Public Cultural Services recommendation System Architecture

Shufeng Ye[1,2], Yi Yang[1], Weixing Huang[1], Jian Wang[1*], Guigang Zhang[1]
1 Institute of Automation, Chinese Academy of Sciences, Beijing China,100190
2 School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing China
{yeshufeng2015,yangyi, weixing.huang, jian.wang, guigang.zhang}@ia.ac.cn
*Corresponding author: Jian Wang

***Abstract* — Public cultural sharing service plays an important role in the public cultural platform. The recommender systems can bring users useful cultural resources and information. For safety and security reasons of the cultural resources, the external applications are not allowed to directly access the original resource data stored in the cultural databases by sending recommendation requests. The cultural recommender system needs a way to isolate the cultural resources from the outside while keeping the recommendation services. To address this issue, this paper proposes Representational State Transfer (REST)-based architecture of the cultural recommender system that can provide loosely coupled structures for system modules while maintaining the security access of the cultural resources.**

***Keywords*—Representational State Transfer, cloud platform, recommended system,resource, JSON**

I. INTRODUCTION

With the advent of the web2.0 era, people have been surrounded by the network, surrounded by a variety of data. The digital construction of the public cultural service system is the basic construction project of the public cultural facilities, products, services, system, and service mode and operation management provided by the government to the society. The project has been planned at the national level since the beginning of this century to start the exploration of public cultural services and management of digital construction[1]. With the development of cloud computing and big data, we can build cloud platform, use large data analysis and forecasting function to realize the maximum sharing of various public cultural resources, and facilitate the collection, feedback and timely allocation of resources in order to enrich the cultural supply.

Public cultural sharing services has a high demand on the platform performance[2]. In the face of massive amounts of data, reasonable resource placement and scheduling are required. For a network system with a security level, it is necessary to efficiently schedule the resources under the premise of ensuring data security. RESTful API can guarantee the normal exchange between data, and it could implement the loose coupling between modules. The REST API is a stateless request mode, which is efficient for resource request access and is maintained for data security and operability. Aiming at the complexity of the integration of public cultural service platform, this paper proposes a system of public cultural service cloud platform based on REST architecture, which regards the system timing update control, user request parameters and website data as resources, using REST architecture component, and describes the structure of the public cultural service platform and the realization of key technologies.

The rest of the paper is organized as follows. Section 2 introduces the related work. Section 3 presents the Restful Architecture for public cultural service recommendation system. Section 4 shows a case study. Finally, conclusion and future work are given in Section 5.

II. Related woRK

A. Public cultural service resources construction

In the background of digitalization, information visualization and globalization, the digitalization of public culture has been an important direction in the construction of public cultural services at home and abroad[3]. In the process of building modern public cultural service system, the application of modern science and technology, communication means, information technology, digital technology and network technology to public cultural service is the inevitable strategic choice.

In 2008, Korea launched the "National Digital Library", which aims to integrate the cultural heritage resources of the Korean library system, private museums, public and private universities, and government agencies. The project consists of five independent agencies led by the Korean government to complete the work[4], the main project content includes: rich digital product supply, including books, movies, art heritage and so on. At the same time, all the basic research projects that contribute to the adaptation and application of digital technology in all areas of the cultural industry are funded. The European Digital Library (EDL)[5] project, which aims to achieve the sharing of European cultural and scientific information, has now integrated data from more than 2,000 cultural institutions in more than 20 countries.

At present, there have been a lot of work in the digital culture of public culture[6][7], such as the national cultural information resources sharing project. Zhang et al. [8]built a big data platform for the integration and sharing of the Chinese public cultural resources. Yang et al. [9]introduced a novel visualization approach to present the topic analysis for massive public cultural data. Zhai et al. [10] proposed a recommender system that provided adaptive methods for

Sponsors: the Support Program of the National '12th Five-Year-Plan of China' under Grant No. 2015BAK25B04.

978-1-5386-2072-4/17 $31.00 © 2017 IEEE
DOI 10.1109/QRS-C.2017.73

recommending public cultural resources. The above techniques can handle the massive public cultural data.

Yang et al. [11]established a public cultural knowledge graph platform that can extract and explore cultural knowledge by analyzing a knowledge graph.

B. Recommendation System

The mainstream recommendation algorithm is divided into three categories[12]: content-based recommendation, collaborative filtering recommendation and hybrid recommendation. Collaborative filtering is mainly divided into three types, based on user collaborative filtering, project-based collaborative filtering and model-based collaborative filtering. The basic idea of content-based recommendation [13]is to recommend other items that have similarities to the content of a given user. Model-based collaborative filtering is the use of machine learning or data mining algorithms, learning data to learn to identify complex patterns, to get the learning model, and then based on the learning model in the data set for intelligent prediction [14]. The well-known model-based collaborative filtering recommendation algorithm is: Bayesian belief network collaborative filtering model [15], clustering collaborative filtering model [16]. At present, most of the recommended system evaluation indicators have predictive accuracy[17], coverage, recall rate, diversity [18], receiver operating characteristic (ROC)

C. REST technology

Representational State Transfer (REST) is a software architecture style that Roy Fielding proposed in his doctoral thesis in 2000[19]. REST is a web-based architecture for designing distributed hypermedia systems that emphasize the scalability of component interactions, the versatility of interfaces, and the independent deployment of components [20]. Components in the REST system must adhere to the following constraints: First, all valuable and useful things on the network are abstracted as resources, and each resource corresponds to a unique resource identifier; secondly, only through a generic connector The interface operates on the resource, and the operation of the resource does not change the identifier of the resource, and all operations are stateless. However, the serious limitation of REST is to interactively follow a request-response mode that is always initiated by the client. These limitations are addressed by ARRESTED[21], and ARRESTED is a set of extensions that support both distributed and decentralized systems.

III. PUBLIC CULTURAL SERVICE PLATFORM BASED ON THE REST ARCHITECTURE

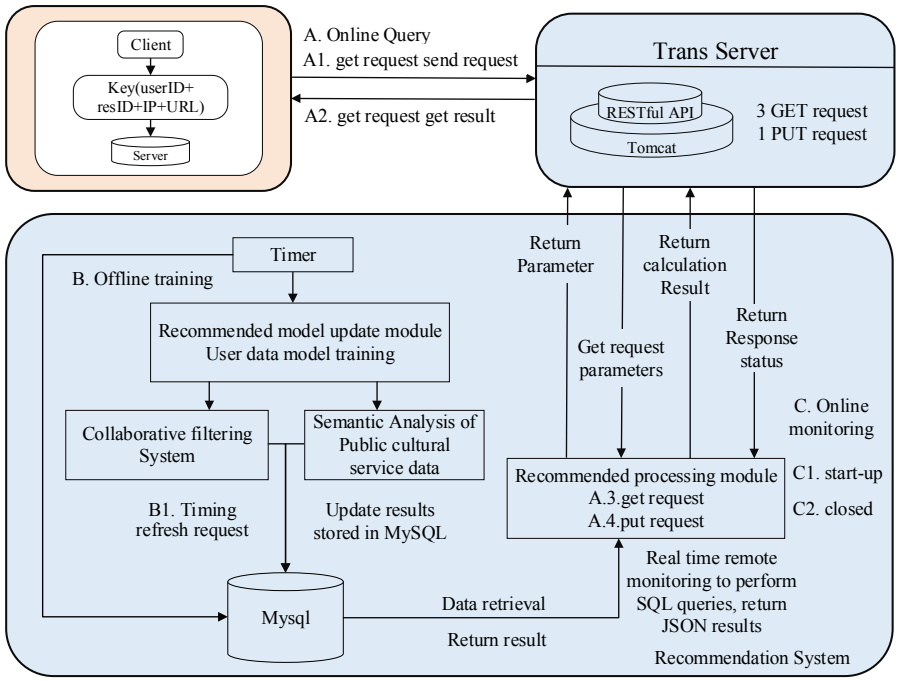

Figure 1 Public Culture Service Cloud Platform API System

A. Public culture service cloud platform

Based on the REST architecture, the public culture service platform is a recommended architecture based on online query and offline update. The recommendation module adopts the collaborative filtering recommendation algorithm based on user behavior[22]-[24], and uses the alternating least squares (ALS) to decompose the matrix[25]. As shown in Figure 1, the REST architecture is used to implement the client-to-server request service, the server's query, and the server's timing update. The platform provides a unified interface for the upper application, providing REST Web service API[26], when the Internet users through the

login and click on the site to select a public cultural service resources, the server will be the user parameters and selected resources The parameters are converted to the request URL call APIs to get the recommended public cultural data for the user. At the same time, the platform system will call the REST APIs to start the recommended processing module, real-time monitoring, the user's request for processing and return. On the other hand, the server also periodically executes the command call to update the recommended APIs to re-train the recommended system.

B. Achieve details for public cultural services cloud platform

1) Resource Description Method

In the concept of REST architectural style, anything that needs to be called is used as a resource[27]. In the public cultural service large data platform, items such as a recommended cultural label introduction, a user's own ID information, a user browsing behavior recommended video, etc., which can be defined as resources, and through the form of URI to be Identified.

As shown in Figure 2, on the basis of the above, calls to the APIs are represented by URIs, which serve as an interface for external client access to public service web resources. For example, URI:

http://recommendation/cultural/online/recresult/1/rec.json(the returned result format is JSON[28]), that is, the user's recommended online query resource result to User which ID is 1. For the module of recommended system update, states such as TopN recommended results, the number of training, the form of return, whether to create a new database table, etc., are resources, are all corresponding URIs. For the module of monitoring the user request, switch to monitor is also a resource.

The Restful APIs for resources represent the important concepts of cloud platform resource management and uniform use. The following will show how to use URI format to represent information semantics and hypermedia relationships between resources. Figure 2 summarizes the REST APIs structure of the Public Culture Service Recommendation System, which represents the hyperlink relationship of the return resource requested by the GET, PUT Command.

/

The root directory resource of the recommended system is a collection of other top-level directory resources that provide high-level links to other top-level resources.

/online/request

The resource represents a collection of each known item and allows access to each of the specified items.

/online/request/uid/resid

The request resource indicates that when the user views an item, it sends a parameter specifying the user and the item bowered, and request for a list of recommendations for the user and the item being accessed.

/online/convey

The resource indicates that the recommendation processing module gets the request parameters from the server.

/online/select

The resource indicates whether or not the identity of the result of the recommended list is returned.

/online/select/uid

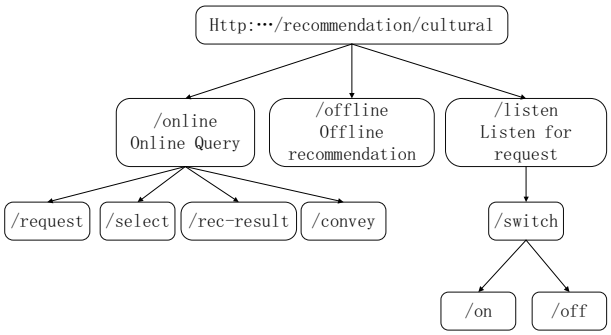

Figure 2 REST APIs presentation

The resource indicates whether the result for the user specified as uid has returned the identity of the recommendation.

/online/recresult

The resource represents the recommended result returned.

/online/recresult/uid

This resource represents the recommended result returned for the user specified as uId.

/offline/

This resource indicates that to start the recommended system with default configuration for offline training.

/offline/TopN/savemode

The resource indicates the TopN recommendation in the training result list and determines the storage mode of the recommended result (stored in mysql or generated json file)

/listen/switch/on

The resource indicates that to start listening the user requests and the data query.

/listen/switch/off

The resource indicates that to stop listening the user requests and the data query.

2) Interface implementation

When defining a RESTful service, it needs to specify the access method in addition to defining the URI of the access resource. Access to the use of HTTP provides the basic operation, corresponding to the public cultural platform resources operations are as follows:

- *Online queries*

GET /online/request/uid/resid

Take uid and resid as parameters to send to the user's request for recommendation.

GET /online/convey/

Get the user's request, and parse uid and resid as key for the query and calculation

GET / online/select/uid

Gets the respond of the server to obtain a confirmation from the uid user request

PUT /online/recresult/uid

Returns the user's recommendation for uid.

- *Offline updates*

PUT /offline

Start the recommended system with default configuration for offline training.

PUT /offline/TopN/savemode

Start an offline update training, the trained results generated as TopN items in the recommended list, and the recommended results are stored in mysql or generated json file.

- *Online monitoring*

PUT /listen/switch/on

Start online monitoring, and constantly send a request to obtain the request parameters, if a new user request is found, the parameters would be parsed and take the query calculation.

PUT /listen/switch/off

Close the online monitoring, do not get the user's request parameters.

3) Resource display

A recommended list for uid and resid, which can be returned in a JSON format through a GET request. The APIs called and the return resource are as follows:

GET /online/request/uid/resid

The returned JSON contains the id, title, and hyperlink information about the current user's browsing item. It also contains information about the relevant tag to the user's interest, the information about the TopN video resources recommended for the user, the resource information of the public venue associated with the resource visited, the similarity value between the relevant tags to the user's interest. The JSON format is represented as follows:

```
{
    "video": {
        "resId": "...",
        ...
        },
    "tag": [
        {
        "tagId": "...",
        ...
        }
    ],
    "recommendation": [
        {
            "userId": "..."
            "resId": "...",
        }
    ]
   ...
}
```

IV. CASE STUDY

1) Online queries

Based on the real use cases, this paper provides user parameters and resource parameters, respectively uid = 221.6.86.252 and resid = 893, to run the process of call the online query APIs.

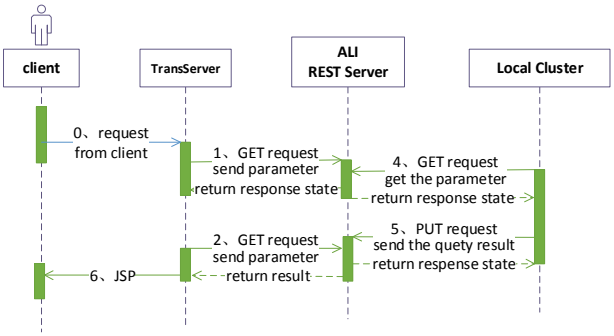

Figure 3 Sequence diagram of the online query

As Figure 3 shown, when a user click on a resource, such as a video, then an API request will be sent from the client, as **GET /online/request/221.6.86.252/893**, then the Trans-Server will receive a request from the browser and the parameters will be placed in the message cache. The online monitoring module API which has been started continue to send API request to the Trans-Server as **GET/online/convey/**, then detect that the server message cache has 221.6.86.252: 893 parameters, and return the parameter to the recommended processing module. The recommended processing module through the query to get the recommended results according the parameters, and then call the RESTful API **PUT /online/recresult/221.6.86.252**, sent result to the Trans-Server results cache. On the other side, the client will send another request **GET /online/select/221.6.86.252** to Trans-Server. If the cache does not have the user's recommendation results, then it would send the request again from time to time, else if the cache has been returned 221.6.86.252 user's recommended query as a result, the result would be returned immediately, and be cleared from the result cache in TransServer in a while of time. The results returned in JSON Type in the browser would be parsed and displayed on the JSP[29].The returned JSON contains keys such as video, tag, recommendation, library, similarity and user preference and so on. Each of the keys has corresponding values that can be also represented as a key-value pair. The JSON is listed as follows:

```
{
    "video": {
        "resId": "893",
        "title": " Olympic Games and Folklore ",
        "videoUrl": "http://zygx2.bjgxgc.cn/573_D_720x576.mp4"
    },
    "tag": [
        {
            "tagId": "600",
            "tagValue": " Chinese spirit "
        }
    ],
    "recommendation": [
        {
            "userId": "221.6.86.252"
        },
        [
            {
                "resId": "891",
                "resName": " Chinese Tea Culture and Beijing "
            }
        ]
    ],
    ...
    ...
}
```


2) Offline updates

The public culture service platform in the server configuration of a timer, from time to time to recommend the update module to send a request for update **PUT /offline/5/mysql**, then call the recommended module to accept the request, and the tables which are used for the user recommended in mysql be copied. The user visited behavior data stored in it is extracted as training data of the collaborative filtering system to generate a new recommended system model. According to each user's top 5 scores to calculate the recommended results, and the results will be stored in mysql. At the same time replace the original user recommended data table, ensure that the system's online query function.

V. CONCLUSION

This paper introduces a rest-based API for controlling the public cultural service cloud platform. We present the API as a novel programming platform for online recommender systems. We believe that as long as there is sufficient adjustment, the proposed API can be applied to any similar system. The main idea is to provide a support for the creating REST interface, monitoring and adapting to user requests that can be deployed across a heterogeneous set of execution hosts (including server Web server and Web browser client). The final RESTful API provides a set of basic resources for online queries, offline updates, and online monitoring. The API is used to build and support the recommender system to ensure the automatic load balancing and fault tolerance.

Restful technology is based on the http protocol, is also stateless. Just a way of architecture, so its security features are required to achieve our own, there is no ready. In a REST system, the server does not save any status about the client. In other words, the client itself is responsible for the maintenance of user status, future work will be in the rest of the security features and user status maintenance to do further research.

ACKNOWLEDGMENT

We would like to thank all colleagues and students who helped for our work. We thank the National Key Technology Support Program of the National '12th Five-Year-Plan of China' under Grant No. 2015BAK25B04 for partially supporting our research.

REFERENCES

[1] Wang X. On the Construction of Public Cultural Service System. Journal of Nanyang Normal University, 2007.

[2] Wang R. The Public Library Service Positioning in the Public Cultural Service System. Library & Information, 2009.

[3] Zhang G, Wang J, Huang W, et al. Big Data Collection and Analysis Framework Research for Public Digital Culture Sharing Service[C]// IEEE International Conference on Multimedia Big Data. IEEE, 2015:196-199.

[4] Lee H. Collaboration in cultural heritage digitisation in East Asia. Program, 2010, 44(4):357-373.

[5] Europeana. European Commission welcomes European Parliamentsupport.2014-03-10.

[6] Shoulong Z. On the Public Culture Service and Model Choice in China. Jiangsu Social Sciences, 2008.

[7] Cao S J, Ting-Hua G U, Wang Z H. Research Progress and Feature Analysis of China's Public Digital Culture Construction and Service. Library Tribune, 2015.

[8] G. Zhang, Y. Yang, X. Zhai, W. Huang and J. Wang, "Public Cultural Big Data Analysis Platform," 2016 IEEE Second International Conference on Multimedia Big Data (BigMM), Taipei, 2016, pp. 398-403.

[9] Y. Yang, J. Wang, W. Huang and G. Zhang, "TopicPie: An Interactive Visualization for LDA-Based Topic Analysis," 2016 IEEE Second International Conference on Multimedia Big Data (BigMM), Taipei, 2016, pp. 25-32.

[10] X. Zhai, F. Jin, J. Wang, Y. Yang, Q. Yao, Q. Qiu and G. Zhang, "A Kind of Precision Recommendation Method for Massive Public Digital Cultural Resources: A Preliminary Report," 2016 IEEE Second International Conference on Multimedia Big Data (BigMM), Taipei, 2016, pp. 56-59.

[11] Y. Yang, G. Zhang, J. Wang, S. Ye, and J. Hu, "Public Cultural Knowledge Graph Platform," The International Workshop on Big Data for Intelligent Services and Applications (BDISA 2017) in conjunction with 2017 IEEE International Conference on Semantic Computing (ICSC 2017), in press.

[12] Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(6): 734-749.

[13] Balabanović M, Shoham Y. Fab: content-based, collaborative recommendation[J]. Communications of the ACM, 1997, 40(3): 66-72.

[14] Su Xiaoyuan, Khoshgoftaar T M. A survey of collaborative filtering techniques[J]. Advances in Artificial Intelligence,2009.

doi:10.1155/2009/421425.
[15] Ungar L H, Foster D P. Clustering methods for collaborative filtering[C]//Proceedings of the AAAI Workshop on Recommendation Systems, Madison, USA, Jul 26-27, 1998. MenloPark, CA, USA: AAAI, 1998: 114-129.
[16] Ma H, Liu C, King I, et al. Probabilistic factor models forWeb site recommendation//Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, Jul 24-28,2011. New York, NY, USA: ACM, 2011: 265-274.
[17] Massa P, Avesani P. Trust-aware recommender systems//Proceedings of the 2007 ACM Conference Recommender Systems, Minneapolis, USA, Oct 19-20, 2007. New York,NY, USA: ACM, 2007: 17-24.
[18] Smyth B, McClave P. Similarity vs. diversity//Case-Based Reasoning Research and Development. Berlin, Heidelberg: Springer, 2001: 347-361.
[19] Fielding R T. Architectural styles and the design of network-based software architectures. University of California Irvine, 2000,64(3):303.
[20] Sundvall E, Nyström M, Karlsson D, et al. Applying representational state transfer (REST) architecture to archetype-based electronic health record systems. BMC Medical Informatics and Decision Making, 2013, 13(1):57.
[21] R. Khare and R. N. Taylor. Extending the representational state transfer (rest) architectural style for decentralized systems. In ICSE '04: Proceedings of the 26th International Conference on Software Engineering, pages 428–437, Washington, DC, USA, 2004. IEEE Computer Society.
[22] Chen L, Chen G, Wang F. Recommender systems based on user reviews: the state of the art. User Modeling and User-Adapted Interaction, 2015, 25(2):99-154.
[23] Beel J, Genzmehr M, Langer S, et al. A comparative analysis of offline and online evaluations and discussion of research paper recommender system evaluation[C]// International Workshop on Reproducibility and Replication in Recommender Systems Evaluation. ACM, 2013:7-14.
[24] Shani G, Gunawardana A. Evaluating Recommendation Systems. Recommender Systems Handbook, 2011:257-297.
[25] De Lathauwer L, Nion D. Decompositions of a Higher-Order Tensor in Block Terms—Part III: Alternating Least Squares Algorithms. Siam Journal on Matrix Analysis & Applications, 2009, 30(3):1067-1083.
[26] Masse M. REST API Design Rulebook. 2011.
[27] Chee Er C. Realization of resource-efficient embedded Web services using Representational State Transfer (REST) packaging and roll-back streaming XML (RBStreX) parser. Dissertations & Theses - Gradworks, 2011.
[28] JSON.Internet informational RFC 4627.2006.
[29] Brown S, Dalton S, Raible M. Professional JSP. 2003