International Journal of Software Innovation
Volume 6 • Issue 2 • April-June 2018

Specifying Knowledge Graph with Data Graph, Information Graph, Knowledge Graph, and Wisdom Graph

Yucong Duan, College of Information Science and Technology, Hainan University, Haikou, China

Lixu Shao, Hainan Unviersity, Haikou, China

Gongzhu Hu, Central Michigan University, Mount Pleasant, USA

ABSTRACT

Knowledge graphs have been widely adopted, in large part owing to their schema-less nature. It enables knowledge graphs to grow seamlessly and allows for new relationships and entities as needed. A knowledge graph is a graph constructed by representing each item, entity and user as nodes, and linking those nodes that interact with each other via edges. Knowledge graphs have abundant natural semantics and can contain various and more complete information. It is an expression mechanism close to natural language. However, we still lack a unified definition and standard expression form of knowledge graph. The authors propose to clarify the expression of knowledge graph as a whole. They clarify the architecture of knowledge graph from data, information, knowledge, and wisdom aspects respectively. The authors also propose to specify knowledge graph in a progressive manner as four basic forms including data graph, information graph, knowledge graph and wisdom graph.

KEYWORDS

Graph, Information, Knowledge, Knowledge Data, Wisdom

1. INTRODUCTION

There are different kinds of discrete data in the real world we live in. The data cannot be used if they exist only in the discrete form. However, this is not worth worrying as we can simply make the data meaningful by giving a specific environment. Data are processed to be useful and presented to us in the form of information, then we can get a lot of fragmented expressions. With these fragmented expressions, that is, the conception "information" we mentioned above, we can combine multiple information to answer more complex questions about how to do it. By abstracting and converting information and data in a given context and the application of data and information (Bellinger & Castro, 2004), knowledge shows up. Furthermore, comprehensive knowledge of the same category can be use of making favorable judgments, precisely predicting, and smartly planning. Obviously, the utilization of vested knowledge is beyond its literal meaning of the category, which is what we say, "wisdom". Figure 1 shows the progressive relationship among data, information, knowledge and wisdom. Data existing as discrete elements have no semantics. Information is data after procession of conceptual mapping and relational connection. Users access to information after filtering valuable information and internalize those information into knowledge. When information is adequately assimilated, it produces knowledge which modifies an individual's mental store of information and benefits his/her development and that of the society in which he/she lives.

DOI: 10.4018/IJSI.2018040102

In our previous work (Duan et al., 2017), we clarified the architecture of Knowledge Graph as a whole and extended the existing concept of Knowledge Graph into four aspects including Data Graph, Information Graph, Knowledge Graph and Wisdom Graph. Shao et al. (2017) proposed to answer the Five Ws problems through constructing the architecture of Data Graph, Information Graph and Knowledge Graph. We clarify the architecture of knowledge graph from $\text{Data}_{\text{DIKW}}$, $\text{Information}_{\text{DIKW}}$, $\text{Knowledge}_{\text{DIKW}}$ and $\text{Wisdom}_{\text{DIKW}}$ aspects respectively. Correspondingly, we propose to extend the existing expression of knowledge graph in a progressive manner as four basic forms including $\text{DataGraph}_{\text{DIKW}}$, $\text{InformationGraph}_{\text{DIKW}}$, $\text{KnowledgeGraph}_{\text{DIKW}}$ and $\text{WisdomGraph}_{\text{DIKW}}$. We propose a DIKW approach to support dynamic semantic modeling through a progressive hierarchy of $\text{DataGraph}_{\text{DIKW}}$, $\text{InformationGraph}_{\text{DIKW}}$, $\text{KnowledgeGraph}_{\text{DIKW}}$ and $\text{WisdomGraph}_{\text{DIKW}}$. We define the resources and the four graphs as follows:

Definition 1: Resource elements ($\text{Elements}_{\text{DIKW}}$).
$\text{Elements}_{\text{DIKW}}$: = <$\text{Data}_{\text{DIKW}}$, $\text{Information}_{\text{DIKW}}$, $\text{Knowledge}_{\text{DIKW}}$, $\text{Wisdom}_{\text{DIKW}}$ >;
Definition 2: $\text{Graph}_{\text{DIKW}}$. We extend the concept of existing knowledge graph into four parts: $\text{DataGraph}_{\text{DIKW}}$, $\text{InformationGraph}_{\text{DIKW}}$, $\text{KnowledgeGraph}_{\text{DIKW}}$ and $\text{WisdomGraph}_{\text{DIKW}}$.
$\text{Graph}_{\text{DIKW}}$: = ($\text{DataGraph}_{\text{DIKW}}$), ($\text{InformationGraph}_{\text{DIKW}}$), ($\text{KnowledgeGraph}_{\text{DIKW}}$), ($\text{WisdomGraph}_{\text{DIKW}}$).

We show the progressive forms of resources type in Table 1. $\text{DataGraph}_{\text{DIKW}}$ can be expressed through a variety of data structures including array, list, link and combination of multiple data structures. Semantic modeling of $\text{Data}_{\text{DIKW}}$, providing a platform-independent $\text{Data}_{\text{DIKW}}$ representation will be a major advantage in the cloud space. $\text{InformationGraph}_{\text{DIKW}}$ expresses the interaction between entities in the form of a directed graph. $\text{KnowledgeGraph}_{\text{DIKW}}$ is of free schema and expresses rich semantic relationships which is conductive to have a completing mapping towards requirements described in the form of natural language. $\text{WisdomGraph}_{\text{DIKW}}$ can be used to organize unknown resources inferred through existing known $\text{Data}_{\text{DIKW}}$, $\text{Information}_{\text{DIKW}}$ and $\text{Knowledge}_{\text{DIKW}}$.

In the rest of this paper, we firstly elaborate representations of $\text{DataGraph}_{\text{DIKW}}$, $\text{InformationGraph}_{\text{DIKW}}$, $\text{KnowledgeGraph}_{\text{DIKW}}$ and $\text{WisdomGraph}_{\text{DIKW}}$ in Section 2, 3, 4 and 5 respectively. Then we describe the progressive relationship among $\text{Data}_{\text{DIKW}}$, $\text{Information}_{\text{DIKW}}$, $\text{Knowledge}_{\text{DIKW}}$ and $\text{Wisdom}_{\text{DIKW}}$ in Section 6 and 7. The related works are elaborated in Section 8. And we conclude our work in Section 9.

2. REPRESENTATION OF $\text{DATAGRAPH}_{\text{DIKW}}$

$\text{Data}_{\text{DIKW}}$ is the symbolic representation of observable properties of the world. $\text{Data}_{\text{DIKW}}$ is obtained by observing basic individual item of numbers or other information, but on its own, without context, $\text{Data}_{\text{DIKW}}$ has no meaning. Storing of $\text{Data}_{\text{DIKW}}$ does not change $\text{Data}_{\text{DIKW}}$ itself, but it has many expression forms (Zins, 2010). As Figure 2 shows, $\text{Data}_{\text{DIKW}}$ can be organized in many different types of data structures, including arrays, stacks, list links and so on. $\text{Data}_{\text{DIKW}}$ can be of structured, semi-structured and unstructured, relational or non-relational form. Generally, $\text{Data}_{\text{DIKW}}$ is represented as many discrete elements originally (Johannessen & Fuglseth, 2014). Figure 3 shows a series of original

Figure 1. Progressive relationship among data, information, knowledge and wisdom

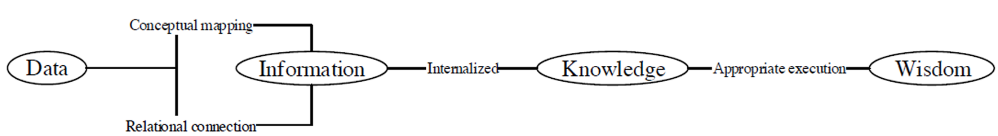


Table 1. Progressive forms of $\text{Elements}_{\text{DIKW}}$ and $\text{Graph}_{\text{DIKW}}$

(right side: general forms)	$\text{Data}_{\text{DIKW}}$	$\text{Information}_{\text{DIKW}}$	$\text{Knowledge}_{\text{DIKW}}$	$\text{Wisdom}_{\text{DIKW}}$
semantic load	not specified for stakeholders/machine	settled for stakeholders/ machine	abstracted on known information	from known towards predicting unknown
format	conceptual collection of data	Conceptual mapping and relational connection of data	probabilistic or categorization of elements and their relationship	(frame or stylish expression)
knowledge answer	who/when/where	what	how	why
usage	identification of existence after conceptualization	communication	reasoning	prediction
graphs	$\text{DataGraph}_{\text{DIKW}}$	$\text{InformationGraph}_{\text{DIKW}}$	$\text{KnowledgeGraph}_{\text{DIKW}}$	$\text{WisdomGraph}_{\text{DIKW}}$

discrete $\text{Data}_{\text{DIKW}}$ points and that we use array, linked list, tree, graph as well as the combination of these four structures to represent $\text{Data}_{\text{DIKW}}$ respectively. Original discrete $\text{Data}_{\text{DIKW}}$ points have no meaning without context. For example, the value 120 can be clinical measurement such as heart rate and it can also indicates the telephone number of the emergency center. Cutting $\text{Data}_{\text{DIKW}}$ from specific context of situations, we cannot determine what $\text{Data}_{\text{DIKW}}$ means for sure. We use a collection $D\ \{d_1, d_2, \ldots, d_i, \ldots, d_n\}$ to represent $\text{Data}_{\text{DIKW}}$ sets where d_i indicates a discrete element. For example, if we input a collection of a series of discrete elements describing risk assessment of software engineering it can be denoted as *D (risk)*. We cannot understand the specific meaning of each element without context and the internal relationship of these elements.

3. REPRESENTATION OF $\text{INFORMATIONGRAPH}_{\text{DIKW}}$

$\text{Information}_{\text{DIKW}}$ is conveyed from $\text{Data}_{\text{DIKW}}$ that has been given meaning by way of relational connection. This "meaning" can be useful, but does not have to be. Items of $\text{Information}_{\text{DIKW}}$ include elements of $\text{Information}_{\text{DIKW}}$ and relationships between elements of $\text{Information}_{\text{DIKW}}$. Elements of $\text{Information}_{\text{DIKW}}$ are displayed as nodes and relationships are represented as lines on $\text{InformationGraph}_{\text{DIKW}}$. $\text{Information}_{\text{DIKW}}$ embodies the understanding of a relationship of some sort and the essence of $\text{Information}_{\text{DIKW}}$ phenomenon has been characterized as the occurrence of a communication process that takes place between a sender and a recipient of a message. The conceptual mapping of different concepts and relationships is called concept mapping (Nadarajan & Chenburger, 2006; Nadarajan & Chen-Burger, 2006). Giving two concepts O_1, O_2 and their associated sets $\{C_1, R_1\}$ and $\{C_2, R_2\}$, conceptual mapping is to identify potential pairs (c_1, c_2) or (r_1, r_2), where $c_1 \in C_1$, $c_2 \in C_2$, $r_1 \in R_1$ and $r_2 \in R_2$. In this way, concept c_1 and relationship r_1 can be translated into instance c_2 and instance r_2 while preserving their original meaning. We define a conceptual mapping function F acting on two concepts C_1 and C_2. The similarity function, denoted as S, is defined in two concepts C_1 and C_2, and a value between 0 and 1 is calculated to indicate the similarity between C_1 and C_2. The logical representation of concept C_i is $L(C_i)$ where function F is evaluated as a similarity representation of the logical representation between C_1 and C_2. The function S is used to define the logical similarity evaluation on $L(C_1)$ and $L(C_2)$.

$$F\left(C_1, C_2\right) = \sum_i w\left(i\right) * S\left(L\left(C_1\right), L\left(C_2\right)\right) \quad (1)$$

Figure 2. Storage of $Data_{DIKW}$ does not change $Data_{DIKW}$ itself

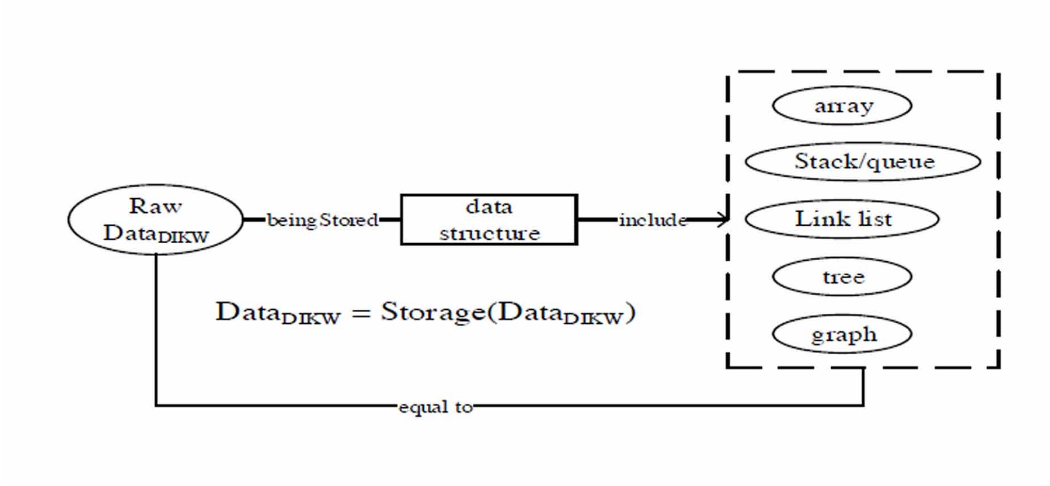
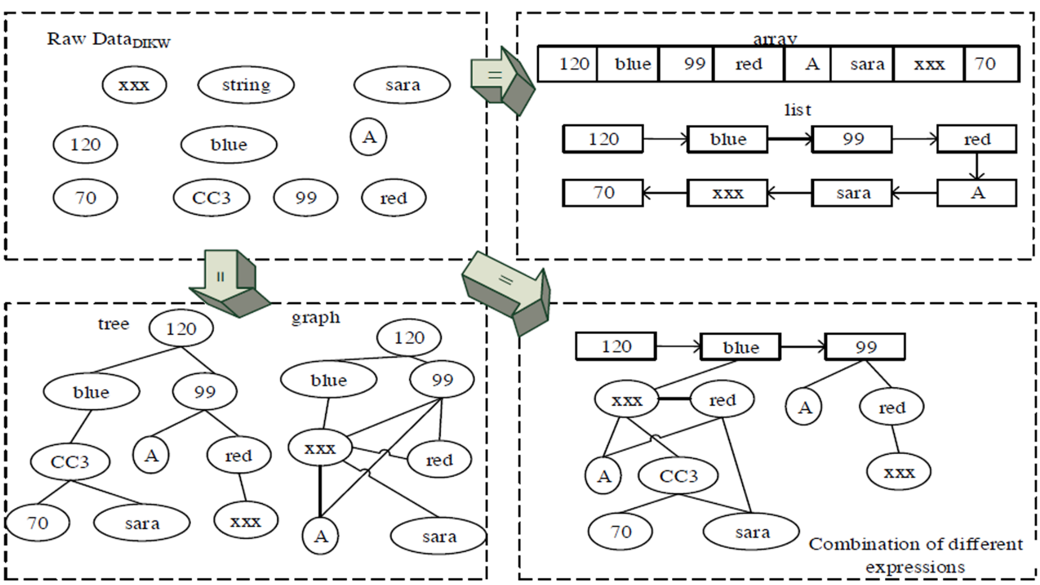


Figure 3. Illustration of $Data_{DIKW}$ expression



There is a need to apply conceptual semantics of professionals from different aspects. Correspondingly, different semantics can be represented by independent logical statements, and function F can be defined exactly as follows:

$$F\left(C_1, C_2\right) = \sum_i w\left(i\right) * S\left(L\left(C_1\right), L\left(C_2\right)\right) \quad (2)$$

where *i* indicates a kind of features and its logical expression is defined as L_i. w(*i*) is an application collaborative function that is applied to *i* (determined by the application professional) to measure the importance of each *i* in evaluating the similarity of C_1 and C_2. Figure 4 illustrates that a series of raw $\text{Data}_{\text{DIKW}}$ points can be converted to $\text{Information}_{\text{DIKW}}$ through conceptual mapping. Relationships between the $\text{Information}_{\text{DIKW}}$ obtained through conceptual mapping are consistent with relationships between original concepts. On $\text{InformationGraph}_{\text{DIKW}}$, there is a simple combination relationship between $\text{Data}_{\text{DIKW}}$ points. The contextual relevance of $\text{Information}_{\text{DIKW}}$ is limited and in different contexts we can establish different classification and combination rules.

In Figure 5 we illustrate the process of mapping $\text{Data}_{\text{DIKW}}$ to $\text{Information}_{\text{DIKW}}$ related to risk factors of system crash and recognize that there are four kinds of raw $\text{Data}_{\text{DIKW}}$ including system type, developing experience, hardware and software configuration and size of memory. Then we have a more complete description of risks that is denoted as D (ST, DE, HSC, MS). ST indicates factors related to system type. DE indicates the developing experience of developers. HSC indicates configuration of hardware and software of a system. MS is the size of memory including 4G, 6G and so on. We identify the raw $\text{Data}_{\text{DIKW}}$ and classify them into concept collections through conceptual mapping. After relational connection, we compute the probability towards different combination of risk factors to obtain $\text{Information}_{\text{DIKW}}$.

Figure 4. Conceptual mapping from $\text{Data}_{\text{DIKW}}$ to $\text{Information}_{\text{DIKW}}$

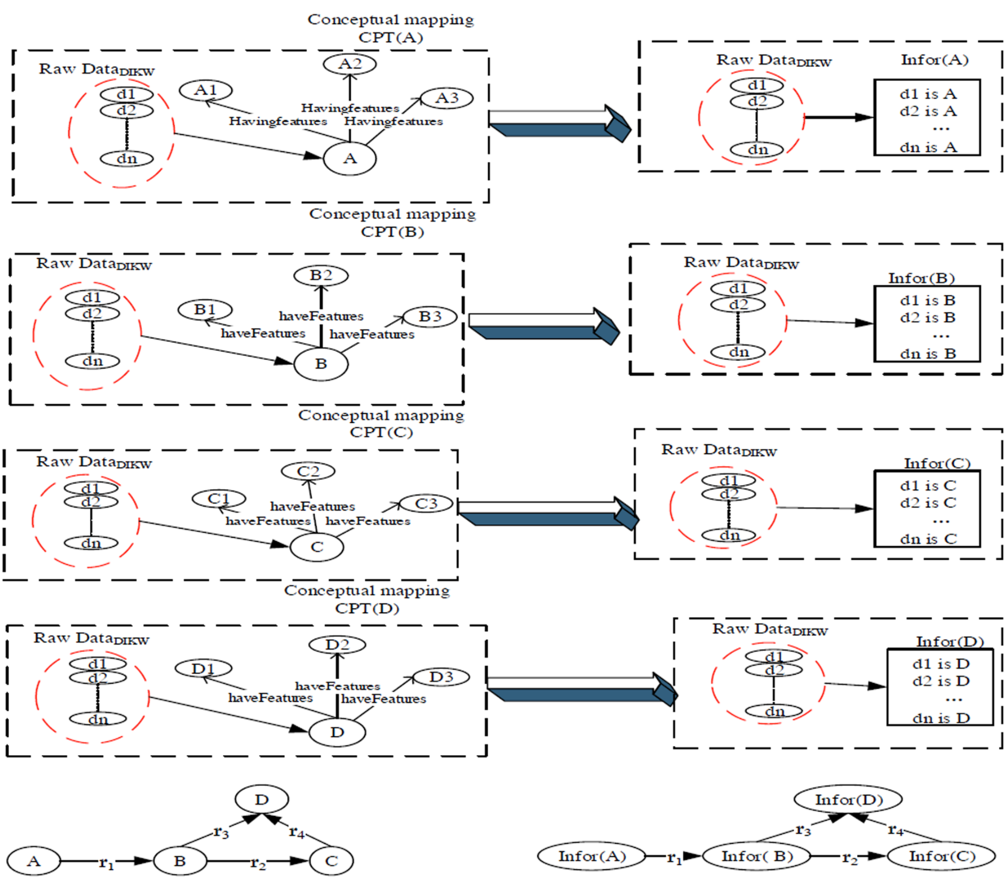
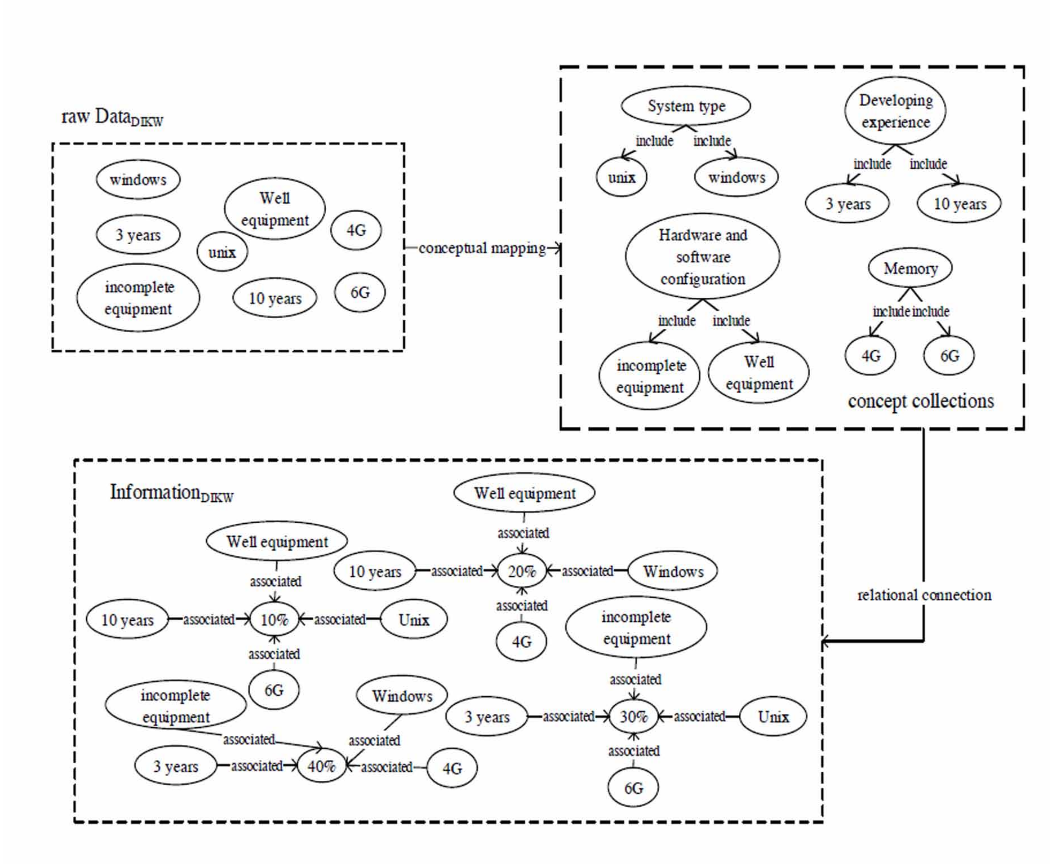


Figure 5. InformationGraph$_{DIKW}$ after relational connection



4. REPRESENTATION OF KNOWLEDGEGRAPH$_{DIKW}$

Knowledge$_{DIKW}$ is Information$_{DIKW}$ that is structured and organized as a result of cognitive processing and validation. Information$_{DIKW}$ is a necessary medium or material for eliciting and constructing Knowledge$_{DIKW}$. Knowledge$_{DIKW}$ being passed on entails encoding Knowledge$_{DIKW}$ into Information$_{DIKW}$ and decoding again into Knowledge$_{DIKW}$. Knowledge$_{DIKW}$ and Information$_{DIKW}$ are not the same, but they have a symbiotic relationship (Lamberti & Wallace, 1990). Knowledge$_{DIKW}$ may be explicit for instance written guidelines and implicit such as people's experience and intuition. Purpose of acquiring Knowledge$_{DIKW}$ is to improve the quality of our life. In the context of risk assessment, purpose of acquiring Knowledge$_{DIKW}$ is to reduce the risk rate or to avoid risks as much as possible for the enterprise and all its stakeholders. Knowledge$_{DIKW}$ representation and reasoning formalism express problems to be solved concerning facts (Chein & Mugnier, 2008). For instance, one may ask with what kind of languages does Mike speaking. Answering such questions requires descriptive Knowledge$_{DIKW}$ but also reasoning capabilities.

4.1. Abstraction on KnowledgeGraph$_{DIKW}$

Data$_{DIKW}$ and Information$_{DIKW}$ are complex, from which we extract valuable Data$_{DIKW}$ and Information$_{DIKW}$ as Knowledge$_{DIKW}$. Thereby we are capable of reducing the available Information$_{DIKW}$ capacity. When stakeholders obtain the description of risk they can screen out valuable Information$_{DIKW}$

and preserve those Information$_{DIKW}$ as Knowledge$_{DIKW}$. As for the example shown in Figure 5, decision maker will choose a program with a lower rate of risk. Page ranking algorithm works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites (Lamberti, Sanna, & Demartini, 2008). We adopt the idea of Page ranking algorithm to filter useless Information$_{DIKW}$ and retain valuable Information$_{DIKW}$. Relevance is measured as the probability that retrieved resource actually contains relationships whose existence is assumed by a user at the time of query requirements. As Figure 6 shows, we give each raw Data$_{DIKW}$ contained in Information$_{DIKW}$ a certain initial weight. Ranks of Information$_{DIKW}$, denoted as $R_{(infor)}$, can be calculated according to (3):

$$R\left(infor\right) = w_{Ai} * w_{Bj} * w_{Ck} * ... * w_{Nn}, i, j, k, n \in \left\{1, 2, ..., n\right\} \tag{3}$$

where A indicates a collection of concepts and Ai indicates the raw Data$_{DIKW}$ elements of concept A. W_{Ai} represents the weight of element Ai. After calculating ranks of Information$_{DIKW}$, we can filter out useless Information$_{DIKW}$ that does not satisfy users' query requirements.

4.2. Transformation on KnowledgeGraph$_{DIKW}$

With Knowledge$_{DIKW}$ stakeholders can make more correct decision. Context of KnowledgeGraph$_{DIKW}$ can be created. KnowledgeGraph$_{DIKW}$ can provide an open Knowledge$_{DIKW}$ access interface and to a certain extent it reflects the real world of inter-entity relationships. The graph structure of KnowledgeGraph$_{DIKW}$ is not restricted by form. KnowledgeGraph$_{DIKW}$ expresses abundant natural semantics and can supplement related Information$_{DIKW}$ among terms. The graph-based nature of KnowledgeGraph$_{DIKW}$ makes possible a linkage to other graphs thus resulting in an easy integrating of multiple kinds of Information$_{DIKW}$ and an enhancement in integrity of Information$_{DIKW}$. As Figure 7 shows, we set an anchor entity to perform entity extraction. Then we find paths between related entities to the anchor entity and compute the weight of edges between them. By exploring KnowledgeGraph$_{DIKW}$, new connections and commonalities between items and users can be discovered and exploited thus the density of nodes and edges of KnowledgeGraph$_{DIKW}$ can be extended. We calculate the rate of correctness of a new relationship obtained through relationship reasoning, denoted as *Cr (E1, R, E2)*, according to (4):

$$Cr(E_1, R, E_2) = \frac{\sum_{\pi \in Q} P(E_1 \to E_2)\theta(\pi)}{\left|Q\right|} \tag{4}$$

where $P(E_1 \to E_2)$ indicates one path between E_1 and E_2, Q indicates all paths starting from E_1 and ending with E_2, $\theta\left(\pi\right)$ represents the weight of a directed path π that can be obtained by training.

5. REPRESENTATION OF WISDOMGRAPH$_{DIKW}$

Wisdom$_{DIKW}$ is an extrapolative process which includes Knowledge$_{DIKW}$ in an ethical and moral framework. Wisdom$_{DIKW}$ is the process by which we discern right from wrong and good from bad. With Wisdom$_{DIKW}$ we can judge from limited to infinity, from known to unknown. Wisdom$_{DIKW}$ is the capacity to decide the most appropriate behavior, taking into account what is known (Knowledge$_{DIKW}$) and what does the most good (ethical and social considerations). Many informed people know what to do, quite a few knowledgeable experts know how to do it, but only a few wise people know and fully

Figure 6. Calculating ranks of Information$_{DIKW}$

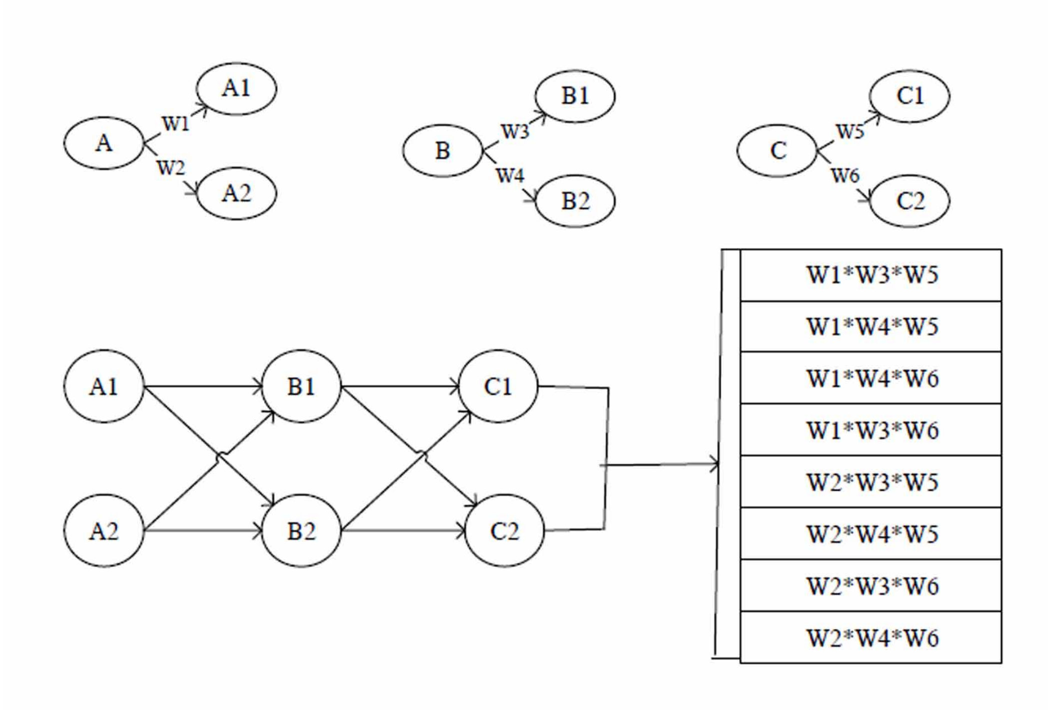
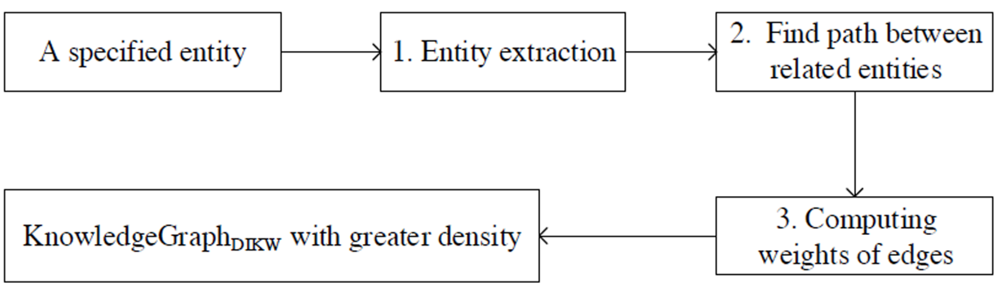


Figure 7. Workflow of extending density of KnowledgeGraph$_{DIKW}$



explicate why something should be done. In line with these ideas the following metaphor applies in Figure 8, Data$_{DIKW}$: "know-who/when/where", Information$_{DIKW}$: "know-what", Knowledge$_{DIKW}$: "know-how" and Wisdom$_{DIKW}$: "know-why". Wisdom$_{DIKW}$ as the ultimate unit of cognition is the result of hierarchical processing of Data$_{DIKW}$, Information$_{DIKW}$, concept, and Knowledge$_{DIKW}$. Knowledge$_{DIKW}$ gives answers to questions guided by how. Wisdom$_{DIKW}$ gives answers to "why" questions. Wisdom$_{DIKW}$ also extends to the application of Knowledge$_{DIKW}$ in action. A simplistic representation of the relationship between Wisdom$_{DIKW}$ and Knowledge$_{DIKW}$ is captured in the following expression: Wisdom$_{DIKW}$ = Knowledge$_{DIKW}$ + Ethics + Action. In Table 2 we take the whole process of risk analysis of software engineering as an example, and list relationships of Data$_{DIKW}$, Information$_{DIKW}$, Knowledge$_{DIKW}$ and Wisdom$_{DIKW}$ in the process of risk analysis. At the initial stage of risk assessment,

Figure 8. Using WisdomGraph$_{DIKW}$ to predict unknown elements

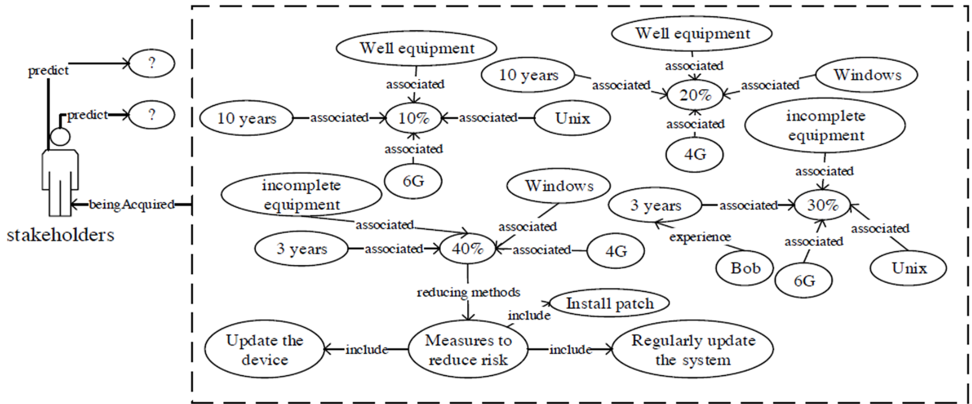


Table 2. Example explanation in Elements$_{DIKW}$ aspects

Aspects	Semantic load	Expression
Data$_{DIKW}$	Input of risk assessment	Array, list link, stack, queue, tree, graph
Information$_{DIKW}$	Risk description	Relational database
Knowledge$_{DIKW}$	Requiring description and make appropriate decision	Nodes and edges with semantic relations
Wisdom$_{DIKW}$	(understand why) the ability to use results of the analysis in a right way	(frame or stylish expression)

we collect various Data$_{DIKW}$ about risk assessment which can be stored in arrays, stacks, lists and other data structures. Based on the Data$_{DIKW}$ we collect, we obtain descriptive Information$_{DIKW}$ about risk through conceptual mapping and relational connection of related Data$_{DIKW}$. And then according to the concept of classification we store Information$_{DIKW}$ in the relational database. Stakeholders can make a favorable decision after gaining this risk description, and ultimately Wisdom$_{DIKW}$ can help stakeholders make future planning and forecasting unknown resources.

6. PROGRESSIVE RELATIONSHIP OF ELEMENTS$_{DIKW}$

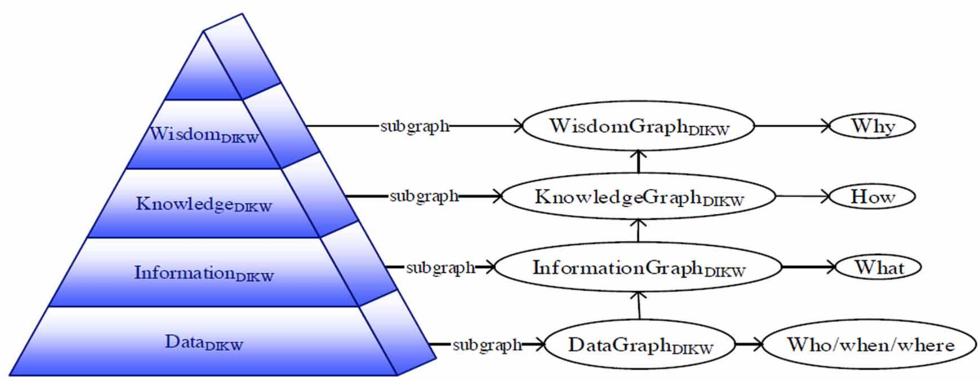
In Figure 9 we show the progressive relationship between Data$_{DIKW}$, Information$_{DIKW}$, Knowledge$_{DIKW}$, and Wisdom$_{DIKW}$ through a pyramid form. Knowledge graph can be used to give answers to the 5Ws problems guided by the interrogative words including "Who /when/ where", "What", "How" and "Why" (Shao, Duan, Sun & Gao, 2017). Norta et al. (2015) proposed to build "Who" model, "Where" model and "What" model towards concepts of agents, the knowledge by agents, and the agents' environment in a business collaboration model. Data$_{DIKW}$ is one of the primary forms of Information$_{DIKW}$. It basically includes recordings of transactions or events that will be used for exchange between human or even with machine. Thus, unless a user understands the context in which Data$_{DIKW}$ is collected, Data$_{DIKW}$ does not make sense. A word, a number or a symbol can be used to describe a business result. It is the context that gives Data$_{DIKW}$ meaning and this meaning makes

$Data_{DIKW}$ informative. $Information_{DIKW}$ extends concept of $Data_{DIKW}$ in a broader context. Therefore, $Information_{DIKW}$ includes combination of $Data_{DIKW}$ but also includes all $Information_{DIKW}$ that a person associates as a member of a social organization in a given physical environment. $Information_{DIKW}$ like $Data_{DIKW}$ is passed by symbol. These symbols have complex structures and rules. $Information_{DIKW}$ has various forms, such as writings, statements, statistics, charts or diagrams. When an individual accepts and retains $Information_{DIKW}$ as a true understanding of reality and an effective explanation of reality, the $Information_{DIKW}$ becomes personal $Knowledge_{DIKW}$. On the contrary, the organization of social $Knowledge_{DIKW}$ exists when it is accepted by the consensus of a group of people. Common $Knowledge_{DIKW}$ does not need to be shared by all members, and in fact it is accepted by a group of insiders that can be considered a sufficient condition. This is also the "public domain" $Knowledge_{DIKW}$ of the real (Syed & Shah, 2008). $Knowledge_{DIKW}$ is a step further on the scale. It involves understanding and ability to make use of $Data_{DIKW}$ and $Information_{DIKW}$ to answer questions, to solve problems, to make decisions and so on. As the human mind uses $Knowledge_{DIKW}$ to choose between alternatives, people's behavior becomes wise. Finally, when values and commitment guide intelligent behavior, behavior may be said to be based on $Wisdom_{DIKW}$. The level of $Wisdom_{DIKW}$ includes all required components such as $Data_{DIKW}$, $Information_{DIKW}$, and $Knowledge_{DIKW}$ to make wise decisions.

7. TYPED RESOURCE MODELLING BASED ON $GRAPH_{DIKW}$ WITH COLLABORATIVE ADAPTATION

We propose a collaborative adaptation approach towards typed resources including $Data_{DIKW}$, $Information_{DIKW}$, $Knowledge_{DIKW}$ and $Wisdom_{DIKW}$ modelling based on $DataGraph_{DIKW}$, $InformaitonGraph_{DIKW}$, $KnowledgeGraph_{DIKW}$ and $WisdomGraph_{DIKW}$ with optimization of temporal complexity and spatial complextity. Temporal complexity is related to the searching efficiency of traversing the resource modelling architecture and spatial complexity is related to the storage efficiency of storing typed resources on $Graph_{DIKW}$. We propose to measure the searching efficiency according to computations about transferring cost of resource types and scale of resources in order to determine which graph should be traversed preferentially. We define typed resources and resources that have been organized on $DataGraph_{DIKW}$, $InformationGraph_{DIKW}$, $KnowledgeGraph_{DIKW}$ and $WisdomGraph_{DIKW}$ as follows:

Figure 9. Relationship among $Elements_{DIKW}$ and corresponding subgraphs



Definition 3. (Typed resources): Typed resources including $Data_{DIKW}$, $Information_{DIKW}$, $Knowledge_{DIKW}$ and $Wisdom_{DIKW}$ are defined as a tuple TR = <TTY, TSC>, where TTY is the type set of typed resources represented by a triad <tty_D, tty_I, tty_K> and TSC is the scale of different kinds of typed resources represented by a triad <tsc_D, tsc_I, tsc_K > where each tsc denotes the scale of resource in the form of tty.

Definition 4. (Resources on $Graph_{DIKW}$): Resources on $Graph_{DIKW}$ are defined as a tuple RoG = <GTY, GSC>, where GTY is the type set of resources on $Graph_{DIKW}$ represented by a triad <gty_D, gty_I, gty_K> and GSC is the scale of different kinds of resources on $Graph_{DIKW}$ represented by a triad <gsc_D, gsc_I, gsc_K > where each gsc denotes the scale of resources in the form of gty.

7.1. Calculation of Resource Type Transferring Cost

We assign values from $Elements_{DIKW}$ to each element of the type set TTY of TR to form the combination case TTY' = {tty_D', tty_I', tty_K'} where tty_D', tty_I' and tty_K' belong to {$Data_{DIKW}$, $Information_{DIK}$w, $Knowledge_{DIKW}$, $Wisdom_{DIKW}$}. The atomic type conversion cost of resources in TR, denoted as *TCost*, is shown in Table 3. Transferring cost of all resources from TR to TTY', denoted as $CostTF_1$, can be calculated according to (5):

$$\mathrm{Cos}\,tTF_1 = \sum_i TCost * tsc_i, i \in \{D, I, K\} \tag{5}$$

We assign values from $Resources_{DIK}$ to each resource of the type set GTY of RoG to form the combination case GTY'= {gty_D', gty_I', gty_K'} where gty_D', gty_I' and gty_K' belong to { $Data_{DIKW}$, $Information_{DIK}$w, $Knowledge_{DIKW}$, $Wisdom_{DIKW}$}. The atomic type conversion cost of elements in RoG, denoted as *GCost*, is shown in Table 4. Transferring cost of all resources from *GTY'* to *RoG*, denoted as $CostTF_2$, can be calculated according to (6):

$$SEC\mathrm{ost} = \sum_i (gsc_i + Cost * gsc_i^{'}) * tsc_i, i \in \{D, I, K\} \tag{6}$$

7.2. Cost of Searching TR in RoG

Different users have different resource requirements. Meanwhile the expansion of resources enable resource searching more inefficient which is called resource overload. In order to solve the problem we propose to adjust storage programs of sharing resources according to computation of searching cost. When resources are needed, users can obtain resources through traversing $DataGraph_{DIKW}$, $InformationGraph_{DIKW}$, $KnowledgeGraph_{DIKW}$ and $Wisdom_{DIKW}$ directly rather than perform related activities to produce the required resources. Table 5 shows the atomic type conversion cost of resource.

Searching cost, denoted as *SECost* can be calculated according to (7):

Table 3. Atomic type conversion cost of resource in TR

	$Data_{DIKW}$	$Information_{DIKW}$	$Knowledge_{DIKW}$	$Wisdom_{DIKW}$
$Data_{DIKW}$	$TCost_{D-D}$	$TCost_{D-I}$	$TCost_{D-K}$	$TCost_{D-W}$
$Information_{DIKW}$	$TCost_{I-D}$	$TCost_{I-I}$	$TCost_{I-K}$	$TCost_{I-W}$
$Knowledge_{DIKW}$	$TCost_{K-D}$	$TCost_{K-I}$	$TCost_{K-K}$	$TCost_{K-W}$
$Wisdom_{DIKW}$	$TCost_{W-D}$	$TCost_{W-I}$	$TCost_{W-K}$	$TCost_{W-W}$

Table 4. Atomic type conversion cost of resource in RoG

	$\mathbf{Data_{DIKW}}$	$\mathbf{Information_{DIKW}}$	$\mathbf{Knowledge_{DIKW}}$	$\mathbf{Wisdom_{DIKW}}$
$Data_{DIKW}$	$GCost_{D\text{-}D}$	$GCost_{D\text{-}I}$	$GCost_{D\text{-}K}$	$GCost_{D\text{-}W}$
$Information_{DIKW}$	$GCost_{I\text{-}D}$	$GCost_{I\text{-}I}$	$GCost_{I\text{-}K}$	$GCost_{I\text{-}W}$
$Knowledge_{DIKW}$	$GCost_{K\text{-}D}$	$GCost_{K\text{-}I}$	$GCost_{K\text{-}K}$	$GCost_{K\text{-}W}$
$Wisdom_{DIKW}$	$GCost_{W\text{-}D}$	$GCost_{W\text{-}I}$	$GCost_{W\text{-}K}$	$GCost_{W\text{-}W}$

Table 5. Atomic type conversion cost of resource

	$\mathbf{Data_{DIKW}}$	$\mathbf{Information_{DIKW}}$	$\mathbf{Knowledge_{DIKW}}$	$\mathbf{Wisdom_{DIKW}}$
$Data_{DIKW}$	$Cost_{D\text{-}D}$	$Cost_{D\text{-}I}$	$Cost_{D\text{-}K}$	$Cost_{D\text{-}W}$
$Information_{DIKW}$	$Cost_{I\text{-}D}$	$Cost_{I\text{-}I}$	$Cost_{I\text{-}K}$	$Cost_{I\text{-}W}$
$Knowledge_{DIKW}$	$Cost_{K\text{-}D}$	$Cost_{K\text{-}I}$	$Cost_{K\text{-}K}$	$Cost_{K\text{-}W}$
$Wisdom_{DIKW}$	$Cost_{W\text{-}D}$	$Cost_{W\text{-}I}$	$Cost_{W\text{-}K}$	$Cost_{W\text{-}W}$

$$SEC\text{ost} = \sum_i (gsc_i + Cost * gsc_i^{'}) * tsc_i, i \in \{D, I, K\} \tag{7}$$

where gsc_i’ indicates the scale of typed resources that are found through traversing $Graph_{DIKW}$ of the different type with initial resources of TR.

7.3. Calculation of Benefit Ratio

The expected investment such as responding time of users that is denoted as $Inve_0$ and the maximum total cost that is denoted as $Total\_cost_0$ are pre-set. After computing $CostTF_1$, $CostTF_2$ *and SECost*, we calculate the total cost of each program, denoted as *Total_Cost*, according to (8):

$$Total\_Cost = CostTF_1 + SECost + CostTF_2 \tag{8}$$

The corresponding investment that is denoted as *Inve* can be computed according to (9) after computing the total cost of each resources searching program:

$$Inve = \gamma * |Total\_Cost_0 - Total\_Cost| \tag{9}$$

where γ represents the required investment of reducing atomic *Total_Cost* that can be obtained through data training. And the ratio of investment and searching cost that is represented by *CostInv* of each program can be calculated according to (10):

$$CostInv = \frac{Inve}{SECost} \tag{10}$$

Then we compare *CostInv* and *Inve* of each program with $CostInv_0$ and $Inve_0$ to determine whether the condition “$CostInv > CostInv_0$ & $Inve < Inve_0$” is satisfied. When *CostInv* is greater than $CostInv_0$, we make $CostInv_0$ equal to the current *CostInv*. If *CostInv* is greater than $CostInv_0$, we perform the next step until the assignment towards TTY and GTY are exhausted. We describe the specific process of investment driven searching sharing resources on resources of $Graph_{DIKW}$ architecture composing $DataGraph_{DIKW}$, $InformationGraph_{DIKW}$, $KnowledgeGraph_{DIKW}$ and $WisdomGraph_{DIKW}$ in Algorithm 1.

Algorithm 1. Calculating *CostInv* of each resource type combination program

Input: TR, RoG, $CostInv_0$.
Output: The maximum *CostInv*.
For each tty **do**
Assign value from $Elements_{DIKW}$;
Compute $CostTF_1$;
For each gty **do**
Assign value from $Elements_{DIKW}$;
Compute $CostTF_2$;
Compute *SECost*;
Compute *Total_Cost*;
If ($CostInv > CostInv_0$ & $Inve < Inve_0$)
$CostInv_0 = CostInv$;

8. RELATED WORKS

Knowledge representation is a critical topic in AI, and currently embedding as a key branch of knowledge representation takes the numerical form of entities and relations to combine the statistical models. However, most embedding methods merely concentrate on the triple fitting and ignore the explicit semantic expression, leading to an uninterpretable representation form (Guu, Miller & Liang, 2015; Minervini, Fanizzi, D'Amato & Esposito, 2016). Traditional embedding methods do not only degrade the performance, but also restrict many potential applications. A semantic representation method for knowledge graph imposes a two-level hierarchical generative process that globally extracts many aspects and then locally assigns a specific category in each aspect for every triple (Xiao, Huang & Zhu, 2016). Because both the aspects and categories are semantics-relevant, the collection of categories in each aspect is treated as the semantic representation of this triple. Sowa (1999) proposed to represent knowledge in logical, philosophical, and computational foundations. As an instance of guidance in the practice of Value Driven Design, a systemic formalization from the value calculation to the design quality measurement binds the modification and change on the design artifacts with the business value strategy through a framework of managed quality properties in a service design process (Duan, Duan, Hu & Sun, 2016).

9. CONCLUSION

Knowledge graph has been widely adopted these years, but the expression of knowledge graph is usually limited to the form of triples. We are increasingly aware of the semantic functions of knowledge graph. We elaborate relationships among $Data_{DIKW}$, $Information_{DIKW}$, $Knowledge_{DIKW}$, and $Wisdom_{DIKW}$, with the aim of clarifying expression and specify the architecture of knowledge graph into four levels that are $DataGraph_{DIKW}$, $InformationGraph_{DIKW}$, $KnowledgeGraph_{DIKW}$, and $WisdomGraph_{DIKW}$. For users with complex resource requirements, they are allowed to express their needs by proposing natural language questions. We use $DataGraph_{DIKW}$ to answer questions guided by "Who / When / Where". $Data_{DIKW}$ is meaningless in the absence of a given context. $Information_{DIKW}$ is a combination of discrete $Data_{DIKW}$ that gives answers to question directed by "What". $Knowledge_{DIKW}$ is an effective combination of abstracted and transformational $Information_{DIKW}$ and capable to answer questions guided by "How". $Wisdom_{DIKW}$ is the ability to criticize or act in a given situation. $Wisdom_{DIKW}$ can provide answers to "Why" questions. Our work lays the foundation for a survey from $Data_{DIKW}$ to $Wisdom_{DIKW}$. In the next stage, we will deal with $Data_{DIKW}$, $Information_{DIKW}$, $Knowledge_{DIKW}$ and $Wisdom_{DIKW}$ under the same context and different context of the 5Ws problem and explore more accurate expression of knowledge graph.

ACKNOWLEDGMENT

This paper is supported by National Natural Science Foundation of China under Grant (No.61363007, No.61662021 and No. 61662019) and Natural Science Foundation of Hainan under Grand No. ZDYF2017128.*refers to corresponding author.

REFERENCES

Bellinger, G., & Castro, D. (2004). Data, information, knowledge, and wisdom. In *Proceedings of the European Congress of Rheumatology Eular* (pp. 276-276). BMJ Publishing Group.

Chein, M., & Mugnier, M. L. (2008). *Graph-based Knowledge Representation: Computational Foundations of Conceptual Graphs*. Springer Publishing Company, Incorporated.

Duan, Y., Duan, Q., Hu, G., & Sun, X. (2016). Refinement from service economics planning to ubiquitous services implementation. In Proceedings of the *IEEE/ACIS International Conference on Computer and Information Science*. IEEE Computer Society.

Duan, Y., Shao, L., Hu, G., Zhou, Z., Zou, Q., & Lin, Z. (2017). Specifying architecture of knowledge graph with data graph, information graph, knowledge graph and wisdom graph. In *Proceedings of the 2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)* (pp. 327-332). IEEE.

Guu, K., Miller, J., & Liang, P. (2015). Traversing knowledge graphs in vector space. *Computer Science*.

Johannessen, T. V., & Fuglseth, A. M. (2014). The effectiveness of data presentation formats: An exploratory study. In *Nokobit 2014, norsk konferanse for organisasjoners bruk av IT*.

Lamberti, D. M., & Wallace, W. A. (1990). Intelligent interface design: An empirical assessment of knowledge presentation in expert systems. *Management Information Systems Quarterly*, *14*(3), 279–311. doi:10.2307/248891

Lamberti, F., Sanna, A., & Demartini, C. (2008). A relation-based page rank algorithm for semantic web search engines. *IEEE Transactions on Knowledge and Data Engineering*, *21*(1), 123–136. doi:10.1109/TKDE.2008.113

Minervini, P., Fanizzi, N., D'Amato, C., & Esposito, F. (2016). Scalable Learning of Entity and Predicate Embeddings for Knowledge Graph Completion. In *Proceedings of the International Conference on Machine Learning and Applications* (pp. 162-167). IEEE.

Nadarajan, G., & Chen-Burger, Y. H. (2006). Mapping Fundamental Business Process Modelling Language to OWL-S. In *Proceedings of the Hellenic Conference on Artificial Intelligence* (pp. 563-566). Springer Berlin Heidelberg. doi:10.1007/11752912_69

Nadarajan, G., & Chenburger, Y. H. (2006). An Ontology-Based Conceptual Mapping Framework for Translating FBPML to the Web Services Ontology. In *Proceedings of the IEEE International Symposium on CLUSTER Computing and the Grid* (Vol. 1, pp. 158-165). IEEE.

Norta, A., Ma, L., Duan, Y., Rull, A., Kõlvart, M., & Taveter, K. (2015). Econtractual choreography-language properties towards cross-organizational business collaboration. *Journal of Internet Services & Applications, 6*(1), 8.1-8.23.

Shao, L., Duan, Y., Sun, X., & Gao, H. (2017). Answering who/when, what, how, why through constructing data graph, information graph, knowledge graph and wisdom graph. In *Proceedings of the International Conference on Software Engineering and Knowledge Engineering* (pp. 1-7). SEKE.

Shao, L., Duan, Y., Sun, X., Zou, Q., Jing, R., & Lin, J. (2017). Bidirectional value driven design between economical planning and technical implementation based on data graph, information graph and knowledge graph. In *Proceedings of the International Conference on Software Engineering Research, Management and Applications* (pp. 339-344). IEEE.

Sowa, J. F. (1999). *Knowledge representation: logical, philosophical and computational foundations*. Brooks/Cole Publishing Co.

Syed, A., & Shah, A. (2008). Data, Information, Knowledge, Wisdom: A Doubly Linked Chain? In *Proceedings of the International Conference on Information & Knowledge Engineering, Ike 2006, Las Vegas, NV* (pp.270-278). DBLP.

Xiao, H., Huang, M., & Zhu, X. (2016). Knowledge semantic representation: a generative model for interpretable knowledge graph embedding.

Zins, C. (2010). Conceptual approaches for defining data, information, and knowledge. *Journal of the Association for Information Science and Technology*, *58*(4), 479–493. doi:10.1002/asi.20508

Yucong Duan is currently a full professor in college of science and information technology, Hainan University, P.R. China. He received a PhD in software engineering from Institute of Software, Chinese Academy of Sciences, China in 2006. He was Post-doc fellow in School of Software, Tsinghua University, China from 2006-2007. He was a Post-doc fellow in the Software Engineering Laboratory at Pohang University of Science and Technology (POSTECH), South Korea, from 2007-2008. He was a lecturer of Biomedical Engineering Institute, Capital University of Medical Sciences, Beijing, China, from 2008 to 2009. He was a Post-doc fellow in Le2i, CNRS, University of Bourgogne, France, from 2009-2010. He was a postdoc in DISCO, University of Milano Bicocca, Milano, Italy, From 2011 to 2012. He has been a member of IEEE since 2005. He joined in the CCF at 2013 and served as a committee member of the Service Computing professional committee of CCF. He has published more than 40 international journal or conference papers. His research interests include: theoretical and empirical software engineering, model driven software development, knowledge engineering, ontology modeling, service science, data cleaning of very large database, service/data quality evaluation, cognitive linguistics, service computing, SOA, cloud computing, etc.

Lixu Shao is a graduate of Hainan University. Her research interests include big data and service computing.

Gongzhu Hu received a BS from Tsinghua University, China, MS from University of Wisconsin-Madison, and a PhD in computer science from Michigan State University. He is currently a professor of the Computer Science Department at Central Michigan University, USA. He was the department chair from 1994 to 2007. He has published over 150 research papers in refereed journals and conference proceedings. His main research interests include data mining and software formal methods. Dr. Hu is a member of ACIS, ACM, senior member of IEEE, life senior member of ISCA, and a member of the Board of Directors of ACIS and ISCA.