# Continual Information Cascade Learning

Fan Zhou*, Xin Jing*, Xovee Xu*, Ting Zhong*, Goce Trajcevski† and Jin Wu*‡
* University of Electronic Science and Technology of China, China
†Department of Electrical and Computer Engineering, Iowa State University, USA
‡Jin Wu is the corresponding author: wj@uestc.edu.cn

***Abstract*—Modeling the information diffusion process is an essential step towards understanding the mechanisms driving the success of information. Existing methods either exploit various features associated with cascades to study the underlying factors governing information propagation, or leverage graph representation techniques to model the diffusion process in an end-to-end manner. Current solutions are only valid for a static and fixed observation scenario and fail to handle increasing observations due to the challenge of catastrophic forgetting problems inherent in the machine learning approaches used for modeling and predicting cascades. To remedy this issue, we propose a novel dynamic information diffusion model CICP (Continual Information Cascades Prediction). CICP employs graph neural networks for modeling information diffusion and continually adapts to increasing observations. It is capable of capturing the correlations between successive observations while preserving the important parameters regarding cascade evolution and transition. Experiments conducted on real-world cascade datasets demonstrate that our method not only improves the prediction performance with accumulated data but also prevents the model from forgetting previously trained tasks.**

***Index Terms*—information cascades, continual learning, catastrophic forgetting, graph neural networks, popularity prediction**

## I. Introduction

Online social networks (OSN) such as Twitter, Weibo, Reddit, Instagram and Facebook have become the main source of information in people's daily life. Various news, events, and posts are disseminated as information cascades spread by users through OSN [1], [2]. In academic community, researchers publish their works in various venues which have been fully digitized and provide unprecedented opportunities to share the scientific results and discover the new ideas for researchers. As a result, predicting the size of (potentially) affected users or authors after a certain time-period has attracted great attention in both academia and industry, which plays a critical role in many down-stream applications – from fake news detection, through epidemic spread identification and improved advertising effect, to suppressing rumor information propagation [3], [4].

**Existing approaches.** While the trajectories of the information items such as microblogs, photos/videos and academic papers are usually structured as information cascades and have been proved to be predictable to some extent [5], a variety of methods have been proposed to analyze, model and predict the popularity of the information cascades. Existing approaches for popularity prediction mainly fall into three categories, i.e., *feature*-based methods, *stochastic process*-based methods and *deep learning*-based methods. Feature-based methods extract the attributes from raw data by hand-crafted feature engineering [5], [6], where the user generated contents, user profiles, structural and temporal features of cascades are widely employed to model the information diffusion process. Therefore, typical machine learning methods such as naïve Bayes, decision trees, neural networks and probabilistic graphical models can be used to perform prediction tasks. Another line of work studies the stochastic process of information diffusion and generally captures the potential rules of the arrival of events (e.g., the adoption by a user), and then predicts the popularity based on the learned diffusion process. These works heavily rely on the assumed stochastic model such as Poisson and Hawkes point processes, whereas the designed self-exciting mechanisms and intensity functions [7], [8] learned in one domain are difficult to be generalized to another domain.

Recent advances in deep neural networks have inspired a few of works learning the information diffusion with various deep learning techniques [2], [9]–[11]. Generally, these methods learn the structural information regarding the evolving process of cascades using graph representation learning techniques and employ recurrent neural networks (RNNs) to model the diffusion steps of cascade. For example, DeepCas [9] borrows the idea of DeepWalk [12] to sample the cascade graphs with random walks. The sampled node sequences are then fed into GRU with attention mechanism to obtain the cascade embeddings and predict cascade popularity in an end-to-end manner. Chen et al. [2] propose to learn the cascade graphs with directed graph convolutional networks (GCN) [13] and employ LSTM with time decay effects to capture the temporal information of diffusion. Recently, VaCas [11] extends the deterministic cascade graph embedding with stochastic node representation and diffusion uncertainty, which allows more robust cascade prediction.

**Challenges.** Typically, current algorithms predict the popularity of a specific information in the future, e.g., 24 hours after posting a tweet or 20 years after the publication of a paper, given a limited number of observations (e.g., 30 minutes for tweets or 3 years for scientific papers). Fig. 1 illustrates a toy example of information diffusion, where $t_{o_1}$ and $t_{o_2}$ are observation times and $t_p$ denotes the prediction time. Previous models usually learn the structural and temporal features of the

This is the author version of the paper presented at IEEE GLOBECOM '20
doi: 10.1109/GLOBECOM42002.2020.9322124

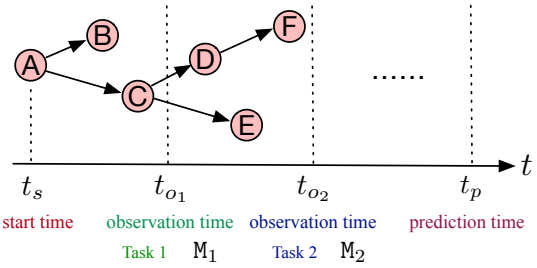



Fig. 1. A toy example of information diffusion and popularity prediction.

cascade given a *fixed* time of earlier observations (e.g., $t_{o_1}$ or $t_{o_2}$) and then predict its popularity at $t_p$.

Despite promising results achieved by current models, existing solutions suffer from a well-known issue called *catastrophic forgetting* [14], i.e., the model's performance may rapidly decreas on previously learned tasks when trained on a new task. Take the case in Fig. 1 for example, we may have trained a model $\mathtt{M}_1$ with the observations by the end of time $t_{o_1}$. As the observation accumulates, $\mathtt{M}_1$ is incapable of adapting to the new observations between $[t_{o_1}, t_{o_2}]$, which means we need to *manually* train another new model $\mathtt{M}_2$ starting from scratch. Though $\mathtt{M}_2$ often performs better than $\mathtt{M}_1$ due to more observations are used, it may not "remember" the parameters of $\mathtt{M}_1$, resulting in significant performance decrease of $\mathtt{M}_2$ on previous task without revisiting the observations before $t_{o_1}$. In real-world applications, the knowledge of the cascade diffusion continually evolves over time and the size of the data often prohibits frequent batch updating and retrospect of the observations in previous tasks, e.g., it is very common that there are millions of retweets for popular articles in a very few hours on Twitter or Weibo. Unfortunately, how to increasingly improve the model performance with new tasks (or incremental observations) while still retaining the performance in previous tasks has not been studied in cascade popularity prediction.

**Present work.** In this paper, we initiate the first attempt to address the continual cascade learning problem. Specifically, we present a novel model CICP (Continual Information Cascade Prediction), which combines graph representation learning and continual learning for dynamic cascade modeling while also adapting to perform well on the entire set of prediction tasks in an incremental way without revisiting the previous data at each stage. In particular, CICP leverages a diffusion graph neural network (GNN) for cascade graph structure modeling and employs LSTM [15] to capture the temporal and sequential diffusion patterns. Moreover, we estimate the distribution of model parameters and the importance of each parameter using Bayesian learning. The posterior of the parameters in previous tasks is inferred with fisher information matrix, which then plays as the prior of the subsequent task where the important parameters (for previous tasks) are largely retained. The main contributions of this work are three-fold:

- We propose a new cascade learning paradigm that is more suitable for practical information dissemination applications, and present the first continual cascade learning framework which can gradually acquire knowledge from streaming of structured diffusion data for incremental popularity prediction.
- We present a novel GNN-based cascade model, which not only captures the structural and temporal diffusion patterns of information cascades, but also allows for effective parameter importance estimation. Our method can largely alleviates the inefficient learning problem when modeling sequential graph representation learning tasks using GNNs.
- We conduct experiments to evaluate the effectiveness of the proposed model on two real-world information cascade datasets. The experimental results demonstrate that CICP learns better diffusion-related knowledge from cascade graphs and preserves the evolving patterns of information, while circumventing catastrophic forgetting issue in existing models.

## II. Related Work

Early efforts mainly focus on characterizing various handcrafted features from raw data, such as cascade graph structures, temporal features, user interests and item contents, etc. The extracted features are then fed into typical machine learning models for prediction. For example, Zhang et al. [16] found user-related features are effective on predicting whether a user would participates in a cascade, while Gao et al. [17] identified structural features and temporal features as more informative predictors for microblogs. However, feature-based methods heavily rely on experts' domain knowledge which makes the learned features hard to be generalized into different scenarios, e.g., a set of features extracted from tweets are usually ineffective for citation cascades.

Stochastic process based approaches assume that the underlying diffusion mechanism is known as a prior and then model the intensity function for information items' arrival times. For example, Lee et al. [18] predict the popularity of online content by borrowing the ideas from survival analysis. Shen et al. [19] proposed a generative probabilistic model to predict the popularity of scientific papers using a reinforced Poisson process, which models items' popularity with reinforcement mechanism, i.e., *rich-get-richer*. Cao et al. [20] transformed cascade into a set of diffusion paths – each of which depicts the process of information propagation for each participant within an observation time – and then model the cascades with self-exciting Hawkes point process. While providing enhanced model interpretability, these methods are network-agnostic and therefore fail to consider the implicit diffusion trajectories and the dynamics regarding cascade graphs that govern the success of popular information items.

Recently, the success of deep learning techniques has spurred a number of neural networks based information cascade models. DeepCas [9] is the first cascade graph representation learning method which models the structural and temporal information w.r.t. cascades using DeepWalk and GRUs, respectively. CasCN [2] samples a series of sequential

sub-cascades and adopts a dynamic multi-directional GCN to learn the structural information of cascades, the learned representations are then fed into LSTM for prediction.

Despite the promising results achieved through deep graph representation learning, existing cascade models are prone to catastrophic forgetting, i.e., training a cascade model with new data would interfere with previously learned knowledge [14]. This issue makes existing models difficult to adapt to evolved cascade graph data and prone to abrupt performance decrease on old tasks, unless storing all previously trained models. Recently, continual learning, also referred to as *lifelong learning* or *incremental learning*, has received increasing attention due to its ability of accommodating new knowledge and/or adapting the learned knowledge to the continuous input [21], [22]. Existing continual learning methods can be roughly categorized as three lines of works, i.e., experience replay based methods, regularization-based methods and parameter isolation based methods, cf. [21], [22] for comprehensive reviews. Our CICP model is build on elastic weight consolidation [23] which, therefore, can be considered as a regularization-based continual cascade learning model. In this vein, this paper provides the first continual information diffusion model that could largely alleviate the forgetting issue through generalizing the diffusion patterns progressively while still retaining the knowledge of previous episodes.

## III. Preliminaries

In this section, we introduce the necessary background of information cascade and formally define the continual cascade learning problem.

**Definition 1: Cascade Graph** – Given an information item $C_i$ and its corresponding cascade graph $\mathbf{G}_i$, which is defined as an evolving sequence of $N$ sub-graphs $\mathbf{G}_i = \{\mathcal{G}_i(t_0), \mathcal{G}_i(t_1), \ldots, \mathcal{G}_i(t_{N-1})\}$. Each sub-graph $\mathcal{G}_i(t_j)$ is composed by a 3-tuple $(\mathcal{V}_i^{t_j}, \mathcal{E}_i^{t_j}, t_j)$, where $\mathcal{V}_i^{t_j}$ and $\mathcal{E}_i^{t_j}$ are nodes and edges in graph $\mathcal{G}_j(t_j)$ added at time $t_j$.

Suppose node $v_i^{t_j}$ is the user who participates in $C_i$ at time $t_j$ and define $\mathcal{V}_i^{t_j} = \{v_i^{t_0}, v_i^{t_1}, \ldots, v_i^{t_j}\}$, we let the set of edges $\mathcal{E}_i^{t_j}$ represents the retweeting (or citing) relationships between nodes in $\mathcal{V}_i^{t_j}$. Following previous works [2], [11], we defined the prediction problem as a regression task, i.e., we aim to predict the numerical future popularity $P_i$ for cascade $C_i$ at time $t_p$ by observing a partial cascade graph $\mathbf{G}_i$ at time $t_o$.

**Definition 2: Information Cascade Popularity Prediction** – Given an information $C_i$ and its partial cascade graph $\mathbf{G}_i$, the *information popularity* $P_i$ is defined as $P_i = |\mathcal{V}_i^{t_p}| - |\mathcal{V}_i^{t_o}|$, where $t_o$ and $t_p$ are the observation time and the prediction time, respectively; and $|\mathcal{V}_i^*|$ denotes the size of the cascade graph, in terms of the number of nodes in $C_i$. Thus, our main objective is to learn a regression function $f : C_i \rightarrow P_i$ that maps cascade $C_i$ to its incremental popularity $P_i$.

Above problem definition is only applicable to learn a single prediction task. In this work, we consider the following continual cascade prediction problem.

**Definition 3: Continual Cascades Prediction** – Continual cascades prediction considers a sequential of tasks. Suppose we have three tasks: task A, B and C, our goal is to learn a model $f : C_i^1 \rightarrow P_i^1, C_i^2 \rightarrow P_i^2, C_i^{\mathcal{T}} \rightarrow P_i^{\mathcal{T}}$, where $C_i^*(* \in [1, \mathcal{T}])$ are different observations of $C_i$, and $P_i^*$ denote the different predicted results made by the model. In another words, the learned model $f$ should retain its performance on a sequence of $\mathcal{T}$ tasks.

## IV. Methodology

We now present the methodology of CICP for addressing continual cascade learning problem.

### A. Modeling Cascade Diffusion with GNN

We model the structure of cascades with graph neural networks, which has received considerable attention due to its capability of transforming, propagating and aggregating node features across the graph. Typical GNNs such as GCN [13] and GAT [24] are only applicable to *undirected* graphs, whereas information diffusion considered in this work requires modeling the *directed* propagation of the information. To this end, we use a directed GNN model suggested by [2] for modeling the structured cascades.

More specifically, for an observed cascade graph $\mathcal{G}_i(t_o)$, $A$ is the weighted adjacency matrix, $D$ is the diagonal degree matrix, then the Laplacian can be therefore determined as $L = D - A = U\Lambda U^T$, where $\Lambda = \text{diag}(\lambda_0, \lambda_1, \ldots, \lambda_{N-1})$ is the diagonal matrix of eigenvalues. When modeling the directional cascade graphs, we use the Laplacian of cascade $\Delta_c$, a.k.a. *CasLaplacian* [2], for modeling the convolution operation over a single cascade signal $X$ as:

$$y = g_\theta * \mathcal{G}X = \sum_{k=0}^{K} \theta_k T_k \left(\widetilde{\Delta}_c\right) X, \tag{1}$$

where $\widetilde{\Delta}_c = \frac{2}{\lambda_{max}}\Delta_c - I_N$ is defined as a scaled Laplacian, $T_k$ is the Chebyshev polynomial, $\theta_k$ denotes a vector of Chebyshev coefficients and $\lambda_{max}$ is the largest eigenvalue of Laplacian.

After obtaining the adjacency representation of sub-cascade graph sequence and corresponding Laplacian matrix $\Delta_c$, we can learn the structural and temporal patterns of the cascade $\mathbf{G}_i$ using any RNN models. Here, we employ an LSTM [15] to model the sequential cascades and use a multi-layer perceptron (MLP) to compute the incremental cascade popularity. Thus, the loss function for training a single cascade popularity task is defined as:

$$\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M}(\log_2 \tilde{P}_i - \log_2 P_i)^2, \tag{2}$$

where $M$ is the total number of cascades in training set, $P_i$ and $\tilde{P}_i$ are the ground-truth and the predicted future popularity of the cascade, respectively.

**Remarks.** Note that the above process of cascade learning is similar to our previous work CasCN [2], which is not the main contribution of this work. For completeness, we use CasCN as our building block for cascade modeling. However, the main architecture of CICP is model-agnostic, which means one can

easily replace CasCN with other cascade learning models, e.g., DeepCas [9], DeepHawkes [20], etc.

### B. Continual Cascade Learning

Above we have presented a GNN-based model for single cascade prediction task. However, as we will show in next section, CasCN, as well as other approaches, would suffer severe model forgetting issue. Suppose that we have two sequential cascade prediction tasks A and B, which are trained on a streaming observed cascades, i.e., the observation time for task B is later than that of task A. During training on A, the model ($\mathtt{M}_A$) parameters are optimized towards the minima of the loss using the observed cascade A. After observing new evolution of the cascade, the model $\mathtt{M}_B$ can be trained using the same method, but it would not adapt to the data for training task A anymore.

To overcome this issue, we present an adaptive training method based on elastic parameter consolidation [23] to preserve the important parameters for model $\mathtt{M}_A$ when training model $\mathtt{M}_B$. Intuitively, the contribution of each parameters is different for various tasks. If the parameters that are important for task A are changed greatly when training on task B, the performance of the new-trained model $\mathtt{M}_B$ would perform poorly on previous task A. Nevertheless, if we know the importance of parameters for task A and decrease the speed of change of the important parameters for task A, the catastrophic forgetting issue will be significantly alleviated.

Without loss of generality, the model parameters $\theta$ are optimized to be their most probable values given the cascade graph $\mathbf{G}$. The conditional probability $p(\theta|\mathbf{G})$ can be described by Bayes' rule:

$$\log p(\theta|\mathbf{G}) = \log p(\mathbf{G}|\theta) + \log p(\theta) - \log p(\mathbf{G}), \quad (3)$$

where $\mathbf{G} = \mathbf{G}_A + \mathbf{G}_B$ denotes the entire dataset. When training the subsequent task B after task A, the conditional probability of parameters $p(\theta|\mathbf{G})$ can be described as:

$$\log p(\theta|\mathbf{G}) = \log p(\mathbf{G}_B|\theta) + \log p(\theta|\mathbf{G}_A) - \log p(\mathbf{G}_B), \quad (4)$$

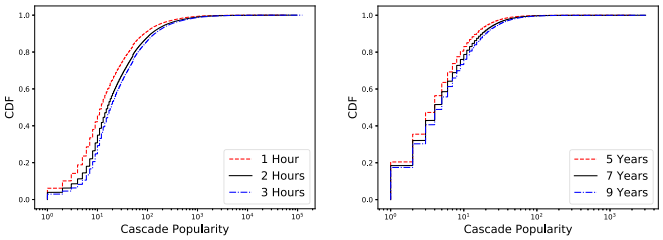
where $\log p(\mathbf{G}_B|\theta)$ represents the loss on task B. In order to consider the influence of task A, the important parameters $\theta^*_A$ for task A should be reflected in the posterior distribution $\log p(\theta|\mathbf{G}_A)$, which can be approximated by Gaussian distribution with averaged parameters $\theta^*_A$ as mean and a diagonal precision estimated by fisher information matrix (FIM) $F_A$ [23], [25]. Following [23], we approximate FIM $F_A$ using the empirical FIM to avoid additional backward pass. Hence, the importance weight $F_{A,i}$ of task A is defined as the squared gradient of loss $\mathcal{L}_A$ for a parameter $\theta_i$:

$$F_{A,i} = \mathbb{E}\left[\left(\frac{\partial \mathcal{L}_A}{\partial \theta_{A,i}}\right)^2\right]. \quad (5)$$

Note that $F_{A,i}$ is only calculated after training task A. Intuitively, parameters with higher value of $F_{A,i}$ have a large curvature in the parameter space. That is, such parameters are important for task A, which, therefore, should be strongly regularized for the next task learning.

TABLE I
STATISTICS OF TWO DATASETS

| Dataset | | Weibo | | | APS | | |
|---|---|---|---|---|---|---|---|
| Task | | A | B | C | A | B | C |
| Time | | 1 hour | 2 hours | 3 hours | 5 years | 7 years | 9 years |
| cascades | train | 2,649 | 3,198 | 3,500 | 2,394 | 3,052 | 3,500 |
| | val | 567 | 685 | 750 | 513 | 654 | 750 |
| | test | 567 | 685 | 750 | 513 | 654 | 750 |
| Avg. nodes | train | 58.39 | 68.08 | 73.78 | 17.52 | 19.44 | 20.59 |
| | val | 60.86 | 72.82 | 69.18 | 17.81 | 19.63 | 22.01 |
| | test | 60.69 | 64.26 | 74.21 | 18.75 | 19.30 | 20.52 |
| Avg. paths | train | 2.23 | 2.28 | 2.30 | 3.08 | 3.27 | 3.41 |
| | val | 2.23 | 2.27 | 2.29 | 3.09 | 3.29 | 3.47 |
| | test | 2.26 | 2.30 | 2.30 | 3.12 | 3.29 | 3.47 |




(a) Weibo (b) APS

Fig. 2. Cumulative Distribution Function (CDF) of cascade popularity.

After training on task A, we can compute the importance of each parameter for task A and specifically constraint the parameters with larger importance weight. While training on task B, the parameters will be updated towards the low place for task B, except for the parameters that have been given strong constraint – which should be changed slightly so that the model performance on task A is still maintained, so that the catastrophic forgetting will slow down, and the loss function for task B is defined as:

$$\mathcal{L}_\mathcal{B} = \frac{1}{M_B}\sum_{i=1}^{M_B}(\log \tilde{P_i^B} - \log P_i^B)^2 + \sum_i \frac{\lambda_A}{2} F_{A,i}(\theta_i - \theta^*_{A,i})^2 \quad (6)$$

where $\lambda$ is a hyper-parameter used to reconcile the importance of previous tasks, $M_B$ is the number of cascades in task B.

After training task B, this continual cascade learning would be proceeded on the subsequent tasks. In this way, the data and parameters for training previous models would not be accessed by later tasks, i.e., our model can significantly save the storage and memory cost that would be very expensive for mining the knowledge from intensive streaming data.

## V. EXPERIMENTS

We now evaluate the proposed CICP model against several state-of-the-art baselines which achieve outstanding performance on single task of cascade prediction.

**Datasets**. We conduct the experiments on two widely used benchmark cascade datasets – Weibo [20] and APS [19] –

TABLE II
PERFORMANCE COMPARISON BETWEEN CICP AND BASELINES ON CONTINUAL CASCADE LEARNING.

| Dataset | Weibo | | | APS | | |
|---|---|---|---|---|---|---|
| Task | A | A→B | A→B→C | A | A→B | A→B→C |
| Metric | AMSLE | AMSLE (APF) | AMSLE (APF) | AMSLE | AMSLE (APF) | AMSLE (APF) |
| Feature-based | 2.441 | 3.970 (1.240) | 4.670 (1.352) | 1.639 | 2.671 (1.160) | 3.380 (1.506) |
| DeepHawkes | 2.236 | 3.615 (1.093) | 4.551 (1.450) | 1.523 | 2.631 (1.489) | 3.739 (2.300) |
| CasCN | 1.950 | 3.366 (1.363) | 4.104 (1.785) | 1.403 | 2.098 (1.026) | 3.270 (2.083) |
| **CICP** | **1.913** | **2.621 (0.600)** | **3.012 (0.640)** | **1.375** | **2.090 (1.000)** | **2.902 (1.695)** |

for evaluating cascade popularity prediction models. **Weibo** is a Twitter-like social networking platform in China. The cascades in Weibo dataset are formed by tweets and their retweets. **APS** dataset contains scientific papers published by 17 American Physical Society journals. The cascades in APS dataset are formed by papers and their citation papers. For each dataset, we select 70% samples as the training set, and the remaining as validation set (15%) and test set (15%). The cumulative distribution function (CDF) of cascade popularity on two datasets are shown in Fig. 2.

To verify the effectiveness of continual cascade learning, we construct three tasks for each dataset, each of which are built by setting different observation times. In our experiments, we set three observation times in Weibo: 1 hour, 2 hours and 3 hours, obtaining three tasks (A, B and C) which represent three cascades of information diffusion. Similarly, we set three observation times in APS: 5 years, 7 years and 9 years. The details of the datasets are shown in Table I.

**Baselines**. To demonstrate the effectiveness of our proposed framework, we compare CICP with the following three different representative models. •**Feature-based**: Feature-based approaches extract many hand-crafted features from information items, such as the texts and images of tweets/papers, structural and temporal features of cascades, etc. Here we use following features: time between original and its first forwarding, time interval of each retweet, average and max path length of sequences, and the average time of the diffusion from the first node to the last node. We then feed these features into a two-layers MLP model for cascade training and prediction.

•**DeepHawkes** [20]: combines the advantages of stochastic processes and deep learning techniques for modeling cascades, which considers three factors from the view of Hawkes point process, i.e., user influences, self-exciting mechanism, and non-parametric time decay effects, and learns these parameters in a deep learning way.

•**CasCN** [2]: samples an information cascade graph as a sequence of sub-cascade graphs and jointly models both the directions of diffusion and the time of retweeting with graph convolutions and LSTM.

**Evaluation metric**. For a single task of information popularity prediction, we usually use the mean square logarithmic error (MSLE) to evaluate the model performance (cf. Eq. (2)). However, it is not suitable for evaluating continual cascades learning problem. Inspired by average accuracy and forgetting measure [26] for classification tasks, we define two metrics taking into account both the averaged task prediction performance and the forgetting degree on multiple tasks. We let $m_{k,j}$ denote the MSLE of the $k$-th task after incrementally training the model from tasks 1 to task $j$. The *average AMSLE* (*MSLE*) is defined as:

$$\text{AMSLE} = \frac{1}{j}\sum_{k=1}^{j} m_{k,j}, \tag{7}$$

where $j$ is the number of previously trained tasks. The *average percentage forgetting* (*APF*) measure depicts the relative extent of model forgetting, which is defined as:

$$\text{APF} = \sum_{k=1}^{j-1} \frac{(m_{k,j} - \min m_{k,l})}{\min m_{k,l}}, \tag{8}$$

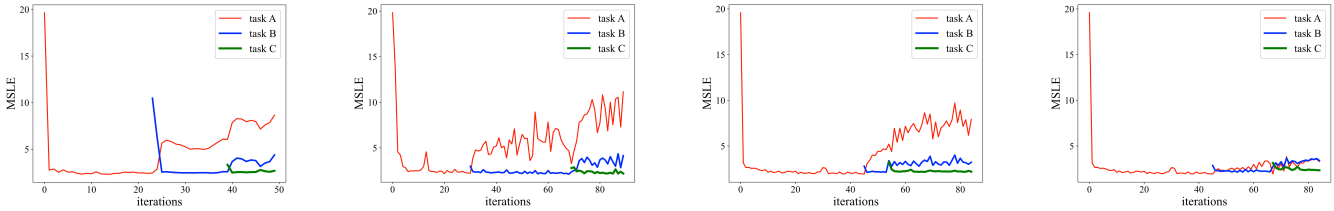
where $l \in [1, j]$. Apparently, the lower the value of APF, the better performance the model achieved on continual cascade learning.

### A. Experimental Observations

**Performance comparison.** Table II reports the overall performance evaluations of all methods on two datasets. We can clearly observe that our proposed CICP model achieves the smallest prediction error AMSLE on both datasets. Apart from the overall superiority of our model, we also have following findings.

Previous cascade modeling methods, both feature-based and deep learning-based, suffer from catastrophic forgetting issue. As the number of tasks increases, their average model performance significantly degrades due to the lack of mechanism for preserving the important model parameters on previous tasks. In contrast, CICP is capable of alleviating the forgetting issues and incrementally adapts the model performance on new coming observations.

Further, our model performs better on Weibo dataset than on APS dataset, which means the microblog platforms are more prone to model forgetting problem. This is because the more intensive observations received on Weibo cascades – e.g., the number of average nodes participated in the Weibo cascades is significantly higher than in the APS cascades. This result suggests that the continual cascade learning model proposed in this paper is, arguably, more suitable to data intensive scenarios.




(a) Feature-based (b) DeepHawkes (c) CasCN (d) CICP

Fig. 3. Convergence of models on Weibo dataset.

**Model convergence.** To better interpret the effectiveness of the proposed model, we plot the training process of CICP and baselines on three tasks, shown in Fig. 3. Clearly, the loss of CICP consistently decreases and converges to a lower value on all three tasks and are remarkably more robust to new tasks, i.e., CICP achieves slightly and "smooth" increase when confronting new tasks. This means our model successfully retains the prediction performance on previous methods while maintaining the performance on new coming observations. Nevertheless, all baseline models typically perform worse on previous tasks compared to CICP due to the problem of parameter forgetting, though they achieve comparable performance on the new tasks (e.g., task C) as CICP. Among the baselines, the performance of feature-based method is relatively stable which means the deep learning models are more vulnerable to model parameter forgetting. This result explains the behavior of our model and indicates that CICP can well adapt to continuous incoming participants. In this spirit, it gives intuitive explanations of the motivation of this work, i.e., we should pay more attention to continual information cascade modeling and predicting rather than simply improving the performance of single cascade task learning.

## VI. Conclusions and Future Works

In this work, we formulated a novel learning problem *continual cascade learning* which requires incremental adaptation to new tasks without significant performance degradation on old tasks. We present a novel model CICP for addressing this problem which is capable of estimating and preserving important parameters in learned tasks and adapting to new tasks without sacrificing too much performance on previous ones. We evaluate the proposed model on real-world cascades and the results proves the effectiveness of our model compared to the baseline approaches. We hope this work can be used as a stepping-stone for inspiring more insightful future works on continual cascade learning.

## Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant No.61602097), and NSF grants CNS 1646107.

## References

[1] C. Yang, M. Sun, H. Liu, S. Han, Z. Liu, and H. Luan, "Neural diffusion model for microscopic cascade study," *TKDE*, 2019.

[2] X. Chen, F. Zhou, K. Zhang, G. Trajcevski, T. Zhong, and F. Zhang, "Information diffusion prediction via recurrent cascades convolution," in *ICDE*. IEEE, 2019, pp. 770–781.

[3] X. Gao, Z. Cao, S. Li, B. Yao, G. Chen, and S. Tang, "Taxonomy and evaluation for microblog popularity prediction," *TKDD*, vol. 13, no. 2, p. 15, 2019.

[4] F. Zhou, X. Xu, G. Trajcevski, and K. Zhang, "A survey of information cascade analysis: Models, predictions and recent advances," *arXiv:2005.11041*, pp. 1–41, 2020.

[5] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *ACM Sigmod Record*, vol. 42, no. 2, pp. 17–28, 2013.

[6] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," *Communications of the ACM*, vol. 53, no. 8, pp. 80–88, 2010.

[7] 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 *KDD*, 2015, pp. 1513–1522.

[8] 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 *WWW*, 2017, pp. 735–744.

[9] C. Li, J. Ma, X. Guo, and Q. Mei, "Deepcas: An end-to-end predictor of information cascades," in *WWW*, 2017, pp. 577–586.

[10] X. Chen, K. Zhang, F. Zhou, G. Trajcevski, T. Zhong, and F. Zhang, "Information cascades modeling via deep multi-task learning," in *SIGIR*, 2019, pp. 885–888.

[11] F. Zhou, X. Xu, K. Zhang, G. Trajcevski, and T. Zhong, "Variational information diffusion for probabilistic cascades prediction," in *INFOCOM, in press*, 2020, pp. 1–10.

[12] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *KDD*, 2014, pp. 701–710.

[13] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*, 2017.

[14] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*. Elsevier, 1989, vol. 24, pp. 109–165.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory." *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] Q. Zhang, Y. Gong, J. Wu, H. Huang, and X. Huang, "Retweet prediction with attention-based deep neural network," in *CIKM*, 2016, pp. 75–84.

[17] S. Gao, J. Ma, and Z. Chen, "Effective and effortless features for popularity prediction in microblogging network," in *WWW*, 2014, pp. 269–270.

[18] J. G. Lee, S. Moon, and K. Salamatian, "Modeling and predicting the popularity of online contents with cox proportional hazard regression model," *Neurocomputing*, vol. 76, no. 1, pp. 134–145, 2012.

[19] H. Shen, D. Wang, C. Song, and A.-L. Barabási, "Modeling and predicting popularity dynamics via reinforced poisson processes," in *AAAI*, 2014.

[20] Q. Cao, H. Shen, K. Cen, W. Ouyang, and X. Cheng, "Deephawkes: Bridging the gap between prediction and understanding of information cascades," in *CIKM*, 2017, pp. 1149–1158.

[21] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "Continual learning: A comparative study on how to defy forgetting in classification tasks," *arXiv:1909.08383*, 2019.

[22] A. Chaudhry, M. Rohrbach, M. Elhoseiny, T. Ajanthan, P. K. Dokania, P. H. Torr, and M. Ranzato, "Continual learning with tiny episodic memories," *arXiv:1902.10486*, 2019.

[23] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *PNAS*, vol. 114, no. 13, pp. 3521–3526, 2017.

[24] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *ICLR*, 2018.

[25] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, 2019.

[26] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *ECCV*, 2018, pp. 532–547.