DeepCas: An end-to-end predictor of information cascades," in *Proc. Int. Conf. World Wide Web*, 2017, pp. 577–586.

[4] H. Shen, D. Wang, C. Song, and A.-L. Barabási, "Modeling and predicting popularity dynamics via reinforced poisson processes," in *Proc. AAAI Conf. Artif. Intell.*, 2014, pp. 291–297.

[5] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information diffusion through blogspace," in *Proc. Int. Conf. World Wide Web*, 2004, pp. 491–501.

[6] F. Zhang et al., "Understanding WeChat user preferences and "WOW" diffusion," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 12, pp. 6033–6046, Dec. 2022.

[7] J. Cheng, L. Adamic, P. A. Dow, J. M. Kleinberg, and J. Leskovec, "Can cascades be predicted?," in *Proc. Int. Conf. World Wide Web*, 2014, pp. 925–936.

[8] L. Li et al., "Propagation analysis and prediction of the Covid-19," *Infect. Dis. Modelling*, vol. 5, 2020, pp. 282–292.

[9] M.-A. Rizoiu, S. Mishra, Q. Kong, M. Carman, and L. Xie, "SIR-Hawkes: Linking epidemic models and Hawkes processes to model diffusions in finite populations," in *Proc. Int. Conf. World Wide Web*, 2018, pp. 419–428.

[10] C. Yang et al., "Full-scale information diffusion prediction with reinforced recurrent networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 5, pp. 2271–2283, May 2023.

[11] X. Xu, F. Zhou, K. Zhang, S. Liu, and G. Trajcevski, "CasFlow: Exploring hierarchical structures and propagation uncertainty for cascade prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 3484–3499, Apr. 2023.

[12] J. Wang, V. W. Zheng, Z. Liu, and K. C.-C. Chang, "Topological recurrent neural network for diffusion prediction," in *Proc. Int. Conf. Des. Mater.*, 2017, pp. 475–484.

[13] X. Chen, F. Zhou, K. Zhang, G. Trajcevski, T. Zhong, and F. Zhang, "Information diffusion prediction via recurrent cascades convolution," in *Proc. IEEE 35th Int. Conf. Data Eng.*, 2019, pp. 770–781.

[14] J. Qiu, J. Tang, H. Ma, Y. Dong, K. Wang, and J. Tang, "DeepInf: Social influence prediction with deep learning," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 2110–2119.

[15] Y. Rubanova, R. T. Chen, and D. Duvenaud, "Latent odes for irregularly-sampled time series," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 5321–5331.

[16] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A review on generative adversarial networks: Algorithms, theory, and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 3313–3332, Apr. 2023.

[17] L. Girin, S. Leglaive, X. Bie, J. Diard, T. Hueber, and X. Alameda-Pineda, "Dynamical variational autoencoders: A comprehensive review," 2020, *arXiv: 2008.12595*.

[18] Z. Xie, C. Liu, Y. Zhang, H. Lu, D. Wang, and Y. Ding, "Adversarial and contrastive variational autoencoder for sequential recommendation," in *Proc. Web Conf.*, 2021, pp. 449–459.

[19] Y. Wang, H. Zhang, Z. Liu, L. Yang, and P. S. Yu, "Contrastvae: Contrastive variational autoencoder for sequential recommendation," in *Proc. 31st ACM Int. Conf. Inf. Knowl. Manage.*, 2022, pp. 2056–2066.

[20] X. Sun, J. Zhou, L. Liu, and Z. Wu, "Castformer: A novel cascade transformer towards predicting information diffusion," *Inf. Sci.*, vol. 648, 2023, Art. no. 119531.

[21] C. Yang, M. Sun, H. Liu, S. Han, Z. Liu, and H. Luan, "Neural diffusion model for microscopic cascade study," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 3, pp. 1128–1139, Mar. 2021.

[22] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2256–2265.

[23] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, "Neural ordinary differential equations," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 6572–6583.

[24] F. Zhou, X. Xu, K. Zhang, G. Trajcevski, and T. Zhong, "Variational information diffusion for probabilistic cascades prediction," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 1618–1627.

[25] R. Wang et al., "DyDiff-VAE: A dynamic variational framework for information diffusion prediction," in *Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2021, pp. 163–172.

[26] H. Li, C. Xia, T. Wang, S. Wen, C. Chen, and Y. Xiang, "Capturing dynamics of information diffusion in SNS: A survey of methodology and techniques," 2021, *arXiv:2110.14245*.

[27] X. Chen, F. Zhou, F. Zhang, and M. Bonsangue, "Modeling microscopic and macroscopic information diffusion for rumor detection," *Int. J. Intell. Syst.*, vol. 36, no. 10, pp. 5449–5471, 2021.

[28] M.-A. Rizoiu, L. Xie, S. Sanner, M. Cebrian, H. Yu, and P. Van Hentenryck, "Expecting to be HIP: Hawkes intensity processes for social media popularity," in *Proc. Int. Conf. World Wide Web*, 2017, pp. 735–744.

[29] Q. Cao, H. Shen, K. Cen, W. Ouyang, and X. Cheng, "DeepHawkes: Bridging the gap between prediction and understanding of information cascades," in *Proc. Conf. Inf. Knowl. Manage.*, 2017, pp. 1149–1158.

[30] B. Shulman, A. Sharma, and D. Cosley, "Predictability of popularity: Gaps between prediction and understanding," in *Proc. Int. AAAI Conf. Web Social Media*, 2016, pp. 348–357.

[31] Y. Liu, K. Zeng, H. Wang, X. Song, and B. Zhou, "Content matters: A GNN-based model combined with text semantics for social network cascade prediction," in *Proc. Pacific-Asia Conf. Knowl. Discov. Data Mining*, 2021, pp. 728–740.

[32] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, "SEISMIC: A self-exciting point process model for predicting tweet popularity," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 1513–1522.

[33] Q. Cao, H. Shen, J. Gao, B. Wei, and X. Cheng, "Popularity prediction on social platforms with coupled graph neural networks," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2020, pp. 70–78.

[34] X. Xu, F. Zhou, K. Zhang, and S. Liu, "CCGL: Contrastive cascade graph learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 4539–4554, May 2023.

[35] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1530–1538.

[36] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proc. Int. Conf. Learn. Representations*, 2014.

[37] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[38] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *Proc. Int. Conf. Learn. Representations*, 2019.

[39] S. Su, J. Song, L. Gao, and J. Zhu, "Towards unsupervised deformable-instances image-to-image translation," in *Proc. Int. Joint Conf. Artif. Intell.*, 2021, pp. 1004–1010.

[40] Jarzynski, "Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach," *Phys. Rev. E*, vol. 56, no. 5, 1997, Art. no. 5018.

[41] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *Proc. Int. Conf. Learn. Representations*, 2021.

[42] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, pp. 8780–8794.

[43] C. Meng, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "SDEdit: Image synthesis and editing with stochastic differential equations," in *Proc. Int. Conf. Learn. Representations*, 2022.

[44] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "DiffWave: A versatile diffusion model for audio synthesis," in *Proc. Int. Conf. Learn. Representations*, 2021.

[45] Y. Tashiro, J. Song, Y. Song, and S. Ermon, "CSDI: Conditional score-based diffusion models for probabilistic time series imputation," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, pp. 24804–24816.

[46] W. Grathwohl, R. T. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud, "FFJORD: Free-form continuous dynamics for scalable reversible generative models," in *Proc. Int. Conf. Learn. Representations*, 2019.

[47] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.

[48] J. Zhang, Y. Dong, Y. Wang, J. Tang, and M. Ding, "ProNE: Fast and scalable network representation learning," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, pp. 4278–4284.

[49] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.

[50] Z. Song, X. Yang, Z. Xu, and I. King, "Graph-based semi-supervised learning: A comprehensive review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 11, pp. 8174–8194, Nov. 2023.

[51] S. Wang, J. Tang, Y. Wang, and H. Liu, "Exploring hierarchical structures for recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 6, pp. 1022–1035, Jun. 2018.

[52] Z. Hou, Y. Cen, Y. Dong, J. Zhang, and J. Tang, "Automated unsupervised graph representation learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 3, pp. 2285–2298, Mar. 2023.

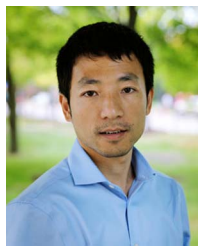
[53] D. Wang et al., "Modeling co-evolution of attributed and structural information in graph sequence," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 1817–1830, Feb. 2023.

[54] L. Yu, L. Sun, B. Du, C. Liu, W. Lv, and H. Xiong, "Heterogeneous graph representation learning with relation awareness," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 6, pp. 5935–5947, Jun. 2023.
[55] C. Donnat, M. Zitnik, D. Hallac, and J. Leskovec, "Learning structural node embeddings via diffusion wavelets," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 1320–1329.
[56] M. Lechner and R. Hasani, "Learning long-term dependencies in irregularly-sampled time series," 2020, *arXiv: 2006.04418*.
[57] Q. Kong, M.-A. Rizoiu, and L. Xie, "Exploiting uncertainty in popularity prediction of information diffusion cascades using self-exciting point processes," 2020, *arXiv: 2001.11132v1*.
[58] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 6840–6851.
[59] L. Weng, F. Menczer, and Y.-Y. Ahn, "Virality prediction and community structure in social networks," *Sci. Rep.*, vol. 3, no. 1, pp. 1–6, 2013.
[60] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," *Commun. ACM*, vol. 53, no. 8, pp. 80–88, 2010.
[61] H. Pinto, J. M. Almeida, and M. A. Gonçalves, "Using early view patterns to predict the popularity of youtube videos," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2013, pp. 365–374.
[62] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[63] X. Lu, S. Ji, L. Yu, L. Sun, B. Du, and T. Zhu, "Continuous-time graph learning for cascade popularity prediction," in *Proc. Int. Joint Conf. Artif. Intell.*, 2023, pp. 2224–2232.
[64] P. Jiao, H. Chen, Q. Bao, W. Zhang, and H. Wu, "Enhancing multi-scale diffusion prediction via sequential hypergraphs and adversarial learning," in *Proc. AAAI Conf. Artif. Intell.*, 2024, pp. 8571–8581.
[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.
[66] X. Gao, Z. Cao, S. Li, B. Yao, G. Chen, and S. Tang, "Taxonomy and evaluation for microblog popularity prediction," *ACM Trans. Knowl. Discov. Data*, vol. 13, no. 2, pp. 1–40, 2019.
[67] L. Van Der Maaten, "Accelerating T-SNE using tree-based algorithms," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3221–3245, 2014.
[68] F. Zhou, L. Yu, X. Xu, and G. Trajcevski, "Decoupling representation and regressor for long-tailed information cascade prediction," in *Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2021, pp. 1875–1879.

Zhangtao Cheng received the MS degree in electronic and information engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, Sichuan, China, in 2023. He is currently working toward the PhD degree with UESTC. His current research interests include social network analysis, Spatio-temporal data mining, and multimodal learning.

Fan Zhou (Member, IEEE) received the BS degree in computer science from Sichuan University, Chengdu, China, in 2003, and the MS and PhD degrees in computer science from the University of Electronic Science and Technology of China, Chengdu, in 2006 and 2012, respectively. He is currently a professor with the School of Information and Software Engineering, University of Electronic Science and Technology of China. His research interests include machine learning, neural networks, spatio-temporal data management, graph learning, recommender systems, and social network data mining.

Xovee Xu (Graduate Student Member, IEEE) received the BS and MS degrees in software engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, Sichuan, China, in 2018 and 2021, respectively. He is currently working toward the PhD degree in computer science with UESTC. His research focuses on understanding information diffusion, user-generated content, multimodal learning, and human behaviors in social network. His work has been presented at top-tier conferences and journals such as *IEEE Transactions on Knowledge and Data Engineering*, INFOCOM, AAAI, SIGIR, SIGKDD, CIKM, NeurIPS, and ACM CSUR. He has served as a program committee member for conferences like SIGKDD, CIKM, ICASSP, and SDM.



Kunpeng Zhang received the PhD degree in computer science from Northwestern University. He is currently an associate professor in the Department of Information Systems with the University of Maryland, College Park. His research focuses on developing machine/deep learning algorithms to analyze unstructured data for better business decisions. Specifically, I am interested in multi-modal representation learning + LLMs in business. He published papers on top conferences and journals. He serves as program committees for many conferences and currently an associate editor for INFORMS Journal on Computing.

Goce Trajcevski (Member, IEEE) received the BSc degree in informatics and automation from the University of Sts. Kiril i Metodij, Skopje, North Macedonia, in 1989, and the MS and PhD degrees in computer science from the University of Illinois at Chicago, Chicago, IL, USA, in 1995 and 2002, respectively. He is currently an associate professor with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA. His research has been funded by the NSF, ONR, BEA, and Northrop Grumman Corporation. In addition to a book chapter and three encyclopedia chapters, he has coauthored more than 140 publications in refereed conferences and journals. His main research interests are in the areas of spatiotemporal data management, uncertainty and reactive behavior management in different application settings, and incorporating multiple contexts. Dr. Trajcevski was the General Co-Chair of the IEEE International Conference on Data Engineeing 2014 and ACM SIGSPATIAL 2019, the PC CoChair of the ADBIS 2018 and ACM SIGSPATIAL 2016 and 2017, and has served in various roles in organizing committees in numerous conferences and workshops. He is an associate editor of the ACM Transactions on Spatial Algorithms and Systems and the Geoinformatica Journals.

Ting Zhong received the BS degree in computer application and the MS degree in computer software and theory from Beijing Normal University, Beijing, China, in 1999 and 2002, respectively, and the PhD degree in information and communication engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2009. She is a full professor of UESTC. Her current research interests include deep learning, social networks, and cloud computing.

Philip S. Yu (Life Fellow, IEEE) received the BS degree in E.E. from National Taiwan University, and the MS and PhD degrees in E.E. from Stanford University, and the MBA degree from New York University. He is a distinguished professor in computer science with the the University of Illinois, Chicago, and also holds the wexler chair in Information Technology. He has spent most of his career with IBM, where he was a manager of the Software Tools and Techniques group with the Watson Research Center. His research interest is in Big Data, including data mining, data stream, database and privacy. He has published more than 1680 papers in refereed journals and conferences. He holds or has applied for more than 300 US patents. He is a fellow of the ACM. He is the editor-in-chief of ACM Transactions on Knowledge Discovery from Data. He is on the steering committee of the IEEE Conference on Data Mining and ACM Conference on Information and Knowledge Management and was a member of the IEEE Data Engineering steering committee. He was the editor-in-chief of IEEE Transactions on Knowledge and Data Engineering (2001-2004).