Discovering Experts, Experienced Persons and Specialists for IT Infrastructure Support

Girish Keshav Palshikar, Harrick M. Vin
Tata Research Development and Design Centre (TRDDC), Tata Consultancy Services Limited,
54B Hadapsar Industrial Estate, Pune 411014, India
{gk.palshikar, harrick.vin}@tcs.com

V. Vijaya Saradhi
Department of Computer Science,
Indian Institute of Technology, Guwahati 781039, India
saradhi@iitg.ernet.in

Mohammed Mudassar
Tata Consultancy Services Limited,
54B Hadapsar Industrial Estate, Pune 411014, India
mohammed.mudassar@tcs.com

Workforce analytics (i.e., statistical analysis, modeling and mining of HR data) is particularly important in service industries. Service industries are people-intensive and the knowledge and expertise of the people within an organization is a strategic resource critical for success. Performance of employees in a service organization is directly related to the customer satisfaction and creation of value. In this paper, we adopt a domain-driven data mining approach and begin by raising specific business questions in workforce analysis with a focus on IT Infrastructure Support (ITIS) services and then propose solutions for them. We distinguish between three aspects of what makes people valuable in an ITIS organization: expertise, specialization and experience. We propose novel formalizations of these notions and discuss rigorous statistical and optimization based algorithms to discover these 3 types of people (along with their work areas). In particular, for the important problem of expert discovery, we propose two separate algorithms: one statistical and another based on data envelopment analysis technique from optimization. The approaches have been implemented and have produced satisfactory results on more than 25 real-life ITIS datasets, one of which we use for illustration.

Key words: expert discovery; workforce analytics; human resource management; support services; IT infrastructure support; data mining

History: Received Apr. 20, 2010; Received in revised form Aug. 02, 2010; Accepted Aug. 17, 2010; Online first publication Nov. 12, 2010

1. Introduction

Effective human resource management (HRM) is well-known to be a key ingredient in the success of an organization. Standards such as People CMMI (Curtis 2009) have helped HR management practices to get aligned with business objectives and to systematize programs of continuous workforce development. To help automate HR processes and practices, modern organizations now deploy sophisticated e-HRM systems (Bondarouk et al 2009), (Strohmeier 2007), which collect vast amounts of data about the work history and other activities of the employees. Statistical analysis, modeling and mining of HR data is now emerging as a leading application (Connors and Mojsilovic 2010), (Naveh et al 2007), (Richter et al 2007), (Lu et al 2006), (Hu et al 2008), (Patterson 2003).

As a special case, managing human resources in service-oriented organizations is challenging due to several special factors (Alvesson 2004). Service industries are people-intensive and the knowledge and expertise of the people within an organization is a strategic resource critical for success. Performance of employees in a service organization is directly related to the satisfaction of customers and creation of value (Swart et al 2003). Service organizations also face higher attrition and hence effective management of the people and their knowledge is
important. For these reasons, the service science is emerging as an important discipline bordering HR management, computing and analytics; see (Katzan Jr. 2008), (Maglio and Spohrer 2007), (Spohrer et al 2007) for overviews of the fundamental concepts of service science. Standard for service organizations, such as CMMI for Services (CMMI-SVC) (Forrester 2009) and IT Infrastructure Library (ITIL), are helping establish systems for effective HR management within service organizations. This paper can be considered as producing outputs which are closely aligned with several processes in ITIL, in particular the continual improvement, operations management and service level management processes in ITIL (Cannon and Wheeldon 2007), (Spalding and Case 2007).

As argued extensively in Service Science research literature (Vargo and Akaka 2009), any analysis of quality of service and performance measures pre-supposes collection of good quality (and quantity) of basic operational data. Due to the systematization of the workflows and processes in services organizations due to standards such as CMMI for Services and ITIL, a lot of data about the operations of these processes gets collected. Given the availability of basic measurements of the service processes, the focus should also be on the statistical analysis of this data with the aim of answering specific business questions.

We focus on the analysis of data about IT Infrastructure Support (ITIS) services. An aim of this paper is to analyze this data for answering specific business questions related to performance (or value) of the IT infrastructure support personnel. We distinguish between three types of valuable people in an IT infrastructure support organization: experts, specialists and experienced people.

(a) How do we identify “experts” in the organization in an objective and rigorous manner, along with their work areas of expertise?
(b) How do we identify “specialists” in the organization in an objective and rigorous manner, along with their work areas of specialization?
(c) How do we identify “experienced” people in the organization in an objective and rigorous manner, along with their work areas of experience?

Identifying experts or specialists is difficult in most domains because of the subjective nature of these notions. The key contributions of this paper are the rigorous formalizations of these notions (which take advantage of the availability of performance data in ITIS) and statistical algorithms to answer the above business questions. The proposed techniques are fairly general and should be usable in any domain that has performance related data; e.g., the customer-facing “front office” in any enterprise (e.g., a bank or a travel agency). We have used these techniques to analyze data from more than 25 different ITIS organizations. The results have been found to be consistent and useful all across them. Further, we have extended and applied these techniques to answer similar questions in other domains such as sports.

One approach would have been to use standard data mining algorithms, a large number of which are now available and which extract different types of knowledge from given historical data. But this variety has led to technology fatigue among end-users. First, there is no unified representation for the many different types of knowledge extracted (decision rules, clusters, associations etc.), making it difficult for users to relate extracted knowledge elements. Next, even the volume of extracted knowledge tends to be large, making it difficult for users to filter and use it. Research on knowledge filtering (Palshikar et al 2007) and knowledge selection using interestingness measures (Geng and Hamilton 2006) are useful, though the problem is far from solved. Lastly, there is a huge gap between the extracted knowledge and practical business goals of the user (e.g., the questions stated above). Extracted knowledge is not aligned to (and hence not directly usable for meeting) the user’s business goals. It becomes the user’s responsibility to understand and make use of the extracted knowledge to solve a business problem. This requires expertise in setting up appropriate data-mining experiments, selecting and understanding the extracted knowledge and applying it to solve the business problem. This demands data-mining expertise as well as domain expertise - a rare combination. In this paper, we attempt to bridge this gap by adopting a domain-driven data mining approach, which is getting increasing acceptance from end-users (Yu 2007), (Yu 2008). This consists of designing special purpose algorithms for stated problems in workforce analytics in ITIS domain. (Phua et al 2005) and (Harding et al 2006) survey domain-driven data mining for fraud detection and manufacturing respectively. Instead of domain-neutral data-mining, we discuss problem-specific data-mining i.e., instead of trying to fit a data-mining technique to a business problem, we adopt a top-down approach where we begin with a business goal.

This paper is organized as follows. Section 2 provides overview of some of the related literature. Section 3 discusses motivation for the specific business questions related to performance of people in service organizations. Section 4 is an overview of the IT infrastructure support systems that we have analyzed. Section 5 discusses a sample real-life dataset that we use throughout the paper to illustrate the approaches. Section 6 formalizes the notion of work area. Sections 7 to 9 present rigorous formalizations of the above notions and our approaches for answering the above questions. Section 10 presents conclusions and further work.
2. Related Work

In this paper, we have focused on the performance of people in a service organization, which is directly related to the service quality in the IT infrastructure support domain. In general, measurement of service quality is an active area of research (Schneider and White 2004), (Rowley 2006). The SERVQUAL model for measuring service quality (typically through customer surveys) based on various dimensions such as reliability, responsiveness, competence etc. has seen wide usage (Buttle 1996), (Nyeck 2002). Standards such as ITIL and CMMI include service quality parameters; see (dos Santos 2009) for an application to IT service providers. A more specific area that has attracted attention of researchers is the measurement of the e-service quality (e.g., quality of the web portals for e-businesses).

Apart a statistical approach to identify experts, we have also proposed an optimization/operations research based approach for the same (expert as an efficient person) and demonstrate a remarkable agreement between their results. The idea of efficiency of decision making units is formalized as a ratio of outputs produced to inputs consumed by the field of data envelopment analysis (DEA) in operations research (Charnes et al 1994); see also (Boussofiane 1991). DEA has been used to evaluate and compare the performance of decision making units (DMU) in different domains: manufacturing plants, bank branches, hospitals; e.g., see (Charnes et al 1985). In this paper, we have used the idea of cross-efficiency as proposed in (Doyle and Green 1994). See (Youn and Park 2009) for a model-based approach for comparative efficiency analysis of 60 Korean universities in terms of 4 development models.

We have already mentioned the emergence of service industry related standards such as CMMI of Services (Forrester 2009) and ITIL (Cannon and Wheeldon 2007), (Spalding and Case 2007), which has led to systems that collect much data about the operations. The analysis proposed in this paper uses such datasets. This paper can be considered as producing outputs which are closely aligned with several processes in ITIL, in particular the continual improvement, operations management and service level management processes in ITIL.

IT infrastructure support services, analyzed here, are often outsourced. (Behara and Bhattacharya 2008) underscore the importance of training and other related factors (as emphasized in this paper also) as critical to the success of knowledge-intensive outsourced services. See (Choi and Yoo 2009) for a review of the analysis of IT services, in particular as related to software quality.

This paper proposes specific problems in workforce analytics and approaches to solve them. Workforce analytics is an active research area (Connors and Mojsilovic 2010). Many specific problems in workforce analytics have been investigated; e.g., optimal team formation for which a range of techniques from operations research to constraint programming and stochastic networks have been used (Naveh et al 2007), (Richter et al 2007), (Lu et al 2006), (Hu et al 2008). Another important problem in workforce analytics is analysis of employee satisfaction surveys; see (Palshikar et al 2009) and (Van De Voorde et al 2010). There is work on various other problems in workforce analytics; e.g., see (Murray and Young 2008) for optimal call routing in support services.

Identification of experts is a well-known research problem and many different techniques have been used. An entire track in (TREC 2005) was devoted to this problem. In most of the problem formulations, an expert is discovered using a corpus of documents authored by people (who are candidate experts) and its access patterns. Further, in much of the work (not all) the task is really “Who are the experts on topic X?” Our problem formulation is different: experts are identified using their performance on handling of specific problems in support tickets. We answer a more general question: “Who are the experts and what are their areas of expertise?” As another contribution, we have proposed two different techniques for expert discovery: statistical and optimization based, which have not been used for this task to our knowledge. In the literature, an expert’s profile includes topics of expertise and connectivity to other experts (which is not applicable in a support scenario reported here).

(Balog and de Rijke 2007) use similarity measures between given set of expertise topics with given document corpus to identify topic profile of experts. Techniques such as co-nomination (Nevada et al 2007) and social network analysis (Chentharamarakanthan 2009) have been used to identify social profiles of experts; see also (Becerra-Fernandez 2000) and (Craswell et al 2001). (Campbell et al 2003) used email communication patterns for the task of expert discovery. A commercial tool called ExpertFinder (www.teezir.com) uses authorship and access patterns for documents written by people to identify experts. We have proposed two different techniques for identifying experts in ITIS domain, which appear to be new. Similarly, the problems of identifying specialists and experienced proposed in this paper people appear to be new in workforce analytics. See (Groysberg and Lee 2009) for a discussion about effective utilization of experts in service industry. A topic related to expert discovery is extracting and reusing expert knowledge, for which techniques such as case-based reasoning have been used particularly in support domain; see (Cheetham 2003) and (Simoudis 1992) as examples of this work.
Effective management of knowledge assets is closely related to expertise, experience and specialization in a service organization and has received much attention from HR management perspective; e.g., see (Macey et al 2009), (Doorewaard and Meihuizen 2003), (Swart and Kinnie 2010).

3. Motivation

Realistically speaking, employees differ in terms of their “value” to the organization. In any organization, and in particular in service organizations, there is a subset of “highly valuable” employees, though it may never be explicitly identified. In a successful organization, a large fraction of the workforce is “highly valuable”. At a broad level, an employee is “highly valuable” if that person’s contribution to the success of the organization is high. There is no unique way to measure the success of an organization because it depends on the perspective and the goals of the analysis. In particular, when evaluating the success of service organizations (or support functions within an organization), quality of service measures are much more appropriate than, say, financial measures such as the revenue earned.

Even assuming a way for quantifying the success of an organization, there is probably no unique way to formalize and measure the notion of the value of employees, in a broad manner which is applicable across a wide variety of organizations. A major difficulty is how to formalize and measure the impact or contribution of an employee on the success of an organization. However, in a more indirect approach, we do not measure such a contribution and instead associate certain qualities with “highly valuable” employees. We adopt such an approach and propose to use 3 specific dimensions for understanding the notion of “value”. Before we discuss these dimensions, we would like to emphasize that several other dimensions, such as flexibility, adaptability etc., could (and should) be taken into account to more completely formalize the notion of the “value”. We ignore the issue of the monetary contributions of the employee to the company’s business. Further, this notion of “value” is different from that of the “lifetime value” (used for customer in sales and marketing disciplines) in the sense that we do not project the value into the future i.e., we do not consider the expected future contributions of an employee in computing that person’s current “value”.

We begin with the observation that a “valuable” employee has some knowledge which that person is demonstrably using towards the benefit of the organization. We propose to use three dimensions to measure the knowledge and hence the “value” of an employee: experience, expertise and diversity in work (Figure 1). An employee who measures high on any one (or more) of these dimensions is considered “valuable”. A mentioned earlier, it is possible to add more dimensions. These 3 dimensions are somewhat independent of each other, in an informal sense.

An expert is a person who is high on the expertise dimension. An expert has attributes such as broader domain knowledge, specialized skills and knowledge, demonstrably superior performance on specific tasks, problem-solving ability etc. An expert may or may not be highly experienced. In most service-oriented situations in practice, an expert is not characterized only by the knowledge but in terms of demonstrably consistent and high performance levels on specific tasks.

An experienced person is high on the experience dimension. An experienced person may have only limited specialized skills and knowledge acquired through high experience and generally does not possess very high levels of the other attributes that characterize an expert. In particular, an experienced person need not be a “star performer”. In terms of this dichotomy, it is easy to understand why experts who are also experienced persons are more valuable for any organization.

A generalist is a person high on the work diversity dimension. A generalist is a person having high work diversity (or just diversity) i.e., one who has worked with a much larger variety of tasks or problems as compared to most other employees in the organization. In contrast, a specialist is a person low on the work diversity dimension. A specialist is a person having low work diversity i.e., one who has worked with a much smaller variety of tasks or problems as compared to most other employees in the organization. Specialists tend to be good in (or prefer) handling a small and fixed number of tasks or problems. On the other hand, generalists tend to be good in (or prefer) handling a much larger number of tasks or problems. Of course, some employees could neither be specialist nor generalists and may fall in between these two classes. Both specialists and generalist are valuable in a service-oriented organization. Note that a specialist (or generalist) need not be very highly experienced and need not be a “star performer” in any of the tasks or problems.

For any employee in these classes (experienced, expert, specialist, generalist), we need to also identify the associated focus areas of work (in terms of tasks or problems). For example, for an experienced employee, we need to identify the tasks or problems in which that person is experienced; similar considerations apply for an expert, a
specialist or a generalist. Note that these 3 dimensions are not mutually exclusive; e.g., a person can be an expert in one work area and an experienced person in another work area. Following questions are now interesting from a services science perspective:

1. How to characterize the work areas for any given employee?
2. How to formalize and measure the notion of experience of people on a given work area?
3. How to formalize and measure the notion of expertise of people on a given work area?
4. How to formalize and measure the notion of work diversity of people in terms of the work areas?

Figure 1 Dimensions to Characterize Employee “Value”

Following sections discuss approaches to answer these questions in the context of a specific service industry: IT infrastructure support.

4. IT Infrastructure Support

Customer Support (also called help desk, service desk, call center etc.) is an important function across a broad range of organizations which provide a service or sell a product to people. The user/customer brings the problem (or difficulty) faced when using a service or a product to the attention of the Customer Support function. Contact with customers may be through e-mail, chat, telephone, interactive voice system, fax or by interaction with a special-purpose system where the customer can lodge a complaint (also called ticket) describing the problem. Upon receiving the customer’s complaint, an appropriate business process is invoked within the Customer Support function to resolve the customer’s problem. Virtually all types of organizations include a Customer Support function – banks, insurance, power distribution, telecom, manufacturing, and even government bodies.

As another class of a support function, many organizations include an IT Infrastructure Support (ITIS) function which is responsible for maintaining the organization’s internal hardware and software infrastructure. In such a case, the employees in the organization are the internal customers of the ITIS function. In this paper, we focus on the ITIS function.

An IT resource – also called configuration item (CI) – consists of a specific IT service, a hardware product or a software product. When using an IT resource, users sometimes face errors, faults, difficulties or special situations that need attention (and solution) from experts in the ITIS function. Since smooth functioning of IT infrastructure is business critical, effective ITIS is vital. With the emergence of ITIL standard and associated tools for management of ITIS, the processes in ITIS are standardized and well-monitored. A lot of data is collected about all stages and aspects of the various ITIL processes. Apart from service level management and service measurement and reporting processes, ITIL standard includes a continual service improvement (CSI) process to improve effectiveness, efficiency and cost of ITIS function as a whole.

Typical business goals for ITIS process improvements are:
(i) improve service quality e.g., reduce service time (ST), reduce variation in ST;
(ii) improve productivity e.g., number of tickets handled per person per month;
(iii) reduce operational costs e.g., reduce number of people, reduce spare parts consumed;
(iv) reduce SLA violations;
(v) reduce efforts e.g., handle some tickets automatically, do preventive maintenance.
(vi) improve people utilization e.g., improve ticket allocations, reduce ticket reassignments, reduce escalations etc.
(vii) improve user satisfaction levels; e.g., reduce ticket reopening, reduce downtime
(viii) improve people management e.g., improve team formulation, improve shift design, improve ticket allocation policies.

To effectively implement many of these types of improvements, it is important to identify, in a rigorous and objective manner, the experts, experienced persons and specialists/generalists in the ITIS function, along with the focus areas for each. Once identified, they can be used in several ways to design and implement the improvement plans:

- define and establish best practices
- identify shortcomings and bottlenecks in current support processes
- identify candidate trainees for specific types of tickets and conduct training sessions for the trainees (e.g., to improve their performance)
- improve ticket allocation (allocate as many tickets as possible to the most experienced persons or specialists relevant to the work area for that ticket)
- allocate difficult tickets to an expert relevant to the work area for that ticket
- maintain a team (or a shift) at location, which has an optimal mix of specialists and generalists
- identify and fill gaps (e.g., work areas for which no experts or specialists are available)

5. IT Infrastructure Support

To illustrate these concepts, we consider a specific database of tickets handled in the ITIS function of a company. Each ticket has the attributes shown in Table 1. The database contains 15538 tickets created over a period of 89 days (with status = ‘Closed’), all of which were handled at level L1. In standard ITIS terminology, simple problems are handled at level L1, problems requiring expertise and special skills are handled at level L2. The hardest problems requiring complex solutions are handled at level L3. Summary statistics for various information elements is shown in Table 2.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>#distinct values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ticket_id</td>
<td>Unique ID for each ticket</td>
<td>15530</td>
</tr>
<tr>
<td>Client_id</td>
<td>Unique ID of the end user who is facing the problem denoted in a ticket</td>
<td>8022</td>
</tr>
<tr>
<td>Client_location</td>
<td>Unique ID of the office location or business unit to which the client belongs</td>
<td>25</td>
</tr>
<tr>
<td>Severity</td>
<td>Degree of severity of the problem</td>
<td>2</td>
</tr>
<tr>
<td>Priority</td>
<td>Priority with which the ticket was requested to be processed</td>
<td>3</td>
</tr>
<tr>
<td>Problem_type</td>
<td>Broad area of the problem; e.g., Hardware, Software, Networking, Messaging, Security etc.</td>
<td>10</td>
</tr>
<tr>
<td>Problem_item</td>
<td>Sub-area of the problem; e.g., for problem_type = ‘Networking’, these are: Access rights, DNS, Proxy, Router, ISDN, Dialup, Leased line, Wireless etc.</td>
<td>44</td>
</tr>
<tr>
<td>Problem_subitem</td>
<td>Further details about the problem; e.g., for problem_type = ‘Networking’ and problem_item = ‘Access rights’ these are: Home folder access, Project folder access, Server access etc.</td>
<td>103</td>
</tr>
<tr>
<td>Resolver_id</td>
<td>Unique ID for the person who resolved the ticket</td>
<td>66</td>
</tr>
<tr>
<td>Resolver_location</td>
<td>ID of the group or location to which the resolver belongs</td>
<td>24</td>
</tr>
<tr>
<td>Create_dt</td>
<td>Datetime stamp when the ticket was created by the end-user (client)</td>
<td></td>
</tr>
<tr>
<td>Solve_dt</td>
<td>Datetime stamp when the ticket was finally resolved</td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>Estimated total service time spent in resolving the ticket (in minutes)</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2(a) shows a histogram for ST. We have aggregated the long tail on the right into a separate bin to demonstrate that there are a lot of tickets with very high ST and they form an opportunity to improve service quality. The peaks at the left and right ends clearly show noticeable fractions of the tickets having either short ST (< 30 minutes) or large ST (> 240 minutes). Figure 2(b) shows a histogram of the “experience” levels among the 66 resolvers for this 3 month window, as measured by the number of tickets handled by them. The resolvers’
experience before this period is not taken into account here. Again, we have aggregated the long tail on the right into a separate bin to demonstrate that there are a lot of resolvers with very high number of tickets. The peaks at the left and right ends clearly show noticeable fractions of the resolvers having either little experience (< 50 tickets) or sizeable experience (> 400 tickets), thereby implying a team with mixed experience levels (high, medium, low). Figure 2(c) shows the histogram of number of tickets arriving daily. The x-axis shows the number of tickets and y-axis shows the number of days; e.g., the number of tickets arrived was between 0 and 50 for 27 days. Figure 2(d) shows the time plot of the number of tickets arriving daily. Note the “weekend” effect in Figure 2(c) and weekly cycle in Figure 2(d). There is usually no noticeable difference between the ST characteristics of tickets raised on weekends or weekdays. This is possibly because most ITIS organizations function in a 24 x 7 manner.

Figure 2 Some Charts for the Sample Dataset

![Figure 2(a) ST histogram](image)

![Figure 2(b) Histogram of experience (number of tickets)](image)

![Figure 2(c) Histogram of number of tickets arriving per day](image)

![Figure 2(d) Time plot of daily ticket arrival](image)

Table 2 Summary Statistics for Various Information Elements for the Tickets Dataset

<table>
<thead>
<tr>
<th>Information Element</th>
<th>Average</th>
<th>STDEV</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST (minutes)</td>
<td>360</td>
<td>4797</td>
<td>21</td>
<td>72</td>
<td>169</td>
</tr>
<tr>
<td>Number of tickets arriving per day</td>
<td>172.6</td>
<td>111.7</td>
<td>23</td>
<td>217</td>
<td>251</td>
</tr>
<tr>
<td>Total number of tickets handled by a resolver</td>
<td>235.3</td>
<td>176.5</td>
<td>84</td>
<td>245</td>
<td>335</td>
</tr>
<tr>
<td>Team size per day (number of resolvers who closed at least 1 ticket on a day)</td>
<td>27.5</td>
<td>16.5</td>
<td>7</td>
<td>36</td>
<td>41</td>
</tr>
<tr>
<td>Resolver productivity (number of tickets resolved per resolver per day)</td>
<td>5.94</td>
<td>4.33</td>
<td>3</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>Resolver efficiency (average ST per ticket for a resolver)</td>
<td>2862.0</td>
<td>19705</td>
<td>136</td>
<td>212</td>
<td>387</td>
</tr>
</tbody>
</table>

ST: service time; STDEV: standard deviation; Q1, Q2, Q3: first, second and third quartiles
6. Formalization of Work Area

As discussed earlier, we need to formalize the notion of the focus area for a resolver. Clearly, a resolver has a lot of responsibilities and work areas other than those represented in a tickets dataset. For simplicity, we assume that work of each resolver consists only of handling tickets and that each possible work area is represented by means of a selector over the relevant ticket attributes. A selector (also called a problem or a type of tickets) characterizes a subset of tickets.

Let \( A = \{A_1, A_2, \ldots, A_n\} \) denote the set \( n \) of problem attributes useful for characterizing work areas. In our example, \( A = \{\text{problem_type}, \text{problem_item}, \text{problem_subitem}\} \), though other attributes such as severity and priority could also be added to this set. For simplicity, we assume that each attribute \( A_i \in A \) has a finite domain of possible values, denoted \( \text{DOM}(A_i) \). This assumption is clearly true in our example (Table 1). If an attribute has an infinite domain, then we assume that the domain discretized suitably to make it finite.

An attribute-value pair is an expression of the form \( A_i = v \), where \( A_i \in A \) and \( v \in \text{DOM}(A_i) \). This definition can be generalized to include other operators and more complex expressions. A selector \( \theta = X_1 \land X_2 \land \ldots \land X_k \) over the given set \( A \) of attributes is a conjunction (logical AND) of one or more attribute-value pairs \( X_i \) \((1 \leq i \leq k)\) such that any attribute from \( A \) appears in at most one attribute-value pair. In our example, some selectors are as follows:

\[
\begin{align*}
\text{problem_type} &= \text{'Networking'} \\
\text{problem_type} &= \text{'Networking'} \text{ and problem_item} = \text{'Dialup'} \\
\text{problem_type} &= \text{'Networking'} \text{ and problem_item} = \text{'Access Rights'} \text{ and problem_item} = \text{'Home Folder access issue'} \\
\text{problem_item} &= \text{'Mailbox Movement / Renaming Issues'} \text{ and problem_subitem} = \text{'Mail Box Movement'}
\end{align*}
\]

It is possible to generalize the definition of selectors to allow complex logical expressions using other logical operators such as OR and NOT. For simplicity, we stick to the conjunctive form of the selector.

Let \( D \) denote the given dataset of tickets. For any selector \( \theta \) over the given set \( A \) of problem attributes, we use \( D[\theta] \) to denote the subset of tickets in \( D \), each of which satisfies the condition specified by \( \theta \).

7. Identifying Expert Resolvers

As mentioned earlier, we need to identify all experts in the organization along with the focus area for each. We first discuss an algorithm to automatically identify expert resolver for a given work area i.e., for a given selector. We later discuss a more general approach to discover experts along with their focus areas. Automatically identifying experts for a given problem quickly and in an objective manner is important because experts are scarce and valuable resources. Their expertise can be used in many ways, as discussed earlier.

7.1 A Statistical Approach to Identify Experts

We had earlier characterized an expert as a “star performer” on a given problem. As per our assumptions, work for a resolver consists of handling tickets and hence a natural measure of performance of the resolver is the ST taken on the tickets handled by that person. Let \( R \) denote the set of all resolvers i.e., domain of the resolver ID attributes. Informally, a resolver is an expert for a given problem if he/she has worked on ”sufficiently many” tickets and has "reasonably low and uniform” ST (low average and low standard deviation).

Given a problem (i.e., a selector \( \theta \)) and a specific resolver \( r \in R \), we can define several different expertise score functions each of which assigns a score to \( r \). Conceptually, given a selector \( \theta \) and a score function \( f : R \rightarrow [0, 1] \), the expert finding algorithm computes the score for each resolver in \( R \) over tickets characterized by the selector \( \theta \) and handled by \( r \). One can then select those resolvers whose score is more than an automatically decided threshold. One such expertise score function is based on the use of hypothesis testing using the Students t-test (Wasserman, 2004), which is used to check whether the ST of a resolver \( r \) (for tickets of type \( \theta \)) are “significantly” lower than those for all other resolvers put together.
Let $D[r, 0]$ and $\overline{D}[r, \theta]$ denote the set of all tickets of type $\theta$ in $D$ which are handled by resolver $r$ and all resolvers other than $r$ respectively. We use the 2-sample 2-tailed Student’s $t$-test to compare the multisets $\text{ST}[r, \theta]$ and $\overline{\text{ST}}[r, \theta]$ consisting of only the ST values for these two sets of tickets. The null hypothesis $H_0$ is that the two sets of ST values are drawn from the same distribution (more precisely, the means of the two populations, from which each sample of ST values is drawn, are equal). The alternative hypothesis $H_1$ is that the two sets of ST values are drawn from two different distributions. If the null hypothesis is rejected then it means that the mean of the population of ST values (from which $\text{ST}[r, \theta]$ are drawn) is statistically significantly different (lower), as per the $t$-test, than the mean of the population of ST values (from which $\overline{\text{ST}}[r, \theta]$ are drawn). In other words, resolver $r$ has a significantly better performance on tickets of type $\theta$ than others and hence $r$ can be called an expert for tickets of type $\theta$. We illustrate the approach with an example.

In the example dataset, suppose a selector $\theta$ is defined as follows:

```plaintext
problem_item = 'Account Management' and problem_subitem = 'Password Reset'
```

The subset corresponding to this selector contains 3821 tickets with ST average = 110, STDEV = 428.71. A total of 46 resolvers worked on these tickets, all of whom handled 10 or more tickets of this type. Consider a resolver $r = '221384'$, who has handled 142 tickets of this type with an average ST of 17 and ST STDEV of 44.44. The remaining 45 resolvers put together handled 3679 tickets with ST average = 113 and STDEV = 436.43. Let $n_1$, $m_1$, $s_1$ denote the number of tickets, ST average and ST STDEV of tickets handled by the resolver $r$. Similarly, $n_2$, $m_2$, $s_2$ denote the number of tickets, ST average and ST STDEV of tickets handled by all resolvers other than $r$. Then the $t$-statistic is computed as:

$$
T = \frac{m_1 - m_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} = \frac{17 - 113}{\sqrt{\frac{44.44^2}{142} + \frac{436.43^2}{3679}}} = -11.85
$$

The degrees of freedom (DF) are calculated approximately as $df = n_1 + n_2 - 2 = 142 + 3679 - 2 = 3819$; there are better ways of calculating the DF, such as the Welch-Satterthwaite formula. One can specify a value called significance level $\alpha$, which is typically 0.1, 0.05, 0.01 or 0.005 (lower values make the test more stringent). Now we can use the standard table of the $t$-values to obtain the expected $t$-value for the given degrees of freedom and given value of $\alpha$. Suppose that the user specifies $\alpha = 0.01$. Then as per the standard table of $t$-values, for $df = \infty$ and $\alpha = 0.01$, the expected $t$-value is 2.58. Since the calculated $t$-value (11.85 ignoring the sign) exceeds the tabulated $t$-value, the probability (p-value) of obtaining this large a value (11.85) for the $t$-statistic is less than the given significance level (0.01) and hence we reject the null hypothesis. In other words, the two means $m_1$ and $m_2$ are significantly different at the given significance level $\alpha$, i.e., the service times for resolver $r$ and all other resolvers put together (for tickets of type $\theta$) are statistically different at the given significance level. In that case, since $m_1 < m_2$ (i.e., $r$ has a lower mean ST than other resolvers), then $r$ can be called as an expert on tickets of type $\theta$.

For $r = '218101'$, we have $n_1 = 112$, $m_1 = 103$, $s_1 = 441.89$ and for the remaining 45 resolvers $n_2 = 3709$, $m_2 = 110$, $s_2 = 428.37$. The value of the $t$-statistic is -0.16, whose magnitude is less than the expected $t$-value 2.58 for $df = \infty$ and $\alpha = 0.01$. Thus the null hypothesis cannot be rejected; hence performance of $r$ on tickets of type $\theta$ is not significantly different from others and $r = '218101'$ is not an expert. Algorithm `identify_experts_for_problem` systemically examines each resolver and classifies that person as an expert or not for the given problem, using the approach explained above. Figure 3 shows the expert resolvers identified by the algorithm for the above problem, which includes ‘221384’ but not ‘218101’ as expected. Each expert has a much less average ST and much less variation in the ST for this type of tickets, than remaining resolvers. Most (but not all) experts have above average experience (> 3821/46 = 83 tickets). The algorithm combines all 3 aspects of a resolver’s performance (number of tickets, average ST, standard deviation in ST) into a single expertise score and then statistically selects experts as those resolvers whose expertise score is above an automatically determined threshold.

```plaintext
algorithm identify_experts_for_problem
input D // tickets table
input $\theta$ // selector for specifying a type of tickets
input RX // set of resolvers
input m // min. number of tickets of type $\theta$ that a resolver must have worked for being considered as an expert // of this type of tickets
input $\alpha$ // significance level 0.1, 0.05, 0.01, 0.005 or 0.001
```
output $O$ // set of expert resolvers for problem (selector) $\theta$

Compute $X = D[\theta]$ subset of tickets in $D$ that are of type $\theta$

$O = \emptyset$ // initially no expert resolvers are known

for each resolver $r \in RX$
do
  if $|D[r, \theta]| < m$ then continue end if // $r$ has not handled at least $m$ tickets of type $\theta$
  Compute $AX = \text{(multi)set of service times of tickets of type } \theta \text{ handled by } r$
  Compute $BX = \text{(multi)set of service times of tickets of type } \theta \text{ not handled by } r$
  Compute $n_1, m_1, s_1$ (number of tickets, mean ST and STDEV ST of $AX$);
  Compute $n_2, m_2, s_2$ (number of tickets, mean ST and STDEV ST of $BX$);
  Apply $t$-test at given $\alpha$ to $AX$ and $BX$ to check whether $m_1$ and $m_2$ are statistically different
  if yes and $m_1 < m_2$ then $O = O \cup \{r\}$ end if // $r$ is an expert on tickets of type $\theta$
do
end for

return $O$

Figure 3 Expert Resolvers Identified for the Example Problem

We now consider the problem of finding all experts along with the focus areas for each expert. The naïve brute force approach is, for each resolver $r$, to examine all possible problems (selectors) $\theta$ and check whether or not $r$ is an expert in $\theta$; if yes, then report $(r, \theta)$ and do not consider any further refinement of $\theta$. This brute force approach is infeasible due to the exponential number of selectors that needs to be examined. We adapt the well-known “beam search” heuristic for efficient search (Clark and Niblett, 1989). The algorithm performs a level search among selectors, where level $i$ consists of selectors containing exactly $i$ different attributes. At $i=1$, the algorithm examines all selectors of the form $A = v$, for all attributes $A$ and all values $v \in \text{DOM}(A)$. If the given resolver $r$ is an expert on any such selector $\theta$ then $\theta$ is added to the output set $S$ (which contains all areas of expertise for $r$); otherwise, $\theta$ is added to the set $F$. After examining all selectors, another set $C$ is formed by keeping only top $N$ selectors in $F$ (ordered as per the number of tickets which satisfy a selector). Thus $C$ contains the set of candidate selectors available for exploration in the next level.

At level $i+1$, we extend each “candidate” selector $\theta \in C$ by adding an attribute-value pair $(B = v)$, where $B$ is an attribute not already present in $\theta$ and $v \in \text{DOM}(B)$ is a possible value for $B$. If $r$ is an expert for this extended selector $\theta'$ then we add $\theta'$ to $S$. If $\theta'$ “contains” any of the previously known problems for which $r$ is an expert (as per set $S$), then $\theta'$ is not considered. For example, if $\theta'$ is $\text{problem_item} = \text{Account Management}$ and $\text{problem_subitem} = \text{Password Reset}$ and if $\theta$, is $\text{problem_subitem} = \text{Password Reset}$, is already known to be an area of expertise for the given resolver then $\theta'$ is not considered at level 2 because it “contains” $\theta$. If $r$ is not an expert for the extended selector $\theta'$ then $\theta'$ is added to a set $F$ of candidate selectors. After examining all possible extensions of selectors, we reset $C$ to contain only top $N$ selectors from $F$ (e.g., ordered according to the number of tickets). Increasing the beam size $N$ improves the quality of the results but slows down the algorithm.

algorithm identify_experts_problems_BS
7.2 Data Envelopment Analysis to Identify Experts

Here, we propose another notion of what constitutes an expert resolver. Broadly, an expert resolver for a given problem \( \theta \) is one who is among the most efficient ones for handling tickets of problem \( \theta \). Clearly, different resolvers will differ in their efficiency of handling tickets of the same problem \( \theta \). These differences in their efficiencies arise due of several reasons, such as differences in the experience levels among the resolvers. There are several ways to formalize the notion of the efficiency of a resolver. Informally, one possible way is to measure both the outputs due of several reasons, such as differences in the experience levels among the resolvers. There are several ways to formalize the notion of the efficiency of a resolver. Informally, one possible way is to measure both the outputs produced and inputs consumed by each of the resolvers and then the output divided by the input is one measure of the efficiency of each location. A highly efficient resolver would minimize the inputs consumed and maximize the outputs produced. To give example from another domain, it can be argued that a country that participated in 5 events in an Olympics and obtained 2 medals is as “efficient” as another country that participated in 100 events and obtained 40 medals, though the latter country is ranked much higher in terms of a simple medal tally.

This idea is formalized by the field of data envelopment analysis (DEA) in operations research (Charnes et al. 1985), (Charnes et al. 1994), (Boussofiane 1991). DEA has been used to evaluate and compare the performance of decision making units (DMU) in different domains: manufacturing plants, bank branches, hospitals. In this paper, we propose to use each resolver as a DMU (Figure 4). For illustration, we use: (a) average ST \( x_i \) and STDEV ST \( x_j \) of the tickets handled by a resolver as the input parameters; and (b) number of tickets handled \( y_i \) and number of users served \( y_j \) as output parameters. We have also used other models that include more input or output parameters. Let \( n \) be the total number of resolvers. The efficiency of a specific resolver \( r_0 \) is defined as the ratio of
the weighted sum of the outputs to the weighted sum of the inputs, \( \frac{v_1y_{10} + v_2y_{20}}{u_1x_{10} + u_2x_{20}} \), where \( x_{10}, x_{20}, \) are the average ST and STDEV ST of \( r_0, y_{10}, y_{20} \) are the number of ticket handled and number of users served by \( r_0 \) and \( u_i, v_j \) denote the unknown weights associated with \( i^\text{th} \) input and \( j^\text{th} \) output respectively.

**Figure 4 Resolver as a DMU**

One way to understand the choices of inputs and outputs is as follows. Inputs are the quantities that need to be minimized and outputs are the quantities that need to be maximized. An “efficient” DMU (resolver, in our case) would minimize the input quantities and maximize the output quantities. In our case, average ST and STDEV ST clearly need to be minimized and hence are called “inputs”. Similarly, #tickets and #users served clearly need to be maximized and hence are called “outputs”. Seen another way, time is a resource that is consumed when handling tickets and hence needs to be minimized. #tickets and #users served is the value delivered and hence these need to be maximized. To find the weights, we set up the following optimization program where the efficiency of \( r_0 \) is maximized subject to the constraint that the efficiencies of all resolvers are \( \leq 1 \) and the weights are non-negative.

Maximize \( \frac{v_1y_{10} + v_2y_{20}}{u_1x_{10} + u_2x_{20}} \) Subject to

\[
\frac{v_1y_{10} + v_2y_{20}}{u_1x_{10} + u_2x_{20}} \leq 1 \text{ for all } 1 \leq i \leq n
\]

\[
u_1x_{10} + u_2x_{20} = 1
\]

By removing the fractional functions, we transform the above into a linear program (LP):

Maximize \( v_1y_{10} + v_2y_{20} \) Subject to

\[
u_1x_{10} + u_2x_{20} = 1
\]

\[
v_1y_{10} + v_2y_{20} - u_1x_{10} + u_2x_{20} \leq 0 \text{ for all } 1 \leq i \leq n
\]

By solving the above LP, we obtain the weights for resolver \( r_0 \) that maximize the efficiency of \( r_0 \). By solving it \( n \) times (setting each resolver as \( r_0 \) in turn), we obtain the efficiency for each resolver. Resolvers with efficiency = 1 are “efficient” and those with lesser efficiency are “inefficient”. The dual of this LP can be used to find the benchmark DMUs for \( r_0 \) i.e., those which use less inputs but produce the same level of output.

One problem with the above LP (analyzed in [Doyle and Green, 1994]) is that the weights assigned by the solver may be “skewed” for some DMUs, who may weigh some favourable parameters very high and ignore the rest. Such DMUs are not efficient in an overall sense; they are good only some niche areas. Cross-efficiency method is designed to identify overall good performers and produce a true ranking of the DMUs in terms of their efficiency. Intuitively, cross efficiency can be thought of as “peer-appraisal”, and the efficiency as “self-appraisal”. Cross efficiency first obtains the most favorable weights for a DMU \( k \) and then evaluates these weights against its peers to compute the average efficiency; i.e., the efficiency of DMU \( k \) as seen by peer DMUs. Formally, an \( n \times n \) cross-efficiency matrix (CEM) is produced in which the entry \( c_{ij} \) (called cross-efficiency) is the efficiency of \( i^\text{th} \) DMU when evaluated with the optimal weights of \( j^\text{th} \) DMU, as identified by the above LP. A DMU is an overall good (bad) performer if it has several high (low) cross-efficiency scores in its column. Thus column average is useful to identify good and bad performers and to produce an effective ranking of the DMUs. A limitation of the cross-efficiency scheme is that the weights obtained after solving the above LP may not be unique.
Table 3 shows the 42 resolvers, along with their cross-efficiencies computed using the above model, who have handled at least 25 tickets of the following problem:

\[ \text{problem\_item} = \text{‘Account Management’} \text{ and } \text{problem\_subitem} = \text{‘Password Reset’} \]

The cross-efficiency of \( j \)th resolver is the average of the \( j \)th column in the cross-efficiency matrix. The question now is: how can we use these cross-efficiencies to identify expert resolvers for this problem? Again, we use one-sample Student’s \( t \)-test. The average and standard deviation of these 42 cross-efficiency values are: \( m = 0.175 \) and \( s = 0.242 \). The \( t \)-value for the resolver ‘221384’ is 19.11, thus rejecting the null hypothesis and clearly identifying this resolver as an expert having an unusually high cross-efficiency score. The 7 resolvers identified as experts using the \( t \)-test on cross-efficiencies are: 241406, 221384, 212929, 175447, 195027, 211063, 217637. All these, except 217637 are identified as the expert for this problem in section 7.1, indicating a high-level of consistency between the two approaches despite the techniques being very different. DEA is an operations research / optimization technique whereas the earlier technique is statistical. Note that the earlier \( t \)-test based technique used only sample size, average ST and STDEV ST whereas our DEA model used \#users as an additional parameter.

### Table 3 Some Resolvers and their Cross-Efficiencies

<table>
<thead>
<tr>
<th>Rank</th>
<th>resolver id</th>
<th>Cross-efficiency</th>
<th>Rank</th>
<th>resolver id</th>
<th>Cross-efficiency</th>
<th>Rank</th>
<th>resolver id</th>
<th>Cross-efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>241406</td>
<td>0.935</td>
<td>15</td>
<td>239180</td>
<td>0.152</td>
<td>29</td>
<td>197857</td>
<td>0.043</td>
</tr>
<tr>
<td>2</td>
<td>221384</td>
<td>0.8904</td>
<td>16</td>
<td>192603</td>
<td>0.1381</td>
<td>30</td>
<td>229093</td>
<td>0.0423</td>
</tr>
<tr>
<td>3</td>
<td>212929</td>
<td>0.8362</td>
<td>17</td>
<td>218101</td>
<td>0.1088</td>
<td>31</td>
<td>229112</td>
<td>0.0423</td>
</tr>
<tr>
<td>4</td>
<td>175447</td>
<td>0.6604</td>
<td>18</td>
<td>221369</td>
<td>0.0972</td>
<td>32</td>
<td>230655</td>
<td>0.0389</td>
</tr>
<tr>
<td>5</td>
<td>195027</td>
<td>0.6075</td>
<td>19</td>
<td>201151</td>
<td>0.0965</td>
<td>33</td>
<td>212534</td>
<td>0.0386</td>
</tr>
<tr>
<td>6</td>
<td>211063</td>
<td>0.3036</td>
<td>20</td>
<td>239467</td>
<td>0.0902</td>
<td>34</td>
<td>210980</td>
<td>0.0382</td>
</tr>
<tr>
<td>7</td>
<td>217637</td>
<td>0.2898</td>
<td>21</td>
<td>205136</td>
<td>0.0798</td>
<td>35</td>
<td>218972</td>
<td>0.031</td>
</tr>
<tr>
<td>8</td>
<td>221366</td>
<td>0.2147</td>
<td>22</td>
<td>201150</td>
<td>0.0651</td>
<td>36</td>
<td>198159</td>
<td>0.0283</td>
</tr>
<tr>
<td>9</td>
<td>212994</td>
<td>0.2107</td>
<td>23</td>
<td>205142</td>
<td>0.0638</td>
<td>37</td>
<td>213058</td>
<td>0.0267</td>
</tr>
<tr>
<td>10</td>
<td>205347</td>
<td>0.1801</td>
<td>24</td>
<td>218099</td>
<td>0.0611</td>
<td>38</td>
<td>205939</td>
<td>0.024</td>
</tr>
<tr>
<td>11</td>
<td>197184</td>
<td>0.1791</td>
<td>25</td>
<td>199563</td>
<td>0.0556</td>
<td>39</td>
<td>229003</td>
<td>0.0238</td>
</tr>
<tr>
<td>12</td>
<td>175446</td>
<td>0.1612</td>
<td>26</td>
<td>208552</td>
<td>0.0477</td>
<td>40</td>
<td>206950</td>
<td>0.0219</td>
</tr>
<tr>
<td>13</td>
<td>179307</td>
<td>0.1602</td>
<td>27</td>
<td>213062</td>
<td>0.0467</td>
<td>41</td>
<td>221374</td>
<td>0.0174</td>
</tr>
<tr>
<td>14</td>
<td>221370</td>
<td>0.1543</td>
<td>28</td>
<td>210246</td>
<td>0.0444</td>
<td>42</td>
<td>218060</td>
<td>0.017</td>
</tr>
</tbody>
</table>

### 8. Identifying Specialist and Generalist Resolvers

As mentioned earlier, we need to identify all specialist and generalist resolvers in the organization along with the focus area for the specialists. As earlier, we assume that we are given a set \( A \) of problem columns and a problem then is a selector (or subset) of tickets, which uses all attributes in \( A \). In the example dataset, \( A = \{ \text{problem\_type, problem\_item, problem\_subitem} \} \) for example. Then a problem is any selector that involves all the 3 columns. A selector that does not involve all 3 columns is not valid. For example, the following is a valid problem:

\[ \text{problem\_item} = \text{‘Access Rights’} \text{ and } \text{problem\_subitem} = \text{‘Home Folder access issue’} \text{ and } \text{problem\_type} = \text{‘Networking’} \]

A resolver \( r \) is a specialist if he/she has handled “very few” problems; \( r \) is a generalist if he/she has handled “too many” problems; \( r \) can be neither a specialist nor a generalist. Note that a specialist is not necessarily an expert in the problems that he has worked on. The notion of expertise depends on the “performance” of the resolver on tickets of a particular problem; whereas the notion of specialist examines the “diversity” in the problems handled by a resolver. Note also that a specialist in problem \( P \) is not necessarily “highly experienced” in \( P \) i.e., he does not necessarily have \( P \) as his/her area of experience (AOE).

Formally, we determine upper and lower limits \( U \) and \( L \), on the number of problems. Any resolver who has handled less than \( L \) problems is a specialist and any resolver who has handled more than \( U \) problems is a generalist. \( r \) is neither a specialist nor a generalist if the number of problems handled by him/her fall between \( L \) and \( U \). We use one-sample Student’s \( t \)-test to determine the values of \( L \) and \( U \) for the given dataset. We illustrate the approach using an example.

Following SQL query computes the number of distinct problems (made up of all the 3 columns in \( A \)) for each resolver:
A total of 54 resolvers have worked on 25 or more tickets. The average number of problems worked on by a resolver is 22 and SDTEV is 8.31. Consider the resolver \( r = '198159' \) who has worked on 382 tickets of 34 distinct problems. We use one-sample Student’s t-test to decide whether 34 is “too many” problems; if yes, then \( r \) is a generalist. The null hypothesis \( H_0 \) is that the sample (of size 54) number of problems for 54 resolvers is obtained from a population with mean 34. Assuming the resolver’s number of problems (34) is the population mean, the value of the \( t \)-statistic is computed as follows (\( n \) is the sample size, \( \mu \) and \( s \) are sample mean and STDEV):

\[
t = \frac{m - \mu}{s} \approx \frac{22 - 34}{8.31} = -10.61
\]

Clearly, probability of obtaining this value of the \( t \)-statistic (for \( df = 52 \)) is less than \( \alpha = 0.01 \). Thus \( H_0 \) is rejected and the resolver \( r \) is a generalist (since 34 > 22). Similarly, we can test whether the resolver \( r = '179265' \), who has handled 185 tickets of only 2 distinct problems. The \( t \)-value for this resolver is 17.7, which is clearly very unlikely; thus this resolver is a specialist (since 2 < 22).

Algorithm identify_specialists_generalists systematically examines all resolvers to find specialists and generalists. Using the algorithm on the sample dataset with \( m = 25 \), we get 15 generalists (including ‘198159’) and 12 specialists (including ‘179265’) out of total 66 resolvers. The team mix is characterized as 12/66 = 18.2% specialists, 15/66 = 22.7% generalists and 39/66 = 59.1% others.

Table 4 shows a summary statistics for top 5 locations (in terms of number of tickets handled). Clearly, specialists tend to have a much lower ST and much lower ST variation (STDEV) as compared to generalists or others. Generalists clearly are most expensive in terms of their ST characteristics, though they are perhaps a “safety net” due to their ability to handle a wide variety of problems. Specialists tend to handle much lower % of tickets; e.g., location ‘YP’ has 17.39% specialists though they handled only 8.65% tickets. The % of tickets handled by generalists is slightly lower than their numbers; e.g., location ‘YP’ has 34.78% generalists who handled 33.12% of tickets. Resolvers who are neither specialists nor generalists tend to handle a much higher workload than their numbers, with ST characteristics which are better than generalists. This suggests an improvement possibility: reduce the % of generalists, and increase that of either specialists or others. All 5 locations have roughly the same % of

\[
\text{select resolver_id, count(*) as n, count(distinct(problem_type + problem_item + problem_subitem)) as d from iSupport_Demo_DBTRDDC_84891..iSupport_ghd where tlevel = 'L1' group by resolver_id having count(*) >= 25 order by count(distinct(problem_type + problem_item + problem_subitem)) asc}
\]
generalists (16%). An important question, that we do not discuss in this paper, is how far can this number be reduced?

Table 4 Summary Statistics for Top 5 Locations

<table>
<thead>
<tr>
<th>client_location</th>
<th>VP</th>
<th>GP</th>
<th>VEP</th>
<th>JMB</th>
<th>BGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>#resolvers</td>
<td>46.00</td>
<td>46.00</td>
<td>46.00</td>
<td>46.00</td>
<td>43.00</td>
</tr>
<tr>
<td>#Specialists</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>6.00</td>
</tr>
<tr>
<td>%Specialists</td>
<td>17.39</td>
<td>17.39</td>
<td>17.39</td>
<td>17.39</td>
<td>13.95</td>
</tr>
<tr>
<td>#Generalists</td>
<td>16.00</td>
<td>16.00</td>
<td>16.00</td>
<td>16.00</td>
<td>16.00</td>
</tr>
<tr>
<td>%Generalists</td>
<td>34.78</td>
<td>34.78</td>
<td>34.78</td>
<td>34.78</td>
<td>37.21</td>
</tr>
<tr>
<td>#Others</td>
<td>22.00</td>
<td>22.00</td>
<td>22.00</td>
<td>22.00</td>
<td>21.00</td>
</tr>
<tr>
<td>%Others</td>
<td>47.83</td>
<td>47.83</td>
<td>47.83</td>
<td>47.83</td>
<td>48.84</td>
</tr>
<tr>
<td>#tickets</td>
<td>1872.00</td>
<td>982.00</td>
<td>948.00</td>
<td>362.00</td>
<td>348.00</td>
</tr>
<tr>
<td>#Tickets resolved by Specialists</td>
<td>162.00</td>
<td>89.00</td>
<td>83.00</td>
<td>40.00</td>
<td>29.00</td>
</tr>
<tr>
<td>%Tickets resolved by Specialists</td>
<td>8.65</td>
<td>9.06</td>
<td>8.76</td>
<td>11.05</td>
<td>8.33</td>
</tr>
<tr>
<td>#Tickets resolved by Generalists</td>
<td>620.00</td>
<td>338.00</td>
<td>318.00</td>
<td>120.00</td>
<td>117.00</td>
</tr>
<tr>
<td>%Tickets resolved by Generalists</td>
<td>33.12</td>
<td>34.42</td>
<td>33.54</td>
<td>33.15</td>
<td>33.62</td>
</tr>
<tr>
<td>#Tickets resolved by Others</td>
<td>1090.00</td>
<td>555.00</td>
<td>547.00</td>
<td>202.00</td>
<td>202.00</td>
</tr>
<tr>
<td>%Tickets resolved by Others</td>
<td>58.23</td>
<td>56.52</td>
<td>57.70</td>
<td>55.80</td>
<td>58.05</td>
</tr>
<tr>
<td>Avg. ST of Specialists</td>
<td>166.00</td>
<td>109.00</td>
<td>534.00</td>
<td>242.00</td>
<td>184.00</td>
</tr>
<tr>
<td>Avg. ST of Generalists</td>
<td>388.00</td>
<td>336.00</td>
<td>277.00</td>
<td>264.00</td>
<td>396.00</td>
</tr>
<tr>
<td>Avg. ST of Others</td>
<td>285.00</td>
<td>265.00</td>
<td>211.00</td>
<td>302.00</td>
<td>221.00</td>
</tr>
<tr>
<td>STDEV ST of Specialists</td>
<td>590.28</td>
<td>151.72</td>
<td>3032.04</td>
<td>668.07</td>
<td>335.89</td>
</tr>
<tr>
<td>STDEV ST of Generalists</td>
<td>1171.91</td>
<td>1088.41</td>
<td>1426.05</td>
<td>838.08</td>
<td>1014.04</td>
</tr>
<tr>
<td>STDEV ST of Others</td>
<td>2063.18</td>
<td>811.11</td>
<td>727.18</td>
<td>775.18</td>
<td>383.01</td>
</tr>
</tbody>
</table>

9. Identifying Experienced Resolvers

As discussed earlier, there is a need to identify the experienced resolvers in ITIS, along with their areas of experience. The business problem that we attempt to answer is the following. A manager often wants to get an overall idea (summary) about a particular employee’s (say, \( r \)) work history. Basically, the manager wants to know on what problems \( r \) has been spending the bulk of her time and efforts. These problems form the main chunks of the employee’s own experience and we call them the areas of experience (AOE) of that employee. An AOE for a resolver \( r \) is a problem where \( r \) has spent a “significant” part of his/her own time and efforts. An example answer is: “\( r \) has been mostly working on database access permissions, data backup/restore for Oracle and mail client configurations problems.”. However, such answers are currently produced informally. We now discuss a formalization of this business question and produce a rigorous answer that is justified by the historical data. The technique analyzes the tickets handled by the given employee \( r \) and then attempts to find common characteristics patterns among these tickets such that each selector accounts for a “significant” chunk of \( r \)’s work history.

As for experts, an area of experience (AOE) for a resolver is a problem defined by a selector, which is a conjunction (AND) of attribute-value pairs. A problem is called an area of experience for a given resolver \( r \) if \( r \) has spent a “significant” part of his/her “efforts” on handling tickets having this problem. There are several ways to formalize the notion of efforts; e.g., the fraction of the ST spent by \( r \) on tickets of this problem. We use another related notion: the proportion of the number of tickets of a problem handled by \( r \) as the “effort” spent by \( r \) on that problem. We can now use the one-sample \( t \)-test, as before, to quantify whether this proportion is large enough to accept that problem as an AOE for that resolver.
As an example, consider the resolver $r = '221384'$ who has handled 311 tickets. At ‘level’ 3 (i.e., using all the 3 columns `problem_type`, `problem_item`, `problem_subitem` as problem descriptors), this resolver has worked on 22 distinct problems. The numbers of tickets handled by $r$ for these 22 problems are: 142, 44, 37, 21, 11, 10, 10, 8, 5, 3, 3, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1. The average and STDEV of these numbers are: 14 and 30.8. We use one-sample $t$-test to establish whether the problem for which $r$ has solved 44 tickets is an AOE or not. The value of the $t$-statistic (with $\mu = 44$, $m = 14$, $s = 30.8$, $n = 22$) is $-4.57$, which means that the null hypothesis is rejected at the significance level of 0.01. Hence this problem is an AOE for this resolver. The problem for which $r$ has solved 142 tickets is also an AOE for this resolver (Table 5). Note that $r$ was earlier identified as an expert for this problem (section 7.1).

### Table 5 AOE for Resolver $r = '221384'$

<table>
<thead>
<tr>
<th>Problem</th>
<th>#Tickets</th>
<th>%Tickets</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>problem_item = 'Account Management'</code> and <code>problem_subitem = 'Password Reset'</code> and <code>problem_type = 'TCS Global Domain'</code></td>
<td>142</td>
<td>45.66</td>
</tr>
<tr>
<td><code>problem_item = 'Messaging Support'</code> and <code>problem_subitem = 'ID File / Password related issues'</code> and <code>problem_type = 'Global Messaging Support (Lotus Notes)'</code></td>
<td>44</td>
<td>14.15</td>
</tr>
</tbody>
</table>

**Algorithm identify_AOE_for_resolver**

```plaintext
algorithm identify_AOE_for_resolver
input D // tickets table
input A = \{A_1, A_2, ..., A_M\} // a subset of M attributes of table D which are to be used
input r // given resolver
output O // set of AOE for the given resolver
// following step is efficiently computed by select – group by query (use all columns in A)
F = set of all (\(\theta_i, n_i\)) where \(\theta_i\) is a problem, \(n_i\) is the number of tickets of problem \(\theta_i\) handled by resolver \(r\)
Let \(m\) and \(s\) be the average and standard deviation of numbers \(n_i\) in \(F\)
Let \(n =\) number of elements in \(F\) // \(n = |F|\)
Sort numbers \(n_i\) in \(F\) in descending order
\(O = \emptyset\)
for each number \(n_i \in F\) do
    Use one sample Student’s $t$-test to find whether null hypothesis is rejected for $\mu = n_i$
    if null hypothesis is rejected then \(O = O \cup \{\theta_i\}\); else break; end if // no further AOE
end for
return \(O\)
```

Algorithm `identify_AOE_for_resolver` systematically examines all problem selectors at the given level (as specified by the given set $A$ of problem columns) for the given resolver $r$ and finds which problems are an AOE for this resolver. This algorithm can be easily generalized to discover the AOE at different levels, using the beam search approach.

### 10. Conclusions and Further Work

This paper raises and proposes solutions for some specific problems in workforce analytics which are important in service industries, particularly in *IT Infrastructure Support (ITIS)* services. Service industries are people-intensive and the knowledge and expertise of the people within an organization is a strategic resource critical for success. Performance of employees in a service organization is directly related to the customer satisfaction and creation of value. We proposed a distinction between three aspects of what makes people valuable in an ITIS organization: expertise, specialization and experience. The paper’s major contributions are novel formalizations of these notions and rigorous statistical and optimization based algorithms to discover these 3 types of people (along with their work areas). In particular, for the important problem of expert discovery, we proposed two separate algorithms: one statistical and another based on data envelopment analysis technique from optimization. The approaches have been implemented and have produced satisfactory results on more than 25 large real-life ITIS datasets, one of which we used for illustration. The algorithms are efficient and work well on large datasets; in one of case study, we applied them on a tickets dataset containing nearly 500,000 tickets, where it took a few minutes to produce the results.
As stated in the Introduction, a basic motivation behind this work is domain-driven data mining in workforce analytics in ITIS. This goal has been successfully achieved; most users of our tools are ITIS managers, without much expertise in data mining. They were able to produce results without much experimentation and found the results directly useful as well as easy to understand and interpret.

One generic issue that needs some discussions is the possibility of “pigeon-holing” (“flooding” or “self-fulfilling loop”) of specific kinds of tickets to a resolver, once he/she is identified as an expert, specialist or experienced. There are actually two distinct issues here: (a) Different problem may differ in their “inherent” difficulty levels; e.g., “replace faulty hard-disk” is much more difficult (i.e., typically takes much more time to resolve) than “reset password”, for example. Is it possible that a person r who is identified an expert in problem θ gets “flooded” with tickets of type θ, and does not get too many tickets of other “simpler” problems? (b) Different tickets of the same problem may differ in their “inherent” difficulty levels; e.g., “replace faulty hard-disk” may be more difficult for Unix machines than Windows machines (an imaginary example). Is it possible that a person r who is identified an expert in problem θ = “replace faulty hard-disk” gets “flooded” with more difficult tickets of this type θ?

Both these issues critically depend on the strategies used for allocating incoming tickets to resolvers. Allocation of tickets to specific resolvers is a very complex process and involves a lot of different factors such as need for balancing the workloads, persons currently available, people’s interests etc. “Job rotation” is often explicitly encouraged as part of the ticket allocation strategy. Further, the difficulty level associated with a ticket is not always clear up front. In the above example, if the details of the OS are not recorded as part of the ticket data, then it is quite difficult to decide a priori (as a ticket of type “replace faulty hard-disk” arrives) the difficult level of a ticket. Very often, the details of the problem become clear after the resolver begins working on it. Also, if a ticket is “too difficult”, it is usually escalated and handled at a different level. Due to all these reasons it is not easy to “flood” particular persons with “difficult” problems or even “difficult” tickets within a specific problem. It can and does happen to some extent but, generally speaking, not strongly enough.

One area where we need to improve is comparing the results obtained using the algorithms and the “wisdom” on the ground. The tool produces supplementary evidence to justify its output (e.g., to support its claim that resolver X is an expert on problem Y). We have mostly used informal discussions with and feedback from the ITIS managers to compare results. While we have observed very high degree of agreement from ITIS managers on the outputs in most case studies, there have been some cases where the managers were somewhat reluctant to accept the tool’s results. At a broad level, the major reason seems to be that the distribution of difficult tickets (of the same problem) was not uniform across all resolvers, even on the same “level” (say L1). Some resolvers get more difficult tickets than others, for some practical reasons not apparent in the tickets data. We have implicitly assumed that all tickets having the same problem characteristics (e.g., problem_type = 'Networking' and problem_item = 'Access Rights' and problem_subitem = 'Home Folder access issue') are equally difficult. We have further assumed that tickets of various difficulty levels (for the same problem) are uniformly distributed across all resolvers. These assumptions are reasonable but not completely accurate. We are investigating ways to make them more realistic. Additional problem details would have indicated factors which make one ticket more difficult than others. We are working on estimating the difficulty level of a ticket and taking that into account to get a more accurate performance measure. One way is to analyze the textual problem and solution descriptions in tickets.

Another difficulty that we have faced is the fact that the service time for a ticket (as mentioned by a resolver or as computed from the open and close timestamps on it) is not always extremely accurate. This “noise” in ST typically does not alter the results much, since it is likely to be uniformly distributed across resolvers and tickets. Nevertheless, this does cause concern as some tickets appear to be “outliers” in terms of their service times. We have options in the tool to detect and remove such “suspect” outlier tickets from the analysis (we use standard outlier detection algorithms). Nevertheless, for producing more accurate results, we are trying to identify corrections that can be applied to ST. For example, it is easy to identify (using ticket timestamps) how many tickets are pending with a resolver at any particular time. Resolvers typically have ways of prioritizing and time-sharing when they have to work on multiple tickets. We are building a model of this behavior which can be taken into account to improve ST estimates.

One critical factor that we used (e.g., for expert discovery) is the measurements of performance of the people on handling specific types of problems in ITIS. This is a somewhat limited view. Expertise is revealed by means other than performance: learning, knowledge sharing, referral patterns etc. For further work, we propose to extend the framework of expert discovery to include these other aspects of the expertise. Most ticketing systems allow the user to add free-form natural language descriptions of the problem they are facing. Similarly, resolvers can add textual descriptions of the diagnosis (cause analysis) of the problem and solution steps that they applied to solve the
problem. This is a rich source of information. We are working on using text mining techniques to extract more fine-grained characterizations of the problems that people have worked on. This would expand the characterization of the work areas of people. Further, we are interested in investigating inter-dependence and interactions among expertise, experience and specialization.

Identifying generic “star” performers (i.e., generic higher ability) is a different and important problem than finding experts for a specific problem. Star performers in one specific problem may or may not indicate a generic high ability; e.g., expertise in “replace faulty hard-disk” may not indicate a generic ability to handle all hardware related tickets. Identifying generic “star” performers will probably require different techniques than the ones considered in our paper.

Identifying the “ideal” team composition in terms of specialists and generalists (and even in terms of experts and experienced persons) is an interesting problem. Unfortunately, the “ideal” team composition depends on the composition of incoming tickets streams. This relationship seems to be quite subtle. Further, there is an element of forecasting involved in forming an ideal team. If we want to form an ideal team for handling the next week’s tickets, then we must know what kinds of tickets to expect next week (not just numbers but also composition of tickets in terms of problems and complexity). This forecasting is quite a complex problem, not the least because there are dependencies among problem types; e.g., tickets of one kind often lead to tickets of another kind after some lag. All these issues make investigations of “ideal” team formation somewhat complex and hence we have not dealt with this problem in this paper.

Acknowledgement

The authors thank the iSupport team for their excellent work in implementation and deployment as well as for useful discussions. We wish to acknowledge the splendid help we have been receiving from the colleagues in TCS (IS practice and iGTM) in terms of connecting us to the clients, data, and keeping us focused on real issues. The authors sincerely thank the reviewers for their helpful comments. The first author would like to thank Dr. Manasee Palshikar.

References


Patterson, B., 2003. Mining the gold: gain competitive advantage through HR data analysis. *HR Magazine*.


Girish Keshav Palshikar obtained an M.Sc. (Physics) from Indian Institute of Technology, Bombay in 1985 and an M.S. (Computer Science and Engineering) from Indian Institute of Technology, Chennai in 1988. Since 1992, he is working as a scientist at the Tata Research Development and Design Centre (TRDDC), Pune, India, where he heads the Machine Learning Group. He has more than 60 publications in international journals and conferences. His areas of research include machine learning and theoretical computer science.

Dr. Harrick Vin is a Vice President and the Chief Scientist of the Tata Research Development and Design Centre (TRDDC) at Tata Consultancy Services (TCS), India. He is also the Global Head for Innovation and Transformation for IT Infrastructure Services at TCS. He is a member of the TCS Corporate Technology Board that oversees all R&D and innovation activities at TCS. Prior to joining TCS, he was a Professor of Computer Sciences at the University of Texas at Austin for 15 years. At UT Austin, he was the founder of the Distributed Multimedia Computing Laboratory and the co-founder of the laboratory for Advanced Systems Research. Over the past 20 years, Harrick has been involved in developing novel solutions in variety of areas in computing, including: one of the first video-over-IP desktop conferencing and collaboration system (1989); one of the earliest system design and implementation of a large-scale video-on-demand server and multimedia file system (1991); one of the first Quality-of-Service-aware Linux kernel (1996); Internet-scale distributed caching solution (1998); web-based content syndication and personalization solution (1999); a digital media entertainment gateway for homes (2002); and a programming environment for multi-core multi-threaded processor architectures (2005). These innovations led to three venture-backed startups in the US during 1999-2004. Harrick has authored more than 125 papers in leading international journals and conferences. Harrick is a recipient of several awards including the Faculty Fellow in Computer Sciences, Dean's Fellowship, National Science Foundation CAREER award, IBM Faculty Development Award,
Fellow of the IBM Austin Center for Advanced Studies, AT&T Foundation Award, National Science Foundation Research Initiation Award, IBM Doctoral Fellowship, NCR Innovation Award, and San Diego Supercomputer Center Creative Computing Award. He has served on the Editorial Board of ACM/Springer Multimedia Systems Journal, IEEE Transactions on Multimedia, and IEEE Multimedia. He has been a guest editor for IEEE Network. He has served as the conference and program chairperson for the premier ACM and IEEE international conferences in the area of multimedia systems and networks; and served as a technical program committee member for many international conferences. Harrick received his PhD in Computer Science and Engineering from the University of California, San Diego in 1993 and Bachelors in Computer Science from the Indian Institute of Technology (IIT), Bombay in 1987.

Vijaya Saradhi did his PhD and M.Sc. (Engg.) in computer science and engineering from Indian Institute of Technology Kanpur, India and Indian Institute of Sciences Bangalore, India respectively. He pursued B. Tech in civil engineering from Nagarjuna University, India. Presently he is associated with Indian Institute of Technology Guwahati as a computer science faculty. His research interests include machine learning and data mining. Prior to joining IIT Guwahati, he worked for Tata Research Development and Design Center, Pune, India.

Mohammed Mudassar is with Systems Research Lab, Tata Research Development and Design Centre. He is currently working as a Scientist in the Machine Learning Group. He has an overall of 6 years of experience in IT. He did his M.Tech. in Computer Cognition Technology from Department of Studies in Computer Science, University of Mysore, India. He obtained the B.E. in Computer Science from Gogte Institute of Technology under Vishveshwariah Technological University, Belgaum, India. His areas of interest are Machine learning, Data Mining, Pattern recognition, Computer Vision and Statistical Modeling and Analysis.