GraphInf: A GCN-based Popularity Prediction System for Short Video Networks

Yuchao Zhang[1], Pengmiao Li[1], Zhili Zhang[2], Chaorui Zhang[3], Wendong Wang[1], Yishuang Ning[4,5] and Bo Lian[6]

[1]Beijing University of Posts and Telecommunications, Beijing, China
[2]University of Minnesota, Minneapolis, US.
[3]Huawei Company, Hong Kong, China
[4]National Engineering Research Center for Supporting Software of Enterprise Internet Services, Shenzhen 518057, China
[5]Kingdee International Software Group Company Limited, Shenzhen 518057, China
[6]Kuaishou Company, Beijing, China

Abstract. As the emerging entertainment applications, short video platforms, such as Youtube, Kuaishou, quickly dominant the Internet multimedia traffic. The caching problem will surely provide a great reference to network management (e.g., traffic engineering, content delivery). The key to cache is to make precise popularity prediction. However, different from traditional multimedia applications, short video network exposes unique characteristics on popularity prediction due to the explosive video quantity and the mutual impact among these countless videos, making the state-of-the-art solutions invalid. In this paper, we first give an in-depth analysis on 105,231,883 real traces of 12,089,887 videos from *Kuaishou* Company, to disclose the characteristics of short video network. We then propose a graph convolutional neural-based video popularity prediction algorithm called *GraphInf*. In particular, *GraphInf* clusters the countless short videos by region and formulates the problem in a graph-based way, thus addressing the explosive quantity problem. *GraphInf* further models the influence among these regions with a customized graph convolutional neural (GCN) network, to capture video impact. Experimental results show that *GraphInf* outperforms the traditional Graph-based methods by 44.7%. We believe such GCN-based popularity prediction would give a strong reference to related areas.

1 Introduction

In recent years, online short video (or micro-video) platforms are emerging as a new trend to satisfy the fast-paced modern society. They have been widely spreading all over the world, making video traffic dominate the Internet traffic. As of 2019, there are

The work was supported in part by the National Natural Science Foundation of China (NSFC) Youth Science Foundation under Grant 61802024, the Fundamental Research Funds for the Central Universities under Grant 24820202020RC36, the National Key R&D Program of China under Grant 2019YFB1802603, and the CCF-Tencent Rhinoceros Creative Fund under Grant S2019202.

over 200 million active users in Kuaishou and more than 27 million short videos are being uploaded and viewed, on a daily basis [9].

Video popularity prediction has long been considered as an important topic in Internet traffic area, because it can provide a basis for many network management problems such as caching policies [12, 14], reducing the required memory size [29], and modeling videos' lifecycle [28].Existing popularity prediction algorithms [2, 13, 20, 22, 24, 25] work well in traditional multimedia scenario, but they become invalid in short video network due to the following two characteristics.

- **Explosive video quantity.** Kuaishou [9] produced more than 5 billion short videos in half a year in 2019, nearly 3,445,900 times more than the total number of TV series (about 351) and films (less than 1100) [27].
- **Relationship among videos.** Online social networks [4] and user behaviour [3, 15] play important roles in video popularity, making hot topics propagate from one region to another. Such effects become more apparent in short video network due to its strong social interaction and high timeliness.

Several pioneer efforts have been invested to the short video network prediction problem. [17] uses the average watched percentage of videos to predict the popularity, but becomes inefficient in large scale short video network. [12] takes video content into consideration to predict video popularity, but without considering video relationship, it becomes invalid in short video network.With further research, we made a simple comparison of short videos in different regions. For example, there were 5,755 same videos between Henan in the first 20 minutes with Anhui in the last 20 minutes. This shows that videos in different regions influence each other.

In this paper, we propose *GraphInf*, a popularity prediction system in short video network, which is based on a novel customized graph convolutional neural algorithm. *GraphInf* is a highly scalable system that clusters the massive videos into corresponding regions and formulates them by a simple GCN network. The main contributions of this paper are summarized below:

- We disclose the unique characteristics and challenges in short video network by analyzing real data from industry (Section 3.1).
- We present a system called *GraphInf* to address the popularity prediction problem in short video network (Section 4).
- We demonstrate the practical benefits of *GraphInf* by building a prototype, and the results also reveal some useful experiences/lessons that would be instructive to related topics. (Section 5).

The remainder of this paper is organized as follows. Section 2 briefly reviews the state-of-the-art efforts related to popularity prediction in short video network. Section 3 introduces the background and motivation of the proposed problem. Section 4 presents the framework of *GraphInf*, with detailed design. Section 5 demonstrates the setting up of *GraphInf* prototype and shows extensive experiment results from real data evaluations. Finally, Section 6 concludes this work.

2 Related Work

This work relates to a few areas of active research. We structure our discussion along two parts: the specific characteristics of short video network and video popularity prediction.

2.1 Short Video Network

The specific characteristics of short video network, like large scale and the influence between regions, have not been fully considered yet.

Explosive video quantity. It is well known that the number of short videos is huge and the growth rate is quite high. This raises the challenge as to how to cache these videos in limited storage to guarantee the cache hit rate [29]. To reduce the latency due to intermediate caching, [11] proposed a distributed resilient caching algorithm (DR-Cache) that is simple and adaptive to network failures. [26] designed a data store VStore for analytics on large videos. VStore selects video formats catering so as to achieve the target accuracy. [5] designed a parallel processing framework Streaming Video Engine that specially designed to solve the scalability problem, and showed the results of some use cases on data ingestion, parallel processing and overload control.

Relationship among videos. The influence between regions depth of short videos is surprising, due to the impact from user behaviour, online social networks, geo-distributed hot events, etc [7]. [15] proposed an end-to-end framework, DeepInf, which takes user's local network as the input for capturing the latent social representation. [3] discussed the popularity dynamics of videos dynamics of videos in Video Sharing Sites (VSSes) focusing on views, ratings and comments so as to build a emulator which replicates users behaviours in Online Social networks (OSN). These researches focus on the influences between regions model, but didn't make popularity prediction.

In brief, the literature above highlights the special characteristics of short video network that are large scale and the influence between regions.

2.2 Popularity Prediction

Significant efforts have been devoted to exploring item popularity prediction due to the potencial business value [22]. [24] provided the affect the popularity of science communication videos on YouTube. They found that the user-generated contents were significantly more popular than the professionally generated ones and that videos that had consistent science communicators were more popular than those without a regular communicator. [13] proposed LARM that is empowered by a lifetime metric that is both predictable via early-accessible features and adaptable to different observation intervals, as well as a set of specialized regression models to handle different classes of videos with different lifetime. [19] used support vector regression with Gaussian radial basis functions to predict the popularity of an online video measured by its number of views. Although these algorithms have their own advantages, they lack the high performance of operating speed with the rapid growth of the number of short videos.

Overall, we propose a framework *GraphInf* to explore the video popularity in short video network, by taking the special characteristics above into consideration.

3 Motivation

We start by providing some background knowledge of short video network. We point out the uniqueness of such kind of networks by comparing it with traditional online video networks (Section 3.1). In particular, by analyzing real statistical figures and access traces from *Kuaishou*, a popular short video platform in China, we disclose its two challenges, the explosive video quantity and the complex relationship among short videos.

We then show the opportunity of solving these two challenges above by mapping the massive short videos into regions and formulating the origin problem into a graph-based problem (Section 3.2). The graph structure property motivates us to design the proposed *GraphInf*.

3.1 Characteristics of Short Video Network

In this subsection, we show the differences between traditional online video network and short video network. The differences clearly disclose the unique characteristics (also challenges) of popularity prediction problem in short video network.

Explosive video quantity. We introduce the explosive video quantity in two aspects, from the perspective of videos and users, respectively.

- *From the perspective of videos.*

Length of video uploaded: In 2018, the length of all the uploaded traditional online videos is about 100 thousands of minutes, while that is 340 millions of minutes in only one short video platform (*Kuaishou*), which is 2,850 times longer [8].

Videos viewed: In 2018, there are only 46 million views per day in top 10 traditional online video platforms in China [10], while the number is 10 billion in only one short video platform (Toutiao), which is 217 times more than the traditional online videos [18].

- *From the perspective of users.*

Growth rate of usage time: According to QuestMobile's report [16], app usage time of short video grew 521.8% in 2018, while the usage time of online video dropped 12%.

Relationship among videos. To show the video relationship from a macro perspective, we draw the effect matrix between each pair of provinces from *Kuaishou*'s real traces. To quantify the impact from province i to province j, we take the number of overlapped videos from province i (in current time slot) and province j (in the next time slot). We show the normalized value in Figure 1. In the figure, each small cube represents the effect depth from province i to province j. Taking Shanxi province as an example, from the horizontal axis, the column of cubes represents the video effect from Shanxi Province to each of the other provinces, while from the ordinate axis, the row of cubes represents the video effect from each of the other provinces to Shanxi Province. We

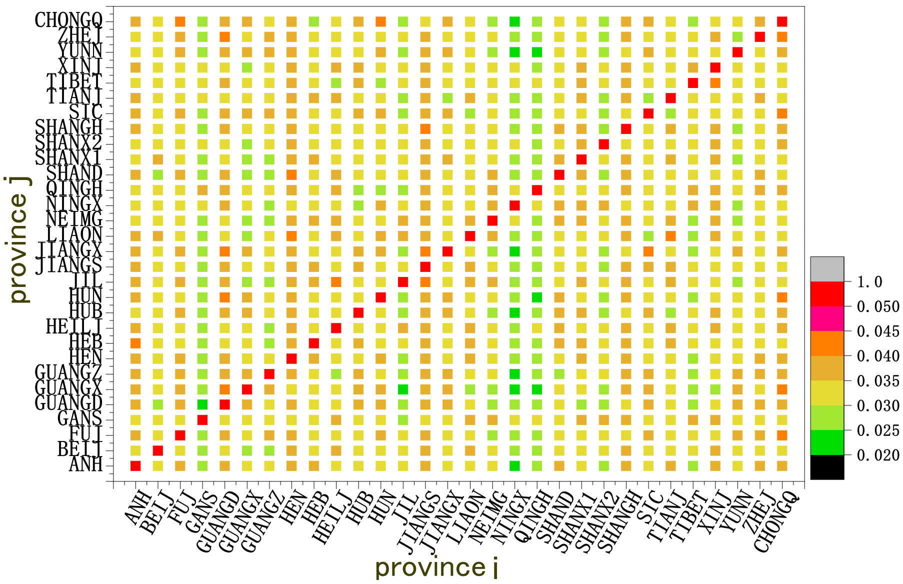

Fig. 1: Popularity effect between any pair of the 31 provinces.

can find that there is influence between different regions, and the influence between different regions is different. What's more, the effect matrix is time-varying due to the timeliness of short video network.

Traditional popularity prediction approaches are usually based on long-term historical access pattern or fixed relationship, which can not reflect the time-varying impact, and therefore become invalid in short video popularity prediction.

3.2 Potential of graph-based scheme

The characteristics above motivate the need for a lightweight prediction approach which can work under huge amount of short videos with influence between regions.

It is known that such relationship among short videos is often modeled by graphs [6, 21, 23]. Inspired by their success, we model the popularity prediction problem into a graph-based network. Instead of considering each video as a node, we design a geo-distributed clustering scheme to reduce the size of the graph. In particular, we cluster all the short videos into several geo-distributed regions (e.g., provinces/states), and formulate each region as a node in the graph (Figure 2 shows a simple example). Such graph-based clustering enjoys two significant advantages when predicting video popularity, and therefore has the potential to solve the above two challenges:

– Reduce the calculation scale. Due to the massive amount, it is impractical to predict the popularity for each short video, while such clustering significantly reduces the calculation scale to the number of regions. Therefore, *GraphInf* could handle much more short videos and keep the calculation overhead unchanged.

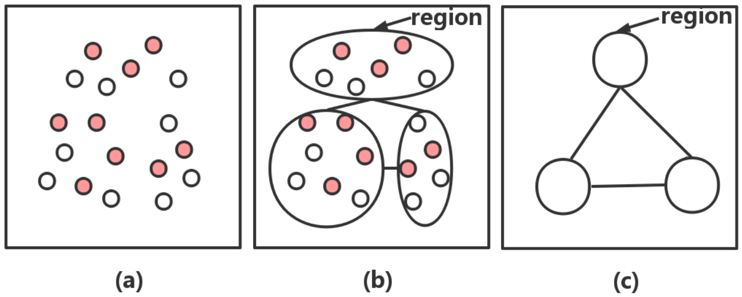

Fig. 2: To predict top n popular videos (red) from N videos (red and white). (a) Predict the popularity for each video. (b) Group the N videos to K regions, and predict top n_i videos in region k_i ($\sum_1^K n_i = N$). (c) Analyze these regions that constitute a simplified graph.

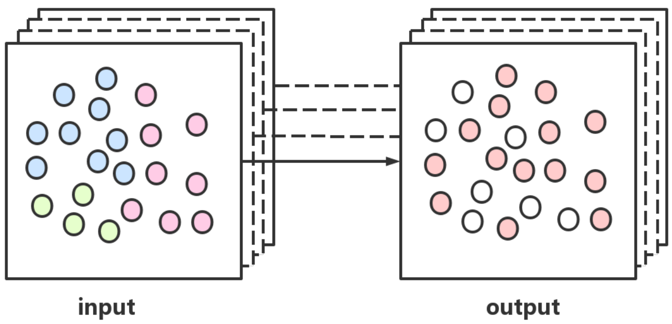

Fig. 3: The input and output of a *GraphInf*

– Get the relationship among videos in different regions. As *GraphInf* formulates the problem into a simple graph network, it thus can get the influence between the short videos (Figure 2(a)), by updating the attributes of nodes and links.

Overall, in such graph-based formulation, we can model a GCN network which takes all the potential videos as input and predicts the future video popularity, as shown in Figure 3. Base on the analysis and observations above, the question becomes that how to achieve short video popularity prediction on a graph, with explosive video quantity and time-varying video influence? We therefore design *GraphInf*, which will be introduced in the next section in detail.

4 *GraphInf* Framework

In this section, we first introduce how to formulate the popularity prediction problem of short videos in a graph-based way in Subsection 4.1. Then, we propose *GraphInf* to deal with the challenges introduced in Section 3.1, and describe the components of *GraphInf* in Subsection 4.2. The notations to be used are shown in Table 1.

4.1 Problem formulation

Generally, different places have their own popular short videos. The popular short videos of each place are mainly dominated by its local information (e.g., human behavior, age structure) and neighbors. In particular, we distinguish the places by administrative regions, where each region could be a city, a province, or a state. By clustering and analyzing all short videos in the region, we are able to handle the large scale problem of the growing short videos. Then, the connections among all regions are described by a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. Each region is denoted by a node and $\mathcal{N}$ is the set of all nodes. We assume that every two regions are virtually adjacent by using an edge with a weight that represents their relationship. All edges are included in the set $\mathcal{E}$. Next, we formally introduce the node and edge information used here.

Node information At each time slice t, every region has its own popular short videos. We divide the short video source into two types of popularity according to request times. One is the popular short videos (PV_{self}) whose request times are over hundreds or even thousands. The other is the sub-popular short videos (PV_{sub}) with a smaller number of request times. Let s_i^t and c_i^t denote the types of the popular and sub-popular short video of region i at time slice t, respectively. Specifically, s_i^t and c_i^t are two vectors with the dimension M, which consists of the request times of short videos. The number M indicates the top M short videos of each type.

Edge information Except the geographically adjacent information between two regions, we think that region i is affected by the top short videos of all other regions. Thus, we use the top short video information of all other regions (PV_{other}) as the edge information of region i. Let o_i^t denote the edge information of region i at time slice t. Mathematically, the representation of o_i^t is

$$o_i^t = [s_1^t; ...; s_{i-1}^t; s_{i+1}^t; ...; s_{\mathcal{N}}^t]$$

with dimension $(|\mathcal{N}| - 1) \times M$.

When given the node and edge information of previous $l \geq 1$ time slices, we aim to predict the popularity in the next time slice $t+1$ in each region. This problem is formally described as below.

$$\begin{aligned}\{s_i^{t+1}; c_i^{t+1}; o_i^{t+1}\} = f(\{s_i^t; c_i^t; o_i^t\}, \{s_i^{t-1}; c_i^{t-1}; o_i^{t-1}\}, ..., \\ \{s_i^{t-l}; c_i^{t-l}; o_i^{t-l}\}),\end{aligned} \tag{1}$$

Table 1: Notations

Symbol	Definition
s_i^t	The popular videos of region i at time slice t
c_i^t	The sub-popular videos of region i at time slice t
o_i^t	The top short videos of all other region of region i at time slice t
$w_{i\leftarrow j}^t$	The number of the same short videos appearing in both region j at time slice $t-1$ and region i at time slice t
W_i^t	$W_i^t = [w_{i\leftarrow 1}^t, ..., w_{i\leftarrow(i-1)}^t, w_{i\leftarrow(i+1)}^t, ..., w_{i\leftarrow \mathcal{N}}^t]$
X_i	The importance vector of each feature at region i, $X_i = [x_i^1, x_i^2, x_i^3]$
R_i^{t+1}	The predicted popular videos of region i at time slice $t+1$
Y_i^{t+1}	The ground truth indicating that the short videos of region i are popular at time slice $t+1$

where $f(\cdot)$ is the popularity prediction function of short videos. The difficulty of solving problem (1) lies in how to obtain an appropriate graph representation of the effect of hot topic between regions and then use it to overcome the relationship among videos in different regions to get popularity videos in explosive video quantity. For this, we propose *GraphInf* to solve this problem.

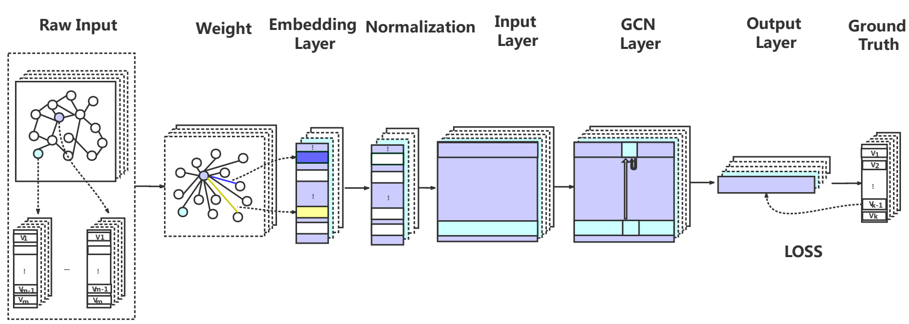

Fig. 4: The architecture of *GraphInf*.

4.2 *GraphInf*: A novel Graph network

GraphInf is tailored to deal with the challenges mentioned above in short videos. In particular, the architecture of *GraphInf* given in Figure 4 includes five layers: embedding layer, normalization layer, input layer, GCN layer, and output layer. Initially, we collect the raw data of node and edge information of all nodes at time slice t, i.e., $\{s_i^t; c_i^t; o_i^t\}, \forall i \in \mathcal{N}$. Next, we describe the five steps in turn.

– Embedding layer. The edge information of region i adopted here is the top short videos of all other regions. To quantify the effect of region j to region i at time slice t, we calculate the weight $w^t_{i \leftarrow j}$ to denote the number of the same short videos appearing in both region j at time slice $t-1$ and region i at time slice t. Thus, we use vector W^t_i denoting all weights of all other nodes, i.e.,

$$W^t_i = [w^t_{i \leftarrow 1}, ..., w^t_{i \leftarrow (i-1)}, w^t_{i \leftarrow (i+1)}, ..., w^t_{i \leftarrow \mathcal{N}}]$$

with dimension $|\mathcal{N}| - 1$. (Line 3 in Algorithm 1)

– Normalization layer. After the embedding step, the raw data of region i at time slice t turns to

$$H^t_i = \begin{bmatrix} s^t_i \\ c^t_i \\ W^t_i o^t_i \end{bmatrix}. \tag{2}$$

The entries in H^t_i are the request times of corresponding short videos. To eliminate the magnitude impact, we carry out the normalization of each entry h of H^t_i following equation (3). (Line 4 in Algorithm 1)

$$h' = \frac{h - \min(H^t_i)}{\max(H^t_i) - \min(H^t_i)}. \tag{3}$$

– Input layer. In the input layer, we define the normalized term $Normalization(H^t_i)$ as $\hat{H}^t_i$ to be the new input of region i at time slice t. (Line 5 in Algorithm 1)

– GCN layer. We customized GCN layer that independent of the original GCN. The overall operation of the GCN unit for region i follows the next three steps.

$$U^t_i = X_i \cdot \hat{H}^t_i, \tag{4}$$

$$\tilde{U}^t_i = \text{Sort}(U^t_i), \tag{5}$$

$$R^{t+1}_{i,m} = \begin{cases} 1 & \text{if } U^t_{i,m} \geq \tilde{U}^t_{i,k} \\ 0 & \text{otherwise} \end{cases}, \tag{6}$$

Each row of $\hat{H}^t_i$ denotes a pre-defined feature, say the popular, sub-popular, and all other popular short videos of region i. We adopt GCN to learn the importance of each feature at region i. Let $X_i = [x^1_i, x^2_i, x^3_i]$ denote the importance of each feature at region i. $R^{t+1}_{i,m}$ is the predicted outcome and indicates whether the mTH video is a hot video in all short videos, 1 if it is, else 0 ,in region i at time $t+1$. Where k represents a number of hot Videos defined.We use the GCN layer to find the hot videos in the mount of short videos, to deal with the explosive video quantity challenge.(Line 6-8 in Algorithm 1)

– Output layer. In the training process, when we obtain the output from the GCN layer, we have to evaluate the loss between the training result and ground truth so as to update all $X_i, \forall i \in \mathcal{N}$. We define the loss function of each region as follows.

$$loss = \frac{|U^t_i (R^{t+1}_i)^T - U^t_i (Y^{t+1}_i)^T|^2}{U^t_i (Y^{t+1}_i)^T}, \tag{7}$$

Algorithm 1 The pesudo code of *GraphInf*

input: The time sequence training dataset $\mathcal{D}^{train}$ including the video request times $\{s_i^t, c_i^t, o_i^t\}, t \in \mathcal{D}^{train}$ (*Here, the data of the next time slice* $t+1$ *is used as the ground truth* Y_i^{t+1}); the initialization $X_i(0)$.

1: **for** t **do**
2:   **for** $i \in \mathcal{N}$ **do**
3:     Calculate W_i^t
4:     Calculate H_i^t
5:     $\hat{H}_i^t = Normalization(H_i^t)$
6:     $U_i^t \leftarrow X_i^t \cdot \hat{H}_i^t$
7:     $U_{i(K)}^t \leftarrow$ The K^{th} value in Sort(U_i^t)
8:     $R_i^{t+1} \leftarrow$ Find the values in U_i^t that are larger than $U_{i,K}^t$ and denote them as one
9:     Calculate the loss compared with the ground truth Y_i^{t+1} by using equation (7)
10:     **if** $loss^i - loss^{i-1} \leq \epsilon$ **then**
11:       Terminate the algorithm
12:     **end if**
13:     Return $X_i = X_i(t)$ and R_i^{t+1}
14:   **end for**
15: **end for**

output: The importance of features $X_i = [x_i^1, x_i^2, x_i^3]$ and the prediction popular videos R_i^{t+1}

where Y_i^{t+1} is the ground truth indicating that the K short videos of region i are popular at time slice $t+1$. If yes, the corresponding value in Y_i^{t+1} is set to be 1, otherwise 0. The superscript T is the operation of matrix transpose. (Line 9 in Algorithm 1)

At last, we summarize *GraphInf* in Algorithm 1.

5 Evaluation

In this section, we evaluate our approach *GraphInf* using real traces, and show the results of applying *GraphInf* versus the existing representative policies.

5.1 Experiment setting.

Algorithms We compare *GraphInf* with three representative solutions.

– RNN-based. As popular short videos of each place are mainly dominated by its local information, there are some works that use historical data to predict item popularity, by designing a recurrent network with memory [14]. So we use the historical hot videos to illustrate the potential of RNN-based schemes.
– Graph-based. As we described in the Section 3.2, some efficient solutions are modeled by graphs [6, 21], so we use the historical sub-hot videos, as comparison of *GraphInf*.
– Embedding. In order to improve the topic prediction accuracy, some works embed specific characteristics of graph node into consideration [1] and achieve more desirable results. We therefore further implement an embedding method as comparison.

Table 2: Information about the dataset.

Dataset	Cache Size	Access #.	Server #.	Video #.
Province 1	$3.96T$	$5,323,508$	30	$1,424,564$
Province 2	$5.12T$	$9,202,618$	72	$1,813,058$
Province 3	$2.51T$	$3,051,059$	10	$876,058$
Province 4	$2.39T$	$2,765,419$	21	$843,219$
Province 5	$2.48T$	$2,828,645$	6	$862,806$
...	...	...	...	...
Total	$78.75T$	$105,231,883$	488	$12,089,887$

Datasets The traces [29] are from 31 provinces with 1,128,989 accesses to 132,722 videos in 1 hours (shown in Table 2). Each trace item contains the timestamp, anonymized source IP, video ID and url, file size, location, server ID, cache status, and consumed time (with no personal sensitive information). We then deploy and evaluate *GraphInf* by comparing with representative algorithms.

5.2 System Performance

We first conduct a series of experiments to show the overall prediction performance. In particular, we show the popular video prediction accuracy in each region, and then deeply look into these popular videos by analyzing their source and the prediction accuracy correspondingly.

Overall Accuracy As described in problem formulation section (Section 4.1), we consider both node information (popular videos PV_{self} and sub popular videos PV_{sub}) and edge information (popular videos from other regions PV_{other}). Here we divide them into: the top popular 300 videos from the same region, the top 301 to 600 videos from the same region, and the top popular 300 videos from other 30 regions, respectively.We use this data as input for *GraphInf*, and the output is a matrix of hot videos.

Figure 5 shows the average prediction accuracy in 19 minutes of all the provinces in China. Figure 6 and 7 show the prediction accuracy comparison of ShanXi and Tibet province in 19 minutes, respectively. The reason why we choose these two provinces is that ShanXi is a populous province with 10 times the population but only 12% area compared with the sparse Tibet (about 124 times the population density). From these three figures, we can see that *GraphInf* exceeds the other three methods in a populous province (6). Surprisingly, we thought it would be easier to predict hot topics in the sparse provinces because its data size is relatively small and the topology is also simpler, but the results show that the accuracy in Tibet is unexpectedly low. In order to figure this out, we further conduct experiments to calculate the specific accuracy by video source (PV_{self}, PV_{sub} and PV_{other}).

Source accuracy To analyze the power of *GraphInf* in detail, here we differentiate these three video sources and check the source accuracy separately. So in each ex-

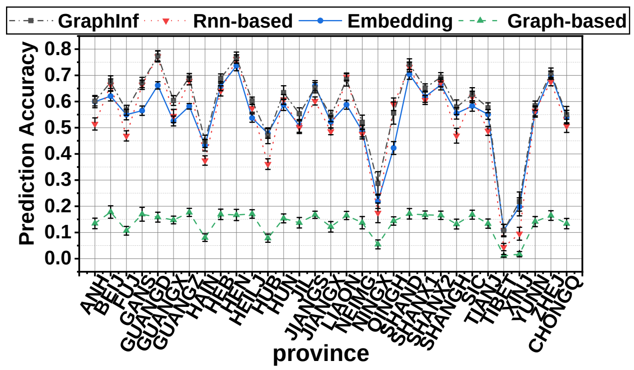

Fig. 5: Prediction accuracy of the four algorithms, in all the 31 cities.

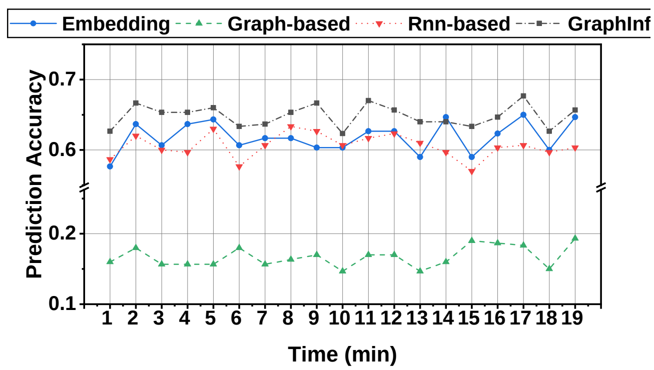

Fig. 6: Prediction accuracy comparison in ShanXi Province, during 19 minutes.

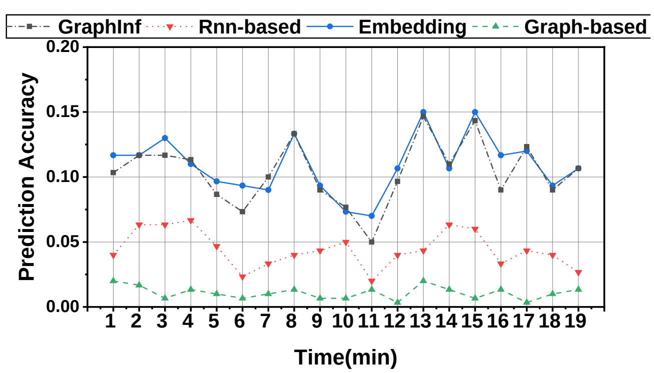

Fig. 7: Prediction accuracy comparison in Tibet Province, during 14 minutes.

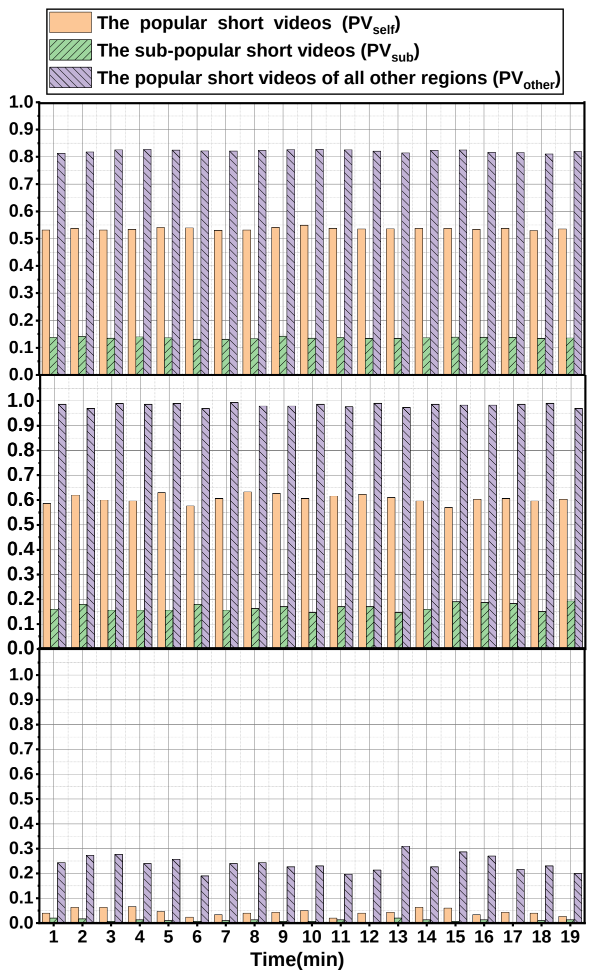

Fig. 8: [Sources accuracy] The sources accuracy w.r.t. different video sources (PV_{self}, PV_{sub} and PV_{other}). Up: the average accuracy of 31 cities. Middle: accuracy of ShanXi Province. Down: acuuracy of Tibet Province.

periment results, there are three bars, denoting the source accuracy from video source PV_{self}, PV_{sub}, and PV_{other}, respectively. The results are shown in Figure 8. There are more than 2,500 videos (1 minutes) to get 300 hot videos. The up 19 series of experiments illustrate the average source accuracy of all the 31 cities (in 19 minutes), the middle 19 series of experiments show the results of ShanXi province, and the down 19 series of experiments show the results of Tibet province. From these results we can see that Tibet is lower (any bar) than the average (in the left 19 series experiments) , but ShanXi is higher than the average. So, we can get a result that we obtain a high accuracy of data sources as input in popular province, and a low accuracy in sparse province. The reason for this phenomenon is that it is hard to formulate the video from those sparse provinces, mainly due to the relatively strong closure. Therefore, compared with other cities, the prediction accuracy decreases significantly.

To summarize, the relationship between hot videos in different regions will have an impact on future popularity. *GraphInf* uses a graph to model different regions so as to extract the influence of hot videos in these regions, that's why *GraphInf* can get higher prediction accuracy when compared with existing algorithms. The prediction accuracy of our algorithm will be further improved in the conditions with strong population mobility or close region relationship..

6 Conclusion

In this work, we shed light on the popularity prediction problem in short video network. We first disclose the specific characteristics of such network from online application data, i.e., explosive video quantity and relationship among videos. Based on the observations, we formulate this problem to a graph and propose a graph-based neural network by incorporating network embedding, normalization and graph convolution. We evaluate the proposed *GraphInf* using real online traces from *Kuaishou* Company and compare the results with three state-of-the-art methods. Experimental results show that *GraphInf* significantly outperforms the baselines with precise topic prediction in short video network.

References

1. Battaglia, P.W., Hamrick, J.B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al.: Relational inductive biases, deep learning, and graph networks. arXiv preprint arXiv:1806.01261 (2018)
2. Chen, J., Song, X., Nie, L., Wang, X., Zhang, H., Chua, T.S.: Micro tells macro: Predicting the popularity of micro-videos via a transductive model. In: Proceedings of the 24th ACM international conference on Multimedia, ACM (2016) 898–907
3. Ghosh, S., Kumar, S.: Video popularity distribution and propagation in social networks. Int. J. Emerg. Trends Technol. Comput. Sci.(IJETTCS) **6**(1) (2017) 001–005
4. Huang, C., Li, J., Ross, K.W.: Can internet video-on-demand be profitable? In: ACM SIGCOMM Computer Communication Review. Volume 37., ACM (2007) 133–144
5. Huang, Q., Liang, C., Talwar, M., Mathur, A., Kulkarni, S., Burke, M., Lloyd, W., Ang, P., Knowles, P., Nykiel, T., Tverdokhlib, I., Yajurvedi, A., Dapolito, P., Yan, X., Bykov, M.: Sve: Distributed video processing at facebook scale. (10 2017) 87–103

6. Hussein, N., Gavves, E., Smeulders, A.W.: Videograph: Recognizing minutes-long human activities in videos. arXiv preprint arXiv:1905.05143 (2019)
7. Jampani, V., Gadde, R., Gehler, P.V.: Video propagation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 451–461
8. Jianshu: Analysis report of tiktok. https://www.jianshu.com/p/2097f2dda7b0 (2018)
9. kuaishou: Kuaishou. https://www.kuaishou.com (2019)
10. lbzuo: The top 10 popular video websites in china. http://www.lbzuo.com/ziyuan/show-15766.html (2018)
11. Li, J., Phan, T.K., Chai, W.K., Tuncer, D., Rio, M.: Dr-cache: Distributed resilient caching with latency guarantees. In: IEEE INFOCOM. (2018)
12. Li, S., Xu, J., Van Der Schaar, M., Li, W.: Popularity-driven content caching. In: IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE (2016) 1–9
13. Ma, C., Yan, Z., Chen, C.W.: Larm:a lifetime aware regression model for predicting youtube video popularity. In: Proceedings of the 2017 ACM on CIKM, ACM (2017) 467–476
14. Narayanan, A., Verma, S., Ramadan, E., Babaie, P., Zhang, Z.L.: Deepcache: A deep learning based framework for content caching. In: Proceedings of the 2018 Workshop on Network Meets AI & ML, ACM (2018) 48–53
15. Qiu, J., Tang, J., Ma, H., Dong, Y., Wang, K., Tang, J.: Deepinf: Social influence prediction with deep learning. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM (2018) 2110–2119
16. QuestMobile: 2018 china mobile internet. http://www.questmobile.com.cn/research/report-new/29 (2019)
17. Tan, Z., Zhang, Y., Hu, W.: Online prediction of video popularity in ovss: A video age-sensitive model with beyond views features. IEEE Transactions on Broadcasting (2019) 1–10
18. Toutiao. https://www.toutiao.com (2019)
19. Trzciński, T., Rokita, P.: Predicting popularity of online videos using support vector regression. IEEE Transactions on Multimedia **19**(11) (2017) 2561–2570
20. Trzciski, T., Rokita, P.: Predicting popularity of online videos using support vector regression. IEEE Transactions on Multimedia **19**(11) (2017) 2561–2570
21. Tsai, Y.H.H., Divvala, S., Morency, L.P., Salakhutdinov, R., Farhadi, A.: Video relationship reasoning using gated spatio-temporal energy graph. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 10424–10433
22. Vasconcelos, M., Almeida, J.M., Gonçalves, M.A.: Predicting the popularity of micro-reviews: a foursquare case study. Information Sciences **325** (2015) 355–374
23. Wang, X., Gupta, A.: Videos as space-time region graphs. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 399–417
24. Welbourne, D.J., Grant, W.J.: Science communication on youtube: Factors that affect channel and video popularity. Public Understanding of Science **25**(6) (2016) 706–718
25. Wu, B., Shen, H.: Analyzing and predicting news popularity on twitter. International Journal of Information Management **35**(6) (2015) 702–711
26. Xu, T., Botelho, L.M., Lin, F.X.: Vstore: A data store for analytics on large videos. CoRR **abs/1810.01794** (2018)
27. xueqiu. https://xueqiu.com/9231373161/136685735 (2020)
28. Yu, H., Xie, L., Sanner, S.: The lifecyle of a youtube video: Phases, content and popularity. In: Ninth International AAAI Conference on Web and Social Media. (2015)
29. Yuchao, Z., Pengmiao, L., Zhili, Z., Bo, B., Gong, Z., Wendong, W., Bo, L.: Challenges and chances for the emerging short video network. In: IEEE International Conference on Computer Communications(Infocom), IEEE (2019)