

Weighted Estimation of Information Diffusion Probabilities for Independent Cascade Model

Yoosof Mashayekhi, Mohammad Reza Meybodi
Computer Engineering and Information Technology
Department
Amirkabir University of Technology
Tehran, Iran
{ymashayekhi, mmeybodi}@aut.ac.ir

Alireza Rezvanian
School of Computer Science
Institute for Research in Fundamental Sciences (IPM)
Tehran, Iran
a.rezvanian@aut.ac.ir

Abstract— In recent years, social networks have become popular among Internet users, and various studies have been performed on analysis of such networks. One of the important issues in analyzing social networks is information diffusion analysis. In this context, users' behavior is assumed to be influenced by other social network users. Several models have been designed to simulate and analyze how information is disseminated in social networks. In this paper, we study the problem of learning the diffusion probabilities for the independent cascade model. We first outline the importance of the subject, and then we propose a method to estimate diffusion probabilities. In this method, we assign a weight to each individual diffusion sample of each link in the network based on its parameters. We propose two weighting schemes to consider the different effects of diffusion samples. Then, we evaluate our method for learning diffusion probabilities with the help of several datasets and present the results. Also, the method presented in this paper is compared with other methods in terms of mean absolute error and training time.

Keywords— *Independent Cascade Model, Information Diffusion, Social Network Analysis*

I. INTRODUCTION

In recent years, online social networks have become a major source of information. The growing importance of social networks is the result of huge amount of data they produce. Users' actions in a social network like sharing a content, retweet, like and comment produce a large amount of information. This information can be used and analyzed due to the understanding of user behavior. Several problems are studied in this area like community detection, link prediction or information diffusion. Study of information diffusion aims at studying how a content propagates in the networks via user interactions.

The study of information diffusion in social networks firstly initiated in epidemiology and social sciences contexts. With growing online social networks among people, understanding user behavior and how the information spreads in the network became an important research area. Users mostly are influenced by their friends, resulting in information cascades [1]. Main tasks in studying information diffusion include detecting popular topics, identifying influential information

spreaders and modeling information diffusion [2]. Many of the information diffusion models are extensions of widely used independent cascade model (ICM) [3] and linear threshold model (LTM) [4]. Information diffuses in these models as an iterative process where the probability of reaching information to a user depends on his neighbors which already diffused the information [5-7]. Some models have been proposed to consider continuous time of diffusion process [8, 9]. These models have a time delay before spreading the content to a node. One of the main challenges of using information diffusion models is learning the parameters of the model. The parameters of information diffusion models are usually learned through observing behavior of the users of social network. After learning the parameters, model can be used in various applications of information diffusion like predicting which users will get infected in a diffusion process, maximizing influence propagation, etc.

In this paper, we focus on ICM and try to learn its parameters. Parameters of ICM are diffusion probabilities, which should be learned from observed cascades in the network. The problem of learning diffusion probabilities for ICM has been studied in [10] and several variants have been proposed which will be described in section 2. Most methods use iterative algorithms to learn probabilities which result in low performance over large networks. We try to learn diffusion probabilities with less error and better training time. We learn diffusion probability of each edge of the graph by assigning weights to diffusion samples derived from observed cascades.

The paper is organized as follows. Section 2 presents previous works on this problem and their strength and weakness. Section 3 describes problem formulation, which includes the diffusion model description and problem definition. Section 4 illustrates the method proposed in this paper. Section 5 presents the evaluation of proposed method for diffusion probability estimation task. Section 6 concludes our work and gives some insights for future works.

II. RELATED WORK

Social network can be modeled as a graph, where graph nodes represent users of social network and edges represent relations between users, like friendship, follow, etc. In recent years, information diffusion has been studied in many social

network analysis works [5, 11-16]. In studying information diffusion, mostly a diffusion starts by a set of users and spreads in the network via user interactions.

Domingos and Richardson [17, 18] were the first to study the propagation of influence in social networks and identifying most influential nodes with data mining approach. Several heuristics like node degree, betweenness, closeness, etc. have been proposed as well for maximizing influence in social network. Kempe et al. [13] formally defined the influence maximization problem as an optimization problem and proved that the optimization problem of selecting most influential nodes in ICM and LTM is NP-hard. They also provided a greedy algorithm with approximation guarantees.

Saito et al. [10] studied learning the parameters of independent cascade model. They formulated the problem as a likelihood function and estimated the probabilities of edges using EM algorithm. In an another primary work on learning diffusion models, Goyal et al. [19] studied learning the parameters of general threshold model. Unlike the method Saito et al. [10] used, their work is scalable, because they don't apply an iterative algorithm like EM for learning model's parameters. Many recent works tried to solve some drawbacks of these two fundamental approaches. Continuous time models AsIC and AsLT have been developed to let the nodes activate in continuous time [8, 9]. Some studies do not consider having the full social network graph and the underlying network is unknown [20, 21].

Some studies use users' information (interests, profile, etc.) and the content which is propagated through social network, to predict diffusion probabilities and users' behavior more accurately [22-25]. Barbieri et al. [24] proposed two models extending ICM and LTM which consider the topic propagated in social network. In a similar research, Wang et al. [25] learned diffusion probabilities for ICM assuming that each message is a distribution over five emotions and the diffusion probabilities vary for each emotion.

Several research works have studied information diffusion dynamics [26-28]. Guille and Hacid [28] proposed a time based asynchronous independent cascade model where diffusion probabilities depend on time of diffusion. They used machine learning techniques to learn diffusion probabilities by features extracted from twitter social network. Such studies do not learn diffusion probability for each link and learn a model based on users' extracted features.

In this paper we tackle the problem of learning diffusion probabilities for ICM. We design a weighting method to estimate diffusion probability based on the information obtained from each previous cascade in the network. As a result, our method is scalable in comparison to baselines. Also result of experiments shows less error compared to other methods.

III. PROBLEM FORMULATION

In this study, we learn diffusion probabilities for ICM. First, we explain ICM. Next, we formulate the problem we try to solve.

A. Diffusion Model

In this paper, we'll try to learn the parameters of the well-known independent cascade model. In ICM, the diffusion starts with a set of seed nodes which are the only active nodes at time zero, and proceeds in discrete time-steps. At the time-step $t+1$, every node which is activated at the time-step t , has a single chance to activate any of its inactive child nodes. Whether the parent node succeeds or fails in activating a child node, it won't have another chance to activate the same node. The diffusion process continues until there is no newly activated node. An assumption in ICM is that an activated node remains active until the end of the diffusion process, in other words, the state of the node will not change afterwards. Any attempt that a parent makes to activate its child, succeeds with the probability which is associated with the edge from parent node to that child node. So ICM should be provided with probabilities of the edges of the graph to proceed. In this paper we tackle the problem of inferring these probabilities.

B. Problem Definition

For specifying the problem, we will follow [10]'s notation closely. For a given directed network which is represented by the graph $G=(V,E)$, let V be the set of nodes and E be the set of edges of graph. We represent each edge with a tuple (u,v) which in this manner there is a link from node u to node v . For a node such as u , let $N^{out}(u)=\{v|(u,v) \in E\}$ denote the set of children of u and let $N^{in}(u)=\{v|(v,u) \in E\}$ denote the set of parents of u .

Let a diffusion episode $D=\langle D(0), D(1), \dots, D(T) \rangle$ be a sequence of node sets, where $D(t)$ denotes the nodes activated at the time t . We assume any node can appear once in a diffusion episode. Which means any node can be activated one time in a single diffusion episode. Thus, for a diffusion episode D , we have $D(i) \cap D(j) = \emptyset$, $i \neq j$. Another assumption is that once a node become active in a diffusion episode, it's state can't be changed to inactive, so it remains active in that episode. In addition, $AT_s(u)$ denotes activation time of node u in diffusion episode D_s . If u is not activated in D_s at all, $AT_s(u)$ is infinity. Also, $T(D_s)$ is last time-step of nodes in D_s .

The problem is, given a directed network graph $G=(V,E)$ and a set of diffusion episodes $S=\{D_1, D_2, \dots, D_n\}$, what are the diffusion probabilities for flowing information from a node to another through the edge between them and how these probabilities can be learned? In this paper, we provide a method with two types of weighting to learn diffusion probabilities.

IV. PROPOSED METHOD

In this section, we'll explain our method to learn diffusion probabilities for ICM. We use information obtained from diffusion episodes to estimate the diffusion probability of each edge. Each diffusion episode provides information about some of the edges of the graph. First, we discuss different types of information presented in each diffusion episode. Next, we describe our method for learning diffusion probabilities with two different weighting scheme.

A. Diffusion Episode Analysis

In ICM, each newly activated node has a single chance to activate each of its inactive children. A diffusion episode provides information about an edge $e=(u,v)$ only if there is a chance that u has attempted to activate v . In other words, if v is activated after u in a diffusion episode, or it is not activated at all, we can infer that u may have attempted to activate v and we can use this information to estimate diffusion probability of edge e . For an edge $e=(u,v)$, a diffusion episode in which u is not activated at all, or v is activated before or simultaneously with u , does not provide information about the diffusion probability of edge e . Three cases can happen in a diffusion episode for an edge which reveal information about the diffusion probability of the edge. We refer to these cases that provide us information about diffusion probability as **diffusion samples**. In other words, we have a diffusion sample $ds=((u,v),D_s)$ if $AT_s(u) \leq T(D_s)$ and $AT_s(v) \geq AT_s(u)+1$. In the following, we will explain three types of diffusion samples which may occur.

The first case is about edges such as $e=(u,v)$ in which u is activated at the time-step t . Then, at the time-step $t+1$, v is activated and the other parents of this node are not activated at the time-step t . Hence, we are sure that u has succeeded in activating v . Formally, this case occurs in diffusion episode D_s if $AT_s(u) \leq T(D_s)$, $AT_s(v)=AT_s(u)+1$ and $|N^{in}(v) \cap D_s(AT_s(v)-1)|=1$. We refer to this case as **certain positive sample** for edge e .

The second case is about edges such as $e=(u,v)$ in which u is activated at time-step t , but the child node v is not activated from time-step 0 to $t+1$. In this case it can be concluded that u definitely failed to activate v . Formally, this case occurs in diffusion episode D_s if $AT_s(u) \leq T(D_s)$ and $AT_s(v) > AT_s(u)+1$. We refer to this case as **certain negative sample** for edge e .

The third case is for edges such as $e=(u,v)$ in which the parent node u is activated at the time-step t and the child node v is activated at the time-step $t+1$. Also, other parents of v are activated at time-step t . In this case, it is unknown which parent activated v , but we know that one of its parents succeeded in activating it and the rest were either failed to activate it or v has become active by another node before they have a chance to activate the node. Formally, this case occurs in diffusion episode D_s if $AT_s(u) \leq T(D_s)$, $AT_s(v)=AT_s(u)+1$ and $|N^{in}(v) \cap D_s(AT_s(v)-1)| > 1$. We refer to this case as **uncertain sample** for edge e .

To illustrate, consider a node v , which has two parents, u and w . In a diffusion episode D_s , if v is activated before u or at the same time-step with u , we are sure that u has never attempted to activate v . Hence, we cannot obtain any information about diffusion probability of edge (u,v) . If u is activated at time-step t and $AT_s(v) > t+1$, we are sure that u has failed in activating v and it is a certain negative sample. If u is activated at time-step t and v is activated at time-step $t+1$, two cases may happen. If $AT_s(w) \neq t$, we are sure that u has activated v and it is a certain positive sample. Otherwise, we do not know that if u has activated v or w has activated it. Thus, it is an uncertain sample.

B. Diffusion Probability Learning

The diffusion probability of an edge $e=(u,v)$ by observing a set of learning diffusion episodes is equal to the number of times u has succeeded in activating v , divided by the number of times u has attempted to activate v . The problem with using the probability definition is when we are not sure which of the active parents of v in the previous step has activated it (uncertain samples). In fact, we do not know the success or failure of the parents to activate it. We also do not know the occurrence of attempts to activate v from either of its parents. To overcome this problem, we suggest using the probability formula considering an assumption and weighting different samples.

Suppose that v is activated at the time-step $t+1$. We assume that all of the parents of v with activation time t , activated v , with the difference that the information obtained from this episode is weighted lower than episodes in which we are sure the parent node u has succeeded or failed in activating the child node v . To achieve this purpose, we define a weight function to assign weight to each edge (u,v) and each diffusion episode D_s which is given in:

$$Weight((u,v),D_s) = \begin{cases} W(N^{in}(v) \cap D_s(AT_s(v)-1)): & ((u,v),D_s) \text{ is uncertain sample} \\ 1: & ((u,v),D_s) \text{ is certain positive sample} \\ 1: & ((u,v),D_s) \text{ is certain negative sample} \\ 0: & \text{Otherwise} \end{cases} \quad (1)$$

In (1), $W(A)$ is a function which takes $A \subseteq V$ and returns a value in range $[0, 1]$. Two different definitions are provided for this function later. Weight is 0 if the diffusion episode does not provide any information about the diffusion probability of edge. We assigned weight 1 to certain positive and certain negative samples and weight $W(N^{in}(v) \cap D_s(AT_s(v)-1))$ to uncertain samples. We also define a function which assigns a value 0 or 1 to each edge (u,v) and each diffusion episode D_s :

$$IsDiffused((u,v),D_s) = \begin{cases} 1: & ((u,v),D_s) \text{ is certain positive sample} \\ 1: & ((u,v),D_s) \text{ is uncertain sample} \\ 0: & \text{Otherwise} \end{cases} \quad (2)$$

In (2), when there is a chance of information diffusion from u to v (certain positive and uncertain samples), the function returns 1. Otherwise, it returns 0 (certain negative samples). Using a set of diffusion episodes $S=\{D_1, D_2, \dots, D_n\}$ for learning, equation (3) estimates diffusion probability of an edge $e=(u,v)$:

$$P(u,v) = \frac{\sum_{D_s \in S} Weight((u,v),D_s) IsDiffused((u,v),D_s)}{\sum_{D_s \in S} Weight((u,v),D_s)} \quad (3)$$

In (3), we divide number of times that u succeeded in activating v by number of times that it failed. We also apply lower weights on uncertain samples in probability formulation. In the followings, we propose two weighting scheme: linear decay and exponential decay weighting.

1) *Linear Decay Weighting*: In the linear decay weighting scheme, We define $W(A)=1/|A|$. In other words, suppose that v is activated at the time-step $t+1$. We assign weight $1/|N^in(v) \cap D_s(AT_s(v)-1)|$ to uncertain samples in episode D_s and 1 to certain positive and certain negative samples. Hence, as the number of parents of v that have the chance to activate it increases, each of the edges between them and v would receive lower weight. We refer to this weighting scheme as LDW.

2) *Exponential Decay Weighting*: In the exponential decay weighting scheme, We define $W(A)=\exp(-(|A|-1))$. In other words, for an edge $e=(u,v)$, we reduce the weight of uncertain samples exponentially in terms of the number of active parents of the node v which were activated in previous time-step of the activation time of the node v . Consequently, We assign weight 1 to certain positive and certain negative samples, and weight $\exp(-(|N^in(v) \cap D_s(AT_s(v)-1)|-1))$ to uncertain samples. We refer to this weighting scheme as EXDW.

C. Improving Robustness

One of the problems of estimating the diffusion probability for an edge $e=(u,v)$ is when there is no certain sample for it in any diffusion episodes. In this case, the only available information is obtained from uncertain samples. As a result, the diffusion probability of such edges will be 1, despite the absence of reliable information for learning diffusion probability. It is clear that this will yield in inaccurate estimation and cannot be trusted. To prevent such behavior, we ignore uncertain samples in these edges and their diffusion probability is chosen uniformly at random.

V. Experimental Evaluation

A. Experimental Settings

Table 1 lists the datasets used in experiments. Self-edges are removed from datasets. The datasets which we used are: Digg [29], Slashdot [30], Facebook-Wall-Posts [31] and CitHepTh [32].

We applied the approach of [10] to choose true diffusion probabilities and create diffusion episodes. For each dataset, we generated 100 diffusion episodes starting from one uniformly randomly selected node for each episode and we tested methods for different size of learning episodes. The diffusion probability of each edge is selected uniformly at random in range [0.1, 0.3].

TABLE I. NETWORK DATASETS

| Network | Node | Edge | Description |
|---------------------|-------|--------|--|
| Digg | 30398 | 86404 | The reply network of the social news website Digg. |
| Slashdot | 51083 | 131175 | The reply network of communication technology website Slashdot. |
| Facebook-Wall-Posts | 46952 | 274086 | The network of a small subset of posts to other user's wall on Facebook. |
| CitHepTh | 27770 | 352807 | The network of publications in the arXiv's High Energy Physics. |

We compare our proposed method for estimating diffusion probabilities with the followings:

- EM: Saito et al [10] designed a method for estimating diffusion probabilities from a set of diffusion episodes. They formulated a likelihood function and used EM algorithm to optimize it. We refer to this method as EM.
- DAIC: Lamprier et al [20] proposed a method similar to method presented in [10] to learn diffusion probabilities for ICM by considering order of nodes' activations rather than exact activation time-steps. In their method the underlying network is unknown and they use complete graph in learning process. The network's graph is known in this paper and we apply their method to this graph. They also considered exponential prior distributions of the diffusion probabilities to improve robustness. We refer to this method as DAIC.
- EM with priors: we applied the idea of using exponential prior distributions of the diffusion probabilities to the method presented in [10]. Thus, maximization step of EM algorithm will be similar to [20]. We refer to this method as EMP.
- Equal weighting: this method is similar to LDW, with the difference that weights of uncertain samples are also 1. In other words, $W(A)=1$. We refer to this weighting scheme as EW.
- Binary weighting: this method is similar to LDW as well, with the difference that uncertain samples are ignored and do not count in probability. In other words, $W(A)=0$. We refer to this weighting scheme as BW.

DAIC and EMP require a parameter λ to determine the degree of using prior distributions of diffusion probabilities. Different values of λ have been tested and the best performance is reported. For all datasets and for both methods, $\lambda=10$ performed better than other values.

We compare methods in terms of mean absolute error of estimated diffusion probabilities from true diffusion probabilities of edges. Moreover, we compare methods in terms of training time.

B. Experimental Results

For the experiment, we learned each method with different size of diffusion episodes, starting from 10 to 100. For each model, we compared the true edge probabilities from estimated diffusion probabilities in terms of mean absolute error. Fig. 1, Fig. 2, Fig. 3 and Fig. 4 represents mean absolute error against number of episodes for each dataset. We can observe that the proposed methods in this paper estimate diffusion probabilities better than the others. Moreover, EXDW have a slightly better performance than LDW in some datasets which means uncertain samples are not promising and their weight should be less than certain samples.

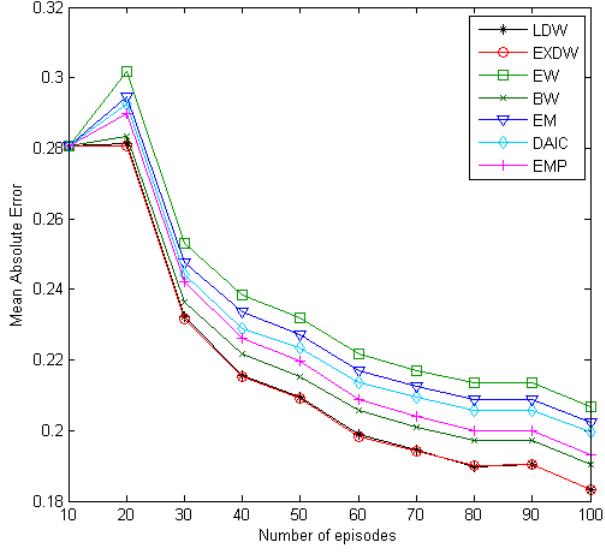


Fig. 1. Mean absolute error against number of episodes in Digg dataset

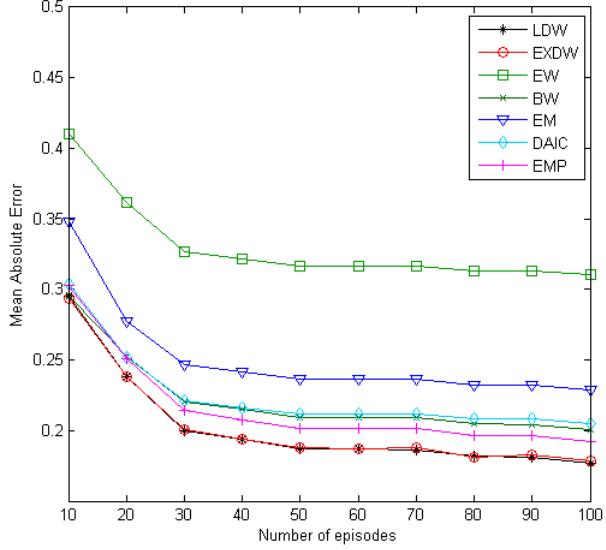


Fig. 2. Mean absolute error against number of episodes in Slashdot dataset

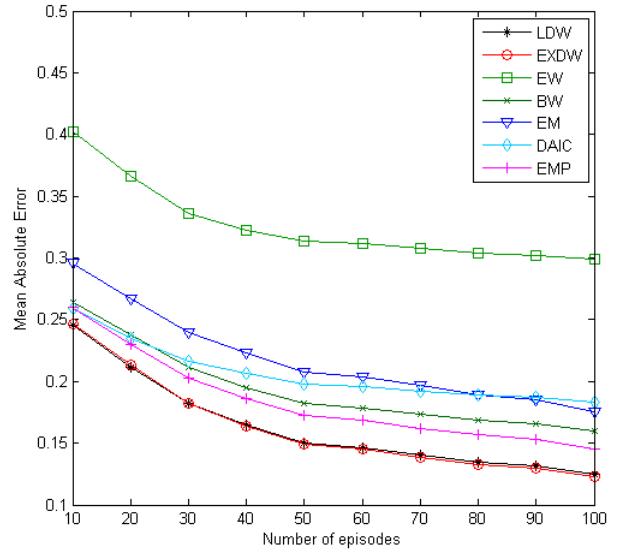


Fig. 3. Mean absolute error against number of episodes in Facebook-Wall Posts dataset

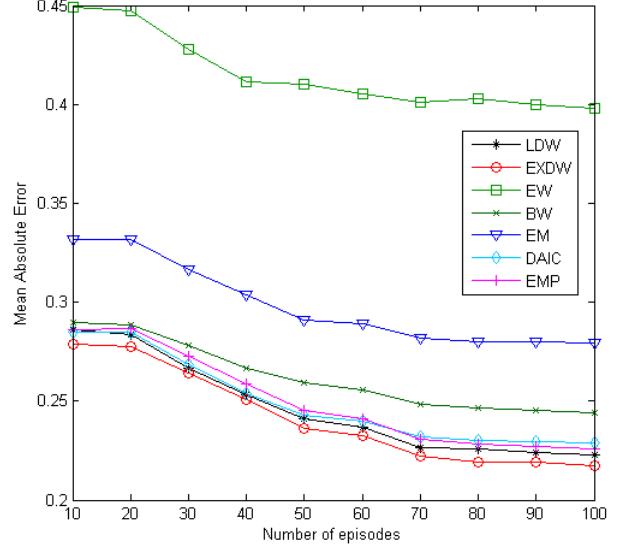


Fig. 4. Mean absolute error against number of episodes in CitHepTh dataset

As is clear in the results, BW has a better performance than EM and in some datasets it outperforms DAIC and EMP too. We claim that EW and BW depend heavily on how diffusion episodes are generated. As BW ignores some samples which show the occurrence of propagation through an edge, the probabilities learned are underestimate. So BW has lower error than EM, DAIC and EMP in some datasets because we selected diffusion probabilities for edges in range [0.1, 0.3] and its mean (0.2) is close to 0. If the mean of the diffusion probability selection range was near 1, we would expect that the results of EW should be better than BW, EM, DAIC and EMP. To illustrate, we conduct another experiment on dataset Digg which we selected the diffusion probability of each edge uniformly at random in range [0.6, 0.8]. Fig. 5 shows the results of this experiment and the error of EW is less than BW, EM, DAIC and EMP. The results also show that in this experiment when number of episodes for learning purpose

increases, both LDW and EXDW have better performance than other methods.

Next, we compared training time of each method at different diffusion episode size. Fig. 6 represents training time against number of episodes for dataset Digg. Other datasets yield similar results. The difference between training time of the methods is significant, because the EM, DAIC and EMP methods are iterative and take more time to converge. On the other hand, the other methods are not iterative, and they process each diffusion episode once. Consequently, training time of our method is much less than the iterative methods.

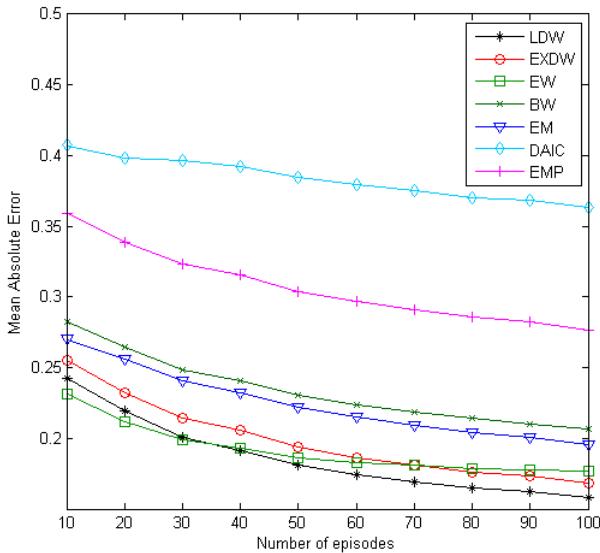


Fig. 5. Mean absolute error against number of episodes in Digg dataset.
Diffusion probability selection range: [0.6, 0.8]

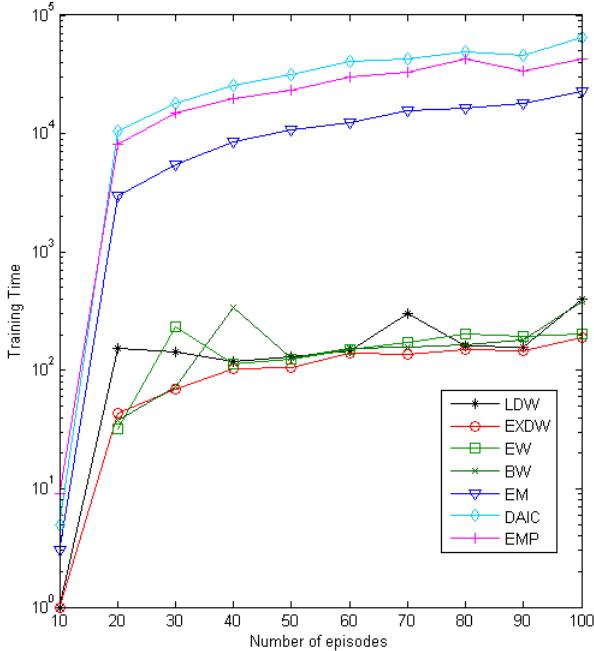


Fig. 6. Training time (log-scale) against number of episodes in Digg dataset

VI. CONCLUSION AND FUTURE WORK

In this paper, we tackled the problem of learning diffusion probabilities for well-known diffusion model, ICM. Our method learns diffusion probabilities from diffusion episodes available for learning purpose. The experiments showed that our method results better outcome in terms of mean absolute error of estimated probabilities from true probabilities placed on graph edges. In terms of training time, the method proposed in this paper learns probabilities in much less time and so, is more scalable.

Promising results with this approach let us expect various development of the proposed method. For instance, we intend to develop methods with similar approach to this paper, to work with data extracted from real social networks like twitter.

REFERENCES

- [1] Q. Bao, W. K. Cheung, Y. Zhang, and J. Liu, "A component-based diffusion model with structural diversity for social networks," *IEEE transactions on cybernetics*, vol. 47, pp. 1078-1089, 2017.
- [2] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *ACM SIGMOD Record*, vol. 42, pp. 17-28, 2013.
- [3] J. Goldenberg, B. Libai, and E. Muller, "Talk of the network: A complex systems look at the underlying process of word-of-mouth," *Marketing letters*, vol. 12, pp. 211-223, 2001.
- [4] M. Granovetter, "Threshold models of collective behavior," *American journal of sociology*, vol. 83, pp. 1420-1443, 1978.
- [5] J. Yang and J. Leskovec, "Modeling information diffusion in implicit networks," in *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, 2010, pp. 599-608.
- [6] M. Gomez Rodriguez, D. Balduzzi, and B. Schölkopf, "Uncovering the temporal dynamics of diffusion networks," in *Proceedings of the 28th international conference on machine learning (ICML- 11)*, ICML'11, ACM, 2011, pp. 561-568.
- [7] G. Ver Steeg and A. Galstyan, "Information-theoretic measures of influence based on content dynamics," in *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013, pp. 3-12.
- [8] K. Saito, M. Kimura, K. Ohara, and H. Motoda, "Learning continuous-time information diffusion model for social behavioral data analysis," *Advances in Machine Learning*, pp. 322-337, 2009.
- [9] K. Saito, M. Kimura, K. Ohara, and H. Motoda, "Selecting information diffusion models over social networks for behavioral analysis," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2010, pp. 180-195.
- [10] K. Saito, R. Nakano, and M. Kimura, "Prediction of information diffusion probabilities for independent cascade model," in *Knowledge-based intelligent information and engineering systems*, 2008, pp. 67-75.
- [11] M. Gomez-Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 5, p. 21, 2012.
- [12] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information diffusion through blogspace," in *Proceedings of the 13th international conference on World Wide Web*, 2004, pp. 491-501.
- [13] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, pp. 137-146.
- [14] X. Yu and T. Chu, "Learning the structure of influence diffusion in the independent cascade model," in *Control Conference (CCC), 2017 36th Chinese*, 2017, pp. 5647-5651.
- [15] K. Zhang, X. Yun, J. Liang, X.-y. Zhang, C. Li, and B. Tian, "Retweeting behavior prediction using probabilistic matrix factorization," in *Computers and Communication (ISCC), 2016 IEEE Symposium on*, 2016, pp. 1185-1192.

- [16] B. Jiang, J. Liang, Y. Sha, R. Li, W. Liu, H. Ma, et al., "Retweeting Behavior Prediction Based on One-Class Collaborative Filtering in Social Networks," in Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, 2016, pp. 977-980.
- [17] P. Domingos and M. Richardson, "Mining the network value of customers," in Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, 2001, pp. 57-66.
- [18] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 2002, pp. 61-70.
- [19] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in Proceedings of the third ACM international conference on Web search and data mining, 2010, pp. 241-250.
- [20] S. Lamprier, S. Bourigault, and P. Gallinari, "Influence learning for cascade diffusion models: focus on partial orders of infections," Social Network Analysis and Mining, vol. 6, p. 93, 2016.
- [21] S. Bourigault, S. Lamprier, and P. Gallinari, "Representation learning for information diffusion through social networks: an embedded cascade model," in Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, 2016, pp. 573-582.
- [22] C. Lagnier, L. Denoyer, E. Gaussier, and P. Gallinari, "Predicting information diffusion in social networks using content and user's profiles," in European conference on information retrieval, 2013, pp. 74-85.
- [23] D. Varshney, S. Kumar, and V. Gupta, "Predicting information diffusion probabilities in social networks: A Bayesian networks based approach," Knowledge-Based Systems, vol. 133, pp. 66-76, 2017.
- [24] N. Barbieri, F. Bonchi, and G. Manco, "Topic-aware social influence propagation models," Knowledge and information systems, vol. 37, pp. 555-584, 2013.
- [25] Z. Wang, J. Zhao, and K. Xu, "Emotion-based Independent Cascade model for information propagation in online social media," in Service Systems and Service Management (ICSSSM), 2016 13th International Conference on, 2016, pp. 1-6.
- [26] D. Li, S. Zhang, X. Sun, H. Zhou, S. Li, and X. Li, "Modeling information diffusion over social networks for temporal dynamic prediction," IEEE Transactions on Knowledge and Data Engineering, 2017.
- [27] K. Zhuang, F. Han, H. Shen, K. Zhang, and H. Zhang, "Information Diffusion Temporal Dynamic Prediction in Microblog System Based On User Influence Learning," International Journal of Hybrid Information Technology, vol. 9, pp. 327-336, 2016.
- [28] A. Guille and H. Hacid, "A predictive model for the temporal dynamics of information diffusion in online social networks," in Proceedings of the 21st international conference on World Wide Web, 2012, pp. 1145-1152.
- [29] M. De Choudhury, H. Sundaram, A. John, and D. D. Seligmann, "Social synchrony: Predicting mimicry of user actions in online social media," in Computational Science and Engineering, 2009. CSE'09. International Conference on, 2009, pp. 151-158.
- [30] V. Gómez, A. Kaltenbrunner, and V. López, "Statistical analysis of the social network and discussion threads in slashdot," in Proceedings of the 17th international conference on World Wide Web, 2008, pp. 645-654.
- [31] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in facebook," in Proceedings of the 2nd ACM workshop on Online social networks, 2009, pp. 37-42.
- [32] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, p. 2, 2007.