

US PATENT & TRADEMARK OFFICE

PATENT APPLICATION FULL TEXT AND IMAGE DATABASE



(1 of 1)

United States Patent Application**20140317079****Kind Code****A1****Obernikhin; Vitaly ; et al.****October 23, 2014**

PERSONNEL RECRUTMENT SYSTEM USING FUZZY CRITERIA

Abstract

A data search system provides access to information about potential job candidates, in particular, their professional skills, work experience, education, professional activity and other aspects, which are of importance to a recruiter. The invention can be employed by recruiting agencies, HR departments, independent experts and other staffing agencies, as well as managers looking for new employees. Since there is usually no direct links between different profiles of the candidate, the system makes some distributed profiles "fuzzy" (i.e., unverified). These "fuzzy" profiles are also displayed during the search, and the user of the system can choose the needed profile from the database.

Inventors: **Obernikhin; Vitaly;** (*Moscow, RU*) ; **Dmitrichenko; Sergey;** (*Moscow, RU*) ; **Galyetov; Ilya;** (*Moscow, RU*)

Applicant: **Name** **City** **State** **Country** **Type**

Amazing Hiring, Inc. Redwood City CA US

Assignee: **Amazing Hiring, Inc.**
Redwood City
CA

Family ID: **51729813**Appl. No.: **14/255659**Filed: **April 17, 2014****Current U.S. Class:****707/706 ; 707/722****Current CPC Class:**

G06F 16/2468 20190101

Class at Publication:**707/706 ; 707/722****International Class:**

G06F 17/30 20060101 G06F017/30

Foreign Application Data

Date	Code	Application Number
Apr 18, 2013	EA	201300375

Claims

1. An automated personnel recruitment system, comprising: a data storage; a database management system; at least one user's workstation; an automated search unit; an Internet data-collection unit configured to collect candidates' profile data from the Internet sources; a data-analyzing unit configured to analyze the candidates' profile data; a profile-comparing unit; an Internet data cache; and a cache data-extracting unit, wherein: the data storage contains verified candidates' profile data; the user's workstation is connected to the automated search unit and configured to receive search query parameters provided to the automated search unit; the database management system is configured to provide a search query input to the database and extract candidate-related data from the database according to the search query parameters; the data cache is configured to store the candidate-related data and group it according to candidates' identification data and network resources, from which the information was cached; the automated search unit is configured to search for data according to the query parameters entered by the user at the workstation and extract the candidate's profile data from the database according to the given criteria; the cache data-extracting unit is configured to search for entries corresponding to a search criteria and compare Internet profiles with the extracted database profiles; the profile-comparing unit is configured to transfer cached data to the database, if the information is verified and corresponds to a real candidate's profile; and the profile-comparing unit is configured to store the data in the cache providing the database with a link to the cache entry, if the data is not verified and the correspondence is fuzzy.
2. The system of claim 1, wherein the automated data-collecting unit connects to a network resource using identification data of a user of the resource.
3. The system of claim 1, wherein the data cache stores data for a limited time and the database includes a storage unit for entries containing data transferred from the cache that have not been verified as corresponding to a profile.
4. The system of claim 3, wherein the data is linked to at least one profile and can be called through search queries, while the completed queries inform the user that unverified data was used during the search.
5. The system of claim 3, wherein the data is transferred from the cache to the database, if it is verified by a user with administrator rights.
6. The system of claim 5, wherein at least one user has a second administrator profile with the right to confirm the data correspondence to the database profiles.
7. The system of claim 1, wherein the search query is used to rank entries according to a level of correspondence of the candidate's data to the search query parameters.
8. The system of claim 7, wherein the correspondence to the search query includes correspondence of the candidate's qualification and wherein the candidates are ranked by their qualification.
9. The system of claim 8, wherein the qualification of the candidate includes a) his/her career progression based on the information in a resume or in a profile on a professional social network; (b) proficiency in a particular technology of companies the candidate worked at; (d) "online reputation" of a candidate.
10. The system of claim 9, wherein the qualification of the candidate further includes (e) for software developers there might be even available source code they have written for open-source projects, and (f) peer-review of the source code.

[0004] 2. Description of the Related Art

[0005] Recruiters have two main problems when they are looking for candidates:

[0006] a. There are not enough candidates

[0007] b. It is hard to assess candidates before an interview: [0008] i. Qualification of a candidate [0009] ii. Probability of a particular candidate to be hired by a particular hiring manager for reasons not directly related to qualification but rather related to "culture fit".

[0010] To find enough candidates, recruiters use different resources:

[0011] A conventional method of searching for candidates is to search through so-called "job boards," where "active" candidates (i.e., those, who are currently looking for job) upload their CVs (resumes) providing a detailed account of their professional skills, education, certifications, work experience at different companies and references. The problem of this conventional method is that the highly qualified candidates usually change their job "by invitation," so that they have no need to write up a CV and post it on a job website. Thus, this conventional method can only be used to look for "active" candidates, which is not sufficient.

[0012] Many candidates have their profiles/accounts on various professional social networks and their own websites. The candidates can also join professional discussions on Internet forums and, in case with programmers, participate in open-source projects. In order to find job candidates, recruiters use various job-seeking services (e.g., dice.com, monster.com, superjob.ru, headhunter.ru, etc.), professional social networks (LinkedIn, Xing, Viadeo, Zerply, Moikrug.ru,), recreational social networks (Facebook, Google Plus, Myspace, VKontakte, Twitter, Tumblr,), Q&A forums (rdsn.ru, stackoverflow.com, quora.com, expert-exchange.com), open-source software repositories (e.g., Github, Google Code, Bitbucket, SourceForge, Launchpad, Redmine).

[0013] Nonetheless, it is often difficult to find a suitable candidate, since they might have no LinkedIn profile at all, or their profile might provide only partial data (e.g., a Github profile may contain only the name of the candidate and a list of projects they are participating in, and a LinkedIn profile may provide only the name and the city of the candidate, but no skills description or projects they have participated in).

[0014] As discussed above, a known method for looking for "passive" candidates using professional social networks (e.g., LinkedIn, Xing, Viadeo, Zerply, Moikrug.ru, etc.) is inefficient. The main problem of this method is that the search yields only the data that the candidate has filled in their profile. Highly qualified "passive" candidates usually have no need to give a full account of their achievements, skills and work experience in their respective field. The reasons are the same as for the lack of their CVs on the job boards. For example, highly qualified software developers can be distinguished mainly by their participation in open-source software development projects or activity in giving answers to difficult problems posted on professional forums dedicated to software development (e.g., Stackoverflow).

[0015] Yet another known method is to search for information about potential job candidates (including "passive" ones) on different Internet resources. The problem of this known method is that each individual profile/account of the candidate might miss specific data needed to identify a qualified candidate, such as their professional skills, work experience, projects they have participated in, and/or contacts. Another problem is that the search might yield different profiles of the same candidate, but from different resources that may have been viewed earlier. Thus, it increases the effort needed staff recruiters, who have to look through various resources and spend much more time checking the same candidates, than if they were monitoring only one resource.

[0016] Accordingly, there is a need in the art for a system, which creates a single "distributed profile" for the candidate, which would combine all candidates' profiles/accounts on different network resources and supply the missing information for each individual profile.

[0017] Assessing candidates before an interview is hard to do, and leads to very broad "recruitment funnels"--

first circle of the user of the given resource. The data cache stores data for a limited time.

[0056] Furthermore, the data can be transferred from the cache to the database, if it is verified by a user with administrator's rights (and at least one user of the system should be identified as having such rights, while at least one user of the system may have a second administrator profile with the right to confirm data correspondence to database profiles). The search query can rank entries according to the level of correspondence of the candidate's data to the search query parameters. The search query can also calculate a degree of correspondence (e.g., as a weight coefficient, based on the number of matches of key criteria). Also, the system can rank search results according to the probability of a candidate responding to a job offer by the recruiter.

[0057] The probability calculation is based on secondary data regarding a desire of the candidate to change his or her current job. The probability can also be calculated based on the candidate's job search activity. Particularly, the probability of the candidate changing their job is based on their network (forum) activity fluctuations, and/or financial indexes of their current employer, and/or information about possible cuts of the workforce by their current employer, and/or update statistics of the candidate's profile in professional social networks, and other relevant data.

[0058] When searching for candidate's profiles/accounts on the Internet resources, the system can employ general search engines, such as Google, Yandex, etc. When the search query result is displayed, the candidates can be ranked according to the number and quality of their published works, such as professional publications and patents in the areas corresponding to the skills and the keywords passed as search parameters. To this end, publishing databases can be used to provide additional data for candidates' profiles.

[0059] The quality of the papers can be accessed through external metrics--e.g., "citation index" (how many papers including patents cited or referenced a particular paper--http://en.wikipedia.org/wiki/Citation_index), reputation of scientific journals where the paper was published etc.

[0060] Also, the candidates can be ranked according to the level of correspondence of their profiles to the profile of a potential hiring manager/supervisor, because if the candidate has similar education and experiences, there is a higher chance of him working successfully with the supervisor. Additionally, the supervisor will be more inclined to interview such candidate and offer him a job. The level of correspondence to the potential supervisor (or a hiring manager) is estimated by the system, which checks whether both the candidate and the supervisor graduated from the same educational institutions and/or worked in the same enterprise (regardless of time periods), participated in the same open-source software development projects, etc.

[0061] The same approach can be implemented to compare candidates' profiles with those of current employees by creating a "generalized" profile of enterprise workers. The "generalized profile" can be created automatically by analyzing profiles of the team which has the opening. For instance, if 80% of the team graduated from the same educational institution, say, MIT, then the generalized profile will have "MIT" as education. Likewise, if, say, 92% of the team worked at Google in the past, then the generalized profile will have "Google" as a previous employer.

[0062] Another example: suppose, that a Job Description says C++ is a required skill, upon analyzing team profiles one can figure out that 83% of the team in addition to C++ know Assembler. As a result, both: C++ and Assembler will be set as "skills" in the generalized profile. The profiles can be included into the database in order to improve the search quality. The generalized "personal profile" of a typical employee can be created using information about the age of employees, educational institutions they attended, enterprises they worked at, music and messages posted in social networks they "like."

[0063] The search query parameters can be based on a real profile, and the system can look for candidates corresponding to that profile and rank them accordingly, while the user of the system can mark more/less desirable points to be considered during the search. The level of correspondence in this case can be estimated based on one or several parameters, such as similar professional skills, education, work experience, published works, similar professional interests, similar things posted and "liked" in social networks.

[0083] account IDs for various resources.

[0084] Candidates' profiles often do not contain links to other profiles on other resources, which makes it difficult to locate the candidate registered on different resources.

[0085] The exemplary embodiment provides a method for creating a "distributed profile" to facilitate candidate search. The system allows to create a distributed profile using several profiles/accounts, if:

[0086] the candidate's individual profiles/accounts are filled in different languages (including Russian, English, German, French, Spanish, Ukrainian, Byelorussian, etc.)

[0087] individual profiles/accounts contain transliterated elements (e.g., a name of the university);

[0088] individual profiles/accounts contain various English and French spelling of the name (e.g., Alexander Shirokov and Alexandre Chirokov may refer to the same person);

[0089] the system is aware of the fact that a person can use same images (pictures) in their various individual profiles/accounts, and thus the image can be used to identify the person's profiles, even if the profile information differs substantially;

[0090] the system can employ external services to generate a distributed profile using databases containing individual profiles/accounts, such as gravatar.com, fullcontact.com, falcon.io;

[0091] the system can use links between profiles provided by the candidate themselves.

[0092] When creating a "distributed profile," the system locates profiles/accounts of the candidate on various resources, identifies the person using various, occasionally "fuzzy" criteria and combines individual profiles/accounts into a single "distributed profile."

[0093] According to the exemplary embodiment, the system operates as follows. If the system has an access to an external candidates' profiles database, it performs indexation and organizes the candidates' data according to their skills, work experience, education, contacts. In case there are multiple external databases, the system performs additional indexation and organization, and then begins to create a distributed profile of a candidate based on their first and last name, location, education institution they attended during a given period of time, enterprises they worked at, given contacts, personal image (photo) and other associated data.

[0094] In case the system is unable to combine two profiles reliably, but there is a high probability that both profiles belong to the same person, the system marks them as such, providing a link between the two for the system users, who verifies the information and either confirms or rejects the link.

[0095] The recruitment agency may opt to include their proprietary internal database to be used only inside the agency. In this case, the system performs indexation of data in the internal database and, then, creates distributed profiles based on CVs from the internal database and external network profiles/accounts. If needed, the information in the internal database is updated using links to external candidates' profiles. In case the system is unable to associate an external profile with the existing one reliably, but there is a high probability it belongs to the same candidate, the system provides a link to the external profile marking it for an additional check by the user (i.e., a recruiter), who then verifies the information and either confirms or rejects the link.

[0096] In order to improve quality of the search, the system can collect data for a given specialization (e.g., software development). Also, the system helps to "assess" candidates according to their professional achievements by supplying missing information or verifying the information provided by candidates, which enhances the preliminary assessment of candidates significantly. For instance, highly qualified or high-ranked candidates may add some "extra" achievements to their profiles, e.g., a number of employees who worked under

the candidate, academic degrees and titles, scientific works, publications, etc. The system can perform a search for the employments, degrees and titles and verify them. Such preliminary assessment can help eliminate sub-par candidates at early stages.

[0097] Low-ranked or not very qualified candidates often have identical profiles, but their "achievements" can be registered on various resources (e.g., a number of bans, "likes," and informative/non-informative posts). According to the exemplary embodiment, the system can have a "profile supplement" feature: if the recruiter is unsure whether to select a given candidate or not, he can ask for additional information regarding key skills (e.g., scientific or literary works, forum posts, etc.). As an option, such information is a subject to copyright laws and will be deleted after viewing, but the links can be saved, if needed. In one exemplary embodiment, the system may employ an "anti-plagiarism" feature to verify the authorship. There are numerous anti-plagiarism systems on the market. Generally, what they do is split a paper into smaller chunks (e.g., paragraphs) and then search for these chunks in other published papers. The inventive system may employ an "anti-plagiarism" feature, search engines (e.g., Google), or scientific paper databases.

[0098] According to the exemplary embodiment, the system uses cloud computing and includes the following elements:

[0099] An Internet data-collecting unit, which interacts with multiple sources: professional social networks, professional forums, open-source software repositories. If the system has been granted access to an external resource, the unit indexes data from this resource and passes it on to data storage and processing units;

[0100] High-capacity data storage and processing units, that store and process data, organize each candidate's information (name, work experience, key skills, etc.) and, then, analyze it in order to conduct search for other network profiles/accounts of the person on other resources: social networks, professional forums, patent databases, scientific publications databases and other sources.

[0101] A natural language text processor, which analyzes messages posted by the candidates in order to find out their key skills, professional interests, completed projects and work experience. In the exemplary embodiment, this unit can be amplified by a speech recognizing unit, which transforms speech into printed text.

[0102] An indexing unit, which sorts and organizes search data for each candidate.

[0103] A unit or a distributed computing environment is used to generate a "distributed profile" of a candidate, or to match network profiles/accounts from various resources belonging to the same person.

[0104] A unit or a distributed computing network used to compare candidate's profiles on various resources in case there is no full information about the candidate. The unit utilizes the technology of machine learning, which, according to an aspect of the invention, enables the system to create "fuzzy" distributed profiles of candidates using information provided in their individual profiles/accounts, add links between profiles and mark them for the user of the system.

[0105] Thus, the user is able to analyze the profile node provided by the system and either confirms the distributed profile by approving the system's "guess" or rejects it giving a feedback suitable for further machine learning. Such positive or negative feedback enhances the precision of the "fuzzy" distributed profiles created by the system.

[0106] An interactive web component for user interactivity, which can be embodied into user's workstation. The workstation can grant access to main control elements, both soft- and hardware, which efficiently provide search and other types of queries, search reports, and other data. The user specifies candidate search parameters, which can contain the following--skills; keywords; enterprises the candidate worked at; number of employees led by the candidate; job positions/titles; location; age; gender; educational institution(s) the candidate graduated from; work experience (in years); latest work duration; whether the candidate is currently looking for a job (i.e., "active" or "passive"); contacts (email address, phone number, Skype, and other means of communication);

the cache is empty, the unit 160 sends queries immediately to the profile database 145.

[0132] Description for users:

[0133] The user, i.e. a recruiter, works at the workstation 175 and fills in a search query by means of the workstation user interface 165, which should contain one or several parameters, such as:

[0134] a skills;

[0135] b. keywords;

[0136] c. enterprises the candidate worked at, particularly:

[0137] the name of the enterprise,

[0138] or

[0139] the industry branch,

[0140] number of employees;

[0141] d. job positions;

[0142] e. location;

[0143] f. age;

[0144] g. gender;

[0145] h. educational institution(s) the candidate graduated from;

[0146] i. work experience (in years);

[0147] j. latest work duration;

[0148] k. whether the candidate is currently looking for a job (i.e. "active" or "passive");

[0149] l. contacts (email address, phone number, skype, and other means of communication), Internet resources containing candidate's other profiles/accounts;

[0150] m. whether the candidate belongs to the first/second circle of acquaintances of the user of the system (the recruiter).

[0151] Any search parameter can be marked as "essential," "desirable," "undesired." Search parameters can be changed or modified after the search results are displayed on the workstation 175 monitor. After the search query has been filled via the workstation user interface 165, it is passed on to the profile database access unit 160 and to the full text search unit 150. In case there is no data stored in the cache, the profile database access unit 160 takes the query and processes the query immediately in the profile database 145. Note that the workstation 175 can be implemented on a cloud 170.

[0152] FIG. 2 illustrates a flow chart of the working algorithm of the system. In step 201, the "Robot" conducts searches through both external resources and internal database provided by the user to find new data, which can be used to search for candidates and define their professional qualities. Newly found profiles are stored in the system cache, in step 202, to be subsequently processed (by a parser) and analyzed.

yields no results from the profile database, the system can redirect the query immediately to the Internet data-collecting unit. The unit sends the query to external network resources to find profiles/accounts of the person.

[0173] If there are some, the Internet data-collecting unit transfers the documents to the data analysis unit, which extracts organized information about the candidate and, then, transfers it to the profile matching unit, which, in turn, may address the Internet data-collecting unit with a request to find associated candidate data on other network resources to match network profiles/accounts found and create a "distributed profile" of the candidate.

[0174] There are three types of high-level tasks which the system performs: (1) indexing the information on the Internet; (2) building a "distributed" profile; and (3) ranking search results based on different criteria.

[0175] Indexing the information might seem like a simple task, but if it requires a high-performance, then it becomes a complex computational task.

[0176] Much of the difficulty in implementing algorithms and components lies in #2--building distributed profile based on incomplete information and #3--ranking search result--assessing candidates prior to an interview.

[0177] With reference to FIG. 4, an exemplary system for implementing the invention includes a general purpose computing device in the form of a computer or a server 175 or the like, including a processing unit 21, a system memory 22, and a system bus 23 that couples various system components including the system memory to the processing unit 21.

[0178] The system bus 23 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. The system memory includes read-only memory (ROM) 24 and random access memory (RAM) 25. A basic input/output system 26 (BIOS), containing the basic routines that help transfer information between elements within the computer 175, such as during start-up, is stored in ROM 24.

[0179] The computer 175 may further include a hard disk drive 27 for reading from and writing to a hard disk, not shown, a magnetic disk drive 28 for reading from or writing to a removable magnetic disk 29, and an optical disk drive 30 for reading from or writing to a removable optical disk 31 such as a CD-ROM, DVD-ROM or other optical media.

[0180] The hard disk drive 27, magnetic disk drive 28, and optical disk drive 30 are connected to the system bus 23 by a hard disk drive interface 32, a magnetic disk drive interface 33, and an optical drive interface 34, respectively. The drives and their associated computer-readable media provide non-volatile storage of computer readable instructions, data structures, program modules and other data for the computer 175.

[0181] Although the exemplary environment described herein employs a hard disk, a removable magnetic disk 29 and a removable optical disk 31, it should be appreciated by those skilled in the art that other types of computer readable media that can store data that is accessible by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, random access memories (RAMs), read-only memories (ROMs) and the like may also be used in the exemplary operating environment.

[0182] A number of program modules may be stored on the hard disk, magnetic disk 29, optical disk 31, ROM 24 or RAM 25, including an operating system 35. The computer 175 includes a file system 36 associated with or included within the operating system 35, one or more application programs 37, other program modules 38 and program data 39. A user may enter commands and information into the computer 175 through input devices such as a keyboard 40 and pointing device 42. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner or the like.

[0183] These and other input devices are often connected to the processing unit 21 through a serial port interface 46 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, game port

