Case-Intelligence Recommendation on Massive Contents Processing through Dynamic Computing

Rui Li
Department of Information Engineering
Anhui Communications Technical College
Hefei, China
Liruilary@gmail.com

Jianyang Li, Benkun Zhu
School of Computer and Information
Hefei University of Technology
Hefei, China
lijianyang@sina.com

Abstract—How to suggest a valid recommend within a reasonable time is the greatest technical challenge for the recommendation system, for which tremendous user cases with high dimension are generated while it runs in real time, and these massive data are too difficult to compute directly. This paper proposes a case -intelligence system framework along with a feature -based multi -layer feed -forward neural networks (MFNN) to succeed case- retrieval based on dynamic computing, which constructs the neural networks dependence on the real input vectors instead of the fixed and dull networks structure presupposed, and can apply many kinds of knowledge granularity from various levels effectively to help users for information retrieval and case adaptation. Our subsequent experimental results indicate that it is capable of handling the massive personalized data, and our covering algorithm can decrease the complexity of MFNN algorithm for dynamic computing, which performs adaptable knowledge granularity to enhance the system's efficiency of reasoning.

Keywords- **case-intelligence recommender; dynamic computing; covering algorithm; MFNN*; system efficiency***

I. INTRODUCTION

The acquisition of personalized need is the key to effective recommender, which can capture user's information demand exactly, and promote the wide application of E-commerce. Many intelligent tools have been used to help users search, locate and manage web documents, such as data mining and other artificial intelligence technology used to collect data, obtain their behaviors in e-commerce, and generate interests in the products for consumer's purchase [1]. Thus, more and more recommendation systems have been developed to fit for e-commerce use [2], which could be distinguished into three different types: rule-based filtering, content based filtering, and collaborative filtering. Personalized information acquirement is an important research center naturally, for they realize that efficient access and quality of service are helpful to attract more visitors [3].

But the statistics report from ACM indicates that the current recommender can not meet the large-scale e-commerce applications, and it has poor real-time problem accompany with the problem of weak quality in accuracy. As well known, the personalized information- the real user behavior data from websites, can accumulate up to millions or even billions for the recommendation system running, the processing of massive user data is the greatest challenge, for which involves system performance deeply [4]. Because of the recommended system is a data priority, the more accumulation of data, and the higher accuracy the recommender can explore [5]. How to suggest a valid recommend in a reasonable time people need from the mass merchandise has become increasingly difficult, which is exactly the same with case intelligence as we have described in the paper [6].

Normally, recommendation system uses low level analogy reasoning, which is an important cognitive model of human sense [7]. The "low level" means simple analogy, which is not inferring in different domain data. For the incomplete knowledge implicit in the reasoning, the conclusions of analogy may be effective or invalid, which must require an objective confirmation or readjust until new knowledge or contradiction comes. Generally, CBR system implements four processes well known as the 4R - Retrieve Reuse, Revise, and Retain, to solve new problems [8]. The former cases can also be used to evaluate the new issues and new programs of problem-solving [9], and prevent the potential errors in the future. Cases can be reused by similarity computing as case knowledge space conversion, which is the glorious with exciting highlight in the construction of CBR intelligent system, and the characteristic advantage distinguishes CBR systems from RBR systems thoroughly [10].

System flexibility depends on case knowledge space conversion through case- adaptation method, whose process is manipulate the adjustment of space projection based on former knowledge; So that the system outputs a set of most similar cases to guide users' corresponding inputs, from which case- adaptation can get great benefits [11]. This paper focus on such problems by using MFNN to acquire flexible knowledge for our synthesis reasoning, which mainly comes from Granular Computing and Case-Based Reasoning, can combine various reasoning principles and integrate many methods, and enhance the system's efficiency of reasoning by means of dynamic granularity.

II. CASE SELECTION MODELING

The acquisition of user's data is the key to meet the individual needs and infer the learning ability of the recommender. How to obtain personalized demands, expression knowledge to support user from information retrieving and adapting, is the most important task to intelligent recommendation system. The paper [12] has

described that case-intelligence recommendation system can be used for acquiring effective personalized knowledge, besides several other adaptive interfaces attempt to collect user information unobtrusively.

A. *Case retrieval*

Case-Based Reasoning as a cognitive model suggests people learn the best from former problem-solving cases as they solve new problems, which is the simulator of human analogy learning. Case retrieval is the key process of the CBR system, and plays an important role in Machine Learning community. Case is the integrated representation of human sense, logics and creativity, great achievement has been acquired for CBR in the field of knowledge lack, and case intelligent decision techniques is built from CBR can overcome some defects, such as poor flexibility, and provides decision support [13].

Artificial Neural Networks has the natural relationship with CBR, and several successful theories have been put forward to integrate ANN into the CBR system. Similarity assessment plays a key role in lazy learning methods, but the traditional k-nearest neighbor (kNN), which are applicable to any representation for cases gathered, measures similarity between cases are time-consuming. Specifically, the similarity measurement is empirically evaluated on relational data sets of different expressiveness [14]. The case library in the CBR system can be viewed as a CSP, therefore CS-ANN model, such as Schema model, Hopfield model, Boltzmann and Harmony theory can be employed to construct the case library. Theoretically, in the symbolic description model-based CBR system, rules can be elicited by ANN method; and in the quantitative description model-based CBR system, due to the system's flexibility, many mathematical approaches and optimization techniques can be employed in the definition and analysis of similarity measurement and case adaptation criteria, thereby more and more ANN applications can be prevailed in the CBR system.

There have been a lot of very wide and profound researches on this topic, including data integration, query processing, and fine-granularity data sharing. Currently the main way for case-matching is the k-nearest neighbor algorithm, but it can not reflect the relationship between the cases and as well as their attributes, neither can it shows the preference of the customers. The widely used BP Network can be used to create a CBR retrieval model and its most outstanding characteristic is that the retrieving speed and the size of the case library are in a non-linear relationship. But some insurmountable weaknesses remained in these application systems, for example, the weak performance in interpretation and large-scale case library that due to the high complexity of ANN algorithm makes the systems far too complex and hard to be integrated. Especially for the large-scale case library, the retrieving time is unacceptable. Facing these problems, we should employ the MFNN dynamic computing instead.

After investigating the behavior of MFNN together with many kinds of existent algorithms for case retrieval based on MFNN, besides BP, simulated annealing algorithm and their ameliorated algorithms, we found that weaknesses such as having lower speed and local extreme value, are inherent in those algorithms, and cannot be conquered thoroughly. Considering that more and more interests are focusing on data intensive computing and data cloud computing in industry and academia, those methods can not be used in large- scaled case library especially for dynamic computing as intelligent case retrieval techniques. In this paper, we suggest to use MFNN and employ Covering algorithm [15], which is easily understandable and constructed, to effectively decrease the complexity of ANN algorithm, to manipulate the massive personalized users' data in real time.

B. *Personalized features*

In order to achieve personalization services, we must trace down user's behavior to study his interest, which can be collected from three sources: Server, Client, and Proxy, to exploit potential information or patterns useful. There are three types log files to record users' actions: Access logs, Refer logs and Agent logs, also may be Cookie logs. Besides, there are query information, register information, and website structure. These data can be divided into the following categories:

- Content data: the real data which user having read and used, mainly constituted by text and image.
- Structure data: describing how to construct website and organize the webpage. The page can be constructed by HTML, XML representation as the tree structure, and HTML label set as the root for the tree, whose structure can be connected by hyperlinks between the different pages; and web structure data mining refers to a method of mining the structure between different web pages user browsing.
- Usage data: describing web usage pattern, such as IP address, URL, webpage citation, access time and date, which indicates each user's behavior model, and a typical use of data originates from server log. The Usage Data Collector (UDC) is a framework for collecting usage data information, which gathers information about the kinds of things that the user is doing (i.e. activating views, editors, etc.).
- User Profile: relative statistics data from web user, including user registration and personal information, for example, user name, education background, career, position, age, income, personal hobby, etc. Due to the dynamic and complex nature of web users, automatically acquiring user profiles is very challenging.

Recommender is running in much complex environment, each data is regarded as the foundation for knowledge representation, but may be represented in semi-structured or unstructured model, or even in natural language texts. Although the research of personalized interest has become widespread only in recent years, several adaptive interfaces have been developed to describe personalization by observing a user's browsing behavior for promoting recommendation performance.

Web data mining has been widely used for sharing and exchanging of data and resources among numerous computer nodes, and personalized data objects could be identified with high-dimensional feature vectors. Though an investment table feed back by user may be acquired, user behavior extraction is compulsory to estimating the user interest degree. There is a growing trend among

companies, organizations and individuals to gather information through web data mining to utilize that information in their best interest [16]. TM is the topic-keywords matrix of user's weight for the expression of user interest degree in our paper, where n represents the number of user interested topics, and w is the weight. A new approach of web DM technologies to users' data can generate an analysis of customers' behavior, by synthesizing key abstract information that will facilitate and improve the customization of services.

$$TM = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1m} \\ w_{21} & w_{22} & \dots & w_{2m} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nm} \end{bmatrix}$$

C. System construction

The personalized recommendation system that we proposed based on case intelligence mainly construct by three parts: input module, recommendation methods and output module, as shown in Figure 1, which the dynamic computing module with MFNN algorithm as we have described, is added in as retrieve process. Our dynamic computing method will be validated in the sequent chapter, which will be described in detail and be evaluated directly with huge personalized users' data.

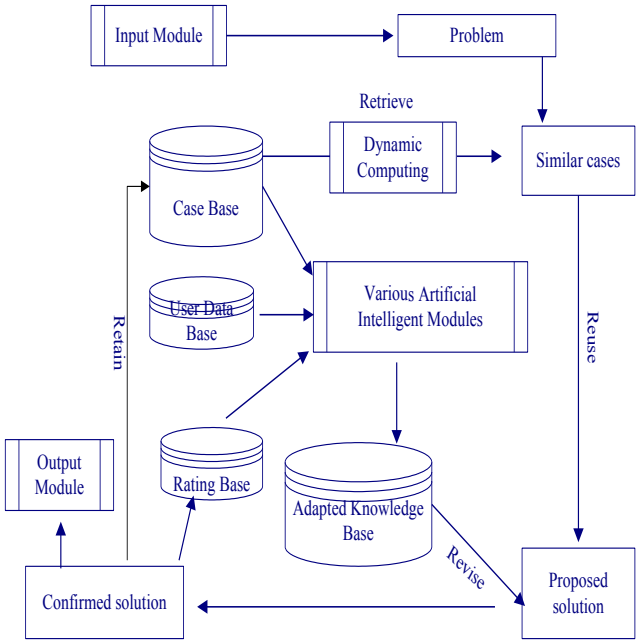

Figure 1. the system frame

Personalized recommendation system involves a serial processes of gathering and storing information about site visitors, managing the content assets, analyzing current and past user interactive behavior, and delivering the right content to each visitor based on its analysis, the details of whose components is not described in this paper, which can be seen in paper [6]. We care only about the dynamic computing method prepared for case matching, revealing and exploiting the most similar users' groups, where the implementation process of personalized recommendation for the same with common recommendation system is also not mentioned.

III. DYNAMIC COMPUTING FOR RECOMMENDER

Such personalized data are difficult for computing, for they are massive with high dimensional and increasing in every moment drilled from websites. While the accumulation of the real user behavior data up to millions or even billions in real time, the greatest technical challenge most organizations face is how to suggest a valid recommend within a reasonable time from the dynamic data-ocean.

A. Dynamic computing algorithm

MFNN consists of an input layer, one or more hidden layers, an output layer, where layers are in order of priority, and the i-layer neurons receive signals only from the i-1 layer neurons without feedback between each layer. It has proved that 3-layer networks can realize any given function for approximate accuracy, and can be used to solve the nonlinear classification. This paper suggests Multi-Layer covering algorithm for dynamic computing, which is a constructive method for ANN, and the foundation of the geometrical representation McCulloch-Pitts neural model.

Our networks construct its layer-structure by means of input training data for itself. Firstly, Covering Algorithm assumes that each input vectors x of an n-dimension can be projected on a bounded set of a certain hyper-sphere S^n of a (n+l)-dimensional space (define the sphere radii is R), it is no doubt that the transformation must be achieved to the aim through widening the vector's dimension. The dynamic algorithm is described as the following steps:

(1)Search for the maximum sample ||r|| from the learning samples X, then project all the points in X to the sphere, which centers at the base point, radius R(R> = r);

(2)Assume i=1, t=0; // i represents the i-th class and regarded as the sample covering center for covering; t is the number of covering domain;

(3)Then let m=mod(i, N):

If $X_m = \varnothing$, goto(11),

Else t++, randomly select a point a_i from X_m ;

(4)Calculate $d_i = \max\{\langle a^i, x\rangle\}$ to make the cover C(a_i), which centers from a_i, θ as the threshold;

(5)Calculate the barycenter of all the points in C(a_i), project it to the sphere and get projective point a'_i , then calculate its threshold θ' and partition the sphere domain C(a'_i).

(6)If the total points which C(a'_i)covers, is more than what C(a_i)covers,

Then let $a'_i \to a_i$, $\theta' \to \theta$, return(5),

Else restore the original parameters;

(7)Calculate $B = \min_{j \notin X_m} \{x \mid d(a, x)\}$,

Where d(a, x)represents the distance between a and x, let the translation of point a '= a;

(8)If |B|>n, (|B| represents the card(B)), goto(10)

Else find the pedal b from a to P(B), let P(K)= P(B),

for each $x \in X /(P(K) \cup Xm)$,

calculate d(x)：d(x)=<a,c-x>/<b,c-x>

Where c is the random point P(K).

If there exists x: <a, cx> = 0,

then let ck + 1 = x, a '= a, P(K + 1)= P(K) ∪ {ck + 1}

else let d =d(x*)= $\min_{x \notin X_m}$ {d(x)},

assume a' =R(a-db)/|a-db|,

where R is the radius of the spherical Sn+1

project vector(a-db)to Sn+1, take ck + 1 =. x *.

(9)P(k+1)= P(k) ∪ {ck+1}(P(k) ∪ {ck +1}

if k+1 > n, then a' is the result, and goto(11)

else project a' to P(k + 1), b=bk+1 , a= a', k++ ,return(8)

(at the beginning, let k = |B|).

(10)Count for the corresponding spherical domain C(a'_i)

If it is more than what C(a_i)covers,

let $a'_i \to a_i$, $\theta' \to \theta$, return(5)

Else restore the original parameters, and get a covering domain C(t)of X_m ;marked with Cp(t) , X_m / Cp(t)→ X_m

(11)Count for nonempty set among X1, X2, ..., XN,

If the number is greater than 1; i++; return(3);

By this way of dimension expansion and space projection, the domain covering for the similar users in the user library will be well achieved, which can be used as the input of the MFNN for case matching in dynamic computing.

B. Experiments with outlook

Two experiments are designed to validate our system algorithm, and the experimental data of "forest cover type" is downloaded from UCI repository, whose main information describes as follows: Number of instances (observations) 581012, Number of Attribute: 54 (12 measures, 10 quantitative variables, 4 binary wilderness areas and 40 binary soil type variables); Number of Class: 7; Missing Attribute: none. Each record represents the user personalized data collected from the websites, which is regarded as a user behavior vector with 54 Attributes, and user data library accumulates to 581,012 users' sessions in real time. Then, the normal Macro-averaging is used to calculate all classes' means $F-score$:

$$F-score=\frac{2*recall*precision}{recall+precision}$$

We can find the data are spare matrix with high dimensional and huge records. The actual forest cover type for a given observation (30 x 30 meter cell) is determined from US Forest Service (USFS) Region 2 Resource Information System (RIS) data; each record is regarded as a user case in our experiments, to decrease the interference in the pretreatment of the real world. The simulation starts from 10,000 to 100,000 user cases, and the user cases are randomly selected.

The first is designed to confirm that our MFNN and its algorithm are reliable for its excellent precision and outlook speed, as Table 1 shows. We can find that each covering domain is a most similar group, and can be represented as a most similar user group in the same interest degree, which can be recommended for the new user with the same interest. Besides, user data accumulating is gradually adding up, which can be calculating in the background, and we can use the results of the previous partition without recalculating the groups.

TABLE I. SYSTEM PERFORMANCE

Records	Domains	F-score(%)	T-partion(s)	Time(ms)
10,000	2468	79.1	14.207	14.81
15,000	3607	81.7	33.225	34.339
20,000	4360	82.9	49.209	50.391
30,000	5646	81.3	68.316	69.577
40,000	6391	82.4	86.05	87.349
50,000	6843	81.6	103.06	104.4
100,000	7651	83.2	272.31	273.75

The subsequent experiment is to validate it for large-scale data in dynamic computing. Original user records are increasing with a serial of new users adding in, where original records represent previous personalized resource that the recommendation system have drilled from websites formerly and saved as the greatest asset. While new users' data is adding in the system, this is just like what the new personalized data is acquired and stored in case library in real time. So the second experiment is divided into two items to simulate dynamic users' action, the one is set with the fixed 2,000 new users adding in, the other with the fixed 10,000 with comparison.

TABLE II. SYSTEM COST IN REAL TIME

Origin	⊿Records	⊿Domains	⊿T-partion(s)	⊿Time(ms)
2,000	2,000	414	61.2	1.5895
4,000	2,000	434	69.1	3.2935
6,000	2,000	545	74.2	3.8923
8,000	2,000	575	79.1	4.997
10,000	10,000	1,139	81.7	19.529
15,000	10,000	753	82.9	16.052
20,000	10,000	1,286	81.3	19.186
30,000	10,000	745	82.4	17.772
40,000	10,000	452	81.6	17.051

As Table 2 indicates, it costs only a little system resource to recommend the most similar users' case in dynamic computing, and states clearly that the cost for system data recalculating is valuable and acceptable. Though the real commendation system cannot recalculate its personalized data library unless system collapsing, the costs in our experiments for new users' data feed in the library are under recalculating to meet the general testing of Machine Learning algorithm demands.

IV. CONCLUSION

Personalized data which are explored from websites, are the greatest asset for recommendation system, but they are massive along with high dimension, and hardly dealt with in real time, especially in dynamic environment. Addressing such problem of efficiency, the paper suggests an idea of intelligent information retrieval processing, whose basic task is to construct a suitable granularity to decrease the system complexity for real time computing; and establishes a user model for personalized recommender based on our dynamic computing algorithm, which has such advantages like clear system structure, feasible component combination, and can effectively help users for information retrieval and case adaptation.

ACKNOWLEDGMENT

This research is supported by the Natural Science Project of Anhui Province under grants KJ2014A050.

REFERENCES

[1] Wei Chu, Seung-Taek. Park, "Personalized Recommendation on Dynamic Content Using Predictive Bilinear Models", WWW2009, pp691-700

[2] Khalid Al-Kofahi, Peter Jackson etc, "A Document Recommendation System Blending Retrieval and Categorization Technologies", AAAI Workshop 2007, pp 9-18

[3] Zurina Saaya, Markus Schaal, Maurice Coyle, Peter Briggs, and Barry Smyth. "Exploiting Extended Search Sessions for Recommending Search Experiences in the Social Web". ICCBR 2012. LNAI, (7466), pp. 369-383

[4] Zurina Saaya, Markus Schaal, Maurice Coyle, Peter Briggs, and Barry Smyth, "Exploiting Extended Search Sessions for Recommending Search Experiences in the Social Web", ICCBR 2012. LNAI, (7466), pp369-383

[5] Ruihai Dong, Markus Schaal, Michael P. O'Mahony, Kevin McCarthy,and Barry Smyth, "Opinionated Product Recommendation", ICCBR2013, LNCS,(7969), pp44-58

[6] Jianyang Li, Xiaoping Liu. "A Case-intelligence Recommendation System on Massive Contents Processing through RS and RBF". ICMTMA 2013. pp1-4

[7] Amira Abdel-Aziz, Weiwei Cheng, Marc Strickert, and Eyke Hüllermeier, "Preference-Based CBR: A Search-Based Problem Solving Framework", ICCBR 2013, pp1-14

[8] Odd Erik Gundersen, "Toward Measuring the Similarity of Complex Event Sequences in Real-Time", ICCBR 2012. LNAI, (7466), pp.107-121

[9] Debarun Kar, Sutanu Chakraborti, and Balaraman Ravindran, "Feature Weighting and Confidence Based Prediction for Case Based Reasoning Systems" , LNAI, (7466), pp. 211-225

[10] Jianyang Li,Xiaoping Liu, Rui Li. "Application of Improved MFNN on Dynamic Computing for Case-Intelligence Recommendation System". IMSNA2012, pp407-410

[11] Bach, K., Althoff, K.-D., Newo, R., Stahl, A. " A Case-Based Reasoning Approach for Providing Machine Diagnosis from Service Reports". ICCBR 2011. LNCS, (6880), pp. 363-377

[12] Jianyang Li, Xiaoping Liu. "Personalized Recommendation System on Massive Content Processing Using Improved MFNN". Springer's LNCS, 7529 (2012), pp183-190

[13] Zhiwei Ni, Jianyang Li, Fenggang Li, Shanlin Yang. "Survey of Case Decision Techniques and Case Decision Support System". Chinese Computer science,2009,36(11), pp18-24

[14] David McSherry, "Conversational Case-Based Reasoning in Medical Decision Making", Artificial Intelligence in Medicine, (52), 59–66 (2011)

[15] ZHANG Ling. "The relationship between Kernel Functions Based SVM and Three-layer Feedforward Neural Networks". Chinese J. Computer, 25(7): 696-700, 2002.

[16] Catherine Havasi, Jason Alonso, Robert Speer, "Reducing the Dimensionality of Data Streams using Common Sense", WWW2010, pp 14-19