International Journal of Database Theory and Application
Vol.8, No.4 (2015), pp.71-80
http://dx.doi.org/10.14257/ijdta.2015.8.4.08

Discovery of Service HyperLinks with User Feedbacks for Situational Data Mashup

Chen Liu[1], Jianwu Wang[2], MeilingZhu[1,3] and Yanbo Han[1]

[1] *Cloud Computing Research Center, North China University of Technology, Beijing, China*
[2] *San Diego Supercomputer Center, University of California, San Diego, U.S.A*
[3] *School of Computer Science and Technology, Tianjin University, Tianjin, China*
liuchen@ncut.edu.cn, jianwu@sdsc.edu, meilingzhu2006@126.com, hanyanbo@ncut.edu.cn

Abstract

Discovery of loose data linkages between data services can help on-demand Web data integration in accordance with situation changes. However, we met the uncertainty challenge when discovering such data linkages with current automatic matchers. To handle the uncertainty problem, this paper develops a synthesized matching algorithm to combine the matching results from multiple automatic matchers with user feedbacks. It also proposes a service hyperlink model to encapsulate such data linkages for further reuse. Experiments show our approach can effectively improve the correctness of discovered data linkages.

Keywords: *data service; mashup; service hyperlinks; uncertainty; schema matching*

1. Introduction

Situational data mashup is a special type of mashup which allows non-professional users to access, process and combine data from various data sources to deal with situational and ad-hoc problems [1]. In recent years, lots of government initiatives and organizations, such as Data.gov[1], Public Data Sets on AWS[2], have started to publish open data for users. To offer a unified abstraction for accessing diverse Web data, “data as service” or “data service” is proposed to provide semantically richer view and advanced querying functionality [2]. With them, data sources can be decoupled from data to be shared. Data service composition has become a promising way to realize data mashup by combining service interfaces [1]. Traditional automatic service composition approaches can be referenced for data service composition [3]. However, current approaches still are not flexible enough to handle situation changes. For example, the investigation of a vicious injury incident is a typical situational data mashup scenario. In this example, police officers need to infer and determine suspects via available clues. However, tracing a clue may lead to the mashup of data from different sources. It means requirements of situational mashup data are continuously evolving along with the situational changes.

Inspired by Web hyperlink, on top of our data service model [4-5], this paper proposes a new abstraction to model linkages between data services, called Service Hyperlink (*SHL*). *SHL*s can be applied in our existing mashup environment, called Mashroom, for intelligent recommendations of follow-up services in a mashup process, which are very helpful to support the situational data mashup. First, it avoids the pre-establishment of a costly mediated schema and brings great flexibility. To respond to situational changes, related data sources can dynamically join or quit a mashup process in the form of data services. With more and more *SHLs* between data services are discovered, a knowledge

[1]Data.gov, http://www.data.gov/
[2] Public Data Sets on AWS,http://aws.amazon.com/publicdatasets/

ISSN: 2005-4270 IJDTA

base can be gradually formed. It can easily evolve along with the situation changes. Second, *SHLs* can also be reused in many repeated scenarios where users can reuse experiences and knowledge left by others when similar situations appear. It can speed up responses to the situation changes.

One main challenge to discover *SHLs* is the uncertainty of automatic semantic matchers [6]. In recent years, although lots of them have been developed, they still cannot assure the correctness of their matching results [7]. Therefore, in this paper, based on our previously developed multiple automatic matchers [5], we use the probability theory to model the uncertain *SHLs* and import user feedbacks as another source to gradually refine them. With our approach, the correctness of discovered *SHLs* can be gradually improved at runtime. The main contribution is our proposed algorithm can effectively combine automatic matching results with multiple user feedbacks. Experiments show that our approach can greatly improve the correctness of discovered service hyperlinks.

2. Discovery of Uncertain Service Hyperlinks

Since we cannot fully depend on automatic matchers to handle the uncertainty challenge, we introduce user feedbacks as effective supplements for automatic matchers. With them, service hyperlinks can be gradually refined and approach the correctness at runtime.

2.1. Definitions of Service Hyperlink

Mashroom[3] is our pre-developed mashup environment [1, 4-5], which aims at offering required agility and expressive power to support on-demand data mashup by end-users. It adopts nested relation model as its unified data model. Relevant definitions such as nested relation, data service can be found in our previous work.

Given two data services ds_1 and ds_2, if input parameters or output schema of a service is fully or partially semantically matched with another service, we say there is a data linkage between them. To be clear, we call an input parameter as well as an attribute or a sub-relation in an output schema by a joint name "**element**". We define a data linkages a set of data mappings among elements from two data services. To specify the data linkage, we first define several types of semantic relationships between elements, which are *equivalence*($\doteq$),*subset-subsumption* ($\Subset$), *superset-subsumption* ($\Supset$), *overlapping* ($\Cap$),*disjointness* ($\perp$), *incompatibility* ($\not\eqsim$). Their definitions can be found in Section 5.1 of our previous paper [5].

With these semantic relationships, a data linkage, which is also called as *Element HyperLink (EHL)* can be definedin Def. 1. Taking the uncertainty challenge into account, a probabilistic model to represent such linkages is established. It is defined as a distribution of probabilities over the set of all possible semantic relationships.Υ represents the set of semantic relationships, *i.e.* $\{\doteq, \Subset, \Supset, \Cap, \perp, \not\eqsim\}$.

Definition 1 (Element HyperLink): Let E_1 and E_2be the two element sets from two data services, e_1 and e_2 be their respective elements. An element hyperlink between e_1 and $e_2$is defined as a triple:

$EHL(e_1, e_2) := < e_1, e_2, P >$, where P is a probability function$P: E_1 \times E_2 \times \Upsilon \rightarrow [0,1]$, such that:$P(e_1, e_2, r) \geq 0$, $r \in \Upsilon$and$\forall(e_1, e_2)\Sigma_{r\in\Upsilon} P(e_1, e_2, r) = 1$.

Based on Def. 1, given two data services, we can define a *SHL* of two data services. We distinguish three types of *EHLs*, which are between input-output, output-input and output-output elements of two data services, respectively. The first two *EHLs* are the data linkages between output schema and input parameters of two data services. It can help users compose and invoke them in sequence. The last one

[3]The trail version of Mashroom environment is at: http://113.11.194.86/DataServiceSpace/index.jsp

is between the output schemas of two data services. It can help users aggregate the outputs of two data services in order to combine the corresponding data.

Definition 2 (Service Hyperlink): Given two data services ds_1 and ds_2, I_1 *and* I_2 are the input parameter set of ds_1 and ds_2 respectively, $O_1$and O_2 are the output schema of $ds_1$and ds_2respectively, a service hyperlink between ds_1 and ds_2 is defined as a 4-tuple: *SHL*(ds_1, ds_2) := <*id*, *oi_ehl*, *io_ehl*, *oo_ehl*>, where: id is a unique identifier for a service hyperlink; *io_ehl*isa set of element hyperlinks between elements in $I_1$and O_2; *oi_ehl*isa set of element hyperlinks between elements in $O_1$and I_2; *oo_ehl* is a set of element hyperlinks between elements in O_1 and O_2.

2.2. Discovery of Service Hyperlink with Multiple Matchers

The discovery of *SHLs* can be divided into automatic phase and user-feedback phase. In the first phase, service hyperlinks can be automatically discovered with automatic matchers. In the second phase, users are allowed to provide their feedbacks when they mashup data services with the discovered *SHLs*. User feedbacks are synthesized with automatic matching results to gradually refine the *SHLs*.

In our previously developed automatic matchers [5], element names and values are the most important features. Around these features, we design two kinds of automatic matchers, which are name-based matchers and structure-based matchers. To be unified, each matcher will take two elements as inputs and a pair (a semantic relationship and a confidence value) as the outputs.

As Def. 1 shows, the key of generating an *EHL* is the design of probability function *P*. We assume each matcher has the same weight on combined match results, and use proportion of each possible semantic relationship in overall confidence value summation to calculate the probability function. Hence, the probability function can be defined as follows.

Definition 3 (Probability Function for Element HyperLink): Given multiple matchers $m_1, m_2, ..., m_n$, let vector R=<$r_1, r_2,..., r_n$>represent the relationships returned by the matchers for element e_1 and e_2, function $conf(m_i, r_j)$ will return the confidence value related to the relation r_i computed by matcher m_i, function $equal(r_i, r_j)$ will return 1 if r_i=r_j, otherwise it returns 0, where$r_i \in \Upsilon,\ i,j = 1..n$; then the probability function P in Def. 1 is defined as follows.

$$p(e_1, e_2, r) = \sum_{i=1}^{n} conf(m_i, r_j) * equal(r_j, r) \Big/ \sum_{i=1}^{n} conf(r_i)$$

Based on Def. 3, we do not try to combine the results from multiple automatic matchers with a math formula to get a synthesized result to weigh the similarity between the elements. It is because although combination of the outcomes from multiple automatic matchers can be utilized in one matching process to get better results, such combination is also uncertain [8]. It also cannot assure the correctness of combination results. Hence, we record the matching result as a distribution of probabilities over the set of all possible semantic relationshipsΥ.

2.3. Improvements with User Feedbacks

In the second phase, we let users participate in and gradually improve the correctness of discovered service hyperlinks by automatic matchers. To do this, we regard a user as an “artificial matcher” with her own knowledge and experiences. Hence, we define a user feedback as a new matching result for two elements.

Definition 4 (User Feedback):Let ds_1 and ds_2 be two data services, e_1 and e_2 be two elements from their input parameters or output schema respectively, Υbe the all possible semantic relationships, a user feedback is defined as: *feedback*(e_1,

e_2):=<rel, conf>, where*rel*is the relationship between $e_1$and $e_2$assigned by a user,$rel \in \Upsilon$;*conf*is the confidence of the user about the correctness of the relationship.

Users can browse the discovered service hyperlinks by automatic matchers. They are allowed to alter them in their own mashups. For a given *EHL*, a user can designate a new semantic relationship. She is also required to provide her confidence about the correctness of this relationship. It will lead to the creation of a new *EHL*.

Note that a user also cannot guarantee the correctness of her feedback. For a given element pair, there might be ambiguities among multiple users. Hence, it is necessary to find an effective way to combine different user feedbacks based on the automatic matching results. We hope that the discovered *EHLs* and *SHLs* can gradually approach the correctness when more and more user feedbacks are combined.

Considering ambiguities among different users, we also depend on the probability model to define the result of combining a given *EHL* and a user feedback. Complying with Def. 1, the combination result is also a distribution of probabilities over the set of all possible semantic relationships. However, for a given element pair and a possible semantic relationship *r*, if more user feedbacks agree with *r*, then the probability that *r* is correct should be higher. Hence, we depend on the following formula in Def. 5 to realize the combination.

Definition 5 (Combination with A User Feedback): Let ds_1 and ds_2 be two data services, e_1 and e_2 be two elements from their input parameters or output schema respectively, the original element hyperlink between e_1 and e_2 is *EHL*(e_1, e_2, *P*), a user feedback about e_1 and e_2 *is f*(e_1, e_2)*:=<rel, c>*, then after combining the feedback *f*, the updated element hyperlink between e_1 and e_2 is: *EHL'*(e_1, e_2, *P'*), *where:*

$$P'(e_1, e_2, r) = \begin{cases} (P(e_1, e_2, r) + c)/(1 + c)\ (r = rel) \\ P(e_1, e_2, r)/(1 + c)\ (r \neq rel) \end{cases}$$

Next, we will analyze the properties of the formula in Def.5. We need to prove this formula can satisfy our requirements. First, as the value range of$P(e_1, e_2, r)$is [0, 1], hence we can easily know the value range of $P'(e_1, e_2, r)$ is also [0, 1]. When $r = rel$and $P(e_1, e_2, r) = 1$$P'(e_1, e_2, r)$ also equals 1. When $r \neq rel$ and $P(e_1, e_2, r) = 0$, $P'(e_1, e_2, r)$ also equals 0.

Second, we can prove the formula in Def. 5 is still a probability distribution function. It means, whatever user feedbacks are combined, the sum of probabilities distributed over the possible semantic relationships is equal to 1.

Property 1: Let ds_1 and ds_2 be two data services, e_1 and e_2 be two elements from their input parameters or output schema respectively, the current *EHL*(e_1, e_2)*:=* < e_1, e_2, *P*>, let *R*={r_1, r_2, ..., r_n} be the already discovered relationships, where: $P(e_1,\ e_2,\ r_i) \geq 0 \wedge \sum_{r_i \in \Upsilon} P(e_1,\ e_2,\ r_i) = 1\, r_i \in R$. After combining a user feedback *feedback*(e_1, e_2)*:=<rel, c>*, $\sum_{r_i \in \Upsilon} P'(e_1,\ e_2,\ r_i)$ still equals 1.

Third, we can find that the value of $P'(e_1, e_2, r)$ is proportional to the number of users who agree with the relationship *r* and their confidences. It is consistent with the actual situations. If more users agree with a relationship and the higher confidence they have, then the possibility of this relationship is correct is higher.

Based on Def. 3 and 5, we design two algorithms to discover and refine service hyperlinks, respectively. Algorithm 1 implements the automatic part and Algorithm 2 implements the user feedback part of our approach.

Algorithm 1. Automatic Discovery of Element HyperLinks

Function: *ADSEL*
Input: two element sets S_1 and S_2 from two data services respectively;
Output: a set of element hyperlinks;

1. set *outputSet* = ∅
2. initialize four automatic matchers $M=\{m_1,\ldots, m_4\}$ defined in Table I;
3. for each element e_1 *in* S_1
4. for each element e_2 *in* S_2
5. for each matcher m_i in M
6. $<r_i, c_i> \leftarrow m_i\ (e_1, e_2)$
7. end for
8. combine each $<r_i, c_i>$ into a probabilistic distribution P in Def. 5;
9. initialize a new element hyperlink $ehl(e_1, e_2, P)$ and put it into the *outputSet;*
10. endfor
11. endfor
12. return *outputSet;*

Algorithm 2. User Feedback Combination Algorithm

Function: *UFCA*
Input: an element hyperlink: $ehl(e_1, e_2, P)$
a user feedback: $fd(e_1, e_2, rel, conf)$
Output: a updated element hyperlink: $ehl'(e_1, e_2, P')$

1. set R ← all discovered relationships for *ehl*
2. Initialize a new probabilistic distribution *P'*
3. for each *r* in *R*
4. //combine user feedback with formula in Definition 7
5. If (r == rel)
6. P'(r) = (conf+P(r)) / (conf+1)
7. else
8. P'(r) = P(r) / (conf+1)
9. endif
10. endfor
11. if (rel∉ R)
12. P'(rel) = conf / (conf+1)
13. endif
14. initialize a new element hyperlink $ehl' \leftarrow (e_1, e_2, P')$
15. return *ehl';*

3. Evaluation

In this section, we conduct a systematic empirical evaluation addressing two questions: How do user feedbacks affect the correctness of the matching results? What effects will be led to when multiple user feedbacks are proposed?

3.1. Experiment Data and Process

To be objective, we choose experiment data from three different sources. We list six groups of data sets to be used in the experiments. These data can be divided into three categories shown in Table 1.

Table 1. Experiment Data Sets

Category	No.	Topic	Description	Scale (Element Size)
I. Extracted from Web	1	news	extracts the part of channel structures from the two news website	29*28
	2	paper	extracts a given paper information from the two websites	8*13
II. I^{3}CON Conference	3	animal	provides heterogeneous descriptions about animals	36*24
	4	people+pets	provides heterogeneous descriptions about people and their animals	96*93
III. OAEI Contest	5	bibtex	composes of a reference bibliographic ontology and a set of real bibtex ontology	165*56
	6	conference	composes of a set of conference ontologies, which are extracted from real conference WebPages	89*78

Based on the above schemas, we design and develop corresponding data services. The above schemas are transformed into the nested relations and become the output schema of developed data services. In the experiment, we try to discover element hyperlinks between output schemas of two data services in each data set. Note that an element hyperlink is actually a probability distribution over possible semantic relationships. Hence, when integrating two elements, we choose the relationship with the highest probability as the final discovered semantic mappings. Furthermore, we think if the final discovered semantic mappings have high correctness, then it means the discovered *SHLs* also have high correctness. Hence, we use precision and recall index of the matching results to evaluate the correctness of final semantic mappings, which also can objectively reflect the correctness of discovered *SHLs* with algorithms I and II.

Definition6: Let A be the all semantic mappings between two data schema, and B be the semantic mappings computed by a given matching algorithm, then:

$$recall = \frac{|A \cap B|}{|A|} precision = \frac{|A \cap B|}{|B|}$$

For each dataset, we invite several graduate students who do not know the data sets beforehand. First, we only compute the values of precision and recall with automatic matchers. Second, only one user provides her feedbacks. Third, we invite two different users to provide feedbacks. They may have different conclusions about the matching result of two elements. Fourth, we invite five users to provide their feedbacks. In this step, the chances to have conflictive feedbacks enlarge. We observe the impacts for correctness when more users are involved. The experiment results can be found in Table 2-3.

3.2. Experiment Results and Analyses

From the results of our experiments, we can see user feedbacks clearly increase the correctness of the matching results. As Table 2 and 3show, with more human feedbacks, both recall indexes and precision indexes of matching results clearly increase for most cases. On average, the precision and recall indexes have the maximal improvements of 20.8% and 19.3%, respectively.

For the data sets, the improvement rates of data set I are the largest where the precision and recall indexes have the maximal improvements of 44.4% and 43.3%, respectively. This is because these data sets are extracted from real web sites. Different from other data sets, they are very lack of normalization. For example, in data set I, websites sometimes use Chinese pinyin to represent a term, such as use “youxi” to represent “game”. In these cases, automatic matchers are incapable of identifying and matching these terms.

Besides, an unexpected outcome is the precision index of data set IV-VI where values decrease with combination of one user feedback. Our analysis shows the main reason is that some users do not provide big enough confidence values for their feedbacks. Lower confidence value may not turn around the wrong mappings from automatic matchers. However, with more user feedbacks are involved, this situation can be greatly improved.

Table 2. Precision Index Values for Different Users

users / Data Set	0	1	2	5
I	0.556	1	1	1
II	0.5	0.625	0.875	0.875
III	0.791	0.947	0.92	0.96
IV	0.881	0.862	0.875	0.875
V	0.797	0.762	0.797	0.788
VI	0.583	0.533	0.611	0.857
AVG	0.685	0.788	0.846	0.893

Table 3. Recall Index Values for Different Users

Users / Data Set	0	1	2	5
I	0.4	0.667	0.667	0.833
II	0.667	0.5	0.7	0.7
III	0.904	0.75	1	0.958
IV	0.967	0.946	0.978	0.963
V	0.77	0.814	0.864	0.881
VI	0.467	0.667	0.916	1
AVG	0.696	0.724	0.854	0.889

We also find another unexpected outcome is the recall index of data set IV. Instead of keeping improvements, it decreases when more user feedbacks are provided. Our analysis shows the main reason is that some users may be confused with the same issue. For example, many users cannot assure whether “type” and “hasType” as well as other similar element pairs should be equivalent in this data set. Many of them think these two elements should not be equivalent as they are not the same part of speech. However, in the provided correct answers, they are regarded as the equivalent. Hence, the value of recall index lowers down when more user feedbacks don’t agree with such result. This sample reminds us if the consensus among multiple users is not correct, then our proposed approach also may get the wrong conclusions.

4. Related Work

Many automatic schema matching and data transformation techniques have been developed. Clip is an XML schema mapping tool distinguished from existing tools in that mappings explicitly specify structural transformations in addition to value couplings [9]. TranSheet proposes an approach to transform spreadsheet data to structured formats required by applications and services [10].

Recently, based on these matchers, researchers have started to propose various approaches to address the uncertain matching problem. Work in [6] models the uncertain schema mappings as possible one-one mappings among database schemas, and uses probability theory as the underlying theoretical model. Rule-based probabilistic relationship is used in [11], which employs a probabilistic extension of datalog to encode uncertain relationships between schema objects.

Similarly, there also have been a few studies on service links. Work in [12] proposes a service data link model, which is a service relationship among schema. It can describe service data correlations, which are data mappings among the input and output attributes of services. Work in [13] propose a HyperService approach to provide a much more flexible way to link and explore existing services for solving various situational problems. With the HyperService approach, a group of relevant services are dynamically searched, ranked and recommended for facilitating future navigations.

5. Conclusion

To handle the uncertainty challenge, this paper proposes a user-feedback oriented approach to discover uncertain data linkages among data services. It tries to synthesize the automatic matching results and feedbacks from multiple users to gradually approach the correctness of discovered uncertain service hyperlinks. To do this, we also regard a user as an “artificial matcher” and design the probabilistic distribution function to model the uncertainty of matching results. We also design an experiment and validate the effectiveness of our approaches. The experiments show user feedbacks can be very helpful to improve the correctness of automatically discovered service hyperlinks. This approach is also applied in our existing Mashroom environment for intelligent recommendation of follow-up services in a mashup process.

In the future, we will mainly improve our works from the following aspects. First, we plan to improve the correctness of each single matcher or import more automatic matchers using different matching techniques. Next, we will also take the difference of users into consideration and answer the question what our approach should do when users provide wrong feedbacks.

Acknowledgements

The research work is partially supported by the following projects: Scientific Research Common Program of Beijing Municipal Commission of Education (Research on a User-oriented Approach to Visualized Aggregation of Data Services, No. KM201310009003); The Project of Construction of Innovative Teams and Teacher Career Development for Universities and Colleges Under Beijing Municipality (No. IDHT20130502).

References

[1] Y. Han, G. Wang, P. Jiand and J. Zhang, “Situational Data Integration with Data Services and Nested Table”, Service Oriented Computing and Applications, vol.7, no.2, **(2013)**.

[2] J.C. Michael, O. Nicola and P. Michalis, “Data Services.Commun.ACM”, vol.55, no.6, **(2012)**.

[3] S. Yang, M. Shang-Pin, K. Jong-Yin and F. Yong-Yi, “A Survey on Automated Service Composition Methods and Related Techniques”, Proceedings ofthe 9th International Conference on Services Computing, **(2012)**; Honolulu, HI.

[4] G. Wang, S. Yang and Y. Han, “Mashroom: End-user Mashup Programming Using Nested Tables”, Proceedings of the 18th international conference on World Wide Web, ACM, **(2009)**; Madrid, Spain.

[5] C. Liu, J. Wang and Y. Han, “Mashroom+: An Interactive Data Mashup Approach with Uncertainty Handling”, J.Grid Comput, vol.12, no.2, **(2014)**.

[6] D. X. Luna, H. Alon and Y. Cong, “Data Integration with Uncertainty”, The VLDB Journal, vol.18, no.2, **(2009)**.

[7] P. Shvaiko and E. Jerome, “Ontology Matching: State of the Art and Future Challenges”, IEEE Trans. on Knowl.and Data Eng., vol.25, no.1, **(2013)**.

[8] A. Gal, “Uncertain Schema Matching”, Synthesis Lectures on Data Management, vol.18, no.2, **(2011)**.

[9] R. Alessandro, B. Daniele, C. Stefano, P. Paolo and A. H. Mauricio, “Clip: a Visual Language for Explicit Schema Mappings”, Proceedings of the 24th International Conference on Data Engineering, IEEE Computer Society,**(2008)**; Cancún, México.

[10] H. Vu, B. Boualem and S.-P.Regis, “Spreadsheet-based complex data transformation”, Proceedings of the 20th ACM International Conference on Information and Knowledge Management, ACM, **(2011)**; Glasgow, Scotland, UK.

[11] N. Henrik and S. Umberto, “sPLMap: a Probabilistic Approach to Schema Matching”, Proceedings of the 27th European Conference on Advances in Information Retrieval Research, Springer-Verlag, **(2005)**; Santiago de Compostela, Spain.

[12] G. Zhifeng, X. Bin and L. JuanZi, “Service Data Correlation Modeling and Its Application in Data-Driven Service Composition”, IEEE Trans. Serv. Comput. vol.3, no.4, **(2010)**.

[13] Z. Chenting, M.Chun'e, Z. Jing, Z. Jun, Y. Li and M. Xinsheng, “HyperService: Linking and Exploring Services on the Web”, Proceedings of the20th International Conference on Web Services, **(2010)**; Santa Clara Marriott, CA, USA.

Authors

Chen Liu, he is an Associate Professor at Research Center for Cloud Computing, North China University of Technology. He has authored more than 20 peer-reviewed research papers in journals, conferences, and workshops. He has lead or participated in several research projects, including the National Natural Science Foundation of China, the National R&D Infrastructure and Facility Development Program of China, National Basic Research Program of China and *etc.* His current research interests include data integration, service modeling, service composition, cloud computing and so on.

Jianwu Wang, he is the Assistant Director for Research at Workflows for Data Science (WorDS) Center of Excellence at San Diego Supercomputer Center (SDSC), University of California, San Diego (UCSD), U.S., and an Assistant Project Scientist at SDSC, UCSD. He is also an Adjunct Professor at North China University of Technology, China. His research interests include Service-Oriented Computing, End-User Programming, Scientific Workflow, Distributed Computing, Data-Intensive Computing. He has published over 40 papers with more than 300 citations. He is associate editor or editorial board member of four international journals, co-chair of three related workshops. He is also program committee member for over 20 conferences/workshops, and reviewer of over ten journals or books.

Meiling Zhu, she is a doctoral student at School of Computer Science and Technology in Tianjin University. Her current research interests include data integration, service modeling, service composition, cloud computing and so on.

Yanbo Han, he is a Full Professor and the Director at Research Center for Cloud Computing, North China University of Technology. He holds a Ph.D. from Technical University of Berlin. His current research interests include Internet Computing, Services Interoperability and Composition, Dependable Distributed systems, Business Process Collaboration and Management. He has authored or coauthored over 140 papers and four books. Dr. Han has organized over 20 academic events as general chairs or program chairs including eight journal special issues.

