High Efficiency Web Page Metadata Search Method for Large Data Applications

Haiyang LV[1], Jingnan LIU[2]

[1]Computer Training Center, Yantai Nanshan University, Yantai Shandong, 265713, China
[2]Electrical Information Experiment Center, Yantai Nanshan University, Yantai Shandong, 265713, China

***Abstract* — We consider a rough set PNN Gaussian block expectation-implanting sorting recommendation algorithm for multi-label webpage. This is aimed to solve the problems of imprecise information and new information in multi-label recommendation algorithm caused by unsatisfactory effects of traditional precise algorithm. Firstly, specific to uncertain information, rough set theory is used to improve the model of traditional PNN and makes it suitable to process the uncertainty of information. Secondly, to solve the problem of fixed PNN in: i) processing multi-label webpage recommendation, ii) bad coverage, iii) large structural redundancy, and iv) new label not being identified rapidly, we use Gaussian block to: i) implant expectation sorting mode, ii) establish the operations of combine, add and delete probabilistic function blocks, iii) improve prediction accuracy and reduce computation complexity and iv) effectively resolve real-time problem of new information prediction. Finally, our comparison experiments in the example of Yahoo multi-label data set show the algorithm above has higher computation accuracy and efficiency.**

Keywords - multi-label; webpage searching; big data; data mining; cluster; webpage recommendation

I. INTRODUCTION

In traditional two-category or multi-category problem processing, only one label or category is associated with every example, which is similar to a precise classification problem. However, in some classification problems, an example may be associated with multiple labels or categories. The former kind of problems is called single-label recommendation and the latter one is called multi-label recommendation [1-2]. In multi-label recommendation problem, one sample can be associated with dozens of categories at most so it is more complex than single-label problem [3]. It is more difficult that the number of multi-label associations is uncertain. In addition, relationship between categories is also uncertain sometimes. Therefore, a large amount of data is required to form relatively mature statistical information. However, required data size [4] cannot be acquired in the beginning. There is no literature and report about PNN multi-label classification recommendation at home and traditional PNN algorithm has two deficiencies in processing multi-label recommendation problem: multi-label individual information is imprecise and network training would lose accuracy easily; fixed PNN model cannot precisely identify information of new category in real time. Specific to two problems above, rough set theory [12] is used to solve the inaccuracy, inconstancy and incompleteness of information and combine, add and delete operations of dynamic Gauss probabilistic function block are add to realize the real-time judgment of category information, reduce algorithm redundancy and improve computation efficiency.

II. MULTI-LABEL CLASSIFICATION INDEX

A. Problem Description.

In multi-label learning, training verification sample set is in form of $TV=\left\{d_1,d_2,\cdots,d_{|TV|}\right\}$, which is generally composed of a certain number of text examples and each example is associated with a subset of category *C*. Combine categories according to features of each example in training set. *TV* is mainly used to train and verity text classification system. Unknown sample examples of automatic classification system are included in test set $Te=\left\{d_{|TV|+1},d_{|TV|+2},\cdots,d_{|\Omega|}\right\}$ and sample examples in the training set are not included. *TV* includes $|TV|$ number of individual sample examples and *Te* includes $|\Omega|-|TV|$ number of sample examples. After training set *TV* trains classification system, examples in test set *Te* can be predicted.

Multi-label classification system is generally expressed by real-valued function, in form of $f:D\times C\to$. The function can return a value of $\left\langle d_j\middle|c_j\right\rangle\in D\times C$ for key value to approximately express the credibility of test example d_j belonging to category $c_j\in C_j$, where $C_j\subset C$. Real-valued function $f(\cdot)$ can be transferred into rank sorting function $r(\cdot)$. If $f\left(d_j,c_1\right)<f\left(d_j,c_2\right)$, then $r\left(d_j,c_1\right)<r\left(d_j,c_2\right)$. Based on such sorting, each sample example can be mapped on $\left\{1,2,\cdots|C|\right\}$. If C_j is suitable category of test example d_j, rank of sample example in category C_j successfully acquired by classification system is higher than the rank of sample example beyond category C_j. What's more, a threshold parameter τ is set here. Only when the classification rank of

the example is higher than threshold τ, test sample can be granted.

B. Evaluation Index

(a) Hamming loss (*hl*) is to indicate times of misclassification of example d_j. Its form can be expressed as follows:

$$hl = \frac{1}{p}\sum_{j=1}^{p}\frac{1}{|C|}\left|P_j \Delta C_j\right| \tag{1}$$

(b)Where, $|C|$ is classification number, Δ is the symmetric set difference between predicted category P_j of test example d_j and reasonable category C_j. Predicted category rank is higher than threshold value τ.

$$E_{error} = \frac{1}{p}\sum_{j=1}^{p} E_{error}^{j} \tag{2}$$

$$E_{error}^{j} = \begin{cases} 0, if\left[\arg\max_{c\in C} f\left(d_j, c\right) \in C_j\right] \\ 1, otherwise \end{cases} \tag{3}$$

Where what returned by $\left[\arg\max_{c\in C} f\left(d_j, c\right) \in C_j\right]$ is the highest category in test example d_j rank sorting.

(c) Coverage index (C_{cover}) is to evaluate the category that can distribute all the possibilities for test example d_j only by reduce category rank threshold value. Its computation formula is:

$$C_{cover} = \frac{1}{p}\sum_{j=1}^{p}\left(\max_{c\in C_j} r\left(d_j, c\right) - 1\right) \tag{4}$$

Where, what returned by $\max_{c\in C_j} r\left(d_j, c\right)$ is the highest-rank category set of test example d_j.

(d) Rank sorting loss (C_{rloss}) is to evaluate reverse sorting fraction of $\langle c_k, c_l\rangle$ to test example d_j. Its computation formula is:

$$C_{rloss} = \frac{1}{p}\sum_{j=1}^{p}\frac{\left|\left\{\left(c_k, c_l\right) \middle| f\left(d_j, c_k\right) \le f\left(d_j, c_l\right)\right\}\right|}{\left|C_j\right| \cdot \left|\overline{C}_j\right|} \tag{5}$$

Where, $\left(c_k, c_l\right) \in C_j \times \overline{C}_j$ and $\overline{C}_j$ is the complementary set of C_j in category set C.

(e) Average precision (C_{avep}) is to evaluate average prediction precision of all the test examples d_j. The form is:

$$C_{avep} = \frac{1}{p}\sum_{j=1}^{p}\frac{1}{\left|C_j\right|}\sum_{k=1}^{\left|C_j\right|} N_{precis}^{j}\left(R_{jk}\right) \tag{6}$$

Where, R_{jk} is category from rank position k to highest rank. For test sample d_j, $c_i \in C_j$ and $N_{precis}^{j}\left(R_{jk}\right)$ is the number of related categories in R_{jk}. In the index definition above, except for average precision index C_{avep}, small index indicates better classification effect. Best effect: $hl = E_{error} = C_{cover} = C_{rloss} = 0$ and $C_{avep} = 1$.

III. GRPNN EXPECTATION IMPLANTING MODEL

A. RBNN Model Description

Rough set theory is mainly to process imprecise, inconstant and incomplete information. It is a method [4] that can integrates multiple AI technologies. In the algorithm stated in the thesis, as collected pipeline vibration information cannot be directly used for training and prediction of neural network, rough set theory shall pre-process the data.

U is a finite non-null range, I is inseparable and equivalent to U and $I(x)$ is one-variable function of x. Inseparable concept means not all the elements are in the range U and U can be identified by specific information. Further, this inseparable relationship is mainly to define the upper and lower boundary approximation of rough set and is generally expressed as:

$$\overline{I}(x) = \left\{x \in U : I(x) \subseteq X\right\} \tag{7}$$

$$\underline{I}(x) = \left\{x \in U : I(x) \cap X \neq \phi\right\} \tag{8}$$

$$BN_l(x) = \overline{I}(x) - \underline{I}(x) \tag{9}$$

Where, $\overline{I}(x)$ and $\underline{I}(x)$ are upper and lower approximate areas of rough set respectively and $BN_l(x)$ is middle approximate area that connects $\overline{I}(x)$ and $\underline{I}(x)$. Decision rule of date mode can be expressed as:

$$IF\ \alpha\ \ then\ \beta \tag{10}$$

Where, α decision-making condition and β is prediction result. Use optimization algorithm to select upper and lower boundary approximation parameters. Gauss-improved harmony search algorithm would be introduced in the following.

In RPNN, a function block establishes communication with adjacent function block by signal flow, which can be shown by arrows and directions: $\downarrow$ is 0, $\uparrow$ and $\rightarrow$ are 1. As such modularization exists, RPNN can extend into a larger network easily. As shown in Figure 2, RPNN is composed of *m*-row and *n*-line signal-flow function blocks. Signal-flow

function block is $B_{i,j}$, where $i=1,\cdots,m$ and $j=1,\cdots,n$. The first line of function blocks $B_{1,1},\cdots,B_{1,n}$ is called input layer of RPNN and the last line $B_{m,1},\cdots,B_{m,n}$ is the output layer of RPNN. According to different connections of function block input and output, there are four structures as shown in Figure 3. Figure 3a and Figure 3d are internal structures of two inputs and two outputs, Figure 3b is configuration structure of one input and one output and Figure 3c is configuration structure of one input and three outputs.

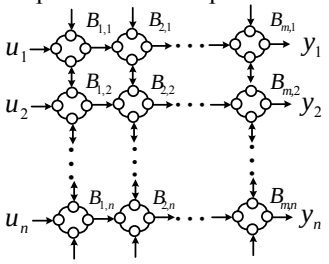

Figure 1. PNN of gauss-mixture model

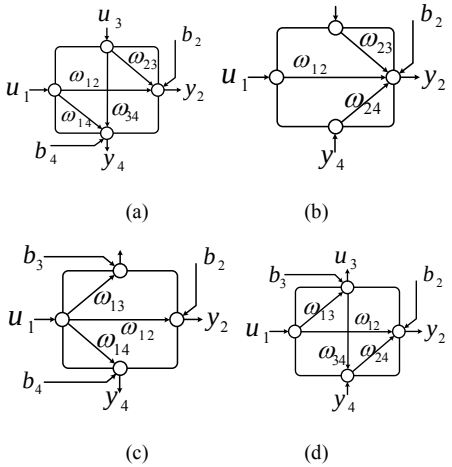

Figure 2. Gaussian kernel structures of different function blocks

In Figure 4d, there are four weight coefficients and two thresholds in two-input and two-output function probability block, which are respectively ω_{13} , ω_{12} , ω_{34} , ω_{24} and b_2 , b_3 . Output of RPNN can be expressed as [8]:

$$y_j=\sum_{i\in I}\omega_{ij}u_i+b_j,\ j\in J \tag{11}$$

Where, u_i is the input data of node *I*, ω_{ij} is the connection weight between node *I* and node *J* and b_j is output threshold. Activation function of each node can be expressed as:

$$\varphi(u)=\alpha\left(\frac{2}{1+e^{-\beta u}}-1\right) \tag{12}$$

Where, α and β are pass parameters.

In conclusion, major parameters involved in PNN of rough set block include: n_l number of hidden layers, network weight ω_{ij} , threshold b_j , pass parameters α and β and boundary parameter of rough set δ . Parameter computation formula can be expressed as:

$$n_{para}=n_l\left(8\times n_{in}-1\right)+4\times\left(n_{in}+1\right) \tag{13}$$

When rough set PNN above is used in multi-label webpage recommendation, as function block of neutral network has no combine and delete mechanisms, modules with small information contribution exist in the network and redundancy of algorithm is too high. So, a method that can dynamically adjust the number of function blocks shall be proposed to improve computation efficiency and computation precision.

B. Parameter Fitting of Gauss Fixture Model

Unknown probability density function (PDF) $p(x)$ of Gauss mixture model can be expressed by the sum of finite Gauss weight:

$$p(x)=\sum_{i=1}^{K}\omega_i f\left(x,\mu_i,\Sigma_i,\varphi_i\right) \tag{14}$$

Where, K is Gaussian kernel number, ω_i is the weight of the i th Gaussian kernel and $\sum_{i=1}^{K}\omega_i=1$. x is the characteristic vector of sample example and dimension is *d*. μ_i,Σ_i are respectively the mean vector and covariance matrix of Gaussian distribution and dimensions are d and $d\times d$. φ_i is perception parameter. Function $f\left(x,\mu_i,\Sigma_i,\varphi_i\right)$ can be expressed as:

$$f\left(x,\mu_i,\Sigma_i,\varphi_i\right)=\frac{1}{\sqrt{2\pi\varphi}}\exp\left(\frac{x'^{\mathrm{T}}\mu'}{\varphi^2}\right) \tag{15}$$

Where, $x'=Ax$, $\mu'=A\mu$, x' and μ' are normalized forms of x and μ respectively and $x'^{\mathrm{T}}x'=\mu'^{\mathrm{T}}\mu'=1$. To reduce complexity to compute Σ^{-1} , generally assume characteristics are mutually independent, which means in covariance matrix Σ , all the elements except for main diagonal elements are null. Computation formula of Σ is:

$$\Sigma=diag\left\{\sigma_1^2,\sigma_2^2,\cdots,\sigma_d^2\right\} \tag{16}$$

Where, $\Sigma^{-1}=AA^T$ and $A=A^T$. For $j=1,\cdots d$ and $i=1,\cdots K$, computation formula of related parameter iteration is:

$$\begin{cases}\mu_{i,j}=\mu_{i,j}+P\left(i|x\right)\left(x_j-\mu_{i,j}\right)\Big/\left(n\omega_i\right)\\ \sigma_{i,j}^2=\sigma_{i,j}^2+P\left(i|x\right)\left[\left(x_j-\mu_{i,j}\right)^2-\sigma_{i,j}^2\right]\Big/\left(n\omega_i\right)\\ \omega_i=\omega_i+\left(P\left(i|x\right)-\omega_i\right)\Big/n\end{cases} \tag{17}$$

Where, computation formula of posterior probability $P(i|x)$ is:

$$P(i|x)=\frac{\omega_i f(x,\mu_i,\Sigma_i,\varphi_i)}{p(x)} \tag{18}$$

Constant n is to control the change rate of parameters entered before. Suppose *count* is the number of training samples, then $n=\min(count,n_{\max})$. r and s are approximate Gaussian kernels. If r accepts training sample x, acceptance domain parameter adjustment process is:

$$\begin{cases}\varphi_{r_{new}}^2=\varphi_r^2+\dfrac{sat_{\varphi_r^2}\left(x'^{\mathrm{T}}\mu_r'-1-\varphi_r^2\right)}{\rho}\\ \varphi_{s_{new}}^2=\varphi_s^2+\dfrac{sat_{\varphi_s^2}\left(\delta^2-\varphi_s^2\right)}{\rho}\end{cases} \tag{19}$$

Where, $sat_h(\cdot)$ is linear saturated transfer function and its value range of its output λ is $-\alpha h\le\lambda\le\alpha h$. Based on Taylor linear expansion (omitted), the computation formula of parameter δ is:

$$\delta=\frac{\varphi_s f_s(x,\mu_s,\Sigma_s,\varphi_s)\left(\varphi_s^2+2D\right)}{\eta\varphi_s^2 f_r\left(x,\mu_r,\Sigma_r,\varphi_{r_{new}}\right)+2D_s f_s(x,\mu_s,\Sigma_s,\varphi_s)} \tag{20}$$

Where, $D_s=x'^{\mathrm{T}}\mu_s'-1$, f_s and f_r are outputs of kernel s and kernel r.

Steps to estimate Gaussian kernel number K:

Step 1: (combine) confirm whether two Gaussian kernels shall be combined. Compare two Gaussian kernels and check whether they are approximate and meet combination condition (threshold). Suppose s' and t' are Gaussian kernels to be combined and r' is new Gaussian kernel after combination. Computation formula of r'-related parameter is:

$$\begin{cases}\omega_{r'}=\omega_{s'}+\omega_{t'}\\ \mu_{r',j}=\left(\omega_{s'}\mu_{s',j}+\omega_{t'}\mu_{t',j}\right)/\omega_{r'}\\ \sigma_{r',j}^2=-\mu_{r',j}^2+\omega_{s'}\left(\sigma_{s',j}^2+\mu_{s'}^2\right)+\omega_{t'}\left(\sigma_{t',j}^2+\mu_{t'}^2\right)/\mu_{t',j}^2\\ \sigma_{r'}^2=\left(\omega_{t'}\omega_{t'}^2+\omega_{s'}\omega_{s'}^2\right)/\omega_{r'}\end{cases} \tag{21}$$

Step 2: (add) confirm whether Gaussian kernel shall be added. If number of network outputs belong to category x exceeds threshold t_1, use formula (17) to update Gaussian kernel or the new kernel u would be added into network. Computation formula of new kernel-related parameter is:

$$\begin{cases}\mu_{u,j}=x_j,\sigma_{u,j}^2=\dfrac{\sum_{i=1}^K\omega_i\sigma_{i,j}^2}{\sum_{i=1}^K\omega_i}\\ \omega_n=\dfrac{1}{n},\varphi_u^2=\dfrac{\sum_{i=1}^K\omega_i\varphi_i^2}{\sum_{i=1}^K\omega_i}\end{cases} \tag{22}$$

To guarantee $\sum_{i=1}^K\omega_i=1$ is satisfied, weight adjustment form of each Gaussian kernel is:

$$\omega_i=\omega_i-1/nK,i=1,\cdots,K,i\ne u \tag{23}$$

Step 3: (delete) define threshold τ_2, when Gaussian kernel weight is lower than the threshold, delete it. It is favorable to remove kernels with small information contribution, reduce network redundancy and improve computation efficiency.

IV. EXPERIMENT ANALYSIS

Algorithm classification experiment was carried out on 11 multi-label data sets [13] from domain name yahoo.com. Contrast algorithms are joint probabilistic multi-label classification algorithm (JPMLC) [8], integrated multi-label recommendation algorithm based on fine grained random graph [14] (IMLR-ML) and Steepest descent RBF neutral network multi-label weight optimization [15] (SDRBF) were selected. Experiment hardware: CPU: i5-760k 3.33GHz; RAM: G ddr3-1600 GHz; test software: Matlab2013a.

Firstly, characteristics of each data set above were selected briefly and dimensions and document number of each data set are reduced on the basis of that. Only about 2% entries and highest-frequency test were selected and others were deleted. Each text composed a vector and each dimension referred to the times (frequency) of a word showing in the text. 2000 samples were trains and 300 samples executed test in each data set and average category number was 30. See table 1 for other parameters.

TABLE 1. EXPERIMENT DATA

Serial No.	Data set	C	T	DC(%)	MNC	RC(%)
1	Arts	26	452	44.48	11	19.23
2	Business	30	443	42.19	10	50.00
3	Computers	33	683	29.58	17	39.39
4	Education	33	553	33.47	7	57.58
5	Entertainment	21	639	29.29	9	28.57
6	Health	32	613	48.07	7	53.13
7	Recreation	22	611	30.18	13	18.18
8	Reference	33	796	13.76	5	51.52
9	Science	40	753	34.75	7	35.00
10	Social	39	1017	20.95	9	56.38
11	Society	27	646	41.87	13	25.93

In Table 1, *C* is category number, *T* is entry number, *DC* is proportion of samples belonging to multiple categories, *MNC* is the number of maximum category distributed to

individual samples and *RC* is percentage of rare category. Firstly, data set was divided into 1500 training sets and 1000 data were used to establish classifier and 500 data were used for test. Adjustment parameters in algorithm include φ_{ini}, τ_1 and τ. Other parameter setting: $n_{\max} \to \infty$, $\rho = 100$, $\alpha = 0.2$, $\eta = 0.8$ and $\tau_2 = 1/300$. These parameters were selected subject to parameters in the single-label sorting. Actually, in multi-label sorting recommendation, it has relatively small impact on algorithm performance. Experiment comparison index was the evaluation index of algorithm stated in Section 1.2. Simulation result is as shown in Figure 3-Figure 7.

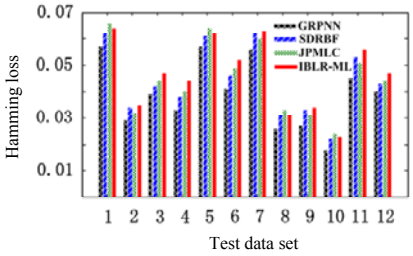

Figure 3. Hamming loss index

Hamming loss indexes of contrast algorithms are shown in Figure 3. Smaller the index refers to better the algorithm. X-coordinates 1-11 in the Figure correspond to serial numbers of data sets in Table 1 (similarly hereinafter). X-coordinate 12 is the average Hamming loss index of data sets above. According to the loss index, GRPNN algorithm is better than the contract algorithms.

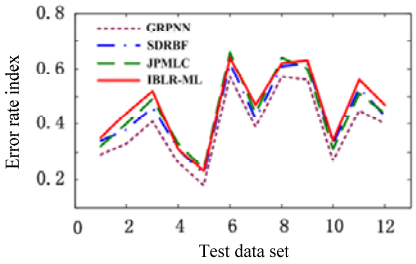

Figure 4. Error rate index

Comparison of Error rate indexes of contrast algorithms in the data sets above is shown in Figure 4. The comparison result shows error rate of GRPNN algorithm stated in the thesis is less than contrast algorithms so it is better than comparison algorithm.

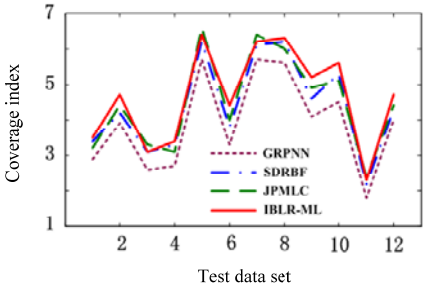

Figure 5. Coverage index

Comparison of coverage indexes of contrast algorithms in data sets above is shown in Figure 5. The coverage here is to evaluate categories that distribute all the possibilities for test example d_j by reducing category rank threshold so smaller the coverage refers to better the algorithm. It can be seen from the algorithm that GRPNN algorithm is better than the contrast algorithms.

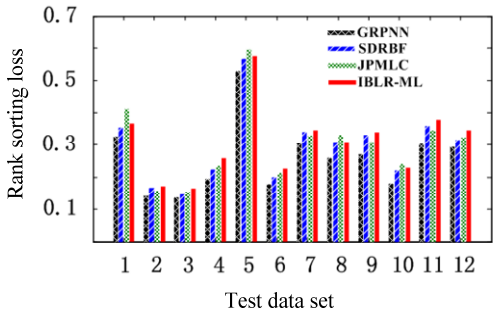

Figure 6. Rank sorting loss index

Comparison of rank sorting indexes of the contrast algorithms in the data sets above is shown in Figure 6. Smaller the index refers to better the algorithm. If can be seen from the Figure that GRPNN algorithm is better than the contrast algorithms.

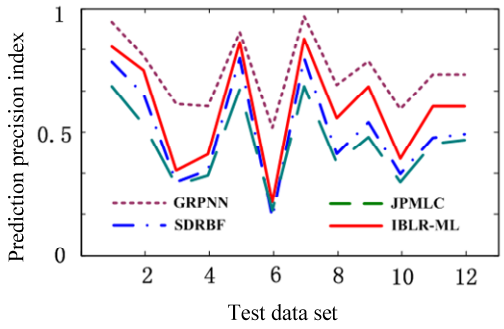

Figure 7. Prediction precision index

Comparison of prediction precision indexes of the contrast algorithms in the data sets above is shown in Figure 7. The comparison result shows prediction precision of GRPNN algorithm is higher than that of contract algorithms so it is better than contrast algorithms.

TABLE 2. COMPARISON OF ALGORITHM OPERATION TIME

Data set	IBLR-ML	SDRBF	JPMLC	GRPNN
Training time (s)				
Yahoo	41.32	38.64	10.19	12.67
Test time (s)				
Yahoo	64.36	55.45	36.59	20.19

Comparison of algorithm operation times is shown in Table 2. The comparison data in the Table shows GRPNN algorithm is better than contrast algorithms in training time and test time, which indicates the outstanding operation efficiency of the algorithm.

V. CONCLUSIONS

A rough set PNN Gaussian block expectation-implanting sorting recommendation algorithm for multi-label webpage is proposed to solve imprecise information and more new information in multi-label recommendation algorithm. Rough set theory is used in the algorithm to effectively process uncertainty of information. Dynamic Gaussian block expectation implantation sorting is adopted to reduce structure redundancy of algorithm and improve computation efficiency. Simulation test for 11 multi-label data sets of yahoo.com domain name shows the algorithm mentioned is better than contrast algorithms in all test indexes and it has faster operating speed and simulation shows effectiveness of the mentioned algorithm.

REFERENCES

[1] Masamoto K, Haizaki N, et al. A Web page of Herbaceous Plants for Searching Plant Species on Plains of Kumamoto City[J]. Memoirs of the Faculty of Education Kumamoto University Natural Science, 2007, 56:53-59.

[2] Zuzana Nevěřilová. Metadata Visualization in Digital Libraries[C]// International Conference on Theory and Practice of Digital Libraries: Research and Advanced Technology for Digital Libraries. Springer-Verlag, 2011:442-445.

[3] Dhingra V, Bhatia K K. Metadata: Towards Machine-Enabled Intelligence[J]. International Journal of Web & Semantic Technology, 2012, 3(3).

[4] Hedstrm U. Automatic Generation of Metadata for Indexing, Searching and Navigating in an Intranet[J]. 2001.

[5] Papoutsidakis M, Sampati S, Ledakis I, et al. Guidelines for Web Search Engines: From Searching and Filtering to Interface[C]// Metadata and Semantics, Post-Proceedings of the, International Conference on Metadata and Semantics Research, Mtsr 2007, Corfu Island in Greece, 1-2 October. 2007:383-392.

[6] Jinyu Hu and Zhiwei Gao. Distinction immune genes of hepatitis-induced heptatocellular carcinoma[J]. Bioinformatics, 2012, 28(24): 3191-3194.

[7] Liu, Y., Yang, J., Meng, Q., Lv, Z., Song, Z., & Gao, Z. (2016). Stereoscopic image quality assessment method based on binocular combination saliency model. Signal Processing, 125, 237-248.

[8] Jinyu Hu, Zhiwei Gao and Weisen Pan. Multiangle Social Network Recommendation Algorithms and Similarity Network Evaluation[J]. Journal of Applied Mathematics, 2013 (2013).

[9] Yishuang Geng, Jin Chen, Ruijun Fu, Guanqun Bao, Kaveh Pahlavan, Enlighten wearable physiological monitoring systems: On-body rf characteristics based human motion classification using a support vector machine, IEEE transactions on mobile computing, 1(1), 1-15, Apr. 2015

[10] Lv, Z., Halawani, A., Feng, S., Ur Réhman, S., & Li, H. (2015). Touch-less interactive augmented reality game on vision-based wearable device. Personal and Ubiquitous Computing, 19(3-4), 551-567.

[11] Jinyu Hu and Zhiwei Gao. Modules identification in gene positive networks of hepatocellular carcinoma using Pearson agglomerative method and Pearson cohesion coupling modularity[J]. Journal of Applied Mathematics, 2012 (2012).

[12] Jiang, D., Ying, X., Han, Y., & Lv, Z. (2016). Collaborative multi-hop routing in cognitive wireless networks. Wireless personal communications, 86(2), 901-923.

[13] Sillence E, Briggs P, Harris P, et al. A framework for understanding trust factors in web-based health advice[J]. International Journal of Human-Computer Studies, 2006, 64(8):697-713.

[14] Donnelli, Robert M. Computer system, method and software for creating and providing an individualized web-based browser interface for wrappering search results and presenting advertising to a user based upon at least one profile or user attribute[J]. 2007.

[15] Laliberte D, Hamilton M. Experimental HTTP methods to support indexing and searching[J]. Journal of Biomolecular Structure & Dynamics, 1984, 1(6):1543-1546.

[16] Kallipolitis L, Karpis V, Karali I. Semantic search in the World News domain using automatically extracted metadata files[J]. Knowledge-Based Systems, 2012, 27(3):38-50.

[17] Alexander C. Thurman. Metadata Standards for Archival Control: An Introduction to EAD and EAC[J]. Cataloging & Classification Quarterly, 2005, 40(3):183-212.

[18] Gokturk S B. System and method for enabling image recognition and searching of remote content on display[J]. 2014.

[19] Lamprianidis G, Pfoser D. Jeocrowd: collaborative searching of user-generated point datasets[C]// ACM Sigspatial International Symposium on Advances in Geographic Information Systems, Acm-Gis 2011, November 1-4, 2011, Chicago, Il, Usa, Proceedings. 2011:509-512.

.