Data Mining Package Report

Kandati Vishnu Sai, Midhilesh E., Sanjay Seetharaman

November 2020

Abstract

With more than 2 billion monthly active users, social networking websites have dramatically changed the lives of people. With a range of features and formats, they cater to the interests of a wide range of users. People build social networks or social relationships with other people who share similar personal or career interests, activities, backgrounds or real-life connections. The availability of services on a wide range of devices such as mobile phones, laptops, and computers has made it more accessible than ever. A user can easily access other users' behaviors and is in turn influenced by them. As a result, effective social influence prediction is vital for a variety of applications such as online recommendation and advertising.

In this report, we discuss DeepInf[1], an end-to-end framework that learns users' latent feature representation for predicting social influence.

[1] Qiu, Jiezhong and Tang, Jian and Ma, Hao and Dong, Yuxiao and Wang, Kuansan and Tang, Jie. 2018. "*Deepinf: Social influence prediction with deep learning*" in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp.2110–2119.

Contents

Chapter 1

Introduction

1.1 Social Network

A **Social Network** is a network that reflects the social structure of its nodes and their inter-dependency, such as friendship of people, co-authorship of researchers, and collaboration between different parties. Its structure made up of a set of social actors (such as individuals or organizations), sets of dyadic ties, and other social interactions between actors. The scale of a social network is usually very large. The dramatic development in technology has led to more and more people engage in a variety of social activities like chatting, blogging, commenting, and shopping. This has led to researchers having far more questions than answers.

A social network can be represented and treated as an abstract network of interconnected nodes, and researchers generally adopt a graph or matrix to describe the networks. However, this representation is not scalable. For large social networks, we can expect millions of nodes and edges. It is challenging to decompose or compress such huge graphs. If at all there is a provision to maintain such huge graphs, it is difficult to describe the dynamical features of social networks over such representations.

1.2 Social Influence

Social Influence is a relationship established between two entities for a specific action. In particular, one entity influences the other entity to perform an action. Usually, the first entity is called the *influencer*, the second entity is called the *influencee*. It is a function of uncertainty because the influence may not have any idea on because the influencee would perform the action. The typical examples are viral marketing, influential bloggers finding, online advertising, social healthcare, expert finding, personalized commendation, citation networks, and so on.

Figure 1.1: A social network diagram

Influence is *asymmetric*: the fact that Alice influences Bob does not necessarily mean that Bob also influences Alice. Influence is *transitive*: if Alice influences Bob and Bob influences Carlos, this implies that Alice influences Carlos indirectly. Influence is *propagative:* information can be passed from one member to another in a social network, creating influence chains. Spreading of influence in social network is the basis of "word of mouth" propagation of information for humans

The research on social influence analysis - still in infancy - interconnects with the other features of social networks, such as influence properties, evaluation metrics of influence, collection and processing social networking big data, and selection of most influential nodes.

Analyzing social influence helps us: 1) in terms of sociology, it is helpful to understand people's social behaviors; 2) in terms of public services, it is helpful to provide a theoretical basis for public decision making and public opinion guidance; 3) in terms of political factors, it is helpful to promote national security, economic stability, economic progress, and so on. Hence, social influence analysis is significant and valuable.

There are many challenges in measuring social influence: 1) no mathematical definition; 2) uncertainty; 3) no effective way to integrate external factors; 4) characterizing the relationship.

In this report, we focus on user level influence - predicting the action status of a user given the action statuses of her near neighbors and her local structural information. For example, in Figure 1.2, consider vertex v to represent Alice. Suppose her friends Bob, Carlos, and Dave (represented by the black vertices) perform an action, say bought a product, will Alice perform the action in the future?

Researchers have proposed solutions to make such predictions, but they consider complicated hand-crafted features, which require extensive knowledge of specific domains and are usually difficult to generalize to different domains.

DeepInf is an deep learning based end-to-end approach to discover hidden

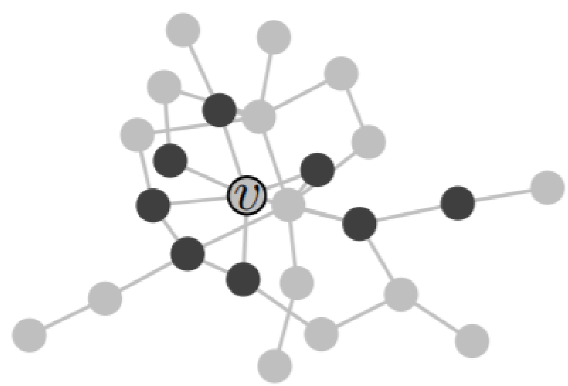


Figure 1.2: A motivating example of social influence locality prediction

and predictive signals in social influence automatically. Using techniques like network embedding, graph convolution and graph attention mechanism into a unified network, it represents both influence dynamics and network structures in a latent space.

Chapter 2

Graph Mining

Hi! This is a new section.

Chapter 3

DeepInf: Social influence prediction

3.1 Introduction

Deepinf focus on the prediction of user-level social influence. It aims to **predict the action status of a user given the action**. It is a deep learning based framework to represent both influence dynamics and network structures into a latent space and tries to minimize the negative likelihood that was defined in the section 1.

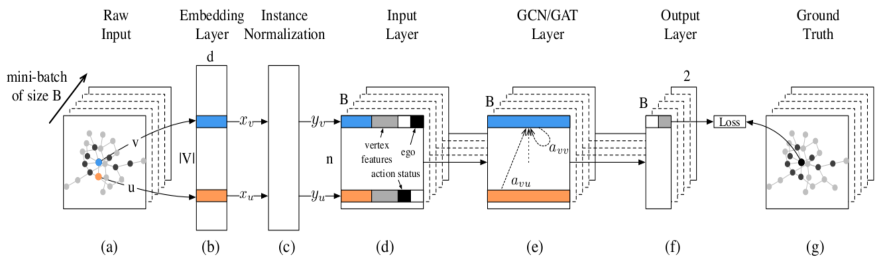
3.2 Model framework

3.2.1 Sampling Near Neighbour

Give a user v, a r-ego network $\mathcal{G}_v^r$ is extracted using breadth-first search (BFS) starting from user v. However, $\mathcal{G}_v^r$ may have different size due to the small world property in the social network. Since most deep learning models expects fixed size data, the graph $\mathcal{G}_v^r$ can be sampled to fixed size.

For sampling a fixed size graph **random walk the restart** (RWR) was used. RWR algorithm is defined as following steps.

- Start random walks from either the ego user v or one of her active neighbors randomly.
- The random walk iteratively travels to its neighborhood with the probability that is proportional to the weight of each edge.
- At each step, the walk is assigned a probability to return to the starting node, that is, either the ego user v or one of v's active neighbors.



- Run the algorithm until a fixed number of vertices denoted by $\hat{\Gamma_v^r}$ with $|\hat{\Gamma_v^r}|$=n.

After running this algorithm, a sub-graph $\hat{\mathcal{G}_v^r}$ and denote $\hat{S}_v^t$= { s_u^t : $u \in \hat{\Gamma_v^r}$ } be the action statuses of v's sampled neighbours. Therefore we re-define the optimization objective in section 1 as:

$$\mathcal{L}(\theta) = -\sum_{i=1}^{N} \log(P_\theta(s_{v_i}^{t_i+\Delta t}|\hat{\mathcal{G}_{v_i}^t}, s_{v_i}^{t_i})) \tag{3.1}$$

3.2.2 Neural Network

With the retreived $\hat{\mathcal{G}_{v_i}^t}$ and $\hat{S}_v^t$ for each user, an effective neural network model to incorporate both the structural properties in $\hat{\mathcal{G}_{v_i}^t}$ and action statuses in $\hat{S}_v^t$.

Deepinf neural network model consist of network embedding layer, instance normalization layer, input layer and GCN layer as described in the above diagram.

Embedded layer

Network embedding technique encode network structural properties into low dimensional matrix $\mathbf{X} \in \mathcal{R}^{\mathcal{D}\times|V|}$. Deepinf uses Deepwalk algorithm for mapping each users into $\mathcal{R}^{\mathcal{D}}$ space.

Instance Normalization

Instance normalization can remove instance-specific mean and variance, which encourages the downstream model to focus on users' relative positions in latent embedding space rather than their absolute positions. It also prevents overfitting.

Let x_u be the low dimension representation for the user $u \in \hat{\Gamma_v^r}$, the instance normalized vector y_u is obtained by

$$y_{ud} = \frac{x_{ud} - \mu_{ud}}{\sqrt{\sigma_d^2 + \epsilon}} \tag{3.2}$$

for each embedding dimension d = 1...$\mathcal{D}$, where

$$\mu_d = \frac{1}{n} \sum_{u \in \hat{\Gamma_v^r}} x_{ud}, \ \sigma_d^2 = \frac{1}{n} \sum_{u \in \hat{\Gamma_v^r}} (x_{ud} - \mu_{ud})^2 \tag{3.3}$$

Input Layer

Input later constructs feature vector for each user. Along with the output from the instance Normalization, input layer adds two binary variable. The first variabe indicates user's action status and second variable indicates whether the user is the ego user.

GCN

3.3 Evaluation Metrics