ABSTRACT
From the past, several studies in the information network mining have been mainly designed for single-typed objects and links, called the homogeneous information network (HoIN). These HoIN-based approaches are definitely unsuitable for multi-typed objects and links, known as the heterogeneous information network (HIN). There is no doubt that most of the real-world networks are not only composed in a complex heterogeneous manner but also are extremely large in size. The big size of these networks is one of the most challenging issues that influence directly the system’s performance. In this paper, our studies are mainly focused on improving the topic-driven weighted similarity measurement between same-typed objects in HIN, based on the meta-path-based mechanism, called W-PathSim. Moreover, our contributions in this paper also aim to optimize the performance of the W-PathSim model in the manner of very large-scaled HIN by combining the proposed W-PathSim model with the approach of distributed computing of ‘graph-frames’ on Spark, called DW-PathSim. The DW-PathSim not only supports in tackling the problem of weighted meta-path-based similarity searching in HINs but also the distributed computing problem on the big networked data. We test the DW-PathSim model with the real-world DBLP dataset in order to demonstrate the effectiveness of our proposed models.

Introduction
This paper is an extended version of the ACIIDS 2018 conference paper (Pham, Do, & Ta, 2018). Depending on the achievements of the proposed W-PathSim model, our works in this paper are concentrated on improving the model’s capability in handling a large-scaled network. The new model is called DW-PathSim, which enables topic-driven similarity measure between nodes in large-scaled networks by using the Spark GraphFrames.
We all know that most of real-world data entities are interconnected via multiple types of relationships within the large-scaled networks, called the information network (Chang et al., 2015; Shi et al., 2017; Sun & Han, 2012, 2013). The real information networks consist of a large number of interacting, multi-typed components such as social networks, biological gene networks, WWW, etc. In the past, most of the studies only considered information networks in the view of the homogeneous aspect, which means that they only contain one type of object and link. These HoIN (homogeneous information network)-based studies are unsuitable for applying in HIN (heterogeneous information network)-based mining tasks due to the failures in distinguishing different types of objects and links in the networks. Due to the existing drawbacks of previous HoIN-based approaches, in recent times, more and more researchers have begun to consider the differences in types of objects and relationships in HINs and are developing a suitable mechanism for leveraging the HIN-based mining tasks. Moreover, as in the case of the big networked data, the developed approaches are also needed to be capable for working on the distributed computing infrastructure. There is no doubt that with the large size of recent real-world networks such as common social networks (Facebook, Twitter, etc.) with the billions of nodes and an uncountable number of relationships, the standalone-based computing mechanism approaches are definitely unaffordable in this case.

**Preliminaries and backgrounds**

In an information network (see Definition 1) mining, evaluating the relationships and computing relevant score between same-typed objects are the fundamental problems. Similarity measurement is considered as a baseline task for other important mining tasks in information networks. Moreover, evaluating the similarity between objects in the information networks also supports the involvements in constructing the object-based information retrieval and recommendation system (Yu et al., 2014). Most of the approaches support for similarity measuring between two same-typed objects are based on evaluating the relationships between linked objects.

**Definition 1** (Information Network (IN)) (Sun & Han, 2012, 2013). The information network is defined as a directed/undirected graph-based structure, denoted as $G = (V, E)$. Each vertex $(v, v \in V)$ represents the object and edge/link $(e, e \in E)$ represents the relationship between
objects in \((V)\) (as illustrated in Figure 1(A)). An IN contains the set of objects’ types \((A)\) and relations \((R)\). There are two mapping functions, including \(\phi: V \rightarrow A\), \(\phi(v) \in A\) and \(\psi: E \rightarrow R\), \(\psi(e) \in R\). In general, a specific IN belongs to two main types:

- **Definition 1a.** Homogeneous Information Network (HoIN). HoIN contains only one type of object and relationship between objects (where \(|A| = 1\) and \(|R| = 1\).
- **Definition 1b.** Heterogeneous Information Network (HIN). On the other hand, HIN is a complex type of IN, with the number of objects’ types and relationships’ types more than 1 (where \(|A| > 1\) and \(|R| > 1\).

**Definition 2 (Network Schema (NS))** (Sun & Han, 2012, 2013). Similar to the schema structural paradigm of the relational table database, the information network is defined as the ‘meta template’ or ‘network structural schema’ (as shown in Figure 1(B)). The network schema of a given network is denoted as \(\text{NS}_G = (A, R)\), representing how a specific object/entity type is connected with the other objects in different or same type.

Multi-typed object and link in HIN mining. In an HIN, the network schema (see Definition 2) is used in order to specify the constraints of type on the sets of existed objects and their relationships. Similarity measurement is only meaningful when evaluated on the set of same-typed objects, definitely we can calculate the similarity scores of different-typed pairwise objects, such as ‘author’ vs. ‘paper’ or ‘paper’ vs. ‘venue’, etc. Moreover, the similarity scores between pairwise same-typed objects are computed also based on evaluating how they are linked to each other via different paths in the given network. However, the paths that are used to link same-typed objects might be not the same due to the differences in the relationship and sequential order of linked nodes within the path. For example, in the DBLP bibliographic network, two ‘author’ objects might be linked via different type of paths such as author \(\xrightarrow{\text{work at}}\) affiliation \(\xleftarrow{\text{work at}}\) author, author \(\rightarrow\) paper \(\xleftarrow{\text{write}}\) author, etc. and these types of paths cannot be treated as the same because they are different in the meaning, such as the first path indicates the co-worker relationship between two authors while the left one represents the co-authorship relation between two authors. Therefore, Sun et al. (2011) have proposed an approach of using ‘meta-path’ (see Definition 3) to indicate the semantic relations between pairwise objects which support to leverage the similarity measure as well as another mining task in HIN.

**Definition 3 (Meta-path).** In HIN mining, a meta-path \((P)\) is defined as a sequential order of objects and relation with \((l)\) in length, denoted as \(P = A_1 \xrightarrow{R_1} A_2 \cdots \xrightarrow{R_{l-1}} A_l\) or \(P = (A_1, A_2, \ldots, A_n)\), in short.

In HIN-based mining techniques, the meta-path is considered as the principle paradigm for discovering relationships in the context of the multi-typed object network. Totally different from HoIN-based similarity search approaches, two same-typed objects in a HIN can be linked via different meta-paths and every meta-path is different in the physical semantic meaning (as examples in Figure 2 and Table 1). Following the defined meta-path \((P)\), there might be one or several concrete paths: \(p_1p_2 = (a_1a_2\ldots a_i a_{i+1})\) which linked two specific objects: \(a_1\) and \(a_{i+1}\) is called a ‘path instance’ of meta-path \((P)\).
Remaining issues and challenge statements

In recent times, most of the HIN-based approaches for similarity measurement between same-typed nodes encounter three main problems, which include:

Binary relation-based similarity evaluation challenge. In prior studies, there was no method which focuses on considering the weighted attributes of the relation between same-typed objects while evaluating the similarity. The approach of the PathSim model (Sun et al., 2011) is considered the most well-known meta-path-based binary link evaluation approach. The similarity between two same-typed objects is obtained mostly depending on evaluating how many these same-typed pairwise objects are connected, following the paradigm that the more they are linked, the more they are similar to each other. This type of method encounters the issue of shortage in considering rich attribute values of nodes and links for the similarity evaluation process. For instance, in the DBLP bibliographic network, there are multiple object’s types which have their own attribute such as ‘author’ objects have own attribute like ‘gender’, ‘age’, ‘home-town’ ‘education background’, etc. or ‘paper’ objects have ‘topic’, ‘category’, etc. attributes. There is no doubt that these attributes play an important role in the processes of extracting relevant features that are used as the baseline for computing the similarity between same-typed objects.

Non/less-connected objects similarity evaluation challenge. Moreover, as a subsequent result of the link-based similarity evaluation approach, the level of relevancy between objects is obtained by counting the number of paths that connect these objects together; hence, the less-linked objects will be considered as less relevant. This situation is not correct in some cases, for example, assuming that two ‘authors’ work in different affiliations but they mostly contributed their works to the ‘bioinformatics’ field, and they
never or rarely submit their papers to the same venues or journals. So, within the link-based approach, the similarity score between these two authors is very low or even zero, and definitely this result seems unreasonable in this case.

**Big networked data processing challenge.** Most of the real-world information networks are very large in size in nature, typically such as social networks, WWW, biological gene database, chemical compounds database, bibliographic network, etc. with the number of network’s nodes and edges being in billions. Therefore, traditional standalone-based approaches for the large-scaled information network are unaffordable due to low-performance and time-consuming as well as challenges related to upgrading the computing resources. Therefore, the need for constructing new methods, which are affordable for handling the large-scaled information networks, is totally necessary.

**Proposed solutions and contributions**

As aforementioned problems in this paper, our contributions concentrate on tackling the problems related to binary link-based similarity evaluation of previous approaches as well as big networked data handling. We propose the DW-PathSim model, which is the extension of the previous W-PathSim (Pham et al., 2018) content-based HIN similarity measurement model in the distributed computing framework of Apache Spark with Graph-frames for large-scaled graph processing. For the W-PathSim model, it supports in the thorough evaluation of the similarity in the topic of content-based objects such as ‘paper’ objects in a bibliographic network combining with the classic meta-path-based approach of PathSim. This combination can help to overcome mentioned challenges of the link-based similarity measurement technique. Overall, our works in this paper are three-fold, including:

- First of all, we use the LDA model to extract the topic distributions from content-based objects within the networks. Then, these topic probabilistic distributions are used to compute the topic similarity between these content-based objects via the cosine similarity metric.

- After that, we present the methodology of W-PathSim and DW-PathSim models which is the combination of the meta-path-based similarity measurement with the topic similarity between content-based objects following the pre-defined meta-paths. For the DW-PathSim model, this model is designed to work on the distributed computing environment of Apache Spark in order to handle the large-scaled heterogeneous network.

- Finally, we conduct empirical studies on our proposed models by comparing with previous approaches that are applied both in HoIN-based and HIN-based similarity measurement. The experimental outputs demonstrate the effectiveness of our proposed models in both the enrich-schema information network similarity measurement as well as time-consuming performance.

The rest of our paper is structured in four main sections. We discuss previous works as well as motivations in the second section. In the third section, we present the related background concepts, our methodologies and the implementation of both the proposed W-PathSim and DW-PathSim models. For experimental studies, in the fourth section, we describe our experimental environment scenario, set-up as well as evaluation metric.
usage; in this section, we also give thorough discussions about the experimental outputs. And the last section is the conclusion about our works and next improvements.

**Related works and motivations**

Following the development of networked data mining, recently the HIN mining has attracted many researchers in multiple disciplines due to its wide applications. From the baselines of the heterogeneous structure mining problem which was first introduced by J. Han and Sun Y., the HIN mining problem has been more and more popular recently. In the complicated dynamic real-world network context, the HIN-based mining tasks are needed to rapidly develop in order to meet the complex knowledge mining requirements. It is unnecessary to say the similarity search is a primitive task for solving all the other techniques that are used to discover the knowledge from the information network, such as object/node clustering, object/node classification, link prediction, recommendation system, etc. The key principle of the HIN-based mining technique is using the ‘meta-path’(s) also called the ‘semantic path’(s) for guiding the processes of similarity evaluation between pairwise nodes in HIN. Several approaches have been developed under the paradigm of meta-path-based mining such as the most well-known RankClus model (Sun et al., 2009), which is the in-cluster-ranking model for the bi-type network and the proposed NetClus (Sun, Yu, & Han, 2009) for solving the start-network clustering problem. In the next development, the proposed PathSim (Sun et al., 2011), PathSelClus (Sun et al., 2013) algorithms were considered as a principal baseline for meta-path-based similarity measurement with strengthened theories of meta-path-based random walks in HIN. However, as the aforementioned problems, most of the previous models are considered as the link-based similarity measurement technique, which is mostly based on the relationships between objects to compute the level of relevance. The shortage of considering object or relation own attributes while evaluating the similarity might lead to the decrease in the output accuracy. Moreover, most of the contemporary HIN-based similarity measurement approaches are only suitable for small-scaled information networks that fail to apply in the parallel and distributed computing environment which is commonly used to handle the big networked dataset.

**Methodology and system design**

In this section, we present the approaches of W-PathSim and DW-PathSim models. The W-PathSim model is designed to run on the standalone-based environment, while the DW-PathSim is aimed to work on the distributed computing environment which supports handling of very large-scaled information network. Both the proposed models are the combination of previous meta-path-based and topic attribute-based similarity between same-typed objects by using the LDA topic model, in a given content-based heterogeneous network.

**Topic similarity evaluation between content-based objects in HIN**

Within the content-based HIN such as the DBLP bibliographic network, there are rich-text objects such as ‘papers’ which can be used to extract the latent topic distributions by the support of topic modelling techniques, such as LSI, pLSI and LDA. In this paper, we select
the LDA topic model (Blei, 2012; Blei, Ng, & Jordan, 2003; Steyvers & Griffiths, 2007) to obtain the topic distributions of content-based objects in HIN. In general, the LDA model supports to extract the latent topic ($T$) distributions over content-based objects, denoted as ($d$). The probabilistic distribution of the ($j$)th topic over a specific content-based object ($i$), donated as $P(t_j|d_i) = \theta_{ij}$ with ($i$:$t \in T$). For the set of probabilistic distributions of a set of latent topics ($T$), we can represent the content-based object ($d_i$) as a $|T|$-dimensional vector $\vec{d}_i$ (as shown in Equation (1)):

$$\vec{d}_i = [P(t_1|d_i) \ldots P(t_T|d_i)] = [\theta_{i1} \ldots \theta_{iT}]$$

where

- $\vec{d}_i$ is the set of probabilistic topic distributions of the content-based object ($d_i$) which is represented as a $|T|$-dimensional feature vector.
- $P(t_j|d_i)$ represents the probabilistic distribution of the ($j$)th topic at a content-based object ($d_i$).

Finally, we use these latent topic distributed proportions as the weights between the document/paper in HINs. For obtaining the similarity score between two same-typed content-based objects, denoted as ($d_x$) and ($d_y$), we use the cosine similarity. The cosine similarity between two objects ($d_x$) and ($d_y$), denoted as $\cos \text{sim}(\vec{d}_x, \vec{d}_y)$ is computed as the following equation (as shown in Equation (2)):

$$\cos \text{sim}(d_x, d_y) = \frac{\vec{d}_x \cdot \vec{d}_y}{\|\vec{d}_x\| \cdot \|\vec{d}_y\|} = \frac{\sum_{t=1}^{T} \theta_{itx} \theta_{ity}}{\sqrt{\sum_{t=1}^{T} \theta_{itx}^2} \sqrt{\sum_{t=1}^{T} \theta_{ity}^2}}$$

where

- $\vec{d}_x$ and $\vec{d}_y$ represent for the $|T|$-dimensional feature vector of content-based objects ($x$) and ($y$), respectively.
- $\theta_{itx}$ and $\theta_{ity}$ represent the component of the feature vector $\vec{d}_x$ and $\vec{d}_y$, respectively, merely the probabilistic distribution value of the ($i$)-th topic at each object.

Following the given equation, the topic relevance between two content-based objects such as papers in DBLP is calculated by the cosine similarity metric. Normally, most of the content-based objects in HINs are not directly connected. Their relationships are represented as meta-path such as two ‘paper’ objects are indirectly linked in common meta-paths like A–P–V–P–A, V–P–A–P–V, etc. Then in the next section, we define the combination of meta-path-based and topic similarity of content-based objects within meta-path to leverage the process of object similarity measurement.

**W-PathSim model: the combination of meta-path-based and topic attribute-based similarity measurement in content-based HIN**

By obtaining the topic similarity between content-based objects, such as ‘paper’ in the DBLP network, we can use them as the weighted attributes to leverage the process of
similarity evaluation between other objects’ types such as ‘author’ or ‘venue’ following the predefined meta-paths, such as \( A \xrightarrow{\text{write}} P \xrightarrow{\text{submit}} V \xleftarrow{\text{publish}} P \xrightarrow{\text{write}} A, \ A \xrightarrow{\text{write}} P \xrightarrow{\text{cite}} V \xrightarrow{\text{cite}} P \xleftarrow{\text{write}} A, \ V \xrightarrow{\text{publish}} P \xrightarrow{\text{write}_{\text{by}}} A \xrightarrow{\text{write}} V, \) etc. Giving a general symmetric meta-path \((P)\), with length is \((l)\), \(P = A_1 \xrightarrow{R_1} \cdots A_c \xrightarrow{R_2} \cdots A_l \xleftarrow{R_{l-1}} \cdots A_2 \xleftarrow{R_1} A_1\), the meta-path \(P\) is composed of at least one content-based object, denoted as \((A_c)\) inside. The W-PathSim model is restricted to use only for meta-paths which contain the content-based object. The fact is that this limitation does not seem a considerable problem due to the following reasons. First of all, there is no doubt that most of the rich-text HINs often have a significant amount of content-based objects, such as: ‘comment’, ‘post’, ‘news’, etc. on the social networks such as Facebook, Twitter, etc. ‘researching paper/article’ on the bibliographic network such as DBLP, DBIS, etc. Moreover, these content-based objects definitely appear in most of common meta-path(s) as well as have considerable influences on the other object’s types. Therefore, our proposed models can be widely applied in multiple cases of HIN mining.

In our proposed W-PathSim model for two same-typed objects, denoted as \((x)\) and \((y)\), the weighted similarity score of this pairwise object follow the meta-path \((P)\). We re-define the PathCount function of the PathSim model (Sun et al., 2011) as a topic-driven weighted similarity path counting mechanism, denoted as \( W \)-PathCount as follows (as shown in Equation (3)):

\[
W \text{- PathCount} (x \bowtie y, P) = \sum_{p,p\in|P|} w_{e(x\bowtie y)} \times \cos \_ sim(x_c, y_c)
\]

where

- \( p \) and \(|P|\) represent the specific path and a total number of path instances between two same-typed objects \((x)\) and \((y)\), respectively.
- \( w_{e(x\bowtie y)} \) is the total weight of link which connects two same-typed objects \((x)\) and \((y)\), following the meta-path \((P)\), because most of the relation’s types in HIN are binary relations, therefore the value of \( w_{e(x\bowtie y)} \) is often set to 1.
- \( x_c \) and \( y_c \) represent for the content-based objects of the object \((x)\) and \((y)\) following the meta-path \((P)\), respectively. Both \( (x_c)\) and \( (y_c)\) content-based objects are in the same type, with \( \phi(x) = A_c \) and \( \phi(y) = A_c \).

For the previous approach of PathSim, the path count function, which supports to calculate the total path instances between two objects, is not clearly evaluated. For a specific meta-path, all the path instances’ weights are assigned as 1 (binary relation). The traditional PathSim model depends mostly on the number of paths that connect two same-typed objects. It does not thoroughly investigate the importance of each path. The more two objects connect, the more they are similar. This assumption is not totally correct in some cases. For example, as illustrated in Figure 3, with the pre-defined meta-path A-P-V-P-A, there are two path instances between two authors (1) and (2), starting from author (1), which are (path 1): [author 1 → paper 1 → venue 1 → paper 3 → author 2] and (path 2): [author 1 → paper 2 → venue 1 → paper 3 → author 2]. We can see that most of the edges in path-1 and path-2 are the binary relation, such as author–
paper, paper–venue, so the overall weights for these paths are all 1. Hence, the link-based approach cannot help to distinguish what are the differences between these path instances. As shown from the example in Figure 3, following the previous link-based approach, the output will indicate that author (3) is much more similar than author (1) because the number of path instances between authors (1) and (3) is higher than authors (1) and (2). But thoroughly looking at the inside content of their submitted papers, most of the papers of author (3) cover the topics/subtopics of ‘artificial intelligence and robotics’ while authors (1) and (2) share the same interests on ‘data mining and retrieval’ fields. Subsequently, concluding author (3) is much more similar with author (1) than author (2) which does not make sense in this case.

Tables 1 and 2 show examples of path instances for meta-path A–P–V–P–A and V–P–A–P–V, respectively. Therefore, we need to add the topic weighting attributes of pairwise content-based objects to control the output following on the relevancy of topics. Finally, the final function of W-PathSim is changed by the path-counting function which is described in Equation (3) as follows (as shown in Equation (4))

\[
W - \text{PathSim}(x \leadsto y, P) = \frac{2 \times W - \text{PathCount}(x \leadsto y, P)}{W - \text{PathCount}(x \leadsto \neg x, P) + W - \text{PathCount}(y \leadsto \neg y, P)} \tag{4}
\]

where

- \( x \leadsto \neg x, P \), represents all path instances, following meta-path \((P)\), which starting form object \((x)\) to other objects which are in the same type with \((x)\), and with \( \phi(x) = \phi(\neg x) \).
- \( y \leadsto \neg y, P \), represents all path instances, following meta-path \((P)\), which starting form object \((y)\) to other objects which are in the same type with \((y)\), with: \( \phi(y) = \phi(\neg y) \).

Figure 3. Example of meta-path A–P–V–P–A represents the set of common authors who often submit their papers on same specific venues.
For the overall information network, the W-PathSim model enables one to capture the similarity score between two same-typed objects which depend on not only how much these pairwise objects are connected but also on how their incorporated (interlinked) objects are similar to each other. This fact is reasonable in the realistic world as most relevant authors often research and publish their papers on the same topic fields, in the bibliographic network. Or a set of friends on the social network, like as the Facebook, often post comments or feeds about the same social topics, etc. Therefore, the combination of link-based and topic attribute-based approach in the W-PathSim model is capable to leverage the similarity searching task in the context of a heterogeneous network (Tables 4 and 5).

**Table 2.** Examples of path instances for ‘Christos Faloutsos’ author with other authors via the meta-path A–P–V–P–A in the DBLP bibliographic network.

| A (author)       | P (paper)                                      | V (venue)                  | P (paper)                                      | A (author)       |
|------------------|-----------------------------------------------|----------------------------|-----------------------------------------------|------------------|
| Christos Faloutsos | Fast subsequence matching in time-series databases | SIGMOD                     | Automatic subspace clustering of high dimensional data for data mining applications | Rakesh Agrawal   |
| Christos Faloutsos | FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets | SIGMOD                     | Closet: An efficient algorithm for mining frequent closed itemsets. | Jiawei Han       |
| Christos Faloutsos | Graphs over time: densification laws, shrinking diameters and possible explanations | SIGKDD                     | Mining concept-drifting data streams using ensemble classifiers | Philip S. Yu     |

**Table 3.** Examples of path instances for ‘SIGMOD’ venue with other venues via meta-path V–P–A–P–V in the DBLP bibliographic network.

| Venue (V) | P (paper) | A (author) | P (paper) | Venue (V) |
|-----------|-----------|------------|-----------|-----------|
| SIGMOD    | Closet: An efficient algorithm for mining frequent closed itemsets. | Jiawei Han | CMAR: Accurate and efficient classification based on multiple class-association rules | ICDM       |
| SIGMOD    | FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets | Christos Faloutsos | Graphs over time: densification laws, shrinking diameters and possible explanations | SIGKDD     |

For the overall information network, the W-PathSim model enables one to capture the similarity score between two same-typed objects which depend on not only how much these pairwise objects are connected but also on how their incorporated (interlinked) objects are similar to each other. This fact is reasonable in the realistic world as most relevant authors often research and publish their papers on the same topic fields, in the bibliographic network. Or a set of friends on the social network, like as the Facebook, often post comments or feeds about the same social topics, etc. Therefore, the combination of link-based and topic attribute-based approach in the W-PathSim model is capable to leverage the similarity searching task in the context of a heterogeneous network (Tables 4 and 5).

**Meta-path-based traversal via BFS for information network discovering**

**Definition 4** (Commuting matrix \( (M_P) \) (Sun et al., 2011)). The commuting matrix, denoted as \( (M_P) \) for specific meta-path \( (P) \), with \( l \) being the meta-path’s length, the meta-path is given as \( P = A_0^l A_2^l \ldots A_l^l \). \( M_P \) is defined as \( M_P = W_{A_1A_2} \ldots W_{A_lA_{l+1}} \), where \( W_{A_iA_j} \) represents for the adjacency matrix between object’s type \( A_i \) and \( A_j \), and \( M_{ij}^{(l)} \) represents the number of path instances between two root objects \( (x_i, \delta(x_i) \in A_i) \) and \( (y_j, \delta(y_j) \in A_j) \).

Obtaining the number of path instances between two same-typed pairwise nodes over a given information network is quite complex in the heterogeneous context of multi-typed objects and relationships. It is true because in the circumstance of dealing with a very large network we cannot figure how many relation’s types that each object has as well as the types of target objects which are linked to the current investigated object. Therefore,
we need to use the meta-path to guide the system travel through specific routes only. In this section, we define the meta-path-based traversal via applying the BFS strategy which is used for computing the number of path instances between two same-typed pairwise nodes and constructing the commuting matrix (Sun et al., 2011) (see Definition 4). Overall processes are described in Algorithms 1 and 2.

**Algorithm 1.** Pseudo code for computing the number of path instances of a specific object in a given heterogeneous network via BFS following defined meta-path ($P$).

**Table 4.** Top 5 authors whose works are similar to ‘Christos Faloutsos’ by meta-path A–P–V–P–A via PathSim and W-PathSim approaches.

| PathSim         | W-PathSim          |
|-----------------|--------------------|
| Rakesh Agrawal  | Rakesh Agrawal     |
| Jiawei Han      | Jiawei Han         |
| Philip S. Yu    | Wei Wang           |
| Hans-Peter Kriegel | Philip S. Yu      |
| Haixun Wang     | Hans-Peter Kriegel |

**Table 5.** Top 5 venues/conferences that are similar to ‘SIGMOD’ by meta-path V–P–A–P–V via PathSim and W-PathSim approaches.

| PathSim         | W-PathSim          |
|-----------------|--------------------|
| KDD             | KDD                |
| ICDM            | Data Min. Knowl. Disc. |
| TKDE            | TKDE               |
| ICDE            | Knowl. Inf. Syst.  |
| Knowl. Inf. Syst. | ICDM               |
Algorithm 2. Pseudo code for overall processes of constructing a full commuting matrix for all pairwise same-typed objects in a given heterogeneous network following the defined meta-path (P).

**Input**: the heterogeneous network, denoted as \( G = (V, E) \), with pre-defined meta-path \( (P) \), \( P = A_1;A_2;\ldots,A_l \) with \( l \) being its length.

**Output**: the commuting matrix \( (M_P) \).

1: **Function** Constructing \( M_P \) \_BFS \((G, P)\):
2: **Create**: candidates = list\_node\_by\_type\((G, \phi(A_1))\)
3: **Create**: \( M_P \) = \([|\text{candidates}|] \times [|\text{candidates}|]\)
4: **For** candidate in candidates:
5: **Set**: next = candidate;
6: **MetaPath\_BFS\_Traversal\((G, P, M_P, \text{candidate}, \text{next})\);**
7: **End for**
8: **Return** \( M_P \)
9: **End function**

**DW-PathSim: the distributed computing framework for W-PathSim in Spark and the Graph-frames environment**

We all known that most of the real-world information networks are very complicated in their structure as well as huge in size, such as more than 1.8 billion linked websites in WWW, approximately 500 million users and company in the LinkedIn network, or over 1.74 billion users along with uncountable relationships in Facebook social network, etc. Therefore, it is true that most of the traditional of the standalone-based processing method seems unsuitable for handling tasks related to knowledge discovering from these such large-scaled networks. In order to overcome the challenges which incorporate big networked data, the approaches of distributed parallel computing are taken into consideration, one of the trusted available solutions for handling scalable distributed graph-based data structure is Apache Spark, firstly developed by Matei Zaharia, at AMPLab in University of California, Berkeley. There are two main components of Spark that support handling graph-based distributed computation, which are GraphX and Graph-frames. Both components’ mechanisms are designed to work for distributed storage and graph processing efficiently in the resilient distributed dataset (RDD) which is affordable for scalable networked data computation. In this section, we demonstrate the approaches of using Spark’s graph-based distributed computing components for meta-path traversal via motifs finding a mechanism, which helps us to fast discover structural patterns of a given heterogeneous network following defined meta-paths. A motif is defined as a sequential order of nodes and edges within the network, denoted as \( G = (V, E) \), as a tuple following specific structural pattern. The general form of a motif is defined as the following equation (as shown in Equation (5)):

\[ s \rightarrow [e] \rightarrow t \text{ with } (s, t \in V) \text{ and } (e \in E), \]  

where

- \( s \) and \( t \) are the source and target nodes of the sequence, respectively, including constraints on the attributes of source and target nodes.
• $e$ represents the direction as well as constraints on the attributes of the relationship between the source ($s$) and target ($t$) nodes.

For representing a longer sequential relation, such as the meta-path(s), we can combine multiple motifs to meet the searching requirement. In this paper, we define the method for constructing the motif querying pattern to find all the paths between two same-typed pairwise nodes ($x$ and $y$) which depends on the structure of a specific meta-path ($P$), with length ($l$), denoted as $\text{motif}_P(x \rightsquigarrow y)$, following the general formulation (as shown in Equation (6)):

$$\text{motif}_P(x \rightsquigarrow y) = \left\{ x \xrightarrow{\sigma(R_2)} \sigma(A_2) ; \ldots ; \sigma(A_{(l-1)}) \xrightarrow{\sigma(R_{(l-1)})} y \right\}$$  \hspace{1cm} (6)$$

where

• $A$ and $R$ denote the type of object and the relationship between two linked objects, respectively.
• $\sigma$ represent the constraint function which supports to restrict the attributes of links and objects.

The number of tuples which are used to compose a specific meta-path is equal to the length ($l$) of that meta-path. For example, with the meta-path: $A$-P-V-P-A (author $\rightarrow$ paper $\rightarrow$ venue $\rightarrow$ author, $l = 4$), with this meta-path, we will have the semantic meaning of relationship like this: an [author 1], as ($s$), submits his/her paper to the same venue as [author 2], as ($t$), with $\phi(s)$ and $\phi(t)$ = ‘author’, we split this meta-path into 4 tuples as following: $[s] \equiv_{\text{write}} a; b \equiv_{\text{submit}} c; d \equiv_{\text{submit}} c$, $[t] \equiv_{\text{write}} d$, with $\phi(a)$ and $\phi(d)$=’paper’, $\phi(c)$ =’venue’. Back to the issue of the meta-path-based path counting problem, in the DW-PathSim model, we apply the motif finding to obtain the total number of path instances between two same-typed pairwise nodes as the following processes (as shown in Algorithm 3):

Algorithm 3. Pseudo code for the overall processes of constructing the commuting matrix via Spark graph-frames motifs finding, following the meta-path ($P$) in the DW-PathSim model.

Input: the heterogeneous network, denoted as $G = (V, E)$, with pre-defined meta-path ($P$), $P = A_1\ldots A_l$, with ($l$) being its length.

Output: the commuting matrix ($M_P$).

1: Function Spark\_GF\_Constructing ($G$, $P$):
2: Create: candidates = list\_node\_by\_type($G$,$A_1$);
3: Create: $M_P$ = [candidates][candidates];
4: Mapping: GraphFrame$G$ $\leftarrow$ $G$;
5: For i in range (0, length(candidates):
6: For j in range (0, length(candidates):
7: Define: $\text{motif}_P(i \rightsquigarrow j)$ (Equation (6))
8: Set: paths = GraphFrame$G$.find ($\text{motif}_P(i \rightsquigarrow j)$);
9: Update: $M_P[i][j]$ = count(paths);
10: End for
11: End for
12: Return $M_P$
13: End function
In the beginning, the system generates the set of pairwise same-typed nodes and created a commuting matrix $M_P$ with the size being equal to the number of candidates. After that the information network is mapped to the graph-frames. Basing the predefined meta-path $(P)$, the motifs are created with the structure following Equation (6). Finally, the motifs are mapped into the graph-frames to find all the matched path instances, the number of paths then update to the created commuting matrix. We use the Scala programming language with the Graph-Frame as follows:

```scala
gf.loadGraphFrame();
gf.
  find('A1-[w1]->(P1);(P1)-[s1]->(V);(A2)-[w2]->(P2);(P2)-[s2]->(V)').filter('w1.type='WRITES' and s1.type='PUBLISHED_AT' and '+w2.type='WRITES' and s2.type='PUBLISHED_AT' and A1.id != A2.id').show();
```

### Experimental studies and discussions

In this section, we demonstrate our experimental studies in order to evaluate the effectiveness of our proposed models. For the proposed W-PathSim model, which is the combination of meta-path-based and topic-driven attribute-based similarity measurement, we test the model with previous approaches of HoIN-based and HIN-based similarity measurement to evaluate the accuracy of W-PathSim with previous models. Moreover, we also conduct experiments for comparing the time-consuming performance of the DW-PathSim model, which is implemented in the distributed computing environment of Apache Spark with the standalone-based W-PathSim model to demonstrate the advantages of the DW-PathSim model in handle large-scaled information networks.

### Experimental dataset and evaluation metric usage

In this paper, we use the real-world DBLP bibliographic network\(^1\) as the main dataset for all experiments. The DBLP bibliographic network contains over 2M authors, 4.1M papers and over 5.3K venues/journals. For the text content of published papers in DBLP, we use the abstract text dataset, which is provided by Aminer.\(^2\) The Aminer corpus contains over 1.5M abstract contents of published DBLP papers. From the 1.5M abstract text document of Aminer, we randomly selected 100K papers as the main corpus for training the LDA topic model. By using LDA, we extract 10 latent topics and 20 keywords for each topic from that text corpus. For evaluating the accuracy of outputs, we used the nDCG (normalized Discounted Cumulative Gain) (Järvelin & Kekäläinen, 2002) metric to compute the accuracy rate. Each output result is intuitively evaluated and the specific relevancy level score is assigned from 1 to 3 (higher the better), where 0 is the non-relevant object to the target object in the user’s query, 1 is quite a relevant object to target one in the query, 2 is closely relevant object to target one in the query and score 3 is very/highly relevant object to target object one in the query.

### Experimental set-up, results and discussions

#### Model accuracy evaluation

In order to evaluate the accuracy of the W-PathSim model, we compare the output results of the W-PathSim model with both HoIN-based models, such as Personalized PageRank (PPR), SimRank and HIN-based PathSim model. For HIN-based
models, we use two main meta-paths, including A–P–V–P–A (to search most relevant authors with a specific author in the query) and V–P–A–P–V (to search most relevant venues with a specific venue in the query). The output results are divided into four output sets: top 5, top 10, top 15 and top 20 results. For finding top \(k\) similar authors in the given experimental dataset, we selected five top-active authors, as the source authors for querying, which include ‘Jiawei Han’, ‘Christos Faloutsos’, ‘AnHai Doan’, ‘Geoffrey E. Hinton’, ‘Christopher D Manning’, then we get the average results for all five authors as the final results. Some output results for two authors: ‘Christos Faloutsos’ and ‘AnHai Doan’ are shown in Tables 6 and 7, and the overall experimental accuracy results are shown in Table 8.

Similar to top \(k\) author searching, for obtaining the set of relevant venues from the current experimental bibliographic network, we also selected five most well-known venues as the source objects for querying, including: ‘KDD’, ‘SIGMOD’, ‘CVPR’, ‘NIPS’, ‘ACL’ (Figures 4 and 5). The accuracy output results are shown in Table 9.

The overall experimental results showed that the W-PathSim outperforms both previous HoIN-based as well as HIN-based approaches. Especially with HoIN-based

### Table 6. Top 10 most similarity authors to ‘Christos Faloutsos’ via PathSim and W-PathSim for the A–P–[V]–P–A meta-path in the DBLP dataset.

| PathSim          | W-PathSim          |
|------------------|-------------------|
| Rakesh Agrawal   | Rakesh Agrawal    |
| Jiawei Han       | Jiawei Han        |
| Philip S. Yu     | Wei Wang          |
| Hans-Peter Kriegel | Philip S. Yu      |
| Haixun Wang      | Hans-Peter Kriegel|
| Wei Wang         | Haixun Wang       |
| Jian Pei         | Jian Pei          |
| H. V. Jagadish   | H. V. Jagadish    |
| Ming-Syan Chen   | Gerhard Weikum    |
| Gerhard Weikum   | Raghu Ramakrishnan|  

### Table 7. Top 10 most similarity authors to ‘AnHai Doan’ computed via PathSim and W-PathSim for the A–P–[V]–P–A meta-path in the DBLP dataset.

| PathSim          | W-PathSim          |
|------------------|-------------------|
| Kevin Chen-Chuan Chang | Sharad Mehrotra   |
| Alon Y. Halevy   | Kevin Chen-Chuan Chang |
| Sharad Mehrotra  | Michalis Vazirgiannis |
| Michalis Vazirgiannis | Alon Y. Halevy |
| Kirthi Ramamritham | Kirthi Ramamritham |
| Stefano Ceri     | Stefano Ceri      |
| Ji-Rong Wen      | Renee Miller      |
| Gio Wiederhold   | Oren Etzioni      |
| RenA©e J. Miller | Ji-Rong Wen       |

### Table 8. nDCG accuracy outputs for top \(k\) similar authors in different approaches.

|        | nDCG Accuracy |
|--------|---------------|
|        | Top 5 | Top 10 | Top 15 | Top 20 |
| PPR    | 0.56324 | 0.49282 | 0.44291 | 0.42123 |
| SimRank| 0.59223 | 0.51291 | 0.45219 | 0.43232 |
| PathSim| 0.78361 | 0.72819 | 0.71289 | 0.68756 |
| W-PathSim| 0.79627 | 0.74829 | 0.73322 | 0.71987 |
Table 9. nDCG accuracy outputs for top k similar venues in different approaches.

|          | nDCG Accuracy |
|----------|---------------|
| Top 5    |               |
| PPR      | 0.67829       |
| SimRank  | 0.69827       |
| PathSim  | 0.84279       |
| W-PathSim| 0.86728       |
| Top 10   |               |
| PPR      | 0.65728       |
| SimRank  | 0.66526       |
| PathSim  | 0.81231       |
| W-PathSim| 0.82231       |
| Top 15   |               |
| PPR      | 0.61231       |
| SimRank  | 0.62354       |
| PathSim  | 0.79962       |
| W-PathSim| 0.81123       |
| Top 20   |               |
| PPR      | 0.59882       |
| SimRank  | 0.61232       |
| PathSim  | 0.78223       |
| W-PathSim| 0.79625       |

Figure 4. Comparisons of different approaches for top k authors search.

Figure 5. Comparisons of different approaches for top k venues search.

Table 10. Experimental infrastructure setups.

| Environment          | Infrastructural description                                                                 |
|----------------------|---------------------------------------------------------------------------------------------|
| Standalone           | Intel Xeon E3-1220 V5, 3.0 GHz, 8 MB Cache, 4 Cores, 4 Threads 4 Gb RAM                      |
| Spark-based distributed cluster | Number of node: 4 (1 master, 3 slaves) Intel Xeon E3-1220 V5, 3.0 GHz, 8MB Cache, 4 Cores, 4 Threads 4 Gb RAM (for each node) |
Figure 6. Comparisons in standalone vs. distributed environments for the meta-path A–P–V–P–A.

Table 11. Execution time for standalone vs. distributed environments for meta-path A–P–V–P–A.

| Dataset size (%) | Execution time(s) Standalone | Execution time(s) Distributed |
|------------------|------------------------------|-------------------------------|
| 10               | 127                          | 421                           |
| 20               | 346                          | 625                           |
| 30               | 519                          | 972                           |
| 40               | 892                          | 1221                          |
| 50               | 1186                         | 1576                          |
| 60               | 1582                         | 1927                          |
| 70               | 2392                         | 2362                          |
| 80               | 3182                         | 2872                          |
| 90               | 4057                         | 3123                          |
| 100              | 4582                         | 3782                          |

Figure 7. Comparisons in standalone vs. distributed environments for the meta-path A–P–A.
approaches of personalized PageRank (PPR) and SimRank, W-PathSim gains better performance in accuracy, over 14.09% in compared with PPR, and approximately 13.71% in compared with SimRank, for both top k authors and venues similarity search. As shown from results, the W-PathSim models also achieve slightly a higher accuracy than the traditional HIN-based PathSim model (2.23% for top k authors searching and 1.85% for top k venues searching).

**Standalone vs. distributed environment evaluation:** For evaluating the effectiveness of applying graph-based distributed computing for the similarity measurement task in the large-scaled network, we implemented the DW-PathSim model, which is a combination of the Spark-based distributed processing environment with the W-PathSim model using the motifs finding mechanism, and it runs on different scales of the experimental dataset (increasing gradually by size – the number of nodes). The overall processes execution times (seconds) are logged to make the comparison with the traditional standalone-based W-PathSim approach. The details about the environmental infrastructure set-up for both W-PathSim and D-PathSim models are described in Table 10.

For testing the performance of standalone-based and distributed-based mechanisms of similarity search in a given network, we split the dataset into 10 parts, following the size (number of nodes), from 10% to 100% (Figures 6 and 7). Both mechanisms are implemented and executed simultaneously on a same dataset’s size, then the overall completion times (in seconds) for each mechanism in each dataset’s size are collected to be compared with each other. The experimental results for meta-path A–P–V–P–A are shown in Table 11 and meta-path A–P–A in Table 12.

The experimental results showed that the distributed-based mechanism of DW-PathSim has gained better performance in time consumption than the standalone-based W-PathSim model. In the context of the large-scaled network, nearly 80% size of the experimental dataset, the DW-PathSim model takes a smaller amount of time duration to complete the processes of meta-path-based traversal and constructing the commuting matrix than the W-PathSim model.

**Conclusion and future works**

In this paper, we present two novel approaches of W-PathSim and DW-PathSim models, which are the combination of topic attribute-based and meta-path-based similarity
searching in HINs. In the context of heterogeneous network mining, the proposed W-PathSim model enables one to thoroughly evaluate not only to the semantic relationships between linked objects following defined meta-paths but also topic relevance which depends on the extracted attributes of content-based objects via the LDA topic model. From the experimental studies, our proposed W-PathSim model has shown capabilities of leveraging the similarity measurement in HIN. We also extend our studies of the W-PathSim model to the aspects of large-scaled networked data handling, called DW-PathSim. The DW-PathSim model is designed to work on the Spark-based distributed computing environment which is capable of handling large-scaled HINs. Through experiments, within the context of large-scaled networks, DW-PathSim is much better in performance than the previous standalone-based W-PathSim model.

Notes
1. DBLP dataset: http://dblp.uni-trier.de/.
2. AMiner dataset: https://aminer.org/.

Disclosure statement
No potential conflict of interest was reported by the authors.

Funding
This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCMC) under the grant number B2017-26-02.

Notes on contributors
Phuc Do is currently an Associate Professor of the Faculty of Information Systems at University of Information Technology (UIT), VNU-HCM, Vietnam. His research interests include data mining, knowledge-based systems and web information exploration, big data analysis and applications.
Phu Pham is currently a Ph.D. student at the Faculty of Information Systems, University of Information Technology (UIT), VNU-HCM, Vietnam. His research interests include data mining, information network mining, big-data analysis, distributed and parallel computing and business information system.

References
Blei, D. M. (2012). Probabilistic topic models. Communications of the ACM, 77–84.
Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. Journal of Machine Learning Research, 993–1022.
Chang, S., et al. (2015). Heterogeneous network embedding via deep architectures. Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 119–128.
Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems (TOIS), 422–446.
Pham, P., Do, P., & Ta, C. D. (2018). W-PathSim: Novel approach of weighted similarity measure in content-based heterogeneous information networks by applying LDA topic modeling. In Asian conference on intelligent information and database systems (pp. 539–549). Cham: Springer.
Shi, C., et al. (2017). A survey of heterogeneous information network analysis. IEEE Transactions on Knowledge and Data Engineering, 29(1), 17–37.

Steyvers, M., & Griffiths, T. (2007). Probabilistic topic models. Handbook of Latent Semantic Analysis, 424–440.

Sun, Y., & Han, J. (2012). Mining heterogeneous information networks: Principles and methodologies. Synthesis Lectures on Data Mining and Knowledge Discovery, 1–159.

Sun, Y., & Han, J. (2013). Mining heterogeneous information networks: A structural analysis approach. ACM SIGKDD Explorations Newsletter, 20–28.

Sun, Y., Yu, Y., & Han, J. (2009). Ranking-based clustering of heterogeneous information networks with star network schema. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 797–806). ACM.

Sun, Y., et al. (2009). RankClus: integrating clustering with ranking for heterogeneous information network analysis. In Proceedings of the 12th international conference on extending database technology: Advances in database technology (pp. 565–576). ACM.

Sun, Y., et al. (2011). PathSim: Meta path-based top-k similarity search in heterogeneous information networks. In Proceedings of the VLDB Endowment (pp. 992–1003).

Sun, Y., et al. (2013). PathSelClus: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. ACM Transactions on Knowledge Discovery from Data (TKDD).

Yu, X., et al. (2014). Personalized entity recommendation: A heterogeneous information network approach. In Proceedings of the 7th ACM international conference on Web search and data mining (pp. 283–292). ACM.