Nowhere to hide: Anonymous User Recognition Based on Multidimensional Trajectory Set

Manfu Ma\textsuperscript{1,2}, Yueshen Lei\textsuperscript{1,2*}, Yong Li\textsuperscript{1,2} and Changqin Wang\textsuperscript{3}

\textsuperscript{1}College of Computer Science and Engineering, Northwest Normal University, Lanzhou, Gansu, 730070, China
\textsuperscript{2}IoT Center of Gansu, Lanzhou, Gansu, 730070, China
\textsuperscript{3}DNSLAB, China Internet Network Information Center, Beijing 100190, China

*Corresponding author’s e-mail: facingworld@nwnu.edu.cn

Abstract. Anonymous user identification is one of the core issues in the field of information security and e-commerce applications. Accurately identifying the implicit characteristics of users' online behavior and the unique identity is important, it has theoretical significance and commercial application value. Based on Massive online clickstream data, this paper proposed AUI algorithms and multidimensional trajectory set model (MTS) to verify the feasibility anonymous user identification. Through the AUI algorithm and vectorization method, a unique multidimensional trajectory set is created for each user. Experiments show that compared with the decision tree and user document methods, the multidimensional trajectory set based on the click data stream has improved the accuracy of recognition and provides an effective solution for user recognition.

1. Introduction
As people become more dependent on the network, the requirements for determining user uniqueness have also increased. Biometrics provides a new solution for traditional identification methods. Each person has unique physiological characteristics such as fingerprints and iris. In addition to these dominant characteristics, everyone also has unique implicit characteristics in behaviour. For example, writers have different writing styles \cite{1}\cite{2}, webpage clicks have unique rules \cite{3}, and users have different habits when using mobile phones to communicate. It is of great significance to research the recognition of implicit features. When a person's dominant features cannot be identified, they need to be identified through implicit features. This research can be used in the areas of customized advertising push \cite{4}\cite{5}, user privacy protection \cite{6}, and network fraud prevention. Therefore, developing user identification for implicit features has positive significance.

At present, research on anonymous network users is mainly realized by users' web browsing information. Data mining and other methods are used to divide anonymous users into user groups with the same characteristics, then mine the key characteristics of this user group and analyse their hobbies \cite{7}\cite{8}, social hotspots \cite{9}, social classes and social relationships \cite{10}, etc.

A Nanda, R Omanwar \cite{11}, etc. proposed a user modelling technique, which implicitly creates a dynamic category interest tree (DCIT). DCIT-based website rankings are usually better than those generated by search engines such as Google. S Jayarathna, APatra et al. \cite{12} researched the importance of combining various implicit and explicit related feedback indicators in a multi-application...
This paper analyses the combination of implicit and semi-explicit relevance feedback and compare it with explicit user ratings.

Most user behaviour researches still pay more attention to the information of the user groups, but ignore the user's accurate identification of the individual. The specific services for the analysis of the user groups may not satisfy the individual user well. In order to be able to perform customized services more accurately, there are also some researches on precise user identification. Y Yang \cite{13} proposed a method to analyse users' web browsing behaviour. This method captures and creates a user document with the strength of the user's behaviour pattern for identifying the user. These user documents can be more accurate than the decision tree in recognition, and can achieve higher efficiency than SVM. Balaji Padmanabhan, Yinghui (Catherine) Yan \cite{14} regarded user identification as an aggregation problem, and then proposed two methods to determine the optimal aggregation level that can accurately predict users. The predicted classifier can uniquely identify individuals. A Jamak, A Savatic \cite{15} and others extracted these features from the text. It is proved that by analysing the number of each essay, the principal component analysis can be used to track the identity of the author.

To sum up, user identification for implicit characteristics has been widely concerned. From the user's web browsing history to the author's unique signature, these implicit characteristics often contain user’s information, and this information can be used as a basis for accurately identifying individuals. When the sample is too large, the accuracy of decision tree is not high. The method based on the multidimensional trajectory set can identify the behaviour patterns of online users, avoiding inaccurate recognition caused by large samples. In previous research, scholars often pay attention to the webpage click behaviour of online users, but rarely pay attention to user’s software click behaviour. This paper uses the implicit characteristics in the click data to complete the following results:

- It verified the unique characteristics of the online users' clickstream data, and answered whether the user's implicit characteristics could be used as the basis for user identification.
- An anonymous user identification method based on user multidimensional trajectory set (MTS) is proposed. Anonymous users can be identified by the user's click on the computer's record.

2. User Behaviour Data

This paper uses the online behavior log data provided by CNNIC as the research object, and thousands of volunteers distributed across the country have installed data collection client programs on personal computers. In the premise of ensuring the user's personal privacy, record User power on / off time, window process name, browser address (partially truncated), program version number of the focus window, program company name, User location information and occupation information (data link: http://www.datatang.com/data/43910). Table 1 is the user occupation classification, and Table 2 is the definition of data parameters.

| Table 1. user categories. | category | category | category |
|--------------------------|----------|----------|----------|
| Industry and service workers | Other | Farmer, Migrant workers |
| Government leader | Company leader | Student |
| Government employee | Company employee | Professional skill worker |
| Self-employed | Retired, Unemployed |

| Table 2. the data parameters. | Parameter | Parameter definition |
|---------------------------|----------|----------------------|
| Last | Interval between shutdown time and startup time |
| L_Start | Startup time |
| t | Interval between the current click time and the start time |
| p | Process name |
| u | URL name |
Each time the user powers on the computer, a corresponding log file is created. The client program scans the focus window at the forefront of the user's computer display every two seconds. If the focus window has changed from two seconds ago, an entry is immediately added to the log.

Due to the variety of software and websites involved by users, the data collection client program may not be able to completely and accurately collect information, so invalid and missing values in the record files are eliminated.

3. AUI Algorithm

The research focus of this paper is to find a multidimensional trajectory set that can capture user behaviour patterns. The high frequency of a user's click behaviour is helpful for user identification, and these special clicks will be recorded in the user's behaviour pattern. In clickstream data, it can be expressed as clicking on certain specific software or web pages (such as 360.exe and baidu.com).

Frequent trajectory set refers to the more frequently occurring set or its subset, and this set consisting of several items. Frequent trajectory sets can already show some features of users. In order to better distinguish users, it is necessary to extract anonymous user association rules from many frequent trajectory sets \[16\]. This paper proposes an improved AUI (Anonymous User Identification) algorithm based on association rules. The algorithm is described as follows:

- Detect abnormal data and eliminate it before generating association rules, it could reduce the time caused by scanning unnecessary data multiple times.
- For the problem that the candidate set \(C_i\) needs to scan the data set multiple times, it is improved to scan \(L_{k+1}\) only once, and on the basis of ensuring accuracy, reduce the number of scanned data sets to save time cost.

**Property 1** (transcendental theorem): If an item set is frequent, then its subset must also be frequent. If an item set is infrequent, all its supersets are also infrequent \[17\].

In the classic algorithm Apriori, after generating a \(C_{k+1}\) candidate set, the entire data set needs to be re-scanned to determine whether the superset generated by \(L_k\) is a frequent item set. As the volume of the data set increases, the efficiency of the Apriori algorithm will become lower and lower \[17\]. The impermanence of the user in operating the computer will be separated from his own behavior because of external factors, such as click on infrequently used websites and handle infrequently used actions.

When analyzing user behavior patterns, this kind of data is useless data, which will increase the time cost. According to property 1, if an item \(a\notin L_1\), it means that \(a\) is not frequent, and \(L_k\) must not contain item \(a\). First calculate the frequent item set \(C_1\) according to formula 1 in the AUI algorithm:

\[
C_1 \rightarrow \sum_{i=1}^{n} a_i (a_i \in D)
\]  

(1)

Iterate through each click information in the data. If an item does not appear in \(C_1\), then add the click information to \(C_1\). Then level one association rule \(L_1\) is generated from frequent item set \(C_1\) according to formula 2,3:

\[
\text{num}_{di} = \text{num}_{di} + 1 (C_{i1} \in d_i)
\]  

(2)

\[
L_1 = L_1 \cup d_i (\text{num}_{di} \geq \text{min Support})
\]  

(3)

For all records \(d_i (d_i \in \text{data})\), if \(C_{i1}(C_{i1} \in C_1)\) is a part of \(d_i\), increase the item's count \(\text{num}_{di}\) \((\text{num}_{di} \geq 0)\), and remove the set that does not satisfy the minimum support. Finally, re-screen the data set \(D\) according to formula 4:

\[
D \rightarrow \sum_{i=1}^{n} d_i (d_i \in L_1)
\]  

(4)

Delete elements in dataset \(D\) that are not in \(L_1\). According to property 1, filtered data will not contain item sets that cannot be the associated rules, which will reduce the time cost of scanning the data set.

**Property 2**: Assume the length of the item set in the dataset be \(l = \{i_1, i_2, \ldots, i_{k+1}\}\). When generating \(L_{k+1}\), if \(l_i\) is greater than \(k + 1\), it is not necessary to scan the item set.

When \(l_i\) is less than \(k + 1\), the candidate set must not be a subset of the set. For example, the candidate
set \{1,2,3,4\} cannot be a subset of the item set \{1,2,3\}. Filter the dataset by formulas 5, 6:

\[
I = \text{len}(d_1) \cup \text{len}(d_2) \cup \ldots \cup \text{len}(d_n)
\]

\[
D = C_0 d_i (I_i > k + 1)
\]

According to property 2, first generate an item set length list \(I\) for the data, and delete the item set with \(I_i\) greater than \(k + 1\) before the candidate set \(C_{k+1}\) scans the data set, thereby reducing the time cost of scanning the data set.

In the original Apriori algorithm, when the candidate set \(C_{k+1}\) generates \(L_{k+1}\), all item sets of the dataset need to be scanned for each candidate to check whether it is a subset of the item set, if it is, the count is increased. But when the candidate set \(C_k\) and the data set are too large, it will inevitably affect the efficiency of the algorithm. According to property 1, it can be known that the subset of frequent item sets must also be frequent item sets. If the subsets of \(C_k\) is less than \(k\) in \(L_{k-1}\), it means that there must be infrequent item sets, and the candidate Item sets must also be infrequent. In this paper, \(L_k\) is calculated by formulas 7, 8:

\[
\text{numl}_i = \text{numl}_i + 1 (C_{ki} \subseteq l_i) \quad (7)
\]

\[
L_k = L_k \cup l_i (\text{numl}_i = k) \quad (8)
\]

After generating \(C_k\), scan \(L_{k-1}\) once. If any item set \(l_i\) in \(L_{k-1}\) is a subset of any item set \(C_{ki}\) in \(C_k\), then add the \(\text{numl}_i\) count which is corresponding to \(C_k\). Finally, the invalid item sets with less numbers than \(k\) in \(C_k\) are deleted.

---

**Algorithm 1:** AUI algorithm.

**input:** \(U_i=\{u_1, u_2, \ldots, u_n\}, P_i=\{p_1, p_2, \ldots, p_n\}\), Support, Confidence

**output:** Anonymous user association rules

1: For each User;

2: \(D_i = U_i + P_i\)

3: \(l = \text{len}(<\text{User}>\))

4: \(x_1 = \text{Find the collection of "1 frequent sets"}

Delete elements in the dataset that are not in \(L_1\)

5: While (\(\text{len}(L[k-2]) > 0\))

For i in data:

if \(I_i < k+1\):

Delete \(D_i\)

For tran in \(C_k\):

For can in \(L_{k-1}\):

if canissubset(tran):

\(\text{count} = \text{count} + 1\)

Delete the elements of count<\(k\) in the dataset

For tran in \(C_k\):

If \(\text{Support}(\text{tran}) < \text{minSupport}\):

delete

……

\(x_k = \text{Find the collection of k-frequent\)

6: For i in \(x_k\):

For m, n in i:
confidence(m, n) = \frac{\text{Support}(m, n)}{\text{Support}(m)}

\text{Support}(m) = \sum_{i=1}^{n} f_i(x, y)

confidence(n, m) = \frac{\text{Support}(m, n)}{\text{Support}(n)}

\text{Support}(n) = \sum_{j=1}^{m} f_j(x, y)

If confidence(m, n) ≥ confidence:

If confidence(m, n) ≥ confidence:

R.append

7: Return R = \{r_1, r_2, \ldots, r_n\}

4. Maximum Entropy Classification

The maximum entropy was proposed by E T Jaynes in 1957. If only part of the information is clear, the selected probability distribution must satisfy these information and have the largest entropy value, which is in the case of partial information to find the most random probability distribution. In the field of natural language processing, the anonymous ID maximum entropy algorithm [18] can be classified as a statistical problem. In this paper, the maximum entropy (MaxEnt-AnonymousId) is used for classification. The feature function is used to represent some known information, and the feature function is expressed as formula 9:

\[ f_{w,y'}(x, y) = \begin{cases} \text{num}(x, w) & y = y' \\ 0 & y \neq y' \end{cases} \quad (9) \]

\text{Num} (x, w) \text{ represents the number of times the word } w \text{ appears in document } x. \text{ The formula 10 used to predict the text is as follows:}

\[ P_w(y|x) = \frac{\exp\left(\sum_{i=1}^{n} w_i f_i(x, y)\right)}{\sum_{y} \exp\left(\sum_{i=1}^{n} w_i f_i(x, y)\right)} \quad (10) \]

W is a word, x is a text, y is a category, and f is a feature function. For a certain text x, the category y with the largest \( P_w(y|x) \) is the desired classification.

\textbf{Algorithm 2: MaxEnt-AnonymousId}

\textbf{input:} \( F = \{f_1, f_2, \ldots, f_n\} \)

\textbf{output:} Similar categories

1: For each \( F = \{f_1, f_2, \ldots, f_n\} \):

2: Feature matrix = \text{Num}_{\text{word}} \times \text{Num}_{\text{category}}

3: similarity = \frac{\exp\left(\sum_{i=1}^{n} w_i f_i(x, y)\right)}{\sum_{y} \exp\left(\sum_{i=1}^{n} w_i f_i(x, y)\right)}

4: list.append(similarity)

5: MaxSimilar = max(list)

6: Similar category = User corresponding to MaxSimilar

5. MTS Model

5.1. User Multidimensional Trajectory Set

This article builds an MTS model (Multidimensional Track Set). The user record data is divided into a training set and a test set (this part of the data only contains click stream and time information, does not contain information that can identify the user explicitly). Both the training set and the test set will generate a multidimensional trajectory set containing their respective click information. Table 3 is the definition of the parameters in the trajectory set.
Table 3. defines the data parameters.

| Parameter   | Parameter definition  |
|-------------|-----------------------|
| D={d_1,d_2,…,d_n} | Multi-parameter set |
| U_i={u_1,u_2,…,u_n} | Single user’s web pages information |
| P_i={p_1,p_2,…,p_n} | Single user’s software click information |
| R_i={r_1,r_2,…,r_n} | Regular information |
| T={t_1,t_2,…,t_n} | Time information for a single user |
| F={f_1,f_2,…,f_n} | User multidimensional track set |

Extract user webpage information U_i = {u_1, u_2, ..., u_n} and software clicks information P_i = {p_1, p_2, ..., p_n} from a user’s clickstream data D = {d_1, d_2, ..., d_n} (d is a collection of multiple parameters that the user contains). The webpage clicks information U_i and software click information P_i are processed by the AUI algorithm to obtain the user’s high-frequency click information and the regular information R_i = {r_1, r_2, ... r_n}.

Table 4. T parameter definition.

| Parameter   | Parameter definition  |
|-------------|-----------------------|
| AM_time     | Morning duration      |
| PM_time     | Afternoon duration    |
| AM_start    | Morning boot time     |
| PM_start    | Afternoon boot time   |
| AM_close    | Morning shutdown time |
| PM_close    | Afternoon shutdown time |

Table 4 is the definition of time parameters. Extract the time information of a single user T_i = {t_1, t_2, ..., t_n} for auxiliary identification, where t contains “AM_time”, “PM_time”, “AM_start”, “PM_start”, “AM_close”, “PM_close”. After averaging these parameters, they are combined into a vector and recorded in t. In addition, the training set needs to add the ID information of the known user, including id number and occupation type information. Finally, the user multidimensional trajectory set F_i = {f_1, f_2, ..., f_n} is established. Figure 1 shows the establishment of a user's multidimensional trajectory set:

5.2. MTS Process

This article adds weight to the user's clicked keywords, while reducing the weight of keywords with low recognition [19]. Adding weights can better organize cluttered data to make it easier to highlight the implicit characteristics. The click behavior of each user in the data is described as a multi-track set vector containing implicit characteristics. Use cosine similarity [20] to match the test set and the training set.

Establishing the user's multidimensional trajectory set stage: Establish f_train for the training set.
and \( f_{test} \) for the test set. Figure 2 shows the matching process:

![Figure 2. Matching process.](image)

**Classification stage**: The MaxEnt-AnonymousId algorithm has proven to be an effective classifier, which can better handle the classification of natural languages. In this paper, the first step is to divide the data into multiple categories. Each category will extract the parameters "U_i" and "P_i" of all users under this class, and then merge them into a category label C after extracting regular information through the AUI algorithm. In the second step, each anonymous user will be matched the most similar class based on the class tag. In the third step, the MaxEnt-AnonymousId algorithm will iteratively select a better way to match until the result is finally stable and no major change. The final output is passed to the next user match stage. It can be found in experiments that individuals in the click data stream have unique behavior characteristics, and user groups also have unique behavior characteristics. For example, students will use more entertainment software such as Thunder.

**User matching stage**: Calculate the similarity of all the \( f_{train} \) and \( f_{test} \) in the classification through the cosine similarity. Calculation formula 11 is as follows:

\[
F_{Similarity} = \frac{\sum_{i=1}^{n}(f_{test_i} \times f_{train_i})}{\sqrt{\sum_{i=1}^{n}(f_{test_i})^2 \times \sum_{i=1}^{n}(f_{train_i})^2}}
\]  

(11)

5.3. **Compensation Matching**

This paper also proposes a method of compensation matching to reduce experimental errors. When each anonymous user uses the multidimensional trajectory set \( f_{test} \) to match the training set, select users with higher matching degree and record it as \( M_1, M_2, ..., M_n \). Calculate the variance \( V \) of the selected user's matching degree. The calculation formula 12 is as follows:

\[
V = \frac{\sum_{i=1}^{n} (M_i - M_{avg})}{n}
\]  

(12)

If \( V \) is large, it indicates that the user is highly distinguishable in recognition, and the recognition result can be determined. If \( V < \gamma \) (0 < \( \gamma < 1 \)), the difference between \( M_1, M_2, ..., M_n \) is small, indicating that the results matched by anonymous users are similar, then compensation matching by the \( T_i \) parameters. The compensation matching formula 13 is as follows:

\[
T_{Similarity} = \frac{\sum_{i=1}^{n}(T_{test_i} \times T_{train_i})}{\sqrt{\sum_{i=1}^{n}(T_{test_i})^2 \times \sum_{i=1}^{n}(T_{train_i})^2}}
\]  

(13)

Similarly, the cosine similarity is used to compare the \( T_{test} \) and \( T_{train} \), the user with the highest match is selected as the final result of the match.
6. Experiments and Results
The experiment set minSupport to 0.5, 0.7, 0.9, and compared the running time of association rules under different data volume (10 days, 15 days, 20 days, 25 days and 30 days). Figure 3(a), 3(b), 3(c) shows the running time of generating frequent item sets under different support and data volume:

![Graph showing running time for different supports](image)

Figure 3. AUI algorithm running time.

The x axis in Figure 3 represents different data volume (days), and the y axis represents the running time (ms). It can be seen from Figure 3 that the AUI algorithm is more efficient than the traditional Apriori algorithm in generating frequent item sets under different support levels and different data volumes. Especially in the click data stream, user may have some data that doesn't fit their behavior patterns, this caused a lot of invalid data, AUI algorithm can be very good to overcome this problem, it will be removed at the beginning of the data set to reducing the time cost.

In the experiment, select 200 users in clickstream data as research objects. Each user contains 30 files recording how to use the computer, and extract 8 parameters from each file. In order to better discover the user behavior pattern in the parameter $U_i$, only keep the main site of the whole page. Each user's monthly record is divided into training set and test set, the data in the training set will construct a multidimensional trajectory set $F_{train}$, and the data in the test set will construct a multidimensional trajectory set $F_{test}$. In the recognition phase, choice of 8 user volume (25, 50, 75, 100, 125, 150, 175, 200) to observe the impact of different user volume on recognition accuracy.

Research has found that extracting different features can affect the accuracy of identifying anonymous users. Figure 4 compares the relationship between different parameters and accuracy:

![Graph showing relationship between parameters and accuracy](image)

Figure 4. Relationship between different parameters and accuracy.

The x axis in Figure 4 represents different user volume, and the y axis represents the improvement in accuracy (%) under different parameters. It can be seen that the experimental results of the combined parameter $U_iP_i$ are better than the single parameter $U_i$ or $P_i$ under each volume. Different parameters can more or less show different behavior between anonymous users, but the combination of different parameters can better reflect the user's implicit characteristics to improve recognition, and the relative accuracy of using $U_iP_i$ increases as the user volume increases, indicating that as the volume of users increases, the combination of multiple parameters can better reflect the implicit characteristics of users. On this basis, in order to improve accuracy and identify users more accurately, the compensation
matching parameter $T_i$ is added in the experiment to increase the advantage of multi-parameters. Table 5 compares the relationship between different parameters and accuracy after adding $T_i$:

Table 5. Influence of different parameters on accuracy.

| Parameter | 25   | 50   | 75   | 100  | 125  | 150  | 175  | 200  |
|-----------|------|------|------|------|------|------|------|------|
| $U_iT_i$  | 81.20| 80.19| 78.20| 77.32| 75.12| 73.20| 71.35| 69.30|
| $P_iT_i$  | 95.10| 94.23| 92.91| 89.73| 90.51| 89.90| 88.32| 86.51|
| $U_iP_iT_i$| 96.20| 95.92| 93.81| 92.30| 91.32| 90.12| 88.80| 88.40|

The x axis in Table 5 represents different user volume, and the y axis represents the influence of different parameters on the recognition degree. The data in the table is accuracy (%). It can be seen that the multi-parameter "$U_iP_iT_i$" performs the best in different volume, indicating that the combination of multi-parameters can better identify anonymous users and has higher recognizability.

In this paper, a multidimensional trajectory set (MTS) is compared with the user documents method (SP)[14] and C4.5 decision tree[21]. The C4.5 decision tree classifier has been proved to be an efficient and accurate classifier, and it can also be well solved for a large number of classification problems. Table 6 shows the accuracy between different methods:

Table 6. Accuracy between different methods.

| Parameter | 25   | 50   | 75   | 100  |
|-----------|------|------|------|------|
| MTS       | 96.20| 95.92| 93.81| 92.30|
| SP        | 90.90| 90.12| 89.49| 87.36|
| C4.5      | 87.98| 83.56| 83.36| 81.01|

The x axis in Table 6 represents different user volume, and the y axis represents different methods. The data in the table is the accuracy (%). It can be seen that in different user volume, the MTS methods have higher accuracy, which is an average improvement of 5% over the SP method, and an average improvement of 10.6% over the C4.5 decision tree. It provides a feasible and simple alternative for the user recognition problem.

7. Conclusions

This paper studies the user's clickstream data as a basis for user identification. Unlike the user's web browsing data, the user's software click data has stronger recognition and more implicit characteristics. In the recognition phase, the average accuracy is improved by 15% compared to web browsing data. This paper proposes an improved AUI algorithm based on association rules. This algorithm has good time efficiency for irregular data. At the same time, a method for identifying anonymous users (MTS) is proposed. Experiments show that the recognition method based on multidimensional trajectory sets improves the accuracy of user recognition. Compared with methods based on decision trees and user documents, it has certain advantages.

More accurate positioning can not only enable merchants to find market positioning and development direction more accurately, but also allow users to contact the advertisement that better fits their needs.

In this experiment, some users are randomly selected in the data for the experiment. In the next work, we will continue to expand the number of users to observe the recognition effect. At the same time, it will explore the recognition of anonymous users by other user features, and find more accurate methods to identify implicit characteristics to improve the matching accuracy.

Acknowledgments

This research was partially supported by the grants from the Natural Science Foundation of China (No.71764025, 61863032, 61662070); the Research Project on Educational Science Planning of Gansu Province, China(Grant No.GS[2018]GHBBKZ021 and GS[2018]GHBBKW007); the Scientific Research Foundation of the Higher Education Department of Gansu Province, China(Grant No.2018A-001); Author contributions: Manfu Ma and Yueshen Lei are co-first authors who jointly designed the
research. Correspondence and requests for materials should be addressed to Yong Li (facingworld@nwnu.edu.cn)

References
[1] Abdi, M. N., & Khemakhem, M. (2015). A model-based approach to offline text-independent Arabic writer identification and verification. Pattern Recognition, 48(5), 1890-1903.
[2] He, S., & Schomaker, L. (2017). Writer identification using curvature-free features. Pattern Recognition, 63, 451-464.
[3] Boratto, L., Carta, S., Fenu, G., & Saia, R. (2016). Using neural word embeddings to model user behavior and detect user segments. Knowledge-based systems, 108, 5-14.
[4] Huang, J., Zhu, K., & Zhong, N. (2016). A probabilistic inference model for recommender systems. Applied Intelligence, 45(3), 686-694.
[5] Fei Zhang, Libo Zhang, Tiejian Luo, & Yanjun Wu. (2018). A feature-based collaborative clustering model. Journal of Computer Research and Development, 55(7), 1508-1524.
[6] Chon, B. S., Lee, J. K., Jeong, H., Park, J., & Park, J. (2018). Determinants of the Intention to Protect Personal Information among Facebook Users. ETRI Journal, 40(1), 146-155.
[7] Bao, J., Zheng, Y., & Mokbel, M. F. (2012, November). Location-based and preference-aware recommendation using sparse geo-social networking data. In Proceedings of the 20th international conference on advances in geographic information systems (pp. 199-208).
[8] Cena, F., Likavec, S., Lombardi, I., & Picardi, C. (2016). Should i stay or should i go? Improving event recommendation in the social web. Interacting with Computers, 28(1), 55-72.
[9] Zehuan Peng, Le Sun, Xianpei Han, & Chen Bo. (2015). Community Hot Weibo Recommendation Research. Journal of Computer Research and Development, 52(5), 1014-1021.
[10] Xin Li, Guiquan Liu, Lin Li, Zongda Wu, & Ding Junmei. (2017). Recommendation algorithm based on social relationship mining in interest circles on LBSN. Journal of Computer Research and Development, 54(2), 394-404.
[11] Nanda, A., Omanwar, R., & Deshpande, B. (2014, August). Implicitly learning a user interest profile for personalization of web search using collaborative filtering. In 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) (Vol. 2, pp. 54-62).
[12] J Jayaratnha, S., Patra, A., & Shipman, F. (2015, June). Unified relevance feedback for multi-application user interest modeling. In Proceedings of the 15th ACM/IEEE-CS Joint Conference on Digital Libraries (pp. 129-138).
[13] Yang, Y. C. (2010). Web user behavioral profiling for user identification. Decision Support Systems, 49(3), 261-271.
[14] Padmanabhan, B., & Yang, Y. C. (2007). Clickprints on the web: Are there signatures in web browsing data?. Available at SSRN 931057.
[15] Can, M. (2012). Teaching Neural Networks to Detect the Authors of Texts Using Lexical Descriptors. Southeast Europe Journal of Soft Computing, 1(1).
[16] Libo Zhang, Yihan Sun, & Tiejian Luo. (2017). A method for calculating semantic similarity based on large-scale knowledge base. Journal of Computer Research and Development, 54(11), 2576-2585.
[17] Harrington, P. (2012). Machine learning in action. Manning Publications Co.
[18] Nigam, K., Lafferty, J., & McCallum, A. (1999, August). Using maximum entropy for text classification. In IJCAI-99 workshop on machine learning for information filtering (Vol. 1, No. 1, pp. 61-67).
[19] Hong, T. P., Lin, C. W., Yang, K. T., & Wang, S. L. (2013). Using TF-IDF to hide sensitive itemsets. Applied Intelligence, 38(4), 502-510.
[20] Kadhim, A. I., Cheah, Y. N., Ahamed, N. H., & Salman, L. A. (2014, December). Feature extraction for co-occurrence-based cosine similarity score of text documents. In 2014 IEEE Student Conference on Research and Development (pp. 1-4). IEEE.
[21] Yimu Ji, Yongpan Zhang, Xianbo Lang, Dianchao Zhang, & Wang Ruchuan. (2017). Parallelization of stream data-oriented decision tree classification algorithm. Journal of Computer Research and Development, 54(9), 1945-1957.