Relevance Search in Heterogeneous Networks

Chuan Shi
Beijing University of Posts and Telecommunications
Beijing, China
shichuan@bupt.edu.cn

Xiangnan Kong
University of Illinois at Chicago
IL, USA
xkong4@uic.edu

Philip S. Yu
University of Illinois at Chicago, IL, USA
King Abdulaziz University
Jeddah, Saudi Arabia
psyu@cs.uic.edu

Sihong Xie
University of Illinois at Chicago
IL, USA
xiesihong1@gmail.com

Bin Wu
Beijing University of Posts and Telecommunications
Beijing, China
wubin@bupt.edu.cn

ABSTRACT

Conventional research on similarity search focuses on measuring the similarity between objects with the same type. However, in many real-world applications, we need to measure the relatedness between objects with different types. For example, in automatic expert profiling, people are interested in finding the most relevant objects to an expert, where the objects can be of various types, such as research areas, conferences and papers, etc. With the surge of study on heterogeneous networks, the relatedness measure on objects with different types becomes increasingly important. In this paper, we study the relevance search problem in heterogeneous networks, where the task is to measure the relatedness of heterogeneous objects (including objects with the same type or different types). We propose a novel measure, called HeteSim, with the following attributes: (1) a path-constrained measure: the relatedness of object pairs are defined based on the search path that connect two objects through following a sequence of node types; (2) a uniform measure: it can measure the relatedness of objects with the same or different types in a uniform framework; (3) a semi-metric measure: HeteSim has some good properties (e.g., self-maximum and symmetric), that are crucial to many tasks. Empirical studies show that HeteSim can effectively evaluate the relatedness of heterogeneous objects. Moreover, in the query and clustering tasks, it can achieve better performances than conventional measures.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications-Data Mining

General Terms

Algorithm

EDBT 2012, March 26–30, 2012, Berlin, Germany.

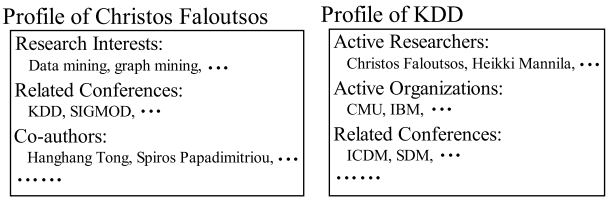
Keywords

Heterogeneous information network, similarity search, pairwise random walk

1. INTRODUCTION

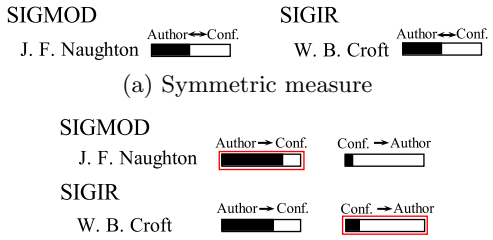
Similarity search is an important task in a wide range of applications, such as web search [17] and product recommendations [10]. The key of similarity search is similarity measure, which evaluates the similarity of object pairs. Similarity measure has been extensively studied for traditional categorical and numerical data types, such as Jaccard coefficient and cosine similarity. There are also a few studies on leveraging link information in networks to measure the node similarity, such as Personalized PageRank [6], SimRank [5], and PathSim [19]. Conventional study on similarity measure focuses on same-typed objects. That is, the objects being measured are of the same type, such as "document-to-document", "webpage-to-webpage" and "user-to-user". There is seldom research on similarity measure on different-typed objects. That is, the objects being measured are of different types, such as "author-to-conference" and "user-to-movie". It is reasonable. The similarity of different-typed objects is a little against our common sense. Moreover, different from the similarity of same-typed objects which can be easily measured on homogeneous situation (e.g., the same feature space or homogeneous link structure), it is hard to effectively define the similarity of different-typed objects.

However, the information of the relatedness of different-typed objects is not only meaningful but also useful in some scenarios. For example, the author J. F. Naughton is more relevant to SIGMOD than KDD. A teenager may like the movie "Harry Potter" more than "The Shawshank Redemption". Moreover, the relatedness measure of different-typed objects are needed in many applications. For example, in a recommendation system, we need to know the relatedness between users and movies to make accurate recommendations. In an automatic profile extraction application as shown in Fig. 1, we need to measure the relatedness of different-typed objects, such as authors and conferences, conferences and organizations etc. Particularly, with the advent of study on heterogeneous information networks [14,



(a) Author profile (b) Conference profile

Figure 1: Examples of automatically extracting academic profile.



(a) Symmetric measure

(b) Asymmetric measure

Figure 2: Examples of relative importance representing by symmetric and asymmetric measures. The rectangle with partially marked black denotes the relatedness of two objects.

20], it is not only increasingly important but also feasible to study the relatedness among different-typed objects. Heterogeneous information networks are the logical networks involving multiple-typed objects and multiple-typed links denoting different relations. It is clear that heterogeneous information networks are ubiquitous and form a critical component of modern information infrastructure. For example, a bibliographic network includes authors, papers, conferences, terms and their links representing their relations. It is essential to provide a relevance search function on different-typed objects in such networks, which is the base of many applications. Since different-typed objects coexist in the same network, so the relatedness measure on different-typed objects is possible through their link structure.

In this paper, we study the relevance search problem in heterogeneous information networks. The aim of relevance search is to effectively measure the relatedness of heterogeneous objects (including objects with the same type or different types). Different from the similarity search which only measures the similarity of same-typed objects, the relevance search measures the relatedness of heterogeneous objects, not limit to same-typed objects. Moreover, the relevance measure should be symmetric based on following reasons. (1) The symmetric measure is needed in many learning tasks, such as clustering and collaborative filtering. (2) The symmetric measure makes more sense in many applications. For example, in some applications, we need to answer the question like who has the similar importance to the conference SIGIR as J. F. Naughton to the SIGMOD. Through comparing the relatedness of object pairs, we can deduce the information of their relative importance. However, it only can be done by the symmetric measure, not the asymmetric measure. It can be explained by the example shown in Fig. 2. For the symmetric measure, we can deduce that W. B. Croft has the same importance to SIGIR as J. F. Naughton to the SIGMOD, since their relatedness scores are close. Suppose we know J. F. Naughton[1] is an influential researcher in SIGMOD, we can conclude that W. B. Croft[2] is also an influential researcher in SIGIR. However, we cannot deduce the relative importance information from an asymmetric measure as shown in Fig. 2(b). From the relatedness of author to conference and conference to author, we will draw conflicting conclusions.

Despite its value and significance, the relevance search in heterogeneous networks has never been studied so far. It faces the following research challenges. (1) Heterogeneous network is more complex than traditional homogeneous network. In heterogeneous networks, different-typed objects and links carry different semantic meanings. So the semantic meanings also contain in search paths that connect two objects through a sequence of relations between object types. Different paths have different semantics. Based on different search paths, the relatedness of two objects may be different. For example, the relatedness of authors and conferences should be different based on the relations of authors publishing papers in conferences and authors's co-authors publishing papers in conferences. As a consequence, a desirable relevance measure should be path-dependent, which can capture the semantics under paths and return different values based on different paths. (2) It is difficult to design a symmetric relevance measure for different-typed objects. In heterogeneous networks, the paths connecting same-typed objects are usually symmetric [19], so it is not difficult to design a symmetric measure based on the symmetric paths. However, the paths connecting different-typed objects are asymmetric. And thus it is more challenging to design a symmetric relevance measure based on an asymmetric path.

Inspired by the intuition that two objects are related if they are referenced by related objects, we propose a novel measure, called HeteSim, to evaluate the relatedness of heterogeneous objects in heterogeneous networks. Based on the path-based relevance framework, HeteSim can effectively capture the subtle semantics of search paths. Based on pairwise random walk, HeteSim treats arbitrary search paths in a uniform way, which guarantees the symmetric property of HeteSim. An additional benefit is that HeteSim can measure the relatedness of different-typed objects as well as same-typed objects. Moreover, HeteSim is a semi-metric measure. In other words, HeteSim satisfies the properties of non-negativity, identity of indiscernibles, and symmetry. It implies that HeteSim can be used in many learning tasks (e.g., clustering and collaborative filtering). The extensive experiments validate the effectiveness of HeteSim. There case studies illustrate the benefits of the relevance search in knowledge discovery of heterogeneous networks: automatically extracting object profile, experts finding through relative importance of object pairs, and relevance search based on path semantics. HeteSim also shows its potential in the machine learning tasks (i.e., query and clustering) where

[1]http://pages.cs.wisc.edu/∼naughton/
[2]http://ciir.cs.umass.edu/personnel/croft.html

HeteSim outperforms other path-based similarity measures.

The rest of the paper is organized as follows. Section 2 introduces the related work, and then the preliminary knowledge of this work is given in Section 3. In Section 4, we present the HeteSim measure. We do extensive experiments to validate the effectiveness of HeteSim in Section 5. Finally, Section 6 concludes this paper.

2. RELATED WORK

The most related work to relevance search is similarity search. Here we briefly summarize these works. Similarity search has been well studied for a long time. These studies can be roughly categorized into two types: feature based approaches and link based approaches. The feature based approaches measure the similarity of objects based on their feature values, such as cosine similarity, Jaccard coefficient and Euclidean distance. The k nearest neighbor is also widely used in similarity measure [2, 9], which aims at finding top-k nearest neighbors according to similarities defined on numerical features. Based on feature similarity, the top-k similarity pair search algorithm (i.e., top-k-join) considers similarity between tuples [23]. This type of approaches does not consider link relation among objects, so they cannot be applied to networked data.

The link based approaches measure the similarity of objects based on their link structures in a graph. The asymmetrical similarity measure, Personalized PageRank [6], evaluates the probability starting from a source object to a target object by randomly walking through the network with restart. It is extended to the scalable calculation for online queries [3, 21] and the top-k answers [4]. SimRank [5] is a symmetric similarity measure, which evaluates the similarity of two objects by their neighbor's similarities. Because of its computational complexity, many follow-up studies are done to speedup such calculations [13, 15]. SCAN [24] measures similarity of two objects by comparing their immediate neighbor sets. Recently, Jin et al. proposed RoleSim to measure the role similarity between any two nodes from networks [8]. These approaches just consider the objects with the same type, so they can not be applied in heterogeneous networks. ObjectRank [1] applies authority-based ranking to keyword search in labeled graphs and PopRank [16] proposes a domain-independent object-level link analysis model. Although these two approaches noticed that heterogeneous relationships could affect the similarity, they do not consider the distinct semantics of paths that include different-typed objects, so they also cannot measure the similarity of objects in heterogeneous networks.

Recently, Sun et al. [19] studied the similarity search on heterogeneous information networks. Considering semantics in meta paths constituted by different-typed objects, they proposed PathSim to measure the similarity of same-typed objects based on symmetric paths. However, many valuable paths are asymmetric and the relatedness of different-typed objects are also meaningful. PathSim is not suitable in these conditions. In information retrieval community, Lao and Cohen [11] proposed a Path Constrained Random Walk (PCRW) model to measure the entity proximity in a labeled directed graph constructed by the rich metadata of scientific literature. Although the PCRW model can be applied to measure the relatedness of different-typed objects, the asymmetric property of PCRW restricts it from being a relevance measure. In our HeteSim definition, users can measure the relatedness of heterogeneous objects based on an arbitrary search path. The good merits of HeteSim (e.g., symmetric and self-maximum) make it suitable for more applications.

3. PRELIMINARY

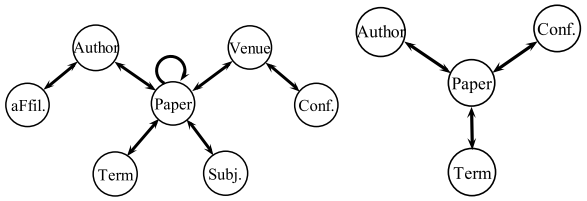
A heterogeneous information network is a special type of information network with the underneath data structure as a directed graph, which either contains multiple types of objects or multiple types of links.

DEFINITION 1. ***Information Network.*** *Given a schema $S = (\mathcal{A}, \mathcal{R})$ which consists of a set of entities types $\mathcal{A} = \{A\}$ and a set of relations $\mathcal{R} = \{R\}$, an information network is defined as a directed graph $G = (V, E)$ with an object type mapping function $\phi : V \to \mathcal{A}$ and a link type mapping function $\psi : E \to \mathcal{R}$. Each object $v \in V$ belongs to one particular object type $\phi(v) \in \mathcal{A}$, and each link $e \in E$ belongs to a particular relation $\psi(e) \in \mathcal{R}$. When the types of objects $|\mathcal{A}| > 1$ or the types of relations $|\mathcal{R}| > 1$, the network is called* ***heterogeneous information network****; otherwise, it is a* ***homogeneous information network****.*

In information networks, we distinguish object types and relation types. As a template for a network, the network schema depicts the object types and the relations existing among object types. For a relation R existing from type A to type B, denoted as $A \xrightarrow{R} B$, A and B are the **source type** and **target type** of relation R, which is denoted as $R.S$ and $R.T$, respectively. The inverse relation R^{-1} holds naturally for $B \xrightarrow{R^{-1}} A$. Generally, R is not equal to R^{-1}, unless R is symmetric and these two types are the same.

EXAMPLE 1. *A bibliographic information network is a typical heterogeneous information network. The network schema of ACM dataset (see Section 5.1) is shown in Fig.3(a). It contains objects from seven types of entities: papers (P), authors (A), affiliations (F), terms (T), subjects (S), venues (V), and conferences (C) (a conference includes multiple venues, e.g., KDD including KDD2010, KDD2009 and so on). There are links connecting different-typed objects. The link types are defined by the relations between two object types. For example, links exist between authors and papers denoting the writing or written-by relations, between venues and papers denoting the publishing or published-in relations. Fig.3(b) shows the network schema of DBLP dataset (see Section 5.1).*

Different from homogeneous networks, two objects in a heterogeneous network can be connected via different paths and these paths have different meanings. For example, in Fig. 3(a), conferences and authors can be connected via "Author-Paper-Venue-Conference" ($APVC$) path, "Author-Paper-Subject-Paper-Venue-Conference" ($APSPVC$) path, and so on. It is clear that semantics underneath these paths are different. The $APVC$ path means that papers written by authors are published in conferences, while the $APSPVC$ path means that papers having the same subjects as the authors' papers are published in conferences. Obviously, the



(a) ACM data (b) DBLP data

Figure 3: Examples of bibliographic network schema.

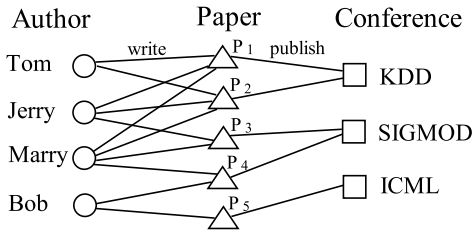


Figure 4: A simple heterogeneous network example.

distinct semantics under different paths will lead to different relatedness. The relatedness under $APVC$ path emphasizes on the conferences that authors participated, while the relatedness under $APSPVC$ path emphasizes on conferences publishing the papers that have the same subjects with authors' papers. For example, assume most papers of an author are published in the KDD, SIGMOD, and VLDB. However, the papers having the same subjects with the author's papers may be published in more wide conferences, such as ICDM, SDM, and CIKM. So the relatedness of objects depends on the search path in the heterogeneous networks. Formally, we define the meta search path as the relevance path.

DEFINITION 2. ***Relevance Path.*** *A relevance path $\mathcal{P}$ is a path defined on a schema $S = (\mathcal{A}, \mathcal{R})$, and is denoted in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$ which defines a composite relation $R = R_1 \circ R_2 \circ \cdots \circ R_l$ between type A_1 and A_{l+1}, where $\circ$ denotes the composition operator on relations. The length of the path $\mathcal{P}$ is the number of relations in $\mathcal{P}$, which is l.*

For simplicity, we can also use type names denoting the relevance path if there are no multiple relations between the same pair of types: $\mathcal{P} = (A_1 A_2 \cdots A_{l+1})$. For example, in Fig. 3(a), the relation, authors publishing papers in conferences, can be described using the length-2 relevance path $A \xrightarrow{writting} P \xrightarrow{published} V$, or short as APV. We say a concrete path $p = (a_1 a_2 \cdots a_{l+1})$ between a_1 and a_{l+1} in network G is a **path instance** of the relevance path $\mathcal{P}$, if for each a_i, $\phi(a_i) = A_i$ and each link $e_i = \langle a_i, a_{i+1} \rangle$ belongs to the relation R_i in $\mathcal{P}$. It can be denoted as $p \in \mathcal{P}$. A relevance path $\mathcal{P}^{-1}$ is the **reverse path** of $\mathcal{P}$, which defines an inverse relation of the one defined by $\mathcal{P}$. Similarly, we define the **reverse path instance** of p^{-1} as the reverse path of p in G. For example, the reverse path of the path APV, which means authors publish papers in venues, is the path VPA which means venues accept authors' papers. Further, a relevance path $\mathcal{P}$ is a **symmetric path**, if the relation R defined by it is symmetric (i.e., $\mathcal{P}$ is equal to $\mathcal{P}^{-1}$), such as APA and $APCPA$. Two relevance paths $\mathcal{P}_1 = (A_1 A_2 \cdots A_l)$ and $\mathcal{P}_2 = (B_1 B_2 \cdots B_k)$ are **concatenable** if and only if $A_l = B_1$, and the concatenated path is written as $\mathcal{P} = (\mathcal{P}_1 \mathcal{P}_2)$, which equals to $(A_1 A_2 \cdots A_l B_2 \cdots B_k)$. A simple concatenable example is that AP and PV can be concatenated to the path APV.

4. HETESIM: A RELEVANCE MEASURE

4.1 Basic Idea

In many domains, similar objects are related to similar objects. For example, similar researchers published many similar papers; similar customers purchase similar commodities. As a consequence, two objects are similar if they are referenced by similar objects. This intuition is also fit for heterogeneous objects. For example, researchers are more relevant to the conferences that publish many papers written by the researchers; and customers are more faithful to brands that manufacture many products purchased by the customers. A more concrete example is shown in Fig. 4. Tom is more relevant to KDD than other conferences, since all of his papers are published in KDD. Although the similar idea has been applied in SimRank [5], it is limited to homogeneous networks. When we apply the idea to heterogeneous networks, it faces the following challenges: (1) The relatedness of heterogeneous objects is path-constrained; (2) The relatedness measure based on an asymmetric relevance path has the symmetric property. In the following section, we will illustrate these challenges and their solutions.

4.2 Path-based Relevance Measure

Different from homogeneous networks, the paths in heterogeneous networks have semantics, which makes the relatedness between two objects different on different relevance paths. Taking Fig. 4 for example, Tom is not related to SIGMOD based on APC path which means authors publishing papers in conferences. However, he is related to SIGMOD based on $APAPC$ path meaning that the coauthors of authors publish papers in conferences. So the relevance measure of objects in heterogeneous networks is based on the given relevance path.

Following the basic idea that similar objects are related to similar objects, we propose a path-based relevance measure: HeteSim.

DEFINITION 3. ***HeteSim:*** *Given a relevance path $\mathcal{P} = R_1 \circ R_2 \circ \cdots \circ R_l$, HeteSim between two objects s and t ($s \in R_1.S$ and $t \in R_l.T$) is:*

$$HeteSim(s,t|R_1 \circ R_2 \circ \cdots \circ R_l) = \frac{1}{|O(s|R_1)||I(t|R_l)|} \sum_{i=1}^{|O(s|R_1)|} \sum_{j=1}^{|I(t|R_l)|} HeteSim(O_i(s|R_1), I_j(t|R_l)|R_2 \circ \cdots \circ R_{l-1}) \quad (1)$$

where $O(s|R_1)$ is the out-neighbors of s based on relation R_1, and $I(t|R_l)$ is the in-neighbors of t based on relation R_l.

When s may not have any out-neighbors (i.e., $O(s|R_1) = \emptyset$) or t may not have any in-neighbors (i.e., $I(t|R_l) = \emptyset$) following the path, we have no way to infer any relatedness between s and t in this case, so we define their relevance value to be 0. Equation (1) shows that to compute $HeteSim(s,t|\mathcal{P})$, we need to iterate over all pairs ($O_i(s|R_1)$, $I_j(t|R_l)$) of (s,t) along the path (s along the path and t against path), and sum up the relatedness of these pairs. Then, we normalize it by the total number of out-neighbors of s and in-neighbors of t. That is, the relatedness between s and t is the average relatedness between the out-neighbors of s and the in-neighbors of t. The process continues until s and t will meet along the path. Similar to SimRank [5], HeteSim is also a pair wise random walk. But it considers the relevance path. As we know, SimRank measures how soon two random surfers are expected to meet at the same node [5]. By contrast, $HeteSim(s,t|\mathcal{P})$ measures how likely s and t will meet at the same node when s follows along the path and t goes against the path.

EXAMPLE 2. *Taking Fig. 4 for example, we calculate the relatedness of Tom and KDD based on APC path.*

$$HeteSim(Tom, KDD|APC) = \frac{1}{|O(Tom|AP)||I(KDD|PC)|} \sum_{i=1}^{|O(Tom|AP)|} \sum_{j=1}^{|I(KDD|PC)|} HeteSim(O_i(Tom|AP), I_j(KDD|PC)) \quad (2)$$

where $O(Tom|AP) = \{P_1, P_2\}$ *and* $I(KDD|PC) = \{P_1, P_2\}$. *So* $HeteSim(Tom, KDD|APC) = 0.5$. *Here we think the relatedness of an object to itself is 1 (the formal definition can be seen in Def. 4). In this case, Tom and KDD both have the possibility 0.5 to reach* P_1 *and* P_2 *along the APC path, respectively. So the possibility of they meeting at same papers along the path is 0.5. In other words, the relatedness of Tom and KDD measures their meeting possibility when they walk along the path.*

4.3 Decomposition of Relevance Path

However, the source object s and the target object t do not always meet at the same objects based on a given path $\mathcal{P}$. For the similarity measure of same-typed objects, the relevance paths are usually even-length, even symmetric, so the source object and the target object will meet at the middle objects. For example, the source and target object will meet at type T based on the $APTPA$ path. However, for the relevance measure of different-typed objects, the relevance paths are usually odd-length. In this condition, the source and target objects will never meet at the same objects. Taking the $APVC$ path as an example, authors along the path and conferences against the path will never meet in the same objects. So the original HeteSim is not suitable for odd-length relevance paths. In order to solve this difficulty, a basic idea is to transform odd-length paths into even-length paths, and thus the source and target objects are always able to meet at the same objects. As a consequence, an arbitrary path can be decomposed as two equal-length paths.

When the length l of a relevance path $\mathcal{P} = (A_1 A_2 \cdots A_{l+1})$ is even, the source objects (along the path) and the target objects (against the path) will meet in the **middle type** object $M = A_{\frac{l}{2}+1}$ on the **middle position** $mid = \frac{l}{2} + 1$, so the relevance path $\mathcal{P}$ can be divided into two equal-length path $\mathcal{P}_L$ and $\mathcal{P}_R$. That is, $\mathcal{P} = \mathcal{P}_L\mathcal{P}_R$, where $\mathcal{P}_L = A_1 A_2 \cdots A_{mid-1} M$ and $\mathcal{P}_R = M A_{mid+1} \cdots A_{l+1}$. For objects in a **self-relation** (denoted as I relation), it is obvious that an object is just similar to itself. So its relevance measure can be defined as follows:

DEFINITION 4. ***HeteSim based on self-relation****: HeteSim between two same-typed objects s and t based on the self-relation I is:*

$$HeteSim(s,t|I) = \delta(s,t) \quad (3)$$

where $\delta(s,t) = 1$, *if s and t are same, or else it is 0.*

When the path length l is odd, the source objects and the target objects will meet at the relation $A_{\frac{l+1}{2}} A_{\frac{l+1}{2}+1}$. For example, based on the $APSPVC$ path, the source and target objects will meet at the SP relation after two steps. In order to let the source and target objects meet at same-typed objects, we can add a middle type object E between the atomic relation $A_{\frac{l+1}{2}} A_{\frac{l+1}{2}+1}$ and maintain the relation between $A_{\frac{l+1}{2}}$ and $A_{\frac{l+1}{2}+1}$ at the same time. Then the new path becomes $\mathcal{P}' = (A_1 \cdots E \cdots A_{l+1})$ which length is $l+1$, an even number. In the aforementioned example, the path becomes $APSEPVC$, which is even-length now. The source objects and the target objects will meet in the **middle type** object $M = E$ on the **middle position** $mid = \frac{l+1}{2} + 1$. As a consequence, the new relevance path $\mathcal{P}'$ can also be divided into two equal-length path $\mathcal{P}_L$ and $\mathcal{P}_R$ as above.

DEFINITION 5. ***Decomposition of relevance path****. An arbitrary relevance path $\mathcal{P} = (A_1 A_2 \cdots A_{l+1})$ can be decomposed into two equal-path path $\mathcal{P}_L$ and $\mathcal{P}_R$ (i.e., $\mathcal{P} = \mathcal{P}_L\mathcal{P}_R$), where $\mathcal{P}_L = A_1 A_2 \cdots A_{mid-1} M$ and $\mathcal{P}_R = M A_{mid+1} \cdots A_{l+1}$. M and mid are defined as above.*

Obviously, for a symmetric path $\mathcal{P} = \mathcal{P}_L\mathcal{P}_R$, $\mathcal{P}_R^{-1}$ is equal to $\mathcal{P}_L$. For example, the relevance path $\mathcal{P} = APCPA$ can be decomposed as $\mathcal{P}_L = APC$ and $\mathcal{P}_R = CPA$. For the relevance path $APSPVC$, we can add a middle type object E in SP and thus the path becomes $APSEPVC$, so $\mathcal{P}_L = APSE$ and $\mathcal{P}_R = EPVC$.

The next question is how we can add the middle type object E in an atomic relation R between $A_{\frac{l+1}{2}}$ and $A_{\frac{l+1}{2}+1}$ in an odd-length path, i.e., between S and P in the previous example on $APSPVC$. In order to contain original atomic relation, we need to keep the R relation be the composition of two new relations. To do so, for each instance of relation R, we can add an instance of E to connect the source and target objects of the relation instance. An example is shown in Fig. 5(a), where the middle type object E is added in between the atomic relation AB along each path instance.

DEFINITION 6. ***Decomposition of atomic relation****. For an atomic relation R, we can add an object type E (called edge object) between the $R.S$ and $R.T$. And thus the atomic relation R is decomposed as R_O and R_I where R_O represents*

the relation between $R.S$ and E and R_I represents that between E and $R.T$. For each relation instance $r \in R$, an instance $e \in E$ connects $r.S$ and $r.T$ and $r.S \to e$ and $e \to r.T$ are the instances of R_O and R_I, respectively.

It is clear that the decomposition has the following property, which is proved in the appendix.

Property 1. An atomic relation R can be decomposed as R_O and R_I, $R = R_O \circ R_I$, and this decomposition is unique.

Based on this decomposition, the relatedness of two objects with an atomic relation R can be calculated as follows:

DEFINITION 7. ***HeteSim based on atomic relation**: HeteSim between two different-typed objects s and t based on an atomic relation R ($s \in R.S$ and $t \in R.T$) is:*

$$HeteSim(s,t|R) = HeteSim(s,t|R_O \circ R_I)$$
$$= \frac{1}{|O(s|R_O)||I(t|R_I)|} \sum_{i=1}^{|O(s|R_O)|} \sum_{j=1}^{|I(t|R_I)|} \delta(O_i(s|R_O), I_j(t|R_I)) \quad (4)$$

It is easy to find that $HeteSim(s,t|I)$ is a special case of $HeteSim(s,t|R)$, since, for the self-relation I, $I = I_O \circ I_I$ and $|O(s|I_O)| = |I(t|I_I)| = 1$. Definition 7 means that we can measure the relatedness of two different-typed objects with an atomic relation R directly, which has never been done before. HeteSim measures their relatedness through calculating the average of their mutual influence.

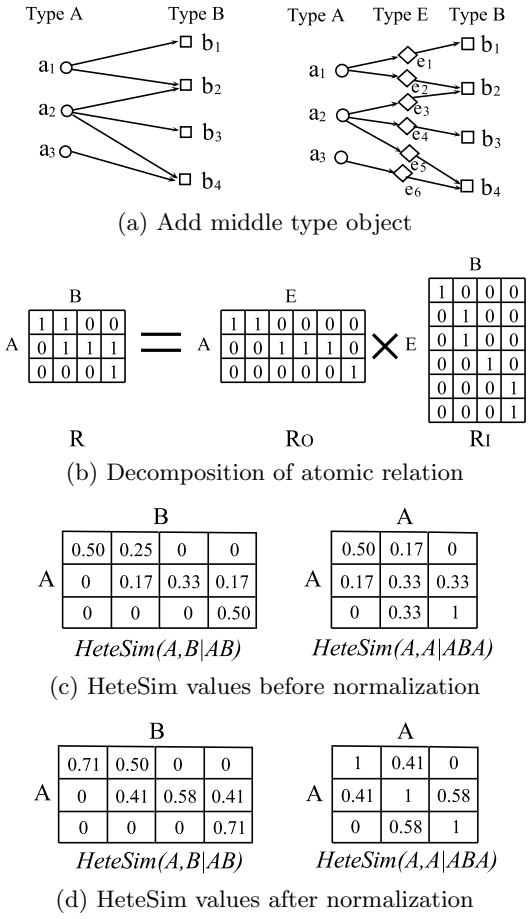
EXAMPLE 3. *Fig. 5(a) shows an example of decomposition of atomic relation. The relation AB is decomposed into the relations AE and EB. Moreover, the relation AB is the composition of AE and EB. The HeteSim is calculated in Fig. 5(c). We can find that HeteSim justly reflects relatedness of objects. Taking a_2 for example, although a_2 equally connects with b_2, b_3, and b_4, it is more close to b_3, because b_3 only connects a_2. This information is correctly reflected in the HeteSim value of a_2:* $(0, 0.17, 0.33, 0.17)$.

We also find that the similarity of an object and itself is not 1 in HeteSim. Taking Fig. 5(c) as example, the relatedness of a_2 and itself is 0.33. It is obviously unreasonable. In the following section, we will normalize the HeteSim and make the relevance measure more reasonable.

4.4 Normalization of HeteSim

Firstly, we introduce the calculation of HeteSim between any two objects given an arbitrary relevance path.

DEFINITION 8. ***Transition probability matrix**. For relation $A \xrightarrow{R} B$, W_{AB} is an adjacent matrix between type A and B. U_{AB} is normalized matrix of W_{AB} along the row vector, which is the transition probability matrix of $A \longrightarrow B$ based on relation R. V_{AB} is normalized matrix of W_{AB} along the column vector, which is the transition probability matrix of $B \longrightarrow A$ based on relation R^{-1}.*



(a) Add middle type object

(b) Decomposition of atomic relation

(c) HeteSim values before normalization

(d) HeteSim values after normalization

Figure 5: Decomposition of atomic relation and its HeteSim calculation.

It is easy to prove that the transition probability matrix has the following property, which can be seen in the appendix.

Property 2. $U_{AB} = V'_{BA}$ and $V_{AB} = U'_{BA}$, where V'_{BA} is the transpose of V_{BA}.

DEFINITION 9. ***Reachable probability matrix**. Given a network $G = (V, E)$ following a network schema $S = (\mathcal{A}, \mathcal{R})$, a reachable probability matrix PM for a path $\mathcal{P} = (A_1 A_2 \cdots A_{l+1})$ is defined as $PM_{\mathcal{P}} = U_{A_1A_2} U_{A_2A_3} \cdots U_{A_lA_{l+1}}$ (PM for simplicity). $PM(i,j)$ represents the probability of object $i \in A_1$ reaching object $j \in A_{l+1}$ under the path $\mathcal{P}$.*

According to the definition of HeteSim, the relevance between objects in A_1 and A_{l+1} based on the relevance path $\mathcal{P} = A_1 A_2 \cdots A_{l+1}$ is

$$\begin{aligned} &HeteSim(A_1, A_{l+1}|\mathcal{P}) \\ &= HeteSim(A_1, A_{l+1}|\mathcal{P}_L\mathcal{P}_R) \\ &= U_{A_1A_2} \cdots U_{A_{mid-1}M} V_{MA_{mid+1}} \cdots V_{A_lA_{l+1}}. \end{aligned} \quad (5)$$

According to Property 2, the relevance matrix can be rewritten in the following way:

$$\begin{aligned} &HeteSim(A_1, A_{l+1}|\mathcal{P}) \\ &= U_{A_1A_2}\cdots U_{A_{mid-1}M}U'_{A_{mid+1}M}\cdots U'_{A_{l+1}A_l} \\ &= U_{A_1A_2}\cdots U_{A_{mid-1}M}(U_{A_{l+1}A_l}\cdots U_{A_{mid+1}M})' \\ &= PM_{\mathcal{P}_L}PM'_{\mathcal{P}_R{}^{-1}} \end{aligned} \quad (6)$$

The above equation shows that the relevance of A_1 and A_{l+1} based on the path $\mathcal{P}$ is the product of two probability distributions that A_1 reaches the middle type object M along the path and A_{l+1} reaches M against the path. For two instances a and b in A_1 and A_{l+1}, respectively, their relevance based on path $\mathcal{P}$ is

$$HeteSim(a, b|\mathcal{P}) = PM_{\mathcal{P}_L}(a,:)PM'_{\mathcal{P}_R{}^{-1}}(b,:) \quad (7)$$

where $PM_{\mathcal{P}}(a,:)$ means the a-th row in $PM_{\mathcal{P}}$.

We have stated that HeteSim needs to be normalized. It is reasonable that the relatedness of the same objects is 1, so the HeteSim can be normalized as follows:

DEFINITION 10. ***Normalization of HeteSim****. The normalized HeteSim between two objects a and b based on the relevance path $\mathcal{P}$ is:*

$$HeteSim(a, b|\mathcal{P}) = \frac{PM_{\mathcal{P}_L}(a,:)PM'_{\mathcal{P}_R{}^{-1}}(b,:)}{\sqrt{\|PM_{\mathcal{P}_L}(a,:)\|\|PM'_{\mathcal{P}_R{}^{-1}}(b,:)\|}} \quad (8)$$

In fact, the normalized HeteSim is the cosine of the probability distributions of the source object a and target object b reaching the middle type object M. It ranges from 0 to 1. Fig. 5(d) shows the normalized HeteSim values. It is clear that the normalized HeteSim is more reasonable. In the following section, the HeteSim means the normalized HeteSim.

4.5 Properties of HeteSim

HeteSim has many good properties, which makes it useful in many applications. The proof of these properties can be found in the appendix.

Property 3: **Symmetric**:
$HeteSim(a, b|\mathcal{P}) = HeteSim(b, a|\mathcal{P}^{-1})$.

Property 3 shows the symmetric property of HeteSim. Although PathSim [19] also has the similar symmetric property, it holds only when the path is symmetric and a and b are with the same type. The HeteSim has the more general symmetric property not only for symmetric paths (note that $\mathcal{P}$ is equal to $\mathcal{P}^{-1}$ for symmetric paths) but also for arbitrary paths. PCRW [12] does not have this symmetric property, since it is based on random walk. This property is important in many applications. For example, the relatedness of any two objects in heterogeneous networks can be measured and the relatedness is symmetric, so the clustering task can be performed on the relevance matrix directly.

Property 4. **Self-maximum**: $HeteSim(a, b|\mathcal{P}) \in [0, 1]$. $HeteSim(a, b|\mathcal{P})$ is equal to 1 if and only if $PM_{\mathcal{P}_L}(a,:)$ is equal to $PM_{\mathcal{P}_R{}^{-1}}(b,:)$.

Property 4 shows HeteSim is well constrained. For a symmetric path $\mathcal{P}$, $\mathcal{P}_L$ is equal to $\mathcal{P}_R{}^{-1}$, so $PM_{\mathcal{P}_L}(a,:)$ is equal to $PM_{\mathcal{P}_R{}^{-1}}(a,:)$. And thus $HeteSim(a, a|\mathcal{P})$ is equal to 1. If we define the distance between two objects (i.e., $dis(s,t)$) as $dis(s,t) = 1 - HeteSim(s,t)$, the distance of the same object is zero (i.e., $dis(s,s) = 0$). As a consequence, HeteSim satisfies the identity of indiscernibles.

Since HeteSim obeys the properties of non-negativity, identity of indiscernibles, and symmetry, we can say that HeteSim is a semi-metric measure [22]. Since HeteSim is a path-based measure, it does not obey the triangle inequality. PathSim is also semi-metric measure. However, it holds only for symmetric paths. Our HeteSim has the property for an arbitrary path. PCRW is not semi-metric measure, since it does not obey the symmetry property. A semi-metric measure can be used in many applications [22].

Property 5. **Connection to SimRank**. For a bipartite graph $G = (V, E)$ based on the schema $S = (\{A, B\}, \{R\})$, suppose the constant C in SimRank is 1,
$SimRank(a_1, a_2) = \lim_{n\rightsquigarrow\infty} \sum_{k=1}^{n} HeteSim(a_1, a_2|(RR^{-1})^k)$,
$SimRank(b_1, b_2) = \lim_{n\rightsquigarrow\infty} \sum_{k=1}^{n} HeteSim(b_1, b_2|(R^{-1}R)^k)$.
where $a_1, a_2 \in A$, $b_1, b_2 \in B$ and $A \xrightarrow{R} B$. Here HeteSim is not normalized.

This property reveals the relation of SimRank and HeteSim. SimRank sums up the meeting probability of two objects after all possible steps: one, two, $\cdots$. HeteSim just calculates the meeting probability along the given relevance path. If the relevance paths explore all possible meta paths among the two objects, the sum of HeteSim based on these paths is the SimRank. So we can say that SimRank is the special case of HeteSim. This property also implies that HeteSim is more efficient than SimRank, since HeteSim only needs to calculate the meeting probability along the given relevance path, not all possible meta paths.

4.6 Discussion

Let us analyze the time and space complexity of computing HeteSim. Suppose the average size of one type of objects is n and there are T types objects, the space requirement of HeteSim is just $O(n^2)$ to store the relatedness matrix. Let d be the average of $|O(s|R_i)||I(t|R_j)|$ over all object-pairs (s, t) based on relation R_i and R_j. For a given l-length relevance path, the time required is $O(ldn^2)$, since node pairs (i.e., n^2) calculate their relatedness along the relevance path. For SimRank, the similarity of node pairs in all types (i.e., $(Tn)^2$) are iteratively calculated at the same time, so its space complexity is $O(T^2n^2)$, and the time complexity is $O(k(T^2d)(Tn)^2)$ (i.e., $O(kdn^2T^4)$), where k is the number of iterations. So the complexity of computing HeteSim is much smaller than SimRank.

Although HeteSim has large computation demand, several approaches can alleviate it. (1) For frequently-used relevance paths, the relatedness matrix $HeteSim(A, B|\mathcal{P})$ can be calculated off-line. The on-line search on $HeteSim(a, b|\mathcal{P})$ will be very fast, since it only needs to locate the row and column in the matrix. (2) The concatenation of partially materialized reachable probability matrix also helps to fasten

Table 1: Automatic object profiling task on author "Christos Faloutsos" on ACM dataset.

Path	*APVC*		*APT*		*APS*		*APA*	
Rank	Conf.	Score	Terms	Score	Subjects	Score	Authors	Score
1	KDD	0.1198	mining	0.0930	H.2 (database management)	0.1023	Christos Faloutsos	1
2	SIGMOD	0.0284	patterns	0.0926	E.2 (data storage representations)	0.0232	Hanghang Tong	0.4152
3	VLDB	0.0262	scalable	0.0869	G.3 (probability and statistics)	0.0175	Agma Juci M. Traina	0.3250
4	CIKM	0.0083	graphs	0.0816	H.3 (information storage and retrieval)	0.0136	Spiros Papadimitriou	0.2785
5	WWW	0.0060	social	0.0672	H.1 (models and principles)	0.0135	Caetano Traina, Jr.	0.2680

Table 2: Automatic object profiling task on conference "KDD" on ACM dataset.

Path	*CVPA*		*CVPAF*		*CVPS*		*CVPAPVC*	
Rank	Authors	Score	Organization	Score	Subjects	Score	Conf.	Score
1	Christos Faloutsos	0.1198	Carnegie Mellon Univ.	0.0824	H.2 (database management)	0.3215	KDD	1
2	Heikki Mannila	0.1119	Univ. of Minnesota	0.0814	I.5 (pattern recognition)	0.1650	VLDB	0.2124
3	Padhraic Smyth	0.1043	IBM	0.0761	I.2 (artificial intelligence)	0.1194	SIGMOD	0.1535
4	Jiawei Han	0.1029	Yahoo! Research	0.0692	G.3 (prob. and stat.)	0.0856	WWW	0.1391
5	Vipin Kumar	0.0966	Univ. of California	0.0683	H.3 (info. storage and retrieval)	0.0653	CIKM	0.0943

the computation. If we pre-computed and stored $PM_{\mathcal{P}_L}$ and $PM_{\mathcal{P}_R{}^{-1}}$, we can calculate the $HeteSim(A, B|\mathcal{P})$ according to Equation 6. For the $HeteSim(a, b|\mathcal{P})$, it only needs to calculate the dot product of two vectors (i.e., $PM_{\mathcal{P}_L}(a, :)$ and $PM_{\mathcal{P}_R{}^{-1}}(b, :)$). Moreover, the different partial paths can be concatenated to many relevance paths. For example, given two pre-stored reachable probability matrix based on paths *CPA* and *APA*, we are able to answer queries for the relevance paths *CPAPA*, *APAPC*, *CPAPC*, *APCPA*, and *APAPA*. (3) Fast algorithms can be designed to speed up the calculation of HeteSim. The related objects to a searched object are a very small percentage of all objects in the target type. The pruning techniques [5, 19] can be used to prune those unpromising objects during the search. We can also apply some approximate algorithms [11] to fasten the search with a small loss of accuracy.

Here, we discuss how to choose relevance path. There are several ways to do it. (1) Users can select proper paths according to their domain knowledge and experiences. (2) The user can try multiple relevance paths, and then make a choice based on his application. (3) Supervised learning can be used to automatically select relevance paths [11]. We can label a small portion of similar objects, and then train the relevance paths and their weights by some learning algorithms. The learning algorithms can automatically choose appropriate relevance paths and the associated weights.

5. EXPERIMENTS

In the experiments, we validate the effectiveness of the HeteSim through three case studies and two learning tasks on two bibliographic networks.

5.1 Data Sets

We use two heterogeneous information networks for our experiments, including ACM dataset and DBLP dataset. They are summarized as follows:

ACM dataset: The ACM dataset was downloaded from ACM digital library[3] in June 2010. The ACM dataset comes from 14 representative computer science conferences: KDD, SIGMOD, WWW, SIGIR, CIKM, SODA, STOC, SOSP, SPAA, SIGCOMM, MobiCOMM, ICML, COLT, and VLDB. These conferences include 196 corresponding venue proceedings (e.g., KDD conference includes 12 proceedings, such as KDD'10, KDD'09, etc). The dataset has 12K papers, 17K authors, and 1.8K author affiliations. After removing stop words in the paper titles and abstracts, we get 1.5K terms that appear in more than 1% of the papers. The network also includes 73 subjects of these papers in ACM category. The network schema of ACM dataset is shown in Fig. 3(a).

DBLP dataset [7]: The DBLP dataset is a sub-network collected from DBLP website[4] involving major conferences in four research areas: database, data mining, information retrieval and artificial intelligence, which naturally form four classes. The dataset contains 14K papers, 20 conferences, 14K authors and 8.9K terms, with a total number of 17K links. In the dataset, 4057 authors, all 20 conferences and 100 papers are labeled with one of the four research areas. The network schema is shown in Fig. 3(b).

5.2 Case Study

In this section, we demonstrate the traits of HeteSim through case study in three tasks: automatic object profiling, expert finding, and relevance search.

5.2.1 Task 1: Automatic Object Profiling

We first study the effectiveness of our approach on different-typed relevance measurement in the automatic object profiling task. If we want to know the profile of an object, we can measure the relevance of the object to our interested objects. For example, we want to know the academic profile of Christos Faloutsos[5]. It can be solved through measuring the relatedness of Christos Faloutsos with related objects, e.g., conference, affiliations, other authors, etc. Table 1 shows the lists of top relevant objects with various types on ACM dataset. *APVC* path shows the conferences he actively participates. Note that KDD and SIGMOD are the two major conferences Christos Faloutsos participates, which are mentioned in his homepage[6]. From the path *APT*, we can obtain his research interests: data mining, pattern discovery, scalable graph mining and social network. Using *APS* path, we can discover his research areas represented as ACM subjects:

[3]http://dl.acm.org/

[4]http://www.informatik.uni-trier.de/~ley/db/

[5]http://www.cs.cmu.edu/~christos/

[6]http://www.cs.cmu.edu/~christos/misc.html

Table 3: Relatedness values of authors and conferences measured by HeteSim and PCRW on ACM dataset.

HeteSim			PCRW					
APVC&CVPA			*APVC*			*CVPA*		
Pair		Score	Pair		Score	Pair		Score
C. Faloutsos,	KDD	**0.1198**	C. Faloutsos,	KDD	0.5517	KDD,	C. Faloutsos	**0.0087**
W. B. Croft,	SIGIR	**0.1201**	W. B. Croft,	SIGIR	0.6481	SIGIR,	W. B. Croft	**0.0098**
J. F. Naughton,	SIGMOD	**0.1185**	J. F. Naughton,	SIGMOD	**0.7647**	SIGMOD,	J. F. Naughton	0.0062
A. Gupta,	SODA	**0.1225**	A. Gupta,	SODA	**0.7647**	SODA,	A. Gupta	**0.0090**
Luo Si,	SIGIR	0.0734	Luo Si,	SIGIR	**0.7059**	SIGIR,	Luo Si	0.0030
Yan Chen,	SIGCOMM	0.0786	Yan Chen,	SIGCOMM	**1**	SIGCOMM,	Yan Chen	0.0013

Table 4: Top 10 related authors to "Christos Faloutsos" based on $APVCVPA$ path on ACM dataset.

	HeteSim		PathSim		PCRW	
Rank	Author	Score	Author	Score	Author	Score
1	Christos Faloutsos	1	Christos Faloutsos	1	Charu C. Aggarwal	0.0063
2	Srinivasan Parthasarathy	0.9937	Philip Yu	0.9376	Jiawei Han	0.0061
3	Xifeng Yan	0.9877	Jiawei Han	0.9346	Christos Faloutsos	0.0058
4	Jian Pei	0.9857	Jian Pei	0.8956	Philip Yu	0.0056
5	Jiong Yang	0.9810	Charu C. Aggarwal	0.7102	Alia I. Abdelmoty	0.0053
6	Ruoming Jin	0.9758	Jieping Ye	0.6930	Chris B. Jones	0.0053
7	Wei Fan	0.9743	Heikki Mannila	0.6928	Jian Pei	0.0034
8	Evimaria Terzi	0.9695	Eamonn Keogh	0.6704	Heikki Mannila	0.0032
9	Charu C. Aggarwal	0.9668	Ravi Kumar	0.6378	Eamonn Keogh	0.0031
10	Mohammed J. Zaki	0.9645	Vipin Kumar	0.6362	Mohammed J. Zaki	0.0027

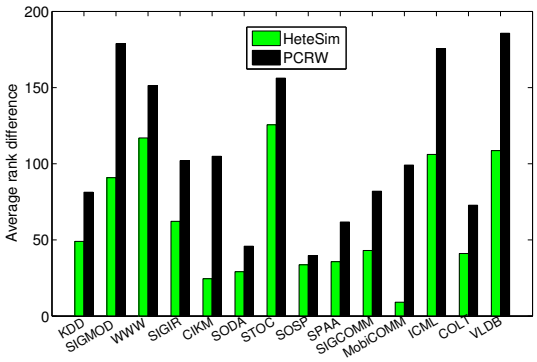


Figure 6: The average rank difference of HeteSim and PCRW on 14 conferences of ACM dataset. The lower the better.

database management (H.2) and data storage (E.2). Based on APA path, HeteSim finds the most important co-authors, most of which are his Ph.D students.

In another case study, we want to find the profile of KDD conference. Table 2 shows the results on ACM dataset. The active researchers in the conference can be found by the $CVPA$ path indicating the relationship of authors publishing papers in conferences. The top five authors are all well-known researchers in data mining area. The $CVPAF$ path reveals the important research affiliations that have published many papers in KDD, such as CMU, IBM, Yahoo! Research. The results of $CVPS$ illustrate that the topics of KDD are database management (H.2), pattern recognition (I.5), and so on. The $CVPAPVC$ path measures the similarity of conferences through their common authors. The conferences that are most similar to KDD are VLDB, SIGMOD, WWW and CIKM. It is reasonable, since these conferences all share many authors whose research areas are data mining and knowledge management.

5.2.2 *Task 2: Expert Finding*

In this case, we want to validate the effectiveness of HeteSim to reflect the relative importance of object pairs through an expert finding task. As we know, the relative importance of object pairs can be revealed through comparing their relatedness. Suppose we know the experts in one domain, the expert finding task here is to find experts in other domains through their relative importances. Table 3 shows the relevance scores returned by different approaches on six "conference-author" pairs on ACM dataset. The relatedness of conferences and authors are defined based on the $APVC$ and $CVPA$ paths which have the same semantics: authors publishing papers in conferences. Due to the symmetric property of HeteSim, we get the same value for both paths. While PCRW returns different values for these two paths. Suppose that we are familar with data mining area, and already know that C. Faloutsos is an influential researcher in KDD. Comparing these HeteSim scores, we can still find influential researchers in other research areas even if we are not quite familiar with these areas. J. F. Naughton, W. B. Croft and A. Gupta should be influential researchers in SIGMOD, SIGIR and SODA, respectively, since they have very similar HeteSim score to C. Faloutsos. Moreover, we can also deduce that Luo Si and Yan Chen may be active researchers in SIGIR and SIGCOMM, respectively, since their HeteSim scores are smaller than that of C. Faloutsos, but not very small. In fact, C. Faloutsos, J. F. Naughton, W. B. Croft and A. Gupta are top ranked authors in their research communities. Luo Si and Yan Chen are the young professors and they have done good work in their research areas. However, if the relevance measure is not symmetric (e.g., PCRW), it is very hard to tell which authors are more influential when comparing these relevance scores. For example, the PCRW score of Yan Chen and SIGCOMM is the largest one in the $APVC$ path. However, the value is the smallest one when the opposite path is considered, i.e., $CVPA$ path.

The relative importance is hard to quantitatively measure. However, we can roughly measure the relatedness of authors and conferences by the number of papers that authors pub-

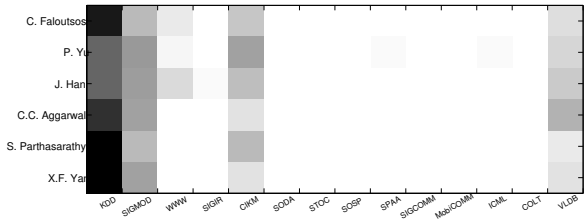


Figure 7: Probability distribution of authors' papers on 14 conferences of ACM dataset.

lish in conferences, and then rank the relatedness as their relative importance (i.e., ground truth). We also compute the relatedness of authors and conferences based on HeteSim and PCRW, and then rank these values. Through computing the average rank difference from the ground truth, we can roughly measure the accuracy of relative importance. For example, C. Faloutsos is ranked 1st on KDD as ground truth, while an approach rank him 6th. So the rank difference is 5. Note that, since PCRW has two rank scores for two different orders, the results are the average rank differences based on these two different orders. Fig. 6 show the average rank difference on the top 200 authors in ground truth on each conference. It is clear that HeteSim more accurately reveals the relative importance of author-conference pairs, since their average rank difference is smaller.

5.2.3 Task 3: Relevance Search based on Path Semantics

As we have stated, the path-based relevance measure can capture the semantics of paths. In this relevance search task, we will observe the effectiveness of semantics capture through comparing HeteSim with other two path-based measures: PCRW and PathSim. Since PathSim can only measure the similarity of same-typed objects, this task will find the top 10 related authors to Christos Faloutsos based on the $APVCVPA$ path which means authors publishing papers in same conferences. The results are shown in Table 4. The PathSim finds the similar peer authors, such as Philip Yu and Jiawei Han. They have the same reputation in data mining field. It is strange for PCRW that the most similar author to Christos Faloutsos is not himself, but Charu C. Aggarwal and Jiawei Han. It is obviously not reasonable. Our conjecture is that Charu C. Aggarwal and Jiawei Han published more papers in the same conferences that Christos Faloutsos published, so Christos Faloutsos has more reachable probability on Charu C. Aggarwal and Jiawei Han than himself along the $APVCVPA$ path. HeteSim's results are a little different. The most similar authors are Srinivasan Parthasarathy and Xifeng Yan, instead of Philip Yu and Jiawei Han.

Let's consider the semantics of the path $APVCVPA$ again: authors publishing papers in the same conferences. Fig. 7 shows the reachable probability distribution from authors to conferences along the path $APVC$. It is clear that the probability distribution of papers of Srinivasan Parthasarathy and Xifeng Yan on conferences are more close to that of Christos Faloutsos, so they should be more similar to Christos based on the same conference publication. Although

Table 6: Clustering accuracy for path-based similarity measures on DBLP dataset.

	Venue NMI	Author NMI	Paper NMI
HeteSim	0.7683	**0.7288**	**0.4989**
PathSim	**0.8162**	0.6725	0.3833

Philip Yu and Jiawei Han have the same reputation with C. Faloutsos, their papers are more broadly published in different conferences. So they are not the most similar authors to C. Faloutsos based on the $APVCVPA$ path. As a consequence, our HeteSim more accurately captures the semantics of the path.

5.3 Performance on Query Task

The query task will validate the effectiveness of HeteSim on query search of heterogeneous objects. Since PathSim cannot measure the relatedness of different-typed objects, in this experiment, we only compare the performance of HeteSim with PCRW. On DBLP dataset which has been labeled, we measure proximity of conferences and authors based on the CPA path. For each conference, we rank its related authors according to their measure scores. We calculate the AUC (Area Under ROC Curve) score based on the label of the authors and conferences in order to evaluate the performances of the ranked results. The larger score means the better performance. We evaluate the performances on 9 representative conferences and their AUC scores are shown in Table 5. We can find that HeteSim consistently outperforms PCRW in all 9 conferences. It shows that our proposed HeteSim method on proximity query task can work better than asymmetric similarity measures.

5.4 Performance on Clustering Task

Due to the symmetric property, HeteSim can be applied to clustering tasks. In order to evaluate its performance, we compare HeteSim with PathSim on the clustering of same-typed objects, since PathSim can only measure the similarity of same-typed objects. These two measures use the same information to determine the pairwise similarity between objects. We evaluate the clustering performance on DBLP dataset which involve 3 clustering tasks: clustering on conferences based on $CPAPC$ path, clustering on authors based on $APCPA$ path, and clustering on papers based on $PAPCPAP$ path. We apply Normalized Cut [18] to perform clustering based on the similarity matrices returned by different algorithms. The number of clusters is set as 4. NMI criterion (Normalized Mutual Information) [20] is used to evaluate the clustering performances on conferences, authors, and papers. NMI is between 0 and 1 and the higher the better. The average clustering accuracy results of 100 runs are summarized in Table 6. HeteSim achieves better performances on two clustering tasks: authors and papers clustering. It shows that HeteSim not only do well on similarity measure of same-typed objects but also has the potential as the similarity metric in clustering.

From Table 6, we can also observe that HeteSim and PathSim both achieve high accuracy on conferences and authors clustering. However, the accuracy on paper clustering is low. We think the clustering accuracy is largely affected by the semantics of relevance paths. The similarity of conferences can be measured by the $CPAPC$ path which means

Table 5: AUC values for the relevance search of conferences and authors based on *CPA* path on DBLP dataset.

	KDD	ICDM	SDM	SIGMOD	VLDB	ICDE	AAAI	IJCAI	SIGIR
HeteSim	0.8111	0.6752	0.9504	0.7662	0.8262	0.7322	0.8110	0.8754	0.6132
PCRW	0.8030	0.6731	0.9390	0.7588	0.8200	0.7263	0.8067	0.8712	0.6068

Table 7: The top 10 most related authors to "KDD" conference under different relevance paths on ACM dataset.

	path	
rank	*CVPA*	*CVPAPA*
1	Christos Faloutsos	Charu C. Aggarwal
2	Heikki Mannila	Philip Yu
3	Padhraic Smyth	Heikki Mannila
4	Jiawei Han	Christos Faloutsos
5	Vipin Kumar	Jiawei Han
6	Philip Yu	Bianca Zadrozny
7	Eamonn Keogh	Padhraic Smyth
8	Kenji Yamanishi	Kenji Yamanishi
9	Mohammed J. Zaki	Inderjit S. Dhillon
10	Charu C. Aggarwal	Vipin Kumar

conferences sharing same authors. Similarly, the $APCPA$ path (authors publishing papers in same conferences) can effectively presents the similarity of authors. Since the similarity on conferences pairs and authors pairs are accurately measured by the relevance paths, the clustering accuracy is high. However, it is not the case for paper clustering. In the $PAPCPAP$ path, the similarity of papers is inferred by the similarity of referenced authors (i.e., the $APCPA$ path), which cannot effectively measure the similarity of papers. So the low-quality similarity of papers leads to the poor clustering accuracy. As a consequence, we need to select appropriate relevance paths to measure the relatedness of objects, which helps to improve the clustering accuracy.

5.5 Semantic Meaning of Relevance Path

We know that different paths have different semantic meanings in heterogeneous networks. Table 7 shows such a case, which searches the most related authors to KDD conference based on two different relevance paths. The *CVPA* path means conferences publishing papers written by authors. It identifies the most active authors to the conference. The *CVPAPA* path means conferences publishing papers written by authors' co-authors. It identities the persons with the most active group of co-authors. In social network setting, this is like identifying the persons with the most active group of friends or potential targets for viral marketing. At first glance, there are no obvious difference between the results returned by these two paths. However, the different ranks of these authors reveal the subtle semantics on the paths. The *CVPA* path returns authors that have high publication records in KDD. For example, Christos Faloutsos published the most papers (32) in KDD. Note that HeteSim does not simply count the number of paths connecting two objects. It also considers the mutual influence of two objects. For example, Jiawei Han and Philip Yu published the second and third highest number of papers in KDD. However, they have wider research interests and published many papers in many other conferences, so their relatedness to KDD decrease based on the *CVPA* path.

By contrast, the *CVPAPA* path emphasizes on the publication records of the co-authors. The results also reflect this point. For example, although Charu C. Aggarwal published 13 papers in KDD, not the highest publication records, he has many co-authors which include many high-publication-record authors (e.g., Philip Yu and Jiawei Han), so he is the first author related to KDD based on *CVPAPA* path. The same thing also happens to other authors. Taking Bianca Zadrozny for example, she only published 6 papers in KDD. However, her co-authors also include many high-publicaton-record authors, such as Philip Yu, Naoki Abe, and Wei Fan. In all, HeteSim can accurately capture the semantics under relevance paths.

6. CONCLUSION

In this paper, we study the relevance search problem which measures the relatedness of heterogeneous objects (including same-typed or different-typed objects) in heterogeneous networks. We propose a novel relevance measure, called HeteSim. As a path-constraint measure, HeteSim can measure the relatedness of same-typed and different-typed objects in a uniform framework. In addition, HeteSim is a semi-metric measure, which can be used in many applications. Extensive experiments validate the effectiveness of HeteSim on evaluating the relatedness of heterogeneous objects.

Acknowledgments.

It is supported by the National Natural Science Foundation of China (No. 60905025, 61074128, 61035003).

7. APPENDIX

Proof of Property 1. According to Definition 6, for each relation instance $a \to b$ in relation $R = AB$ ($a \in A$ and $b \in B$), add an object e ($e \in E$) between a and b, and let $w_{ae} = w_{eb} = \sqrt{w_{ab}}$ where w means the weight of relation instances. Note that for adjacent matrix, $w_{ae} = w_{eb} = \sqrt{w_{ab}}$=1. Since a and b only meet on e, so $R_O(a,:) * R_I(:,b) = w_{ae} * w_{eb} = w_{ab} = R(a,b)$. So $R = R_O \circ R_I$. Since the process is unique, the decomposition is unique.

Proof of Property 2. According to Definition 8, U_{AB} is the normalized matrix of the transition probability matrix W_{AB} along the row vector, which is also the transposition of the normalized matrix of W_{BA} along the column vector (i.e., V_{BA}). So $U_{AB} = V'_{BA}$. Similarly, $V_{AB} = U'_{BA}$.

Proof of Property 3. According to Definition 5, $\mathcal{P} = \mathcal{P}_L\mathcal{P}_R$ and $\mathcal{P}^{-1} = {\mathcal{P}_R}^{-1}{\mathcal{P}_L}^{-1}$. According to Equation 8,

$$HeteSim(a,b|\mathcal{P}) = \frac{PM_{\mathcal{P}_L}(a,:)PM'_{{\mathcal{P}_R}^{-1}}(b,:)}{\sqrt{\|PM_{\mathcal{P}_L}(a,:)\|\|PM_{{\mathcal{P}_R}^{-1}}(b,:)\|}}$$
$$HeteSim(b,a|\mathcal{P}^{-1}) = \frac{PM_{{\mathcal{P}_R}^{-1}}(b,:)PM'_{\mathcal{P}_L}(a,:)}{\sqrt{\|PM_{{\mathcal{P}_R}^{-1}}(b,:)\|\|PM_{\mathcal{P}_L}(a,:)\|}} \quad (9)$$

so $HeteSim(a,b|\mathcal{P}) = HeteSim(b,a|\mathcal{P}^{-1})$.

Proof of Property 4. According to Equation 8, $HeteSim(a,b|\mathcal{P}) = cos(PM_{\mathcal{P}_L}(a,:), PM_{{\mathcal{P}_R}^{-1}}(b,:)) \in [0,1]$. If and on if $PM_{\mathcal{P}_L}(a,:)$ is equal to $PM_{{\mathcal{P}_R}^{-1}}(b,:)$, $cos(PM_{\mathcal{P}_L}(a,:), PM_{{\mathcal{P}_R}^{-1}}(b,:)) = 1$, so $HeteSim(a,b|\mathcal{P}) = 1$.

Proof of Property 5. It is obvious that $SimRank_0(a_1, a_2) = HeteSim(a_1, a_2|I)$ and $SimRank_0(b_1, b_2) = HeteSim(b_1, b_2|I)$. Here $SimRank_i$ means SimRank value after i hop. Let's consider the 1st hop condition.

$$\begin{aligned} &SimRank_1(a_1, a_2) \\ &= \frac{1}{|O(a_1)||O(a_2)|} \sum_{i=1}^{|O(a_1)|} \sum_{j=1}^{|O(a_2)|} SimRank_0(O_i(a_1), O_j(a_2)) \\ &= \frac{1}{|O(a_1)||O(a_2)|} \sum_{i=1}^{|O(a_1)|} \sum_{j=1}^{|O(a_2)|} SimRank_0(b_i, b_j) \end{aligned} \tag{10}$$

$$\begin{aligned} &= \frac{1}{|O(a_1)||O(a_2)|} \sum_{i=1}^{|O(a_1)|} \sum_{j=1}^{|O(a_2)|} HeteSim(b_i, b_j|I) \\ &= HeteSim(a_1, a_2|RR^{-1}) \end{aligned} \tag{11}$$

since $O(a_2) = I(a_2|BA)$, $O(a_1) = O(a_1|AB)$ and $SimRank_0(b_1, b_2) = HeteSim(b_1, b_2|I)$. Similarly, $SimRank_1(b_1, b_2) = HeteSim(b_1, b_2|R^{-1}R)$. Suppose it is correct for k-th hop, let's consider the $k+1$ hop.

$$\begin{aligned} &SimRank_{k+1}(a_1, a_2) \\ &= \frac{1}{|O(a_1)||O(a_2)|} \sum_{i=1}^{|O(a_1)|} \sum_{j=1}^{|O(a_2)|} SimRank_k(O_i(a_1), O_j(a_2)) \\ &= \frac{1}{|O(a_1)||O(a_2)|} \sum_{i=1}^{|O(a_1)|} \sum_{j=1}^{|O(a_2)|} SimRank_k(b_i, b_j) \\ &= \frac{1}{|O(a_1)||O(a_2)|} \sum_{i=1}^{|O(a_1)|} \sum_{j=1}^{|O(a_2)|} HeteSim(b_i, b_j|(R^{-1}R)^k) \\ &= HeteSim(a_1, a_2|R(R^{-1}R)^k R^{-1}) \\ &= HeteSim(a_1, a_2|(RR^{-1})^{k+1}) \end{aligned} \tag{12}$$

Similarly, $SimRank_{k+1}(b_1, b_2) = HeteSim(b_1, b_2|(R^{-1}R)^{k+1})$ So

$$\begin{aligned} SimRank(a_1, a_2) &= \lim_{n \rightsquigarrow \infty} \sum_{k=1}^{n} SimRank_k(a_1, a_2) \\ &= \lim_{n \rightsquigarrow \infty} \sum_{k=1}^{n} HeteSim(a_1, a_2|(RR^{-1})^k) \end{aligned} \tag{13}$$

Similarly,

$$SimRank(b_1, b_2) = \lim_{n \rightsquigarrow \infty} \sum_{k=1}^{n} HeteSim(b_1, b_2|(R^{-1}R)^k) \tag{14}$$

8. REFERENCES

[1] A. Balmin, V. Hristidis, and Y. Papakonstantinou. Objectrank: Authority-based keyword search in databases. In *VLDB*, pages 564–575, 2004.

[2] S. Berchtold, B. Ertl, D. A. Keim, H. peter Kriegel, and T. Seidl. Fast nearest neighbor search in high-dimensional space. In *ICDE*, pages 209–218, 1998.

[3] D. Fogaras, B. Rácz, K. Csalogány, and T. Sarlós. Towards scaling fully personalized pagerank: Algorithms, lower bounds, and experiments. *Internet Mathematics*, 2(3):333–358, 2005.

[4] M. Gupta, A. Pathak, and S. Chakrabarti. Fast algorithms for top-k personalized pagerank queries. In *WWW*, pages 1225–1226, 2008.

[5] G. Jeh and J. Widom. Simrank: a measure of structural-context similarity. In *KDD*, pages 538–543, 2002.

[6] G. Jeh and J. Widom. Scaling personalized web search. In *WWW*, pages 271–279, 2003.

[7] M. Ji, Y. Sun, M. Danilevsky, J. Han, and J. Gao. Graph regularized transductive classification on heterogeneous information networks. In *ECML/PKDD*, pages 570–586, 2010.

[8] R. Jin, V. E. Lee, and H. Hong. Axiomatic ranking of network role similarity. In *KDD*, pages 922–930, 2011.

[9] M. Kolahdouzan and C. Shahabi. Voronoi-based K nearest neighbor search for spatial network databases. In *VLDB*, pages 840–851, 2004.

[10] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. Grouplens: Applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87, 1997.

[11] N. Lao and W. Cohen. Fast query execution for retrieval models based on path constrained random walks. In *KDD*, pages 881–888, 2010.

[12] N. Lao and W. W. Cohen. Relational retrieval using a combination of path-constrained random walks. *Machine Learning*, 81(2):53–67, 2010.

[13] C. Li, J. Han, G. He, X. Jin, Y. Sun, Y. Yu, and T. Wu. Fast computation of simrank for static and dynamic information networks. In *EDBT*, pages 465–476, 2010.

[14] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang. Mining topic-level influence in heterogeneous networks. In *CIKM*, pages 199–208, 2010.

[15] D. Lizorkin, P. Velikhov, M. Grinev, and D. Turdakov. Accuracy estimate and optimization techniques for simrank computation. In *PVLDB*, pages 422–433, 2008.

[16] Z. Nie, Y. Zhang, J. Wen, and W. Ma. Object-level ranking: bringing order to web objects. In *WWW*, pages 422–433, 2005.

[17] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University Database Group, 1998.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transaction of Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[19] Y. Sun, J. Han, X. Yan, P. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. In *VLDB*, 2011.

[20] 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*, pages 565–576, 2009.

[21] H. Tong, C. Faloutsos, and J. Pan. Fast random walk with restart and its applications. In *ICDM*, pages 613–622, 2006.

[22] Q. Xia. The geodesic problem in quasimetric spaces. *Journal of Geometric Analysis*, 19(2):452–479, 2009.

[23] C. Xiao, W. Wang, X. Lin, and H. Shang. Top-k set similarity joins. In *ICDE*, pages 916–927, 2009.

[24] X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger. Scan: an structural clustering algorithm for netowrks. In *KDD*, pages 824–833, 2007.