Volume 6, Issue 9, September 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Development of Service Oriented Situational Web Applications Using Knowledge Discovery

Dr. P. Radhika Raju, Prof. A. Ananda Rao, B. Swarna Jyothi
CSE Department & JNTUACEA, Anantapur,
Andhra Pradesh, India

Abstract— A service can be defined as "an operation that may be invoked independently". A cluster of web services may represent an organization's SOA. The combination of the web application and services computing provides a wide range of scope for the "situational" web application. Dynamic composition of various similar and reusable services results Situational Web Application (SWA). Situational application is software created for a small group of users with a distinctive set of needs. SWA using knowledge discovery from services optimizes the search time and improves the true positives. Available service oriented situational web applications are not appropriate for providing on-the-fly recommendations for classic business enterprises. The down side with available SWAs is inconsistency during searching (true positive rate). An attempt with combined work of situational web applications and knowledge discovery at runtime is one approach for extracting useful information from various existing available web services and for providing viable candidate services. In this paper a prototype application is developed to demonstrate the concept of Knowledge Discovery from Services (KDS) at the runtime of situational web applications. Knowledge discovery from web services is one algorithm which is developed for implementation of SWA using KDS approach. The true positives rate can be increased with respect to optimized searching time compared with available situational web applications.

Keywords— Web Applications, Situational Web Applications, Mashups, KDS, Tagging.

I. INTRODUCTION

With the invent of technologies web 2.0 and web services it is made possible to build web applications that provide rich user experience besides having capabilities to collaborate with other applications. Web 2.0 provides "web as platform" framework. The corporate intranet is evolving into a platform of readily accessible data and services where communities of business users can assemble and deploy Situational applications according to their needs. "Diamia" allows business users to quickly and easily create data mashups that combine data from web and traditional IT resources into feeds that can be consumed by web applications. Increasingly large numbers of situational applications are being created by enterprise business users as a by-product of solving day-to-day problems. Composing available Web delivered services into a single web application is called as "service mashups". Knowledge discovery in Situational Web Applications is a systematic approach there is a customized development life cycle that the software engineers can use to create new applications based mashup development.

The travel planning in Milan City is the existing example of Situational web application. It uses the Tag-based approach and it gives the on-the-fly recommendations to the users for satisfying the user requests. While changing the user situated tags the generated service candidates are not appropriate i.e., increase in false positive rates. With the mean of data-driven approach in situational web applications the searching time is not optimized and the viable candidate service identification is relatively difficult from the repositories.

In this paper we built a technique to develop situational we applications. The technique is based on the concept of development of SWA using knowledge discovery. The application can discover knowledge on various services being rendered and their relevance to the search made. The relevant services are automatically discovered and they are integrated with the proposed application to form a good workflow for interpretation of comprehensive service stack to end users based on their preferences. It also integrated with Google maps in order to provide a graphical view of the search results so as to help end user to have intuitive response.

The remainder sections are organized as follows, section II gives a review about literature survey, section III explains the approach over view and the concept in detail and section IV presents experimental results while section V concludes the paper with future work.

II. LITERATURE SURVEY

Situational web applications are the applications that provide runtime integration with different services as explored in [1, 2, 3]. Service Mashups is an example for a typical situational application which became popular [4, 5]. There is ongoing research and efforts in order to simplify the development of such applications besides developing related tools. Service mashups are used to integrate data that comes from multiple sources in general. There is another approach known

as flow based composition of such applications. A light weight work flow for the same is employed in [5] for making a model which is situational application in generic way.

High level view of business and domain specific language was used in [6] for having application flow for situational context. In this paper our approach is based on the data-driven and tag based which gets flow of different applications and integrated. This kind of work was first proposed in MARRIO system [7]. In the research circles, tag based search is very important and hot topic as of now. It is also there in the data mining and information retrieval domains. Most of the work in the research circles is based on social networking sites like 'flickr' and 'twitter' the tag based approaches are follows. There are some works that are service-oriented as explored in [7, 8, 9]. In this paper the concept makes use of semantic way which for discovering useful tags and then service oriented situational applications can be obtained.

Web services and their features are used in order to have semantic search that can help in finding relevant services and make a work flow. This is achieved by using similarity- based measurement as explored in [10]. By using tags and clustering approach besides semantic knowledge, it was possible to integrate. Lexical dictionary like wordnet is also used in order to achieve the semantic part of the work. More over techniques like SHOP2[11] and AgFlow [12, 13] are also used for achieving satisfying results. In this paper, the semantic and tag based mode which is simple and effective to integrate applications to make them situational applications. Though the tag-based approach isles expensive it is used in our approach in order to design tag-based work flow. Here the customize compositions are used for making a platform for developers in order to design tag-based work flow. Here the customized compositions are used for making a platform for developers in order to help them to build situational web service.

III. ARCHITECTURE OF SWA USING KDS

The concept of this paper is development of situational web application which is based on knowledge discovery. The proposed application discovers the knowledge from deployed web services which are running in distributed environments [15]. The application discovers the relevant tags [14] and passes down them after integrating with situational web application. The architecture of the application is shown in figure.

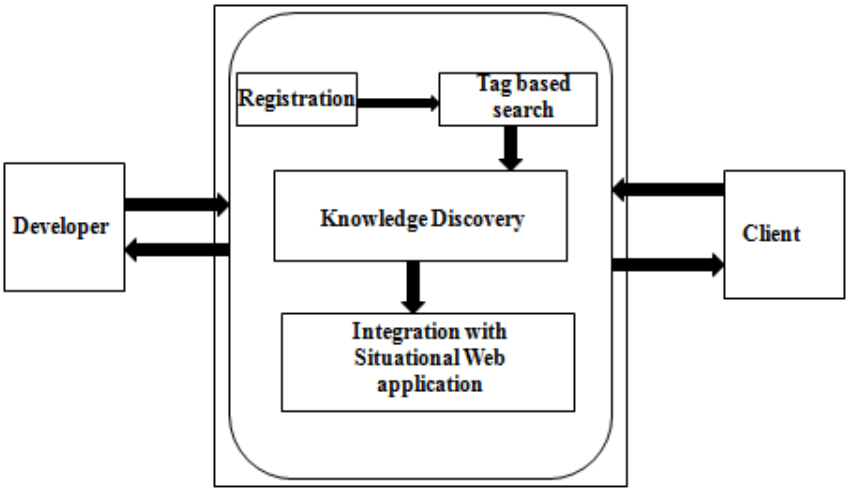

Figure 1 – Architecture of a KDS SWA

As shown in figure the end users can register themselves and they can perform search operation. The searching process is done by using "Tags". They are a set of key words to describe some aspects of a service. Tag extractions are followed by "Textual descriptions of services and Collecting annotations from user experiences about a particular service.(comments, feed backs, user queries) [16]. Useful tags are reside in 'Service name' - the general information about service and the 'service interfaces' and the 'operations, input/output messages'. And these tags are stored in a database.

The tag based search is more effective when combined with knowledge discovery. This concept involves an algorithm known as Knowledge Discovery from Web Services (KDWS). The algorithm is as given below

Algorithm – Knowledge Discovery from Web Services

```
01. Initialize Services Vector SV
02. Initialize Tags Vector TV
03. Initialize Knowledge Vector KV
04. Obtain tags from Tags Data Base
05. Populate TV with obtained tags
06. For each tag in TV
07. Discover services using UDDI API
08. Populate or update SV
09. END For
10. For each Service in SV
```


11. Discover knowledge ' k ' from Service 's'
12. IF' k' is compatible and configurable THEN
13. Integrate 's' with situational web application
14. END IF
15. ELSE
16. Discard service
17. END IF
18. END For
19. Render Situational Web application

The algorithm explains how the process is going to engage KDS with SWA. It initializes the vectors SV, TV, KV. When the user starts searching using tags, with the advantage of web 2.0 tagging users are facilitated with related tags from processed data base. For each tag in TV discover the services using UDDI API by updating the service vector. For each service in SV discover the knowledge 'k' using service names and inputs and outputs of a particular service. We have many discovery trends across the open web service.

Multiple key searching is the one searching strategy which helps the users. By using this strategy user can get the relatively similar and more services for their requirements. The services are going to cluster for multiple key search in a way that, if services are have the same input or same output but the service name is different then that services are grouped into one cluster in similar way by using COP algorithm the services are going to cluster.

When the knowledge is compatible and configurable, that service is integrated with web applications based on the user situated demand. Thus the situational web applications are able to cater to the needs of end users with as many as possible services.

IV. EXPERIMENTAL RESULTS

The results are presented here by using the developed application. The application is build using JAVA platform. The application is purely web based and it demonstrates the concept of Situational Web Applications that make use of Knowledge discovery at runtime based on the needs of end users. By using the Tag based search approach the performance of application is evaluated in terms of time taken to integrate the discovered services with Situational Web Applications using knowledge discovery at runtime. Sometimes the services searched by the user may not be related ones. Generally those searches are called as "False positives". The performance is tabled as given below.

Table1 – Tag based search results

Tag(Id)	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Search Time(ns)	34	86	106	86	42	45	53	59	51	46	45	40	45	51

The above table contains the time taken for different tags in searching services are presented. The performance analysis for the tabulated values is as follows.

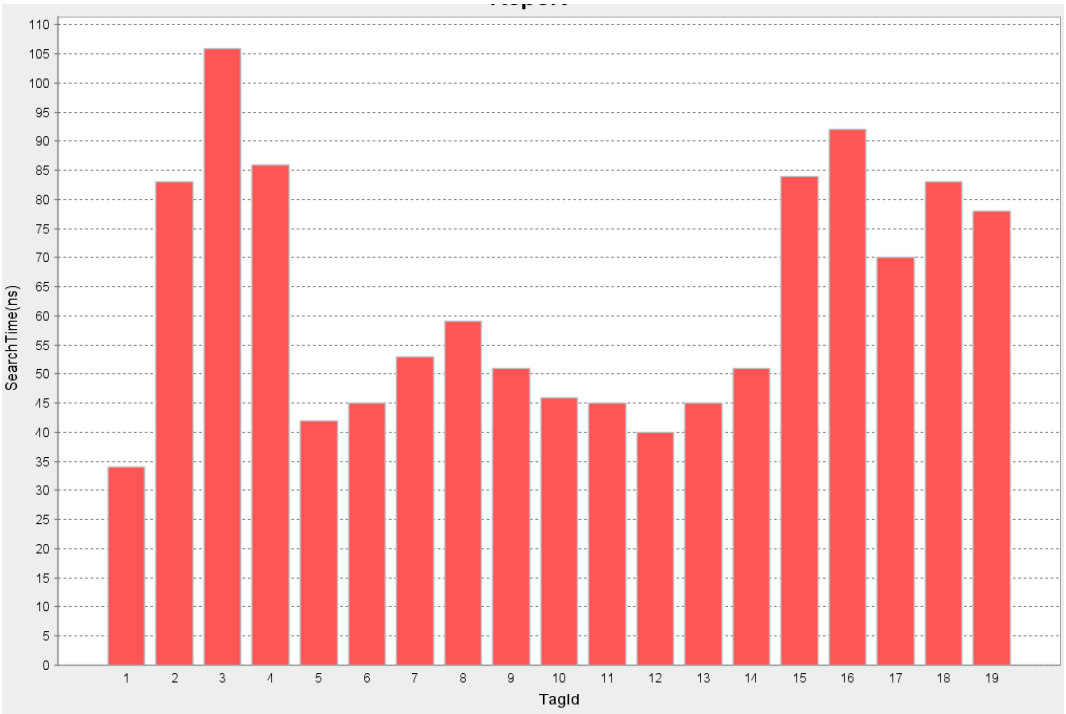

Figure 2 - Searching Analysis

The graph shows the performance analysis for services searching time using tags with runtime KDS. The horizontal axis represents tag id's for searching the services, and the vertical axis represents Searching time for particular tag search. The development of KDS with SWA is improved in terms of true positives and false positives. The table below shows the true positives and false positive comparisons with some real world example application and with the concept of KDS.

Table 2 – Results of measures as true positives and false positives

	Existing	**Proposed**
True Positives	75	85
False Positives	25	15

And the performance analysis for the true positive and false positive of current approach with previous approaches. The graphical representation of performance analysis is as shown below. True positive refers the search operation for discovering most relevant services and integrated with the situational web applications.

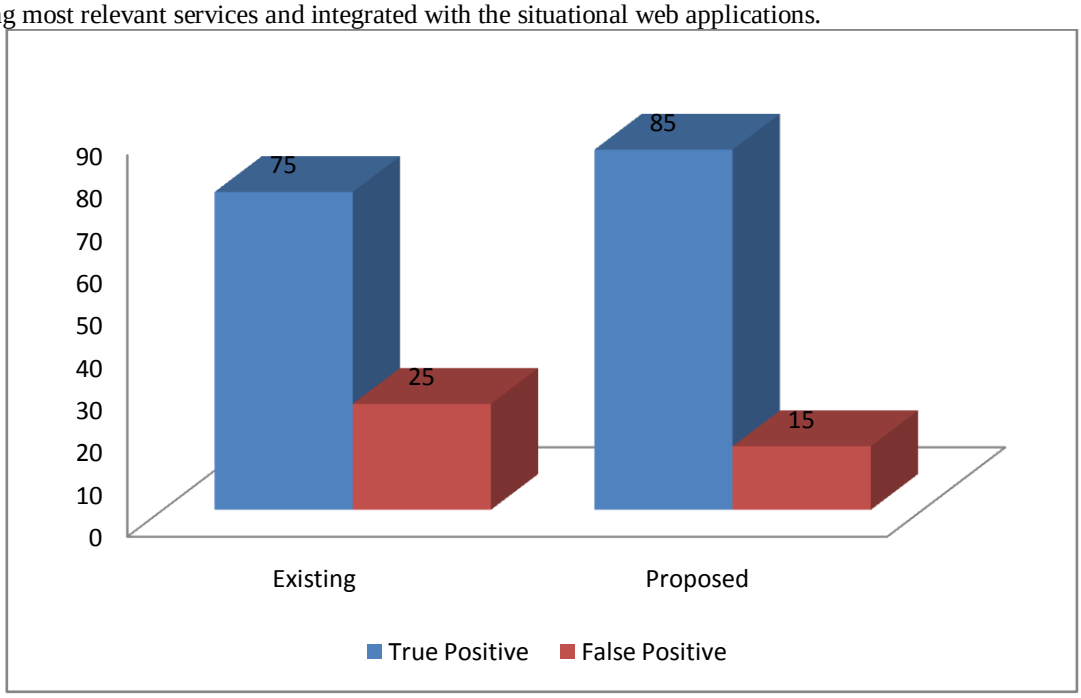

Figure 3 – Comparison graph for true and false positives

V. CONCLUSIONS AND FUTURE WORK

The paper discusses with service oriented situational web applications that are integrated with knowledge discovery concept. And the service composition is made based on the 'on the fly' when request is made from the user. This is possible with the availability of web services that are available in the real world. The dynamic discovery of services and integration of the same with the knowledge discovery can provide complete set of related services to end users. This leads to a way for taking reusability of services to the next level. This research can be extended further by exploring domain specific services and rendering them in the form of situational web applications.

REFERENCES

[1] S. Balasubramaniam, G. Lewis, and S. Simanta, "Situated Software:Concepts, Motivation, Technology, and the Future," IEEESoftware, vol. 25, no. 6, pp. 50-55, Nov./Dec. 2008.

[2] S. Adams, "The Future of End User Programming," Proc. 30th ACM Int'l Conf. Software Eng., 2008.

[3] B.W. Boehm, "A View of 20th and 21st Century Software Engineering," Proc. 28th ACM Int'l Conf. Software Eng. (ICSE '06), pp. 12-29, 2006.

[4] V. Hoyer and M. Fischer, "Market Overview of Enterprise Mashup Tools," Proc. Sixth Int'l Conf. Services Oriented Computing, pp. 708-721, 2008.

[5] F. Curbera, M. Duftler, R. Khalaf, and D. Lovell, "Bite: Workflow Composition for the Web," Proc. Int'l Conf. Service-Oriented Computing, pp. 94-106, 2007.

[6] E. Maximilien, H. Wilkinson, N. Desai, and S. Tai, "A Domain- Specific Language for Web Apis and Services Mashups," Proc. Int'l Conf. Service-Oriented Computing, pp. 13-26, 2007.

[7] E. Bouillet, M. Feblowitz, Z. Liu, A. Ranganathan, and A. Riabov,"A Tag-Based Approach for the Design and Composition of InformationProcessing Applications," ACM SIGPLAN Notices, vol. 43,no. 10, pp. 585-602, Sept. 2008.

[8] J.-R. Falleri, Z. Azmeh, M. Huchard, and C. Tibermacine, "Automatic Tag Identification in Web Service Descriptions," Proc. Sixth Int'l Conf. Web Information Systems and Technologies, vol. 1, pp. 40-47, 2010.

[9] G.A. Aranda Corral and J. Borrego-D_ıaz, "Reconciling Knowledge in Social Tagging Web Services," Proc. Fifth Int'l Conf. Hybrid Artificial Intelligence Systems, pp. 383-390, 2010.

[10] L. Chen, L. Hu, Z. Zheng, J. Wu, J. Yin, Y. Li, and S. Deng, "WTCluster: Utilizing Tags for Web Services Clustering," Proc. Ninth Int'l Conf. Service-Oriented Computing (ICSOC '11), pp. 204- 218, 2011.

[11] D. Wu, E. Sirin, J.A. Hendler, D.S. Nau, and B. Parsia, "Automatic Web Services Composition Using Shop2," Proc. 12th Int'l World Wide Web Conf.—Posters, WWW'03, 2003.

[12] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, H. Chang, and A.H.H. Ngu, "QoS-Aware Middleware for Web Services Composition," IEEE Trans. Software Eng., vol. 30, no. 5, pp. 311- 327, May 2004.

[13] K. Kritikos and D. Plexousakis, "Mixed-Integer Programming for QoS-Based Web Service Matchmaking," IEEE Trans. Services Computing, vol. 2, no. 2, pp. 122-139, Apr.-June 2009.

[14] X. Liu, Q. Zhao, G. Huang, H. Mei, and T. Teng, "Composing Data-Driven Service Mashups with Tag-Based Semantic Annotations," Proc. IEEE Int'l Conf. Web Services (ICWS '11), pp. 243-250, 2011.

[15] M.B. Blake and M.F. Nowlan, "Knowledge Discovery in Services (KDS): Aggregating Software Services to Discover Enterprise Mashups," IEEE Trans. Knowledge and Data Eng., vol. 23, no. 6, pp. 889-901, June 2011.

[16] Xuanzhe Liu, Yun Ma, Gang Huang, Junfeng Zhao, Hong Mei, and Yunxin Liu, "Data-Driven Composition for Service-Oriented Situational Web Applications," IEEE Trans. Services computing, vol. 8, no. 1, January/February 2015.

