METHOD AND SYSTEM FOR PROCESSING RECOMMENDED TARGET SOFTWARE

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] The present application claims the priority and benefit of Chinese Patent Application No. 201210207154.9, entitled "Method and System for Processing Recommended Target Software", filed on June 21, 2012, the disclosure of which is incorporated herein in its entirety by reference.

TECHNICAL FIELD

[0002] The present application relates to software processing technologies of data processing devices, and especially to a method and system for processing recommended target software.

BACKGROUND

[0003] Nowadays, with increasing popularity of data processing devices such as computers, intelligent mobile phones, palm PCs, tablet PCs, etc., all kinds of software that runs on the data processing devices are booming.

[0004] In these data processing devices, a main technical solution for managing software is using a software management system. A software management system is software that manages software on a data processing device. In the industry, currently commonly used software managers for example are 360 software steward, Kingsoft software steward, etc. Currently, the software managers mainly have functions of: software repository, software upgrade, software uninstallation, download management, etc.

[0005] Usually, a software repository in a software management system integrates most of excellent commonly used software in the industry for users to select to install and upgrade. To provide users with more abundant and more comprehensive software installation and upgrade, currently a software repository supports more and more software, covering various categories such as instant messaging, audio-video playing, webpage browsing, input method, etc., and the total number of the software is up to thousands and even to tens of thousands.

Among so much software, the quality of the software may be good or bad. For a user, he tends to hope that a software management system can recommend software with a high quality and more targeted for him. Therefore, currently most of the software management systems are integrated with a function of software recommendation.

[0006] In traditional art, a software management system recommends software based on the number of times of being downloaded or hotness of the software and software that has been installed. A way or channel of recommendation may include: pushing hot target software by using a rank list in a software repository, a recommendation page of special topics in a software repository, a tooltip, etc.

[0007] However, for existing solutions of recommending target software, relevance between recommended software and a specific user is low, which results in that recommendation may not be that accurate, and target software that meets usage features of a user may not be accurately recommended.

SUMMARY

[0008] In view of this, a method and a system for processing recommended software are disclosed to improve relevancy between recommended target software and a specific user, and improve accuracy of recommendation.

[0009] The technical solution is implemented as follows.

[0010] A method for processing recommended target software includes:

[0011] performing a cluster analysis for users based on software using information reported by the users to determine a software list corresponding to each user cluster, and sorting software in the software list according to using condition of the software; and

[0012] determining a user cluster that is the most relevant to a specific user based on software using information of the specific user; and selecting top N pieces of software from a software list corresponding to the user cluster to recommend the selected top N pieces of software to the specific user, where N is a predefined value.

[0013] A system for processing recommended target software includes:

[0014] a cluster analysis module configured to perform a cluster analysis for users based on software using information reported by the users to determine a software list corresponding to each user cluster, and sort software in the software list according to using condition of the software; and

[0015] a recommendation module configured to determine a user cluster that is the most relevant to a specific user based on software using information of the specific user; and select top N pieces of software from a software list corresponding to the user cluster to recommend the selected top N pieces of software to the specific user, where N is a predefined value.

[0016] Compared to that in the traditional art, in the present disclosure, software using information of users is fully considered, and based on this, a cluster analysis is performed for the users, different software lists are recommended to different user clusters; during use of a specific user, a user cluster (equivalent to the using feature type of a user) to which the specific user belongs is determined, and software that is in the front rank of a software list corresponding to the specific user is recommended. In this way, the recommended target software according to the present disclosure may have a high relevancy with the specific user, and the accuracy of recommendation is relatively high.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] For a better understanding of the present disclosure, reference should be made to the Detailed Description below, in conjunction with the following drawings in which like reference numerals refer to corresponding parts throughout the figures.

[0018] FIG.1 is a schematic diagram illustrating a flow of a method for processing recommended target software according to an embodiment of the present disclosure;

[0019] FIG.2 is a schematic diagram illustrating a structure of a system for processing recommended target software according to an embodiment of the present disclosure;

[0020] FIG.3 is a schematic diagram illustrating another structure of the system for processing recommended target software according to an embodiment of the present disclosure;

[0021] FIG.4a is a schematic diagram illustrating a cluster analysis module and a recommendation module being set at a server;

[0022] FIG.4b is a schematic diagram illustrating the cluster analysis module being set at the server and the recommendation module being set at a client;

[0023] FIG.5a is a schematic diagram illustrating an interface of showing recommended target software by an icon;

[0024] FIG.5b is a schematic diagram illustrating an interface of showing the recommended target software by an icon after the recommended target software is clicked on;

[0025] FIG.5c is a schematic diagram illustrating an interface of showing the recommended target software by an icon after the recommended target software is downloaded and installed;

[0026] FIG.5d is a schematic diagram illustrating an interface of showing edit mode of recommended target software by an icon;

[0027] FIG.6a is a schematic diagram illustrating another structure of the system for processing recommended target software according to an embodiment of the present disclosure; and

[0028] FIG.6b is a schematic diagram illustrating another structure of the system for processing recommended target software according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0029] The present disclosure will be further described in detail hereafter in combination with the accompanying drawings and detailed embodiments.

[0030] Fig.1 is a schematic diagram illustrating a flow of a method for processing recommended target software according to the present disclosure. Referring to Fig.1, the

method mainly includes the following process.

[0031] At block 101, a cluster analysis is performed for users based on software using information reported by the users, a software list corresponding to each user cluster is determined, and software in the software list is sorted according to using condition of the software.

[0032] At block 102, when a specific user uses a software management system, a user cluster that is the most relevant to the specific user is determined according to software using information of the specific user. From a software list corresponding to the user cluster, top N pieces of software is selected and is recommended to the specific user, where N is a predefined value.

[0033] Fig.2 is a schematic diagram illustrating a structure of a system for processing recommended target software according to the present disclosure. The processing system is configured to implement the method according to the present disclosure. Referring to Fig.2, the processing system includes a cluster analysis module and a recommendation module.

[0034] The cluster analysis module is configured to perform a cluster analysis for users based on software using information reported by the users, determine a software list corresponding to each user cluster, and sort software in the software list according to a using condition of the software.

[0035] The recommendation module is configured to determine a user cluster which is the most relevant to a specific user according to software using information of the specific user; and select top N pieces of software from a software list corresponding to the user cluster and recommend them to the specific user, where N is a predefined value.

[0036] At block 101, the software using information reported by the users refers to software using information that is reported by total users. The total users do not mean all users, but only sample users who upload their software using information. If the number of the sample users reaches a certain lower number limit (the lower number limit may be set according to requirements of an application), then it is determined that these sample users can represent all the users.

[0037] The sample users refer to users who have a right of reporting data. In a detailed embodiment of the present disclosure, a request of whether or not to agree to report client data is sent to every user of the software management system in advance, and if a user accepts the request, then the user turns into a sample user and has the right of reporting data; or if the user refuses the request, then the user does not has the right of reporting data.

[0038] To be able to collect software using information reported by users, in the processing system according to the present disclosure, a data collection module is included. As shown in Fig.3, the data collection module is set on clients of the software management system, and is configured to collect software using information of the users who have the right of reporting data, and report the software using information to the cluster analysis module and the recommendation module.

[0039] The software using information reported by a user at block 101 includes: software information of software that has been installed on a user client device locally, e.g., information such as software names and/or software categories. The information may be obtained by the data collection module through searching according to information in regedit and in a configuration file on the client device, such as paths and executable files of software that has been installed.

[0040] The cluster analysis process at block 101 will be described in the following.

[0041] Cluster analysis is also called group analysis, which is a multi-level statistical analysis method that classifies sample data according to the principle of "things of one kind come together". The processing object of the cluster analysis is a great amount of sample data, which requires reasonable classification according to features of the sample data.

[0042] The cluster analysis process is a process that classifies the sample data into different clusters. Therefore, objects in a same cluster have a relatively large similarity, while objects in different clusters have a relatively large diversity. The object of the cluster analysis is to perform classification for the sample data based on the similarity.

[0043] In the present disclosure, based on the collected software using information of the total users, features of different user groups are obtained by performing the cluster analysis,

users whose software using features are similar are clustered into a user cluster, and a typical feature of the users in the user cluster is extracted as a criterion for subsequent determination of a cluster to which a specific user belongs.

[0044] In the embodiments of the present disclosure, existing mature cluster analysis methods may be used, for example, hierarchical clustering, partition method, joining method, dynamic clustering, orderly prototype-based clustering, overlapping clustering, fuzzy clustering, etc.

[0045] The cluster analysis module of the present disclosure performs the cluster analysis for the user groups according to the software names and/or category information of the software in installed software lists, clusters the user groups into user groups (user clusters) having different software using features, and selects a software list corresponding to each user cluster.

[0046] The direct data basis of the cluster analysis module is categories of installed software reported by respective sample users. If a sample user only reports a software name, then a software category to which the software belongs can be inquired from the software depository according to the software name, and the software category is used as the direct data basis for the cluster analysis.

[0047] At block 101, the detailed cluster process of the cluster analysis module performing the cluster analysis for the users may vary slightly according to different recommendation requirements. For example, a process of performing the cluster analysis for the users based on the software using information reported by the users according to an embodiment may include the following process.

[0048] At block 111, the amount of respective kinds of installed software that is reported by respective sample users is counted.

[0049] At block 112, categories of software that has been installed the most by the respective sample users are determined.

[0050] At block 113, the sample users are partitioned by using the categories of the

software that has been installed the most as similarity to obtain user clusters of different software using features.

[0051] For example, for n sample users a1~an, a category of software that is installed the most is video. If n is larger than a cluster threshold, then a video user cluster is partitioned. A software using feature of the video user cluster is: video software being installed the most.

[0052] As another example, for m sample users b1~bm, a category of software that is installed the most is game. If m is larger than a cluster threshold, then a game user cluster is partitioned. A software using feature of the game user cluster is: game software being installed the most.

[0053] At block 101, after the cluster analysis is performed for the users, a software list corresponding to each use cluster needs to be determined. The detailed process may be as follows. For a user cluster that is partitioned according to the cluster analysis, software of a category corresponding to it is selected from the software repository of the software management system to constitute a software list, and the software in the software list is sorted according to their software using condition. The amount of the software in the software to be recommended list is also controllable and may be set according to service requirements.

[0054] The software in the software list being sorted according to their software using condition mainly means that the software is sorted according to their overall running hotness. Specifically, the process may include the following process. The number of downloads of the respective software and the number of times of being started and the running time of the respective software on user client devices reported by sample users are collected. For each piece of software, the number of downloads, the number of times of being started, and the running time of the software are respectively multiplied by corresponding weights, and are then summed to obtain the overall running hotness of the software. And the overall running hotness of the software is taken as the using condition of the software. The software in the software list is sorted according to their overall running hotness.

[0055] For example, for the video user cluster, k pieces of video software A1~Ak are selected to constitute a corresponding software to be recommended list, and the k pieces of video software is sorted according to their overall running hotness.

[0056] For the game user cluster, 1 piece of game software B1~B1 is selected to constitute corresponding a corresponding software to be recommended list, and the 1 piece of game software is sorted according to its overall running hotness.

[0057] In the present embodiment, the cluster analysis module is set at the server side. The block 101 is executed by the server.

[0058] At the block 102, when a specific user uses the software management system, usually whether the specific user having a right of reporting data or not needs to be determined, and there are two situations.

[0059] The first situation is as follows. If the specific user has the right of reporting data, then the client of the specific user obtains software using information at the client device, and reports the software using information to the server. The server determines a user cluster that is the most relevant to the specific user according to the software using information reported by the user client, and selects top N pieces of software from a software list corresponding to the user cluster and sends them to the client of the specific user. The client of the specific user shows the top N pieces of software as recommended software. For example, icons of recommended software may be shown on a Qdesk, and light gray is used to identify the software being recommended software.

[0060] The software using information mainly includes information of software that has been installed and/or information of software that is frequently used on the client device. The information of the software that has been installed on the client device may be categories and/or names of the installed software, and it may be obtained by the data collection module through searching according to information in regedit and in a configuration file on the client device, such as paths and executable files of software that has been installed. The information of the software that is frequently used by the client device mainly includes categories and/or names of the software that is frequently used. The software that is frequently used generally is software on a quick launch bar, on a desk, and/or on a DOCK bar, and may be obtained by the data collection module through an application programming interface (API) corresponding to an operating system on the client device.

[0061] In addition, if the specific user has the right of reporting data, then the method

further includes the following process. The client of the specific user sends the using information of the recommended software to the server, and the using information of the recommended software includes at least one piece of the following information: the number of installations of the software, time from starting the software to stopping the software, and the number of secondary start-ups after the software is installed. The information is mainly obtained by the data collection module by counting click operations of the specific user and length of running time. After the server receives the using information of the recommended software, it determines the recommendation hotness of the recommended software. A specific determination method may be respectively multiplying the number of installations of the software, the time from starting the software to stopping the software, and the number of secondary start-ups after the software is installed by weights, and summing them to obtain the recommendation hotness. Then the rank of the recommended software in a software list of a corresponding user cluster is adjusted according to the hotness of the recommended software. For recommended software whose recommendation hotness is larger than a specified threshold, the rank of it is raised, and for recommended software whose recommendation hotness is smaller than the specified threshold, the rank of it is lowered. By this kind of adjustment, a selection dimension of recommending software for users is further added, which further improves the relevancy between target software and users, and improves the accuracy of recommendation.

[0062] In an embodiment of the system for processing recommended target software according to the present disclosure, to implement the method in the first situation, the cluster analysis module and the recommendation module need to be set at the server, as shown in Fig.4a.

[0063] In the second situation, if the specific user has no right of reporting data, then the server sends information of each user cluster and a sorted software list corresponding to the user cluster to the client of the specific user. The client of the specific user obtains the software using information on the client device. The client of the specific user determines a user cluster which is the most relevant to the specific user according to the software using information on the client device, selects top N pieces of software in a software list corresponding to the user cluster, and shows them as recommended software.

[0064] Similar to that in the first situation, the software using information mainly includes information of software that has been installed on the client device and/or information of software that is frequently used by the client device. The information of the software that has been installed on the client device may be categories and/or names of the installed software, and it may be obtained by the data collection module through searching according to information in regedit and in a configuration file on the client device, such as paths and executable files of software that has been installed. The information of the software that is frequently used by the client device mainly includes categories and/or names of the software that is frequently used. The software that is frequently used generally is software on a quick launch bar, on a desk, and/or on a DOCK bar, and may be obtained by the data collection module through an API corresponding to an operating system on the client device.

[0065] In an embodiment of the system for processing recommended target software according to the present disclosure, to implement the method in the second situation, the cluster analysis module needs to be set at the server, and the recommendation module needs to be set at the client, as shown in Fig.4b. The cluster analysis module is further configured to send information of each user cluster and a sorted software list corresponding to each user cluster to the client. The recommendation module is further configured to obtain the software using information of the specific user on the client device from the data collection module on the client device, determine a user cluster which is the most relevant to the specific user according to the software using information of the specific user on the client device, select top N pieces of software in a software list corresponding to the user cluster, and show them as recommended software.

[0066] In an embodiment of the present invention, the software using information of the specific user in block 102 includes information of software that has been installed on the client device and/or information of software that is frequently started by the client device.

[0067] Determining the user cluster which is the most relevant to the user according to the software using information of the specific user specifically includes the following process.

[0068] At block 121, in a software list corresponding to each user cluster, a cluster relevancy degree between each piece of software and the user cluster is determined according to a rank of the software.

[0069] At block 122, according to software lists corresponding to respective user clusters and the cluster relevancy degrees of software in the software lists, user clusters to which the respective software that has been installed and/or the respective software that is frequently started on the user client device belongs are determined, and cluster relevancy degrees between the respective software that has been installed and/or the respective software that is frequently started and the user clusters to which they belong are determined.

[0070] At block 123, cluster relevancy degrees of software that belongs to a same user cluster are added together to obtain a matching score of the user cluster. The process here may further include the following process. A higher weight is allocated to a piece of software which is frequently started than that is allocated to a piece of software that has been installed, and the cluster relevance degrees of the software that belongs to the same user cluster are added together according to software weights, and a weighted sum is used as the matching score of the user cluster.

[0071] For example, if the software that has been installed by the user includes game software X, game software Y, and office software Z, then two relevant user clusters are found, namely, a user cluster of game and a user cluster of office. Then it is found that a relevancy degree of game software X in the user cluster of game is x, a relevancy degree of game software Y in the user cluster of game is y, and a relevancy degree of office software Z in the user cluster of office is z. Then a matching score of the user cluster of game is x+y, and a matching score of the user cluster of office is z.

[0072] At block 124, a user cluster which has the highest matching score is selected to be the user cluster which is the most relevant to the user.

[0073] After the user cluster which is the most relevant to the specific user is determined, e.g., the user cluster of game being determined to be the cluster which is the most relevant to the user, top N pieces of software in a software list corresponding to the user cluster of game is recommended to the specific user.

[0074] In another embodiment of the present disclosure, in step 102, selecting top N pieces of software from the software list corresponding to the user cluster may specifically include the following process. Software that has been installed by the specific user is

removed from a list of software to be recommended of the user cluster, and top N pieces of software is selected from remaining software and is recommended to the specific user. In this way, the recommendation will not be interfered with by the software that has been installed by the user and the accuracy of the recommendation is further improved.

[0075] In a solution, the number of recommended software is N=1. That is, for a specific user, only one piece of software is recommended. In this way, difficulty in selection for the user caused by recommending too much software to the user at one time may be avoided.

[0076] It will be appreciated by those skilled in the art that the modules of the system in the above embodiments may be distributed in the apparatuses as described in the embodiments, or may be changed to be located in one or more apparatuses other than those in the embodiment. The modules in the embodiments may be integrated into one module, or may be further separated into multiple sub-modules.

[0077] In fact, the system for processing recommended target software according to the embodiments of the present disclosure may be implemented in various ways. For example, it may be compiled into a plug-in installed into a software management system in compliance with application interfaces of a certain standard, or may be encapsulated into an application for users to download and use. When it is compiled into a plug-in, it may be implemented by ocx, dll, cab, etc. Or the system for processing recommended target software proposed by the embodiments of the present disclosure may be implemented by specific technic such as a Flash plug-in, a RealPlay plug-in, a MMS plug-in, a MIDI stave plug-in, an ActiveX plug-in, etc.

[0078] Based on the technical solutions disclosed in the above respective embodiments, a computer readable storage medium is further disclosed here. The computer readable storage medium is configured to store instructions that enable a device to execute the method for processing recommended target software disclosed herein. Specifically, a system or apparatus provided with a storage medium may be provided, software program codes that implement functions of any of the above embodiments are stored on the storage medium, and a computer (or CPU or MPU) of the system or apparatus may read and execute the software program codes stored on the storage medium.

[0079] In this situation, since the program codes per se read from the storage medium may implement functions in any instance of the above embodiments, the program codes and the storage medium storing the program codes constitute a part of the mapping management technical solution.

[0080] Embodiments of the storage medium that stores the program codes include floppy disk, hard disk, magnetic disk, disc (e.g., CD-ROM, CD-R, CD-RW, DVD-ROM, DVD-RAM, DVD-RW, DVD+RW), magnetic tape, non-volatile memory and ROM. Alternatively, the program codes may be downloaded from a server computer via a communication network.

[0081] In addition, it should be understood that a part of or all actual operations may be carried out not only by executing the program codes read by a computer, but also by an operating system on a computer according to instructions based on program codes, so as to realize the functions of any of forgoing embodiments.

[0082] For example, Fig.6a is a schematic diagram illustrating another structure of the electronic map distance measurement apparatus according to an embodiment of the present disclosure. As shown in Fig.6a, the system may include a server and a client. The server includes a first memory and a first processor that communicates with the first memory, in which the first memory stores a cluster analysis instruction and a recommendation instruction executable by the first processor.

[0083] The cluster analysis instruction is configured to instruct performing a cluster analysis for users based on software using information reported by clients of the users, determining a software list corresponding to each user cluster, and sorting software in the software list according to using condition of the software.

[0084] The recommendation instruction is configured to instructing determining a user cluster which is the most relevant to a specific user according to software using information of a client of the specific user; and selecting first N pieces of software from a software list corresponding to the user cluster and recommending them to the specific user, where N is a predefined value.

[0085] Accordingly, the client may include a second memory and a second processor that

communicates with the second memory. The second memory stores data a collection instruction that is executable for the second processor and is configured to instruct collecting software using information of users who have a right of reporting data, and reporting data to the server.

[0086] Fig.6b is a schematic diagram illustrating another structure of the electronic map distance measurement apparatus according to an embodiment of the present disclosure. As shown in Fig.6b, the system may include a server and a client. The server includes a first memory and a first processor that communicates with the first memory, in which the first memory stores a cluster analysis instruction that is executable for the first processor.

[0087] The cluster analysis instruction is configured to instruct performing a cluster analysis for users based on software using information reported by clients of the user, determining a software list corresponding to each user cluster, sorting software in the software list according to using condition of the software, and sending information of each user cluster and a sorted software list corresponding to each user cluster to the client.

[0088] The client may include a second memory and a second processor that communicates with the second memory. The second memory stores a data collection instruction and a recommendation instruction that are executable for the second processor.

[0089] The data collection instruction is configured to instruct collecting software using information of users who have a right of reporting data, and reporting the software using information to the server; or instruct collecting software using information of users who have no a right of reporting data.

[0090] The recommendation instruction is configured to instruct obtaining software using information of a specific user on a client device by using data collection instruction on the client device, and instruct determining a user cluster that is the most relevant to the specific user according to the software using information of the client of the specific user; and selecting top N pieces of software from a software list corresponding to the user cluster and showing them as recommended software, where N is a predefined value.

[0091] As described in the above, the technical solution in the embodiments of the

present disclosure may infer a user's preference for using software through a series of processes, and by using the technical solution to recommend software, accuracy of matching between software recommendation and user requirements may be improved, and some relatively outstanding but not well known software may be easily recommended to related users as well.

[0092] There may be many ways to show recommended target software on the client. For example, in an embodiment of the present disclosure, recommended target software is shown by way of icons, and by clicking to execute installation, the overall recommendation flow becomes relatively ethereal and smooth, which simplifies the flow to the largest extent and degrades a user's mental pressure.

[0093] Fig.5a is a schematic diagram illustrating an interface of showing recommended target software by way of icon. Referring to Fig.5a, recommended software may be shown by an icon different from other icons, e.g., by a light gray icon 500 (other icons are in deep color) in an operation panel of a software management system, e.g., an ethereal software start-up operation panel named "Qdesk"; a frequency at which the target software is pushed may be configured at backstage, e.g., a basic frequency being one piece of software one week.

[0094] Fig.5b is a schematic diagram illustrating an interface of showing the recommended target software by an icon after the recommended target software is clicked on. Referring to Fig.5b, when the light gray icon 500 is clicked on, a download and installation procedure is triggered; after the download is completed, the software is silently installed; after the installation is completed, the icon is lightened (i.e., removing the light gray), which indicates that the software can be normally used.

[0095] Fig.5c is a schematic diagram illustrating an interface of showing the recommended target software by an icon after the recommended target software is downloaded and installed. As shown in Fig.5c, after the installation is completed, the software icon 500 is lightened, which can be used as other icons in the "Qdesk" panel.

[0096] If the user does not click on the light gray icon 500, then the light icon 500 may be removed in two ways as follows:

[0097] 1) Common deletion operation: after edit mode is entered, a minus on the top right is clicked on to delete the light gray icon. Fig.5d is a schematic diagram illustrating an interface of showing edit mode of recommended target software by an icon.

[0098] 2) When next software is pushed, if it is detected that the previous light gray icon has not been installed yet, the original icon is replaced automatically.

[0099] False situations may be dealt with in the following ways.

[00100] If a network is not connected normally, a floating window will pop up to prompt "network connection is abnormal, download fails", and when the user clicks on the light gray icon 500 again, downloading the target software is re-triggered.

[00101] If after the target software is downloaded, the installation of it fails, then the light gray icon 500 is kept to be light gray; and a flowing window pops up to prompt "installation fails, please try again", and when the user clicks on the light icon again, silent installation is re-triggered.

[00102] As shown in the Fig.5a to Fig.5d, when software products are recommended in this kind of icon management software panel, it more meets the expectation of users. In this way, the utility of software may be fulfilled, and the feature of software recommendation may be expanded based on icon management.

[00103] It should be understood that the foregoing is only embodiments of the present disclosure and is not intended to limit the present disclosure. Any modification, equivalent substitution, and improvement without departing from the spirit and principle of this disclosure should be covered in the protection scope of the disclosure.

WHAT IS CLAIMED IS:

1. A method for processing recommended target software, comprising:

performing a cluster analysis for users based on software using information reported by the users to determine a software list corresponding to each user cluster, and sorting software in the software list according to using condition of the software; and

determining a user cluster that is the most relevant to a specific user based on software using information of the specific user; and selecting top N pieces of software from a software list corresponding to the user cluster and recommending the selected top N pieces of software to the specific user, where N is a predefined value.

2. The method according to claim 1, wherein the software using information reported by the users comprises: software information of software that has been installed on user client devices; and

performing the cluster analysis for the users based on the software using information reported by the users comprises:

counting the amount of respective categories of installed software that is reported by respective sample users;

determining categories of software that has been installed the most by the respective sample users; and

performing cluster partition for the sample users by using the categories of the software that has been installed the most as similarity to obtain user clusters of different software using features.

3. The method according to claim 1, wherein sorting the software in the software list according to the using condition of the software comprises:

collecting the number of downloads of the respective software and the number of times of being started and running time of the respective software on user client devices reported by sample users;

for each piece of software, respectively multiplying the number of downloads, the number of times of being started, and running time of the software by corresponding weights, and making a sum to obtain overall running hotness of the software, the overall running hotness of the software being the using condition of the software; and

sorting the software in the software list according to overall running hotness of the software.

4. The method according to claim 1, wherein the software using information of the specific user comprises: information of software that has been installed and/or information of software that is frequently started on a client device of the specific user; and

determining the user cluster that is the most relevant to the specific user based on the software using information of the specific user comprises:

in a software list corresponding to each user cluster, determining a cluster relevancy degree between each piece of software and the user cluster according to a rank of the software;

determining user clusters to which the respective software that has been installed and/or the respective software that is frequently started on the client device of the specific user belongs according to software lists corresponding to respective user clusters and cluster relevancy degrees of software in the software lists, and determining cluster relevancy degrees between the respective software that has been installed and/or the respective software that is frequently started and the user clusters to which they belong;

adding cluster relevancy degrees of software that belongs to a same user cluster to obtain a matching score of the user cluster; and

selecting a user cluster that has a highest matching score to be the user cluster that is the most relevant to the user.

5. The method according to claim 4, wherein adding the cluster relevancy degrees of the software that belongs to the same user cluster to obtain the matching score of the user cluster comprises:

allocating a higher weight to a piece of software that is frequently started than a weight allocated to a piece of software that has been installed, adding the

cluster relevance degrees of the software that belongs to the same user cluster according to software weights, and using a weighted sum as the matching score of the user cluster.

6. The method according to claim 1, wherein

performing the cluster analysis for the users based on the software using information reported by the users to determine the software list corresponding to each user cluster, and sorting the software in the software list according to the using condition of the software are executed by a server;

the specific user is a user who has a right of reporting data; and

determining the user cluster that is the most relevant to the specific user based on the software using information of the specific user, and selecting the top N pieces of software from the software list corresponding to the user cluster to recommend the selected top N pieces of software to the specific user, comprise:

a client device of the specific user obtaining the software using information of a client device and uploading the software using information to the server;

the server determining the user cluster that is the most relevant to the specific user based on the software using information reported by the client of the specific user, sending the top N pieces of software selected from the software list corresponding to the user cluster to the client of the specific user; and

the client of the specific user showing the top N pieces of software as recommended software.

7. The method according to claim 6, wherein after the client of the specific user showing the top N pieces of software as the recommended software, the method further comprises:

the client of the specific user sending using information of the recommended software to the server;

the server determining recommendation hotness of the recommended software according to the using information of the recommended software;

adjusting ranks of the recommended software in the software list corresponding to the user cluster according to the recommendation hotness of the recommended

software, raising a rank of recommended software whose recommendation hotness is larger than a specific threshold, and lowering a rank of recommended software whose recommendation hotness is smaller than the specific threshold.

8. The method according to claim 7, wherein the using information of the recommended software comprises at least one of: the number of installations of the software, time from starting the software to ending the software, and the number of secondary start-ups after the software is installed.

9. The method according to claim 1, wherein

performing the cluster analysis for the users based on the software using information reported by the total users to determine the software list corresponding to each user cluster, and sorting the software in the software list according to the using condition of the software are executed by a server;

the specific user is a user who has no right of reporting data;

determining the user cluster that is the most relevant to the specific user based on the software using information of the specific user, and selecting the top N pieces of software from the software list corresponding to the user cluster to recommend the selected top N pieces of software to the specific user comprise:

the server sending information of each user cluster and a sorted software list corresponding to each user cluster to a client of the specific user;

the client of the specific user obtaining software using information on a device of the client; and

the client of the specific user determining the user cluster that is the most relevant to the specific user according to the software using information on the device of the client, selecting the top N pieces of software from the software list corresponding to the user cluster as recommended software, and showing the recommended software.

10. The method according to any of claims 1 to 9, wherein selecting the top N pieces of software from the software list corresponding to the user cluster and recommending the selected top N pieces of software to the specific user comprise:

removing software that the specific user has installed from a list of software to be recommended of the user cluster, and selecting top N pieces of software from remaining software to recommend the selected top N pieces of software to the specific user.

11. The method according to claim 10, wherein N=1.

12. A system for processing recommended target software, comprising:

a cluster analysis module configured to perform a cluster analysis for users based on software using information reported by the users to determine a software list corresponding to each user cluster, and sort software in the software list according to using condition of the software; and

a recommendation module configured to determine a user cluster that is the most relevant to a specific user based on software using information of the specific user; and select top N pieces of software from a software list corresponding to the user cluster to recommend the selected top N pieces of software to the specific user, where N is a predefined value.

13. The system according to claim 12, wherein the system further comprises: a data collection module, which is set at a client, configured to collect software using information of a user who has a right of reporting data, and report the collected software using information to the cluster analysis module and the recommendation module.

14. The system according to claim 13, wherein the cluster analysis module and the recommendation module are set at a server.

15. The system according to claim 13, wherein

the cluster analysis module is set at a server, and the recommendation module is set at the client;

the cluster analysis module is further configured to send information of each user cluster and a sorted software list corresponding to each user cluster to the client; and

the recommendation module is further configured to obtain the software using information of the specific user of a client device from the data collection module on the client device, determine the user cluster that is the most relevant to the specific user according to the software using information of the specific user, select the top N pieces of software from the software list corresponding to the user cluster as recommended software, and show the recommended software.

16. The system according to claim 12, wherein

the software using information reported by the users comprises: software information of software that has been installed on client devices of the users; and

the software using information of the specific user comprises: software information of software that has been installed on a client device of the specific user, and/or software information of software that is frequently started.

17. A system for processing recommended software, comprising a server and a client, wherein the server comprises a first memory and a first processor that communicates with the first memory, and the first memory stores a cluster analysis instruction and a recommendation instruction executable by the first processor;

the cluster analysis instruction is configured to perform a cluster analysis for users based on software using information reported by the users to determine a software list corresponding to each user cluster, and sort software in the software list according to using condition of the software; and

the recommendation instruction is configured to determine a user cluster that is the most relevant to a specific user based on software using information of the specific user; and select top N pieces of software from a software list corresponding to the user cluster to recommend the selected top N pieces of software to the specific user, where N is a predefined value.

18. The system according to claim 17, wherein the client comprises a second memory

and a second processor that communicates with the second memory, wherein the second memory stores a data collection instruction that is executable for the second processor, and configured to collect software using information of a user having a right of reporting data and report the software using information to the server.

19. A system for recommending target software, comprising: a server and a client, wherein the server comprises a first memory and a processor that communicates with the first memory, wherein the first memory stores a cluster analysis instruction that is executable for the first processor; the client comprises a second memory and a second processor that communicates with the second memory, wherein the second memory stores a data collection instruction and a recommended instruction that are executable for the second processor;

the cluster analysis instruction is configured to instruct performing a cluster analysis for users based on software using information reported by the users to determine a software list corresponding to each user cluster, sorting software in the software list according to using condition of the software, and sending information of each user cluster and the sorted software list corresponding to each user cluster to the client;

the data collection instruction is configured to instruct collecting software using information of a user who has a right of reporting data, and report the collected software using information to the server; or instruct collecting software using information of a user who has no right of reporting data; and

the recommendation instruction is configured to instruct obtaining the software using information of the specific user of a client device based on the data collection instruction of the client device, and instruct determining a user cluster that is the most relevant to the specific user based on the software using information of the client of the specific user; selecting top N pieces of software from the software list corresponding to the user cluster as recommended software and showing the recommended software, where N is a predefined value.

20. A computer readable storage medium, storing a computer program, wherein the computer program is configured to execute the method for processing recommended software according to any of claims 1 to 11.