Neurocomputing 217 (2016) 125–132

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Investment behavior prediction in heterogeneous information network

Xiangxiang Zeng [a], You Li [a], Stephen C.H. Leung [b], Ziyu Lin [a], Xiangrong Liu [a,*]

[a] Department of Computer Science, Xiamen University, Xiamen 361005, China
[b] Faculty of Engineering, The University of Hong Kong, Hong Kong

ARTICLE INFO

Article history:
Received 18 October 2015
Received in revised form
3 December 2015
Accepted 15 December 2015
Available online 11 June 2016

Keywords:
Investment behavior
Heterogeneous information network
Meta-path
HeteSim

ABSTRACT

The crowdfunding industry is growing rapidly worldwide and poses new challenges on how to understand investment behavior. Indeed, a key challenge in this area is how to measure the similarity of an investor and a company, or the interest of an investor in a company. Tremendous effort has been made in previous research regarding the single effective factor or homogeneous network model based on link prediction for investment behavior prediction. In this study, we build an investment behavior prediction model of meta-path-based heterogeneous network, which considers multiple entity and relation types associated with the investment behavior of a particular investor. Our investment behavior prediction model provides an effective similarity measure function for meta-path. To validate the proposed model, we perform experiments on real-world data from CrunchBase. Experimental results reveal that our investment behavior prediction model is indeed a useful indicator.

1. Introduction

A new investment style, crowdfunding, is emerging [1]. Crowdfunding is the practice of funding a project or venture by raising monetary contributions from a large number of people, typically via the Internet. Therefore, predicting investor behavior has become another new behavior prediction in the crowdfunding field.

Numerous studies have explored investment behavior. Factors such as psychological differences [2], geographic differences [3], investment experiences [4], and even genetics [5–7] have been proposed to explain factors that spur investments. Moreover, some researchers use social network features, extracted from a homogeneous relational network of investors and companies, to build a predictive model based on link prediction [8]. A homogeneous network considers nodes as the same entity type (e.g., person) and links as the same relation type (e.g., friendship). Nevertheless, in the real world, most networks are heterogeneous [9], where multiple types of objects and links exist. On the one hand, considering nodes as the same type may miss important semantic information. On the other hand, significant schema-level information may also be lost when we treat all nodes in a distinct type. Considering that companies are of the same kind and comparing them with some other kinds, such as categories, is important. Thus, a heterogeneous network may be suitable in capturing the essential semantics of the CrunchBase dataset.

Various methods, such as modeling a link prediction problem, predict the emergence of links between investors and companies in a network based on current or historical network information [10]. However, link prediction methods are designed for homogeneous networks. In the present study, we extend the link prediction to the relationship prediction in heterogeneous information networks. Bio-inspired models and algorithms, such as P systems (inspired by the structure and functioning of cells) [11,12] and evolutionary computation (motivated by Darwinian theory of evolution) [13,14], are also useful methods for investment behavior prediction.

2. Previous related work

Eugene and Daphne [15] explored the possibility that investors invest in companies based on social relationships, whether positive or negative, similar or dissimilar. Their predictive model is based on a homogeneous network, but most networks are heterogeneous where multiple types of objects and links exist. We proposed a method that is based on heterogeneous networks. Guang et al. [16] conducted studies using the CrunchBase dataset and, using profiles and news articles from TechCrunch, predicted company acquisitions with factual and topical features. Their works focus on a different domain of mergers and acquisitions,

* Corresponding author.
E-mail addresses: xzeng@xmu.edu.cn (X. Zeng), xmuyouli@foxmail.com (Y. Li), schleung@hku.hk (S.C.H. Leung), ziyulin@xmu.edu.cn (Z. Lin), xrliu@xmu.edu.cn (X. Liu).

http://dx.doi.org/10.1016/j.neucom.2015.12.139
0925-2312/

and did not use other relations and all the influential factors in investor behavior.

Sun et al. [17], by exploring meta-path-based [18] features, introduced relationship prediction in a heterogeneous information network. To measure the similarity of different types of objects and links, the measure functions used for the meta-path include PathCount, NormalizedPathCount [19], RandomWalk [20], and others. Shi and Kong [21] proposed a novel measure function, HeteSim, that quantifies the topology. The related similarity computation is also used in gene-disease heterogeneous networks [22–24].

The original contribution of this paper is that it proposes the use of a meta-path-based heterogeneous information network as the main way to predict if investments will occur. For example, given an Investor and a Company, we can predict if the Investor will invest in that particular Company by mining the similarity between the Investor and the Company, which merges multiple impact factors together. Experimental results reveal that our model is an effective approach for companies that are seeking investments because these companies have a significant similarity to potential investors in all respects.

3. Meta-path-based relationship prediction

In this section, using the CrunchBase dataset, we introduce how to model the investment behavior by measuring the meta-path-based similarity between Investor and Company in heterogeneous information networks.

3.1. CrunchBase dataset

CrunchBase (http://www.crunchbase.com) is an open dataset that contains information on startups, investors, funders, acquisitions, trends, companies, and related subjects. CrunchBase relies on the online community to provide and edit most of its content. As of May 2014, CrunchBase consisted of 46,015 companies (organizations); 106,075 investments; and 12,068 acquisitions.

Entity types

Investor. Investors consist of persons and companies (organizations), such as Garret Camp, Google, and others. In subsequent calculations, we denote *Inv* as an investor set and Inv_i as investor i.

Company. Companies are simply companies (organizations), such as Google, Uber, AOL, and others. In subsequent calculations, we denote *Com* as company set and Com_i as company i.

Category. The current CrunchBase dataset has 741 categories. For example, the categories of the Google company are software, search, and others. In subsequent calculations, we denote *Cat* as category set and Cat_i as category i.

City. The city is the location of the corporate headquarters. Study [3] revealed that geographic differences spur investments. In subsequent calculations, we employ *Cit* as cities set and Cit_i as city i.

Relationship type

Investment. Investment relationships are created as a result of the investment behavior of an Investor. For example, Google invested in AOL in December 2005. In subsequent calculations, we denote R^{Inv} as an investment relationship set and R_{ij}^{Inv} as investor i who invests in company j.

Market. The market refers to the company's products that belong to a given Category. For example, Uber's business includes automobiles and transportation. In subsequent calculations, we denote R^{Mar} as a market relationship set and R_{ij}^{Mar} indicates the market of company i at category j.

Location. This variable refers to the location of the company headquarters. For example, Uber's headquarters is in San Francisco. In subsequent calculations, we denote R^{Loc} as a location relationship set, and R_{ij}^{Loc} indicates that company i is located in city j.

Acquisition. Acquisition relationships are created as a result of agreements between Companies. For example, Google acquired Adometry in May 2014. In subsequent calculations, we denote R^{Acq} as an acquisition relationship set, and R_{ij}^{Acq} indicates that company i acquires company j.

3.2. CrunchBase heterogeneous information network

3.2.1. Definition of CrunchBase heterogeneous information network schema

A heterogeneous network schema is a special type of information network that differs from a traditional homogeneous network in that the underpinning data structure is in the form of a directed graph. We define a heterogeneous information network as follows:

Definition 1. (Heterogeneous information network schema) *Given a schema $T_G=(\mathcal{A}, \mathcal{R})$, which is a meta template for a heterogeneous network $G=(V,E)$ with the object type mapping function $\phi: V\to\mathcal{A}$ and the link type mapping function $\psi: E\to\mathcal{R}$, which means that each object $v\in V$ meets $\phi(v)\in\mathcal{A}$ and each link $e\in E$ meets $\psi(e)\in\mathcal{R}$, and G is a directed graph defined over object types $\mathcal{A}$, with edges as relations from $\mathcal{R}$. Furthermore, in a heterogeneous network G, the types of object $|\mathcal{A}|$ or the type of relation $|\mathcal{R}|$ or both is more than one.*

In a heterogeneous information network of the CrunchBase dataset, the sets of entity and relation types are $\mathcal{A}$ and $\mathcal{R}$, respectively. R^{Inv}, which represents the investment relationship between *Inv* and *Com*, is denoted as $Inv \xrightarrow[R^{Inv}]{} Com$. *Inv* is the source type and *Com* is the target type that belongs to relation R^{Inv}. The path from *Com* to *Inv* is deemed to be the inverse of the relation R^{Inv} and is denoted as $Com \xrightarrow[(R^{Inv})^{-1}]{} Inv$.

3.2.2. Heterogeneous information network used for experiment

Using the dataset from CrunchBase, we build a heterogeneous information network based on the entity and relationship types mentioned in Section 3.1, where nodes represent entities and edges indicate relationships. Fig. 1 outlines the heterogeneous information network used for experiments. To construct the heterogeneous information network shown in Fig. 1, we have to first select entities as nodes and then connect the nodes based on the relationships between entities. Fig. 1 covers almost all of the usable information on the CrunchBase dataset except the information on URLs, serial numbers, and so on. After establishing the network, we can extract meta-paths and use them to model the similarity between an investor and a company. In the next section, we explain the meta-paths and their measure function in detail.

3.3. Meta-Paths in CrunchBase heterogeneous information network

In this section, we introduce the concept of *Meta-Path* and describe how to apply it to the CrunchBase Heterogeneous Information Network. The meta-path, a special type of path that connects two objects in a Heterogeneous Information Network, is defined as follows:

Definition 2. (Meta-Path) *A meta-path $\mathcal{P}$ is a path defined on the graph of network schema $T_G=(\mathcal{A}, \mathcal{R})$, and is denoted in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_{n-1}} A_n$, which defines a composite relation $R = R_1 \circ R_2 \circ \cdots \circ R_{n-1}$ between type A_1 and A_n, where $\circ$ denotes the composition operator on the relations.*

As mentioned in Definition 2, a meta-path is a path defined

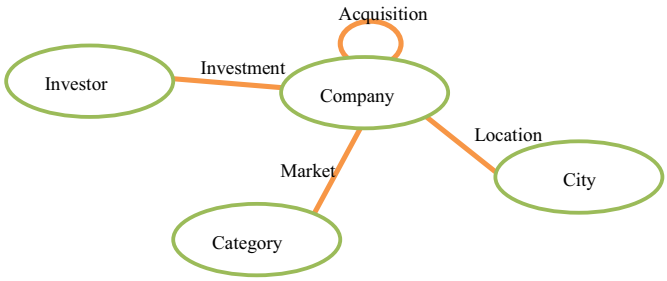

Fig. 1. Heterogeneous information network schema of CrunchBase dataset.

over a network schema, which denotes a composition relationship over a heterogeneous information network. The neighbor set-based and path-based features are defined in a homogeneous network, but both of them can be generalized in a heterogeneous network by considering paths that follow different meta-paths [4]. In the network shown in Fig. 1, we extract all the significant meta-paths within a length constraint less than 4, starting with the type *Inv* and ending with the type *Com*. In our experimental section, we explain in detail that longer meta-paths do not make sense in measuring the similarity between an investor and a company. Meta-paths between investor type and company type can be extracted by traversing the graph of network schema. Then, all the meta-paths between investors and companies are summarized in Table 1.

3.4. Measure function on meta-path

Once the meta-path-based topology (see Table 1) has been generated, the effective measure functions that quantify the topology are required.

Formula (1) is the *HeteSim* [21] between two objects a and b based on meta-path $\mathcal{P}$. *HeteSim* is the cosine of the probability distributions of the source object $a \in Inv$ and target object $b \in Com$, which reach the middle type object M. *HeteSim* ranges from 0 to 1.

$$HeteSim(a, b \mid \mathcal{P}) = \frac{PM_{\mathcal{P}_L}(a,:)PM^{-1}_{\mathcal{P}_R}(b,:)}{\sqrt{\left\|PM_{\mathcal{P}_L}(a,:)\right\|\left\|PM^{-1}_{\mathcal{P}_R}(b,:)\right\|}}, \quad (1)$$

where $PM_{\mathcal{P}}$ is an attainable probability matrix for meta-path p, and $PM_{\mathcal{P}}(i,j)$ represents the probability of object $i \in Inv$ reaching object $j \in Com$ based on meta-path $\mathcal{P}$. $PM_{\mathcal{P}}(a,:)$ is the a-th row in $PM_{\mathcal{P}}$ and is calculated by Formula (2). $PM^{-1}_{\mathcal{P}}$ is the transposed matrix of $PM_{\mathcal{P}}$. $\mathcal{P}_L$ and $\mathcal{P}_R$ are the left and right aspects of the meta-path, respectively.

$$PM_{\mathcal{P}} = U_{R_1}U_{R_2}\cdots U_{R_n}, \quad (2)$$

where U_{R_n} is the normalized matrix of the adjacent relationship matrix R_n. R_n is an element of the adjacent relationship matrix set $\mathcal{R}$, where $(R^*)^{-1}$ is the transposed matrix of R^*. Specifically, in our work, we use the Fig. 1 network schema, where the R_1 must be R^{Inv}.

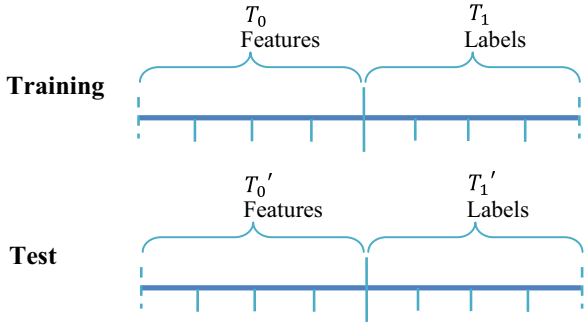

Fig. 2. Investment behavior prediction framework.

Path Count indicates the number of path instances between objects a and b following $\mathcal{P}$, as given by the following equation:

$$PC = |\{p : p \in \mathcal{P}\}|, \quad (3)$$

where p is a path instance between objects a and b.

Random Walk represents the random walk probability that starts from a and ends with b following meta-path $\mathcal{P}$, which is the sum of the probabilities of all the path instances $p \in \mathcal{P}$ starting from a and ending with b, as denoted by the following equation:

$$RW = \sum_{p \in \mathcal{P}} Prob(p). \quad (4)$$

3.5. Investment behavior prediction

As depicted in Fig. 2, the prediction framework consists of a training stage and a test stage. Given the training pairs of investors and companies, we first collect their associated heterogeneous network features extracted from the aggregated network in the time interval T_0. Then, we record the investment behavior building facts, represented as labels in the future interval T_1, that exist between the investors and the companies. Thereafter, we build a prediction model to learn the weights that are relevant to these heterogeneous features.

3.5.1. Investment behavior prediction model

To predict whether an investor will invest in a particular company in a future interval, denoted as y, we use the logistic regression model as the prediction model. For each training pair of investors and companies $<Inv_i, Com_j>$, which follows Bernoulli distribution with probability p_{ij}($P(y_{ij}=1)=p_{ij}$), (1) let $\boldsymbol{X}_{ij}$ be the $(d+1)$-dimensional vector, which includes constant 1 and d heterogeneous network features between investors and companies, and (2) y_{ij} be the label of whether an investor will invest in a company in the future ($y_{ij}=1$ if an investor will invest and 0 otherwise). The probability p_{ij} is modeled as follows:

Table 1
Significant Meta-Path under length 4 in CrunchBase network.

Meta-Path	Semantic meaning of the relation
Inv-Com-Inv-Com	Similarity of Inv_i and Com_j that was invested by Inv_j similar to Inv_i
Inv-Com-Cat-Com	Similarity of Inv_i and Com_j that market at Cat_i that Inv_j is interested in
Inv-Com-Cit-Com	Similarity of Inv_i and Com_j that is located at Cat_i where Inv_i invests more
Inv-Com-Com	Similarity of Inv_i and Com_j that was acquired by Com_i similar to Inv_i
Inv-Com-Com-Inv-Com	Similarity of Inv_i and Com_j that was invested by Inv_j similar to Inv_i and acquired by Inv_j invested Com_i
Inv-Com-Cat-Com-Com	Similarity of Inv_i and Com_j that was acquired by Com_i, which has a common category with Com_k that was invested by Inv_i
Inv-Com-Com-Cat-Com	Similarity of Inv_i and Com_j that market at Cat_i and has common categories with Com_i that was acquired by Com_k similar to Inv_i

Input :
DataSet: training dataset
γ: learning rate
Output:
$\boldsymbol{\beta}$: coefficients of meta-paths
1. Initializations:
(1)$\mathcal{A} = \{Inv, Com, Cit, Cat\}$;
(2)$\mathcal{R} = \{R^{Inv}, R^{Mar}, R^{Loc}, R^{Acq}\}$;
2. Build a heterogeneous information network schema:
(1)T_G = BuildNet($\mathcal{A}, \mathcal{R}$,DataSet);
3. Build significant meta-path set from heterogeneous information network:
(1)$\mathcal{P}$ = Generate(T_G);
4. Calculate $HeteSim(Inv, Com|\mathcal{P})$
(1) **For each** R_i in R
(2) **Begin**
(3) generate normalized adjacency matrix U_{R_i} = Generate(T_G, R_i);
(4) **End for**
(5) **For** each $\mathcal{P}_i$ in $\mathcal{P}$
(6) **Begin**
(7) $PM_{\mathcal{P}_i} = U_{R_1} U_{R_2} \cdots U_{R_n}$;
(8) $HeteSim(Inv, Com|\mathcal{P}_i) = \frac{PM_{\mathcal{P}_i}}{\sqrt{\|PM_{\mathcal{P}_i}\|}}$;
(9) **End for**
5. Train investment behavior prediction model
(1) SampleSet SS = GenerateSamples(DataSet);
(2) LabelSet $\boldsymbol{y}$ = GenerateLabels(SampleSet, Dataset);
(3) $\boldsymbol{\beta}$ = Random();
(4)do
(5) **For** i=1 to $|SS|$
(6) **Begin**
(7) **For** j=0 to $|\mathcal{P}|$
(8) **Begin**
(9) $x_j = HeteSim(Inv, Com|\mathcal{P}_{ij})$;
(10) $\beta_j = \beta_j - \gamma \cdot x_j \cdot (y_i - \exp(\beta_j \cdot x_j)/(1 + \exp(\beta_j \cdot x_j)))$;
(11) **End for**
(12) **End for**
(13)until convergence.

Fig. 3. Prediction model parameters learning.

$$p_{ij}=\frac{e^{\boldsymbol{X}_{ij}\boldsymbol{\beta}}}{e^{\boldsymbol{X}_{ij}\boldsymbol{\beta}}+1}, \tag{5}$$

where $\boldsymbol{\beta}$ is the $d+1$ coefficient weight relevant to the constant and each heterogeneous topological feature. Then, the regression coefficients are optimized in the regularization framework, i.e., the maximum log-likelihood estimate on the training dataset. The formula is described as follows:

$$\max_{\boldsymbol{\beta}} L(\boldsymbol{\beta})=\sum_{i\in Inv, j\in Com}\left[y_{ij}\ log\ p_{ij}+\left(1-y_{ij}\right)log\ p_{ij}\right], \tag{6}$$

where stochastic gradient descent method can be used to learn the coefficients. Finally, we present an integrated investment behavior prediction model of the supervised learning framework in Fig. 3.

3.5.2. *Investment behavior building time prediction model*

Previous studies that modeled investment behavior as link prediction focused on asking whether a link would be built in the future, i.e., "whether an investor will invest in a company." Nevertheless, we are interested in predicting when the investment will be built.

Thus, we propose the GLM-based [25] prediction model, which directly models the investment behavior building time as a function of meta-paths, and provides methods to learn the parameters of the model under different assumptions for investment behavior building time distributions.

The main idea of GLM is to model the response variable y, $E(y)$, as a function (link function) of the linear expression of meta-paths, that is, $\boldsymbol{X}_i\boldsymbol{\beta}$, where $\boldsymbol{X}_i$ is the $(d+1)$-dimensional vector including constant 1 and d meta-paths between investors and companies, and $\boldsymbol{\beta}$ is the coefficient vector. Then, the goal is to learn $\boldsymbol{\beta}$ according to the training data. Under different distribution assumptions for y, usually from the exponential family, $E(y)$ has different forms of parameter set, and the link functions also have different forms. The logistic regression is a special case of GML and y obeys Bernoulli distribution.

We first consider the exponential distribution [26], which is the most frequently used distribution in modeling the waiting time for an event. The probability density function of an exponential distribution is

$$f(y)=\frac{1}{\theta}e^{-\frac{y}{\theta}}, \tag{7}$$

where $y\geq 0$, and $\theta > 0$ is the parameter denoting the *mean waiting time* for the event. The cumulative distribution function is

$$F(y)=P(Y\leq y)=1-e^{-\frac{y}{\theta}}. \tag{8}$$

Then, we consider the Weibull distribution, which is a generalized version of exponential distribution. The probability density function of a Weibull distribution is

$$f(y)=\frac{\lambda y^{\lambda-1}}{\theta^{\lambda}}e^{-(y/\theta)^{\lambda}}, \tag{9}$$

where $y\geq 0$, and $\theta > 0$ and $\lambda > 0$ are two parameters related to *mean waiting time* for the event and *hazard of happening* of the

event along with the time. When $\lambda > 1$, an increasing happening rate occurs along the time (if an event does not happen at an early time, it has higher probability of happening at a later time); and when $\lambda < 1$, a decreasing happening rate occurs along the time (if an event does not happen at an early time, it has less possibility of happening at a later time). In our experiments, we set λ as equal to 0.95. When $\lambda = 1$, the Weibull distribution becomes exponential distribution with mean waiting time as θ, and the happening rate does not change along the time. The Weibull distribution cumulative distribution function is

$$F(y)=P(Y\le y)=1-e^{-(y/\theta)^{\lambda}}. \tag{10}$$

Model under Weibull Distribution. We only need to consider the prediction model with Weibull distribution (when $\lambda\le 1$).

In this case, we assume that investment behavior building time y_i for each train pair is independent of each other, following the same Weibull distribution with the same λ, but with different mean waiting time θ_i. Under this assumption, we can evaluate the expectation for each random variable y_i as $E(y_i)=\theta_i\Gamma(1+\frac{1}{\lambda})$. We then use the link function $E(y_i)=e^{-\boldsymbol{X_i\beta}}\Gamma(1+\frac{1}{\lambda})$, that is $\log\theta_i=-\beta_0-\sum_k^d X_i^k\beta_k=-\boldsymbol{X_i\beta}$, where β_0 is the constant term. Then, we can write the log-likelihood function as follows:

$$log\ L=\sum_i^n (f(y_i\theta_i,\lambda)I_{\{y_i<T\}}+P(y_i\ge T\theta_i,\lambda)I_{\{y_i\ge T\}}), \tag{11}$$

where $I_{\{y_i<T\}}$ and $I_{\{y_i\ge T\}}$ are indicator functions, which are equal to 1 if the predicate holds, or 0 otherwise. Eq. (11) means that if y_i is observed in a future interval, we use its density function; otherwise, we use the probability of $y_i>T$ in the function. In our experiments, we regarded $P(y_i\ge T\theta_i,\lambda)$ as zero.

By plugging in $log\ \theta_i=-\boldsymbol{X_i\beta}$, we can obtain the log likelihood with parameters β and λ as follows:

$$LL(\beta,\lambda)=\sum_{i=1}^n I_{\{y_i<T\}}\ log\ \frac{\lambda y_i^{\lambda-1}}{e^{-\lambda\boldsymbol{X_i\beta}}}-\sum_{i=1}^n\left(\frac{y_i}{e^{-\boldsymbol{X_i\beta}}}\right)^{\lambda}, \tag{12}$$

where LL denotes the log-likelihood function under Weibull distribution [27].

The learning of a model is an optimization problem, which aims to find $\hat{\beta}$ and $\hat{\lambda}$ that maximize the log likelihood. Newton–Raphson method can be used to derive the update formulas.

4. Experiments

To validate the proposed model, we performed our experiments on the real-world data from CrunchBase, and evaluated the prediction more efficiently than did the previous studies. We applied AUC, used in previous research, as our evaluation metric to compare our experimental results. Note that the positive and negative samples are severely imbalanced, and we randomize (equal probability) the negative samples to the usual level.

4.1. Aggregate performance comparison

We compare the experimental results based on meta-path heterogeneous information network with the results based on a homogeneous network. Overall, the performance of all algorithms exceeded the baseline performance of 0.6 for AUC. For the meta-paths extracted from the heterogeneous information network, we used *HeteSim*, *PathCount*, and *RandomWalk* to measure the similarity of the seven meta-paths listed in Table 1. The comparison results are summarized in Fig. 4.

We determine the HomoDT, HomoSVM, and HomoNB model

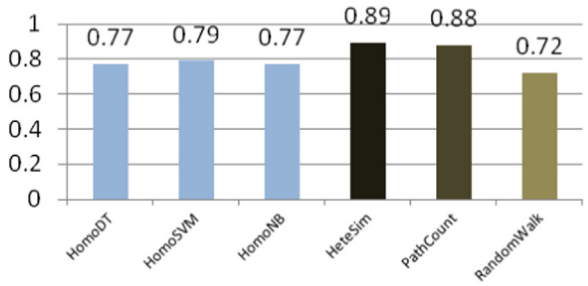

Fig. 4. Investment behavior prediction model comparison of homogeneous and heterogeneous networks.

investment behavior as a link prediction problem, and then extract multiple link prediction features from the homogeneous network. They are distinct from using difference learning algorithms as follows: DT refers to decision tree, SVM refers to support vector machine, and NB refers to naive Bayes. *HeteSim*, *PathCount*, and *RandomWalk* refer to model investment behavior as a heterogeneous information network relationship prediction problem, and we use *HeteSim*, *PathCount*, and *RandomWalk*, respectively, to measure similarity for meta-path. Experimental results using the measure function of *HeteSim* and *PathCount* reveal that our prediction models can predict investment behavior more effectively than modeling that use link prediction in a homogeneous information network.

4.2. Significant of meta-path study

In this section, we show the learned significance of meta-path for each topological feature in deciding the relationship between investors and companies built on CrunchBase. Table 2 presents the coefficients for all the seven meta-paths according to their different measure functions.

Considering the results shown in Table 2, we summarize the following three findings: (1) On the one hand, HeteSim and PathCount obtain better performance than RandomWalk, which indicates that the significant rankings of meta-paths are similar to both measure functions. (2) On the other hand, overall, HeteSim is the best measure of investment behavior prediction. Meanwhile, because the *p*-Value of meta-paths of Inv-Com-Inv-Com, Inv-Com-Cat-Com, Inv-Com-Cit-Com, Inv-Com-Com, and Inv-Com-Com-Inv-Com have positive promotion, we infer that the relationship type of Market, Investment, and Geography [3] prompts investors to invest in companies, but in PathCount and RandWalk the non-significant of Inv-Com-Com reveal that Acquisition may be not directly associated with investment behavior. (3) Finally, according to the experimental results of different measure functions of the six groups, we find that the significance of the meta-path weakens as the length of the path increases.

Table 2
Significant of meta-paths with different measure function.

Meta-paths	*p*-Value		
	HeteSim	**PathCount**	**RandomWalk**
Inv-Com-Inv-Com	< 2e − 16(***)	< 2e − 16(***)	< 2e-16(***)
Inv-Com-Cat-Com	< 2e − 16(***)	< 2e − 16(***)	< 2e-16(***)
Inv-Com-Cit-Com	< 2e − 16(***)	< 2e − 16 (***)	< 2e − 16(***)
Inv-Com-Com	0.0218(*)	0.9964( )	0.0556(.)
Inv-Com-Com-Inv-Com	8.79e − 7(***)	0.0335(*)	0.8813( )
Inv-Com-Cat-Com-Com	0.0968(.)	0.3801( )	0.2682( )
Inv-Com-Com-Cat-Com	0.0807(.)	0.0180(*)	0.2239( )

0: '***'; 0.001: '**'; 0.01: '*'; 0.05: '.'; 0.1: ' '.

Table 3
Detailed time intervals.

Group	T_0	T_1	T_2
T^1	2007/11-2010/11	2010/11-2013/11	2013/11-2014/05
T^2	2007/05-2010/05	2010/05-2013/05	2013/05-2014/05
T^3	2006/11-2009/11	2009/11-2012/11	2012/11-2014/05
T^4	2006/05-2009/05	2009/05-2012/05	2012/05-2014/05
T^5	2005/11-2008/11	2008/11-2011/11	2011/11-2014/05
T^6	2005/05-2008/05	2008/05-2011/05	2011/05-2014/05

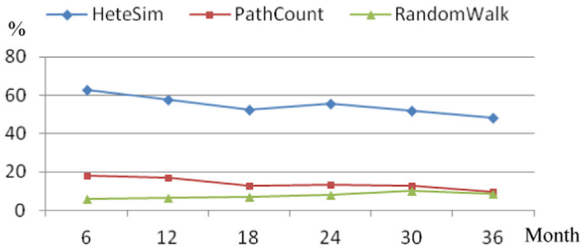

Fig. 5. F-score comparison of different time interval in the test future interval.

4.3. Investment behavior prediction with time

We propose a prediction model based on the linear model introduced in Section 3. Our prediction model learns a function on meta-path features to predict investment behavior building time. Our model uses the maximum log-likelihood estimate method to learn the coefficients of each heterogeneous network feature under different time interval assumptions for investment behavior building-time distributions.

To validate the effect of time on meta-path-based heterogeneous information network investment behavior predictions, we set the training future interval to be unequal to the test future interval ($T^{train} \neq T^{test}$). Therefore, in the test future interval (T), we consider six groups of different time intervals as follows: $T=\{T^1, T^2, T^3, T^4, T^5, T^6\}$. Table 3 shows the details of the time intervals.

In the recommendation system, the F1-score [28] is the most widely used for recommendation quality comprehensive measurement. The evaluation accuracy of the investment behavior prediction algorithm is similar to the evaluation accuracy of the recommendation algorithm. Therefore, in the experimental section, we regard the F-score as performance metric. Fig. 5 shows a summary F-score of four groups, each with a different test future interval and three different measure functions. The results reflect that our prediction model has good generalization power in time. Also, in our model, *HeteSim* has remarkable prediction effectiveness relative to *PathCount* and *RandomWalk*.

Table 4 shows the experimental results of three different measure functions in four groups of future time intervals. The results reveal that *HeteSim* obtains better performance in recall than *RandomWalk*, and a more reasonable performance than *PathCount* both in recall and precision.

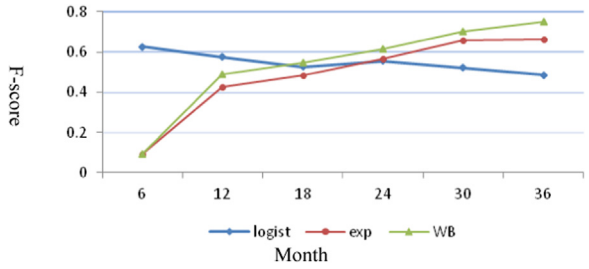

Fig. 6. F-score comparison of different models prediction results.

4.3.1. Different prediction model comparison

In Section 3.5, we denote our model with another different distribution assumption as exponential and Weibull distribution, respectively. To demonstrate the power of using time-involved model in investment behavior prediction, we use logistic regression as the baseline. The output of the logistic regression is a probability that denotes whether a relationship will be built in T_1 for each test pair. In the exponential and Weibull distribution models, the output is the parameter set for distribution of the investment building time. To compare the three models with F-score, we define the following indicator function $I(y_i, T_1)$ as follows:

$$I\left(\hat{y}_i, T_1\right)=\begin{cases} 1, \hat{y}_i \le T_1 \\ 0, \hat{y}_i > T_1, \end{cases} \tag{13}$$

where $I\left(\hat{y}_i, T_1\right)=1$ refers to correct prediction result, or is wrong otherwise; then, we apply *HeteSim* to measure similarity.

As shown in Fig. 6, the exponential and Weibull distribution models can improve their prediction of the longer-term investment building behavior than logistic regression. On the other hand, the exponential and Weibull distribution models can infer much more information rather than a simple probability as to whether an investor will invest in a company.

5. Conclusion

In this study, we designed a data-driven investment behavior prediction framework that models investment behavior prediction as a meta-path-based heterogeneous information network relationship prediction problem. Specifically, we first built a heterogeneous relationship network schema based on entity and relationship types. Then, based on the heterogeneous relationship network, we extracted all significant meta-path lengths that are equal to or less than 4, and provided the effective measure functions to quantify the meta-path topology. Finally, we proposed a supervised learning investment behavior prediction model to

Table 4
Prediction generalization power and effectiveness comparison.

Months	HeteSim (%)			PathCount (%)			RandomWalk (%)		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
6	99.4	45.8	**62.7**	23.3	14.9	18.2	99.5	3.1	6.0
12	99.5	40.6	**57.6**	21.6	14.0	17.0	98.3	3.5	6.7
18	99.6	35.8	**52.7**	20.9	14.1	16.8	99.7	3.7	7.1
24	99.4	38.4	**55.5**	20.0	10.0	13.3	99.5	4.3	8.2
30	99.6	35.3	**52.1**	24.4	8.7	12.8	84.5	5.3	10.0
36	99.5	32.1	**48.5**	23.9	5.3	8.7	98.3	5.1	9.7

predict investment behavior. Experimental results reveal that our model is also a useful indicator to help companies (1) improve their understanding of how and when investors invest, and (2) improve their preparation when they are attempting to seek external investment. The related methods can be extended to the network analysis in other fields, such as bioinformatics [29–32], image processing [33–36], big data [37–41], machine learning research [42–45], neural networks [46–49], and spiking neural model-based optimization [50–54].

Acknowledgement

The work is supported by National Natural Science Foundation of China (Grant nos. 61472333, 61202011 and 61303004), Ph.D. Programs Foundation of Ministry of Education of China (20120121120039).

References

[1] Paul Belleflamme, T. Lambert, A. Schwienbacher, Crowdfunding: tapping the right crowd, J. Bus. Ventur. 29 (5) (2014) 585–609.
[2] P. Giot, U. Hege, A. Schwienbaher, Expertise or Reputation? The Investment Behavior of Novice and Experienced Private Equity Funds. Available at (⟨www.fbfcorporatefinance.fr/medias/cahiers_de_recherche/VCFLOWS_2011-09-19.pdf⟩).
[3] M. Grinblatt, M. Keloharju, The investment behavior and performance of various investor types: a study of Finland's unique data set, J. Financ. Econ. 55 (2000) 43–67.
[4] J.S. Doran, D.R. Peterson, C. Wright, Confidence opinions of market efficiency and investment behavior of finance professors, J. Financ. Mark. 13 (1) (2010) 174–195.
[5] A. Barnea, H. Cronqvist, S. Siegel, Nature or nurture: what determines investment behavior? J. Financ. Econ. 98 (2010) 583–604.
[6] Q. Zou, J. Li, C. Wang, X. Zeng, Approaches for recognizing disease genes based on network, Biomed. Res. Int. 1 (2014) 129–138.
[7] Q. Zou, J. Li, Q. Hong, Z. Lin, H. Shi, Y. Wu, Y. Ju, Prediction of microRNA-disease associations based on social network analysis methods, Biomed. Res. Int. 2015 (2015).
[8] Kleinberg, Liben Nowell Jon, The link-prediction problem for social networks, J. Am. Soc. Inf. Sci. Technol. 58 (7) (2003) 1019–1031.
[9] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu, RankClus: integrating clustering with ranking for Heterogeneous Information Network analysis, in: EDBT, 2009, pp. 565–576.
[10] Y.E. Liang, S.D. Yuan, Investors are social animals: predicting investor behavior using social network features via supervised learning approach, in: Proceedings of the Workshop on Mining and Learning with Graphs(MLG-2013), 2013, pp. 305–312.
[11] X. Zhang, L. Pan, A. Păun, On universality of axon P systems, IEEE Trans. Neural Netw. Learn. Syst. 26 (11) (2015) 2816–2829.
[12] X. Zhang, Y. Liu, B. Luo, L. Pan, Computational power of tissue P systems for generating control languages, Inf. Sci. 278 (10) (2014) 285–297.
[13] X. Zhang, Y. Tian, Y. Jin, A knee point driven evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation, ⟨http://dx.doi.org/10.1109/TEVC.2014.2378512⟩.
[14] X. Zhang, Y. Tian, R. Cheng, Y. Jin, An efficient approach to non-dominated sorting for evolutionary multi-objective optimization, IEEE Trans. Evolut. Comput. 19 (2) (2015) 201–213.
[15] Y.E. Liang, D.Y. Sor-Tsyr, Where's the Money? The Social Behavior of Investors in Facebook's Small World, in: 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, 2012, pp. 158–162.
[16] G. Xiang, Z. Zheng, M. Wen, J. Hong, C. Rose, and C. Liu, A supervised approach to predict company acquisition with factual and topic features using profiles and new articles on TechCrunch, in: ICWSM'12, 2012.
[17] Y. Sun, R. Barber, M. Gupta, C.C. Aggarwal, J. Han, Co-Author relationship prediction in heterogeneous bibliographic networks, in: 2011 International Conference on Advances in Social Networks Analysis and Mining, 2011, pp. 121–128.
[18] Y. Sun, J. Han, Meta-path-based search and mining in heterogeneous information networks, Tsinghua Sci. Technol. 4 (2013) 329–338.
[19] Y. Sun, J. Han, X. Yan, P. S. Yu, T. Wu, Pathsim: Meta path-based top-k similarity search in Heterogeneous Information Networks, in: Proc. 2011 Int. Conf. on Very Large Data Bases (VLDB'11), Seattle, WA, August 2011, pp. 992–1003.
[20] R. N. Lichtenwalter, J. T. Lussier, N. V. Chawla, New perspectives and methods in link prediction, in: Proceedings of the 2010 ACM SIGKDD Conference on Knowledge Discovery and Data Mining(KDD'10), Washington D.C, July 2010, pp. 243–252.
[21] C. Shi, X. Kong, P.S. Yu, S. Xie, B. Wu, Relevance search in heterogeneous networks in: EDBT 2012, 2012, pp. 180–191.
[22] X. Zeng, X. Zhang, Q. Zou, Integrative approaches for predicting microRNA function and prioritizing disease-related microRNA using biological interaction networks, Briefings Bioinforma. 17 (2) (2016) 193–203.
[23] Q. Zou, J. Li, L. Song, X. Zeng, G. Wang, Similarity computation strategies in the microRNA-disease network: A Survey, Briefings in Functional Genomics, ⟨http://dx.doi.org/10.1093/bfgp/elv024⟩.
[24] P. Li, M. Guo, C. Wang, X. Liu, Q. Zou, An overview of SNP interactions in genome-wide association studies, Briefings Funct. Genom. 14 (3) (2014) 143–155.
[25] P. McCullagh, Generalized linear models, Eur. J. Oper. Res. 16 (3) (1984) 285–292.
[26] A.W. Marshall, I. Olkin, A multivariate exponential distribution, J. Am. Stat. Assoc. 62 (317) (1967) 30–44.
[27] A.C. Cohen, Maximum likelihood estimation in the Weibull distribution based on complete and on censored samples, Technometrics 7 (4) (2012) 579–588.
[28] Y. Yang, An evaluation of statistical approaches to text categorization, Inf. Retr. 1 (1999) 69–90.
[29] B. Liu, F. Liu, X. Wang, J. Chen, L. Fang, K.-C. Chou, Pse-in-One: a web server for generating various modes of pseudo components of DNA, RNA, and protein sequences, Nucleic Acids Res. W1 (2015) W65–W71.
[30] S. Li, D. Li, X. Zeng, Y. Wu, G. Li, Q. Zou, nDNA-prot: Identification of DNA-binding proteins based on unbalanced classification, BMC Bioinformatics 15 (17) (2014) 3600 -3600.
[31] Q. Zou, Y. Mao, L. Hu, Y. Wu, Z. Ji, miRClassify: An advanced web server for miRNA family classification and annotation, Comput. Biol. Med. 45 (2) (2014) 157–160.
[32] Q. Zou, X. Li, W. Jiang, Z. Lin, G. Li, K. Chen, Survey of MapReduce frame operation in bioinformatics, Briefings Bioinforma. 15 (4) (2014) 637–647.
[33] J. Li, X. Li, B. Yang, X. Sun, Segmentation-based image copy-move forgery detection scheme, IEEE Trans. Inf. Forensics Secur. 10 (3) (2015) 507–518.
[34] R. Ji, Y. Gao, R. Hong, et al., Spectral-spatial constraint hyperspectral image classifi-cation, IEEE Trans. Geosci. Remote. Sens. 52 (3) (2014) 1811–1824.
[35] B. Zhong, Y. Chen, Y. Shen, et al., Robust tracking via patch-based appearance model and local background estimation, Neurocomputing 123 (2014) 344–353.
[36] B. Zhong, X. Yuan, R. Ji, et al., Structured partial least squares for simultaneous object tracking and segmentation, Neurocomputing 133 (2014) 317–327.
[37] L. Wei, M. Liao, X. Gao, Q. Zou, An improved protein structural prediction method by incorporating both sequence and structure information, IEEE Trans. Nanobiosci. 14 (4) (2015) 339–349.
[38] C. Lin, W. Chen, C. Qiu, Y. Wu, S. Krishnan, Q. Zou, LibD3C: ensemble classifiers with a clustering and dynamic selection strategy, Neurocomputing 123 (2014) 424–435.
[39] L. Wei, M. Liao, Y. Gao, R. Ji, Z. He, Q. Zou, Improved and promising identification of human MicroRNAs by incorporating a high-quality negative set, IEEE/ACM Trans. Comput. Biol. Bioinforma. 11 (1) (2014) 192–201.
[40] T. Ma, J. Zhou, M. Tang, et al., Social network and tag sources based augmenting collaborative recommender system, IEICE Trans. Inf. Syst. E98-D 4 (2015) 902–910.
[41] Q. Zou, Q. Hu, M. Guo, G. Wang, HAlign: Fast multiple similar dna/rna sequence alignment based on the centre star strategy, Bioinformatics, ⟨http://dx.doi.org/10.1093/bioinformatics/btv177⟩.
[42] B. Chen, H. Shu, G. Coatrieux, et al., Color image analysis by quaternion-type moments, J. Math. Imaging Vision. 51 (1) (2015) 124–144.
[43] R. Ji, H. Yao, W. Liu, et al., Task-dependent visual-codebook compression, IEEE Trans. Image Process. 21 (4) (2012) 2282–2293.
[44] B. Gu, V. Sheng, Z. Wang, et al., Incremental learning for ν-support vector regression, Neural Netw. 67 (2015) 140–150.
[45] X. Wen, L. Shao, Y. Xue, W. Fang, A rapid learning algorithm for vehicle classification, Inf. Sci. 295 (1) (2015) 395–406.
[46] T. Song, X. Zeng, X. Liu, Asynchronous spiking neural P systems with rules on synapses, Neurocomputing 151 (1) (2015) 1439–1445.
[47] X. Zhang, B. Wang, L. Pan, Spiking neural P systems with a generalized use of rules, Neural Comput. 26 (12) (2014) 2925–2943.
[48] X. Liu, Z. Li, J. Liu, L. Liu, X. Zeng, Implementation of arithmetic operations with time-free spiking neural P systems, IEEE Trans. Nanobiosci. 14 (6) (2015) 617–624.
[49] X. Liu, J. Suo, S. Leung, J. Liu, X. Zeng, The power of time-free tissue P systems: attacking NP-complete problems, NeuroComputing 159 (2015) 151–156.
[50] T. Song, L. Pan, J. Wang, I. Venkat, K.G. Subramanian, R. Abdullah, Normal forms of spiking neural P systems with anti-spikes, IEEE Trans. Nanobiosci. 4 (3) (2012) 352–359.
[51] X. Zeng, X. Zhang, T. Song, L. Pan, Spiking Neural P Systems with thresholds, Neural Comput. 26 (7) (2014) 1340–1361.
[52] X. Zhang, X. Zeng, B. Luo, L. Pan, On some classes of sequential spiking neural P systems, Neural Comput. 26 (5) (2014) 974–997.
[53] T. Song, L. Pan, Spiking neural P systems with rules on synapses working in maximum spiking strategy, IEEE Trans. Nanobiosci. 14 (1) (2015) 465–477.
[54] X. Zeng, L. Xu, X. Liu, L. Pan, On languages generated by spiking neural P systems with weights, Inf. Sci. 278 (2014) 423–433.

Xiangxiang Zeng received the B.S. degree in automation from Hunan University, China, in 2005, the Ph.D. degree in system engineering from Huazhong University of Science and Technology, China, in 2011. From 2009 to 2010 he spent one year working in the group of natural computing in Seville University, Spain. Currently, he is an associate professor in the Department of Computer Science, Xiamen University. His main research interests include natural computing, neural computing and automaton theory.

You Li is a Master student of the Department of Computer Science at Xiamen University. He received his B.E. in Information and Computing Science from WeiFang University, Shandong, China. His research interests include Data Mining and Machine Learning.

Stephen Leung received his Ph.D. degree in Operational Research and Management Science from City University of Hong Kong, the M.S. degree in Operational Research and the B.S. degree in Mathematical Science. He is a fellow of the Chartered Institute of Logistics and Transport and the Institute of Mathematics and its Applications, and a corporate member of the Hong Kong Society for Transportation Studies and the Institute of Mathematics and its Applications.

Ziyu Lin, Ph.D., received the Ph.D. degree in Peking University, China, in 2009. He is an assistant professor in the Department of Computer Science, Xiamen University. His main research fields focus on data mining and databases.

Xiangrong Liu, Ph.D., professor. He received the B.S. degree in Biomedicine Engineering from Huazhong University of Science and Technology, China, in 2000, the Ph.D. degree in system engineering from Huazhong University of Science and Technology, China, in 2007. From 2007 to 2009 he spent two year working in the department of computer in Peking University for postdoctoral research. From 2009, he is an associate professor in the Department of Computer Science, Xiamen University. His main research fields focus on biomolecular computing, neural computing and bioinformatics.