



2014

# Exploring Customer Specific KPI Selection Strategies for an Adaptive Time Critical User Interface

Ingo Keck

*Technological University Dublin, ingo.keck@dit.ie*

Robert Ross

*Dublin Institute of Technology, robert.ross@dit.ie*

Follow this and additional works at: <https://arrow.dit.ie/scschcomcon>

 Part of the [Artificial Intelligence and Robotics Commons](#), [Databases and Information Systems Commons](#), and the [Graphics and Human Computer Interfaces Commons](#)

## Recommended Citation

Keck, I. and Ross, R. Exploring Customer Specific KPI Selection Strategies for an Adaptive Time Critical User Interface. 2014 International Conference on Intelligent User Interfaces, Israel. ACM Digital Library.

This Conference Paper is brought to you for free and open access by the School of Computing at ARROW@TU Dublin. It has been accepted for inclusion in Conference papers by an authorized administrator of ARROW@TU Dublin. For more information, please contact [yvonne.desmond@dit.ie](mailto:yvonne.desmond@dit.ie), [arrow.admin@dit.ie](mailto:arrow.admin@dit.ie), [brian.widdis@dit.ie](mailto:brian.widdis@dit.ie).



This work is licensed under a [Creative Commons Attribution-NonCommercial-Share Alike 3.0 License](#)



# Exploring Customer Specific KPI Selection Strategies for an Adaptive Time Critical User Interface

Ingo R. Keck

Applied Intelligence Research Centre  
Dublin Institute of Technology  
ingo.keck@dit.ie

Robert J. Ross

School of Computing  
Dublin Institute of Technology  
robert.ross@dit.ie

## ABSTRACT

Rapid growth in the number of measures available to describe customer-organization relationships has presented a serious challenge for Business Intelligence (BI) interface developers as they attempt to provide business users with key customer information without requiring users to painstakingly sift through many interface windows and layers. In this paper we introduce a prototype Intelligent User Interface that we have deployed to partially address this issue. The interface builds on machine learning techniques to construct a ranking model of Key Performance Indicators (KPIs) that are used to select and present the most important customer metrics that can be made available to business users in time critical environments. We provide an overview of the prototype application, the underlying models used for KPI selection, and a comparative evaluation of machine learning and closed form solutions to the ranking and selection problems. Results show that the machine learning based method outperformed the closed form solution with a 66.5% accuracy rate on multi-label attribution in comparison to 54.1% for the closed form solution.

## Author Keywords

Data Analytics; Information Filtering; Machine Learning

## ACM Classification Keywords

H.5.2. User Interfaces: User-Centered Design

;

I.2.1. Applications and Expert Systems: Office Automation

## General Terms

Human Factors; Design

## INTRODUCTION

Key Performance Indicators (KPIs) are numeric or categorical measures which are used to describe the operating performance of an organization or individual [10]. In the area of Customer Relationship Management (CRM) (see for example [4] for an introduction) KPIs are frequently used to characterize the relationship between a client and an organization. KPI

measures in the CRM domain range from long term properties such as the total net sales made to a customer, to short term measures such as the length of time that a customer has been left on hold waiting to speak with a company representative.

KPIs are used by organizations to both assess the organization's performance and to tune services and products for clients. Within an organization's Customer Call Centers for example, appropriate KPI information can facilitate a call centre agent to: (a) provide tailored service to customers; (b) to inform the customer of relevant special offers; and (c) to provide quick and efficient support. The importance of common KPI metrics in the CRM domain has been recognized by the developers of CRM and contact center software who have where possible integrated KPI measures into their software platforms.

As the area of Data Analytics and Information Sciences has expanded over the past ten years, the range and quantity of KPIs available to describe the relationship between customer and organization has expanded enormously (see for example [2]). This expansion in the number of useful KPIs presents a challenge to Contact Center Agents and to the developers of Contact Center Software alike. Namely, while it is possible to collect and provide 100s of KPIs to an agent via a conventionally designed CRM interface, a typical Call Center Agent is required to meet stringent throughput-based service level goals, and does hence not have the time available to review a large number of KPI metrics in the first seconds of answering a telephone or chat based customer inquiry.

In our work we have been investigating solutions to the above class of problem through the use of machine learning techniques and other data analytics techniques. In this paper we introduce a specific solution that we have developed to identify the most important KPIs that should be provided to a call center agent at the initial stage of a customer inquiry. This is a description of work in progress and as such we limit our discussion to a short technical overview of this solution and an initial evaluative comparison of the ML based selection strategy against a closed form analytic solution. We proceed by first providing a brief overview of the state of the art in intelligent user interfaces for Business Intelligence systems. We then present an overview of our proposed solution before introducing a publicly available synthesized data set that we have developed for this study. Following this we provide details on the developed KPI selection strategies, as well as a comparative evaluation of those strategies on the synthesized

Submitted to IUI'14.  
Pre-Review Draft.

data set. A brief discussion is also provided before we conclude and outline potential future work.

## RELATED WORK

The aggregation and presentation of KPI information within CRM or call center systems broadly falls into the class of Business Intelligence (BI) [3] software. User Interfaces design for BI software has historically been less than intelligent but has in recent years slowly embraced intelligent features in order to improve usability for less technical users. This work has been industry and development led rather than research led, and initially focused on the integration of multiple interfaces and the embedding of context-sensitive information displays alongside business process interfaces. The development of alerts based on business measures moving beyond some acceptable bounds continue to be considered an important aspect of *intelligence* in BI software, but such alerts are typically hard coded event triggers [7]. More recently active development in BI interfaces has begun to move towards the incorporation of more intelligent techniques such as the incorporation of improved visualization methods, speech based communication, and multi-touch and multi-modal interfaces[1]. While these developments represent progress, BI interfaces generally remain bloated and often require a user to navigate through many pages of content to find important information.

Adaptive Graphical User Interfaces (see for example [5] for an overview) self-modify to provide users with the most appropriate interface for their needs, and are thus of particular relevance to us. There are a number of different types of adaptations possible. These include: (a) device adaptation where an application adapts and conforms to the parameters of the display; (b) presentation adaptation where visual settings are for example auto modified to accommodate users with eyesight limitations; to (c) content adaptation wherein displayed options available to a user vary based on a model of the user’s previous interactions with the system. The dynamically selected collection of programme menu options available in Windows XP through Windows 7 is a classic example of such content based selection. Within the broad CRM and BI domain, Singh recently examined the role of the adaptive user interface in Enterprise Resource Planning systems [13]. While Singh’s analysis was comprehensive, it did however stay focused on issues far removed from our question of information selection for adaptive prompting and focused instead on issues such as the partial activation of menus based on usage context.

Underlying technologies in the recommender domain (see e.g., [12]) such as collaborative filtering provides a useful foundation for information filtering and provision to business users in the enterprise software domain. For example, personalization and adaptation have been used extensively in the CRM domain for the identification of customized services or products that can be offered to the customer [4]. Indeed one of the great applications of data science in the business domain has been the targeting of products and special offers to specific customers. For our purposes however, the output of recommender systems in the classical sense is inappropriate

for KPI selection. In our case the the selection of appropriate output is linked to the intrinsic properties of current data rather than the specific preferences of an individual customer or service agent. The selection of appropriate KPIs for display is therefore closer from a modeling perspective to the content selection process as used in the natural language or text generation communities [11], i.e., our task is to select the most salient of items to present to the user.

## SOLUTION ARCHITECTURE

In order to investigate alternative strategies for the selection of key information to be provided to service agents, we have developed a prototype KPI recommendation application. Running alongside a traditional CRM solution, the prototype application provides company agents with the most appropriate information generated at run-time and customized to each specific customer and case.

An essential part of the design of the system is that for a given company or even division, the recommendation strategies can be tailored for the given environment. While we provide specific details on the selection strategies later in the paper, it is worth noting at this point that we have adopted a *Managed Selection Strategy*. By this we mean that strategies are not learned based on feedback from individual Contact Center Agents or indeed customers. Instead we have adopted a semi-automatic learning system that can be used by IT personnel and Contact Center management to bootstrap and supervise the assistance provided. Our primary reason for doing so was due to feedback from industry partners who indicated that gathering accurate feedback from Contact Center Agents is rarely feasible in a high throughput environment.

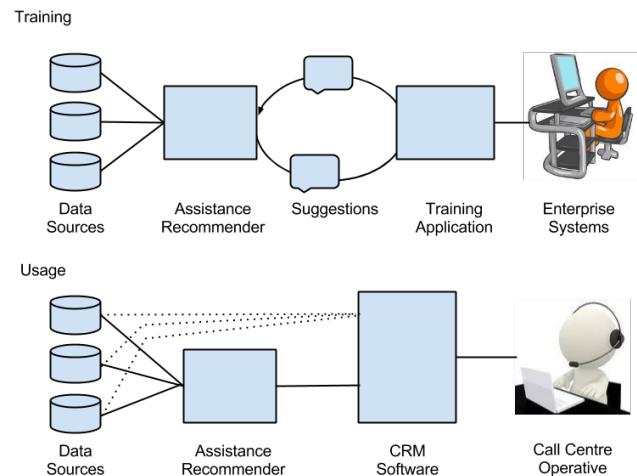


Figure 1. The high-level architecture of the proposed system.

Figure 1 outlines the training and usage models for the enterprise assistance system. During training the assistance recommender is customized by Enterprise Systems or IT personnel through the use of a training application which augments the models used in the assistance system. Once trained, or partially trained, the assistance recommender can then be integrated alongside CRM software to provide key insights from individual customer histories.

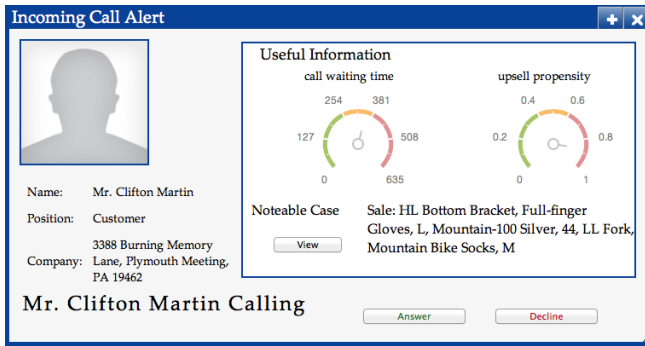


Figure 2. End User Interface to the Selector.

Selected summary items are provided in ‘screen pop’ style information bursts alongside traditional summary information. This information is made available briefly to the sales or service agent on the initiation of a call or chat based interaction by a customer. Figure 2 shows this interface as implemented for the prototype solution. Call center agents can accept the incoming call or chat, reject it, or accept it while clicking through the provided summary items to more comprehensive CRM information screens. While the use of ‘screen pop’ type interfaces are already in use to provide end users with a static selection of information items, our work has concerned the extension of this idea to dynamic content selection and the development of optimal selection strategies.

The selection of relevant metrics and customer history is based on training by IT users or customer service managers. For our prototype design we have implemented an explicit labeling interface which can be used by non-technical users to select the most important items to display to a sales or service agent for a given customer. Specifically this interface displays for a given customer at a given point in time: (a) their key contact and role information; (b) context information such as whether the customer is making a sales or service call; and (c) a selection of metrics that are available to describe the customer-organization relationship at that particular point in time. Designated system developers or managers can then select the most relevant information items for each of a number of training cases.

### DATA CREATION & LABELING

While this project has been developed in part with collaborations with industry partners who have a significant interest in the CRM and call center domains, legal and moral restrictions due to Data Governance and Data Protection legislation mean that it is not always feasible to integrate directly with deployed CRM applications. Moreover, in deploying any solution it is wherever possible useful to evaluate systems and algorithms on publicly available data. In light of these two issues, for this work we have synthesized a comprehensive CRM data set that is indicative of the data used in industry. Here we provide a very brief overview of the synthesized data. It is worth noting that the complete data set is to be made publicly available and published elsewhere.

As indicated earlier the data model itself is typical of many common CRM systems, and has in fact been designed with

a view to easy compatibility with existing systems. With respect to the parametrization or population of the model, the most important issue to consider here relates to the interactive and behavioral aspects of user actions that are captured by the data set. Here the simulated data is based on 5 commonly quoted customer classes (*innovators, early adopters, early majority, late majority and laggards*) with corresponding characteristics. Customers can buy products and services, ask for information and for quotes. Sold products can fail and thus generate customer requests for service or new sales (based on the characteristics of the customer). These constitute contexts for customer call and chat sessions.

For test purposes we simulated a data set of roughly 104,000 interactions, 10,000 customers, 20 agents and 590 products, covering a time span of 2 years. Each customer has 9 KPIs as inherent values (*churn probability, profit generated, upsell propensity, social influence, service load, service contract, sale probability, profit per order and number of sales*), while one KPI value (*call waiting time*) is interaction specific. The distribution type of each KPI varies greatly: Univariate, gaussian, exponential and multimodal distributions are generated in the simulation by the combination of the distinct customer segment characteristics.

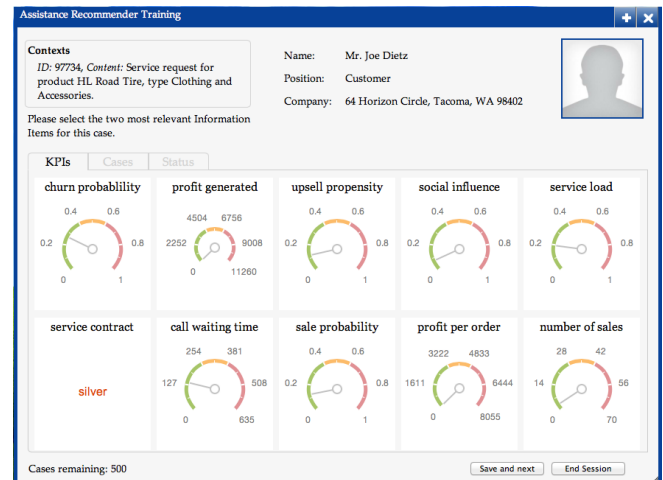


Figure 3. The labeling interface.

A custom front-end has been implemented to aid with the direct labeling of cases from the data set (see Figure 3). Each KPI is presented either in form of a gauge or as text in case of the KPI *service contract*. A human expert can then select up to two KPIs to label as the most important KPIs in the given context of the simulated customer call.

### SELECTION STRATEGIES

From the perspective of intelligent user interfaces the primary technical problem to be addressed is the selection of KPI measures for presentation to the call center agents at the start of every call<sup>1</sup>. For this work we have implemented and evaluated two types of selection strategy for this purpose, i.e., a closed form solution and a classifier based ranking solution.

<sup>1</sup>From this point we are generalizing the use of the term call to telephone based or chat based conversations.

Measure	Annotator Weights	Nelder-Mead-A	Nelder-Mead-B
All Labels	<b>54.2</b>	51.9	54.1
Both Labels	<b>21.6</b>	19.6	<b>21.6</b>
Either Label	<b>86.6</b>	84.2	<b>86.6</b>

Table 1. Summary of Closed Form Solution Results.

### Closed Form Solution

By assuming that KPI measures obey a normal distribution or other well defined distribution we can characterize the problem of most relevant measure selection as a problem of identifying those KPI measures which have the greatest deviation from the location parameter (mean in the case of normal distributions). Namely for a given KPI,  $x$ , we can measure the deviation of the  $i^{th}$  instance of that parameter from the mean for that parameter  $\bar{x}$  as:

$$\delta(x_i) = \frac{x_i - \bar{x}}{\sigma_x} \quad (1)$$

where  $\sigma_x$  is the standard deviation for the parameter  $x$ .

Since the deviation measure  $\delta$  accounts for the relative deviation of a parameter from the mean, we can use  $\delta$  to provide relative prominence values for each KPI. Scoring each KPI measure and ranking the values for  $\delta$  for the given instance we gain a very simple method for estimating the relative deviation of KPI from its expected value. Generalizing over each KPI  $x \in X$  the most prominent KPI is simply that which maximizes the  $\delta$  function, i.e.,

$$\arg \max_x \delta(x_i) = \frac{x_i - \bar{x}}{\sigma_x} \quad (2)$$

Within a given organization all KPIs will not be equally relevant. Moreover, within a given context difference KPIs will have greater or lesser importance in different contexts. For example, within a sales context an *upsell* propensity is more important than *call waiting time*. We therefore assign individual weights  $w_x$  for each KPI,  $x$  as follows:

$$\arg \max_x \delta(x_i) = w_x \frac{x_i - \bar{x}}{\sigma_x} \quad (3)$$

There are a number of advantages to such a closed form solution. Primarily no labeling of data is required for closed form solutions, thus allowing rapid implementation and evaluation of closed form solutions. Moreover, the closed form solution is also extremely transparent and allows business users to explicitly control the weights assigned to different KPIs.

### Classifier Based Ranking Solution

In addition to the closed form solution just outlined, we also implemented a supervised learning based solution to the KPI selection problem. Specifically we can treat KPI selection as a multi-label classification problem [15] where the most relevant  $n$  KPIs for a given instance are labels. This multi-label classification treatment is necessary since it is generally

appropriate to display more than one KPI instance for every given incoming call. In the current case we selected  $n=2$  and subsequently both labeled each instance with the two most applicable labels, and implemented the selection strategy to select the two most appropriate labels. While this constraint is necessary for evaluation purposes, it is generally not necessary to constraint the number of labeled KPIs and selected KPIs in this way.

We first labeled 500 instances from our synthesized data set using the labeling interface introduced earlier. For each of the 500 cases the annotator selected the two most important KPIs based on a specific business rational. This business rational, outlined in the annotation instructions, explained that the priority of the business was to maximize profits while maintaining brand quality perception in customers. Within a real deployment we would expect a more prescriptive and company specific description of objectives to be made available to an annotator in line with business requirements.

The features used for training in the unsupervised learning case were the KPI values themselves and a context feature which defines the business context in which the call was made. Following a standard multi-label classification approach, one binary classifier was then trained per target KPI label. For classification we used Support Vector Machine instances with a sigmoid kernel function [6]<sup>2</sup>. As well as a binary classification decision for the given target KPI, each SVM also provides a class probability based on distance to the class boundary. Comparing class probabilities across SVMs, we may then select the  $n$  most probable KPIs for a given instance.

### RESULTS

The closed form solution described earlier requires relative weights to be assigned to each KPI. We approached the assignment of weights in three ways. In the first approach annotator provided post-hoc estimates were assigned to each KPI weight. In the second approach the weights were instead learned directly from the annotated data using a global Nelder-Mead search [9] starting from 0.5 for all KPI weights (Nelder-Mead-A). As a third method we also applied the Nelder-Mead optimization method but with the annotator provided post-hoc weights as starting point (Nelder-Mead-B). In the latter two cases, 10-fold cross validation was applied in the training and testing process.

For this multi-label classification problem we apply three distinct measures of accuracy in assessing the performance of the KPI selection process:

<sup>2</sup>Our choice of the Support Vector Machine as the baseline classifier was due to its well established accuracy, but other robust high-performance classifiers such as Random Forests [14] would also be applicable.

- **All Label Accuracy** - A count of the total number of correctly attributed labels divided by the total number of attributed labels in the sample.
- **Both Label Accuracy** - A count of the number of cases in which both labels were correctly attributed divided by the total number of cases.
- **Either Label Accuracy** - A count of the number of cases in which either one or both labels were correctly attributed divided by the total number of cases.

Table 1 summarizes the results of the closed form solution application for the three weight assignment methods. We can see from the table that the annotator provided weights alone provided the overall best accuracy, with Nelder-Mead-B or the optimization algorithm with annotator provided weights as starting point coming in second. Referring to the individual measures, both methods performed well on either label accuracy with 86.6% of cases having at least one correctly attributed label. Results for the correct attribution of both labels for a given case are however considerably worse, with only 21.6% of cases having both labels correctly attributed.

For the machine learning based approach we similarly applied 10-fold cross validation for training and testing. Table 2 summarizes accuracy results for the multi-label classification problem. In this 2 label identification problem, 66.5% of labels were identified correctly. In other words, in the case of 50 test cases and hence 100 labels to be identified, on average 66.5 of these 100 labels were correctly attributed. In the case of successfully identifying both labels for a given sample, the raw accuracy rate drops to 39.1% . While this is a poor result if the overall problem is viewed as a single class classification problem, it is less problematic in the multi-label context. This is evidenced when we instead consider the number of cases in which at least one label was identified successfully. For the current data set either of the attributed labels were correct in 93.4% of cases. This means that in the vast majority of cases at least one of the KPIs presented to a call center agent would have been selected by the expert.

Measure	Accuracy
All Labels	66.5
Both Labels	39.1
Either Label	93.4

Table 2. Summary of Multi-Label Classification Results.

While our main consideration is the multi-label results, it is worth also considering results on a class by class basis. Figure 4 presents the F-score (F1) for each individual SVM based classifier. From the results we see that many of the classifiers performed moderately well with F-score values greater than 0.6 in 7 cases. Individual classification for 3 KPIs were however poor. Further investigation shows that this poor performance is most likely due to unbalanced target labels with certain KPIs under represented in the data set. For example KPI-10 accounted for less than 5% of all labels in the current data set.

## DISCUSSION

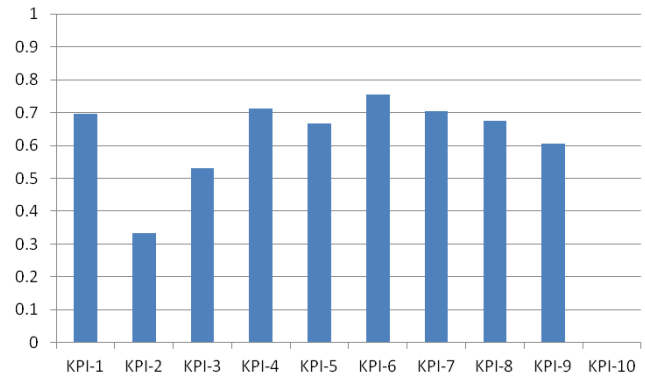


Figure 4. F1 Scores for individual classifiers

The automated provision of business metrics and customer case summary information is far from novel in either a research or industrial setting (consider for example the use of ‘screen pops’ in *Salesforce.com*’s CRM solution<sup>3</sup> for details.). Thus, the question we are investigating in our work is not whether information should be intelligently provided to users through the user interface, but rather what are the best methods which can be used in selecting and summarizing that information for users in time critical environments. The selection of KPIs as outlined here is a starting point in this work.

Considering first the results for the classifier based approach, we see these as a good starting point in terms of KPI recommendation scores. We recognize however that under represented target labels in the training and test data are a problem for the method. With a larger labeled data set and appropriate sampling, we do however expect to lessen this problem. Related to this issue we also recognize the limitation of depending on synthesized data. While we expect real data to be naturally more noisy and necessarily different in nature to the data we have synthesized and worked with, we see the synthesized data as being a necessary and useful first test bed in developing the methods presented here.

The SVM-based supervised learning method considerably outperformed the closed form solution based method in this study. While poor performance in closed form results will be due in part to deficiencies in optimization, it is far more likely that the poor performance is due chiefly to the inaccuracy of annotator based weightings and the invalid nature of the assumptions being made with respect to the data set. Essentially the closed form solution assumes that KPIs are well defined by a normal distribution and that notable values are directly proportional to deviation from a well defined mean. In practice such assumptions do not hold in many cases. Multi-modal data, uniformly distributed data, or data that fits other well defined distributions such as logarithmic distributions will each violate the assumptions made by the basic closed solution model. However we are hopeful that unsupervised learning methods such as those used in novelty detection can provide a more robust framework on which to

<sup>3</sup>See <http://www.salesforce.com>



base an improved closed form solution [8].

### CONCLUSIONS & FUTURE WORK

The model and results presented in this paper are contributed as work in progress rather than a presentation of completed or finalized work. Nevertheless we believe that the work presented constitutes a useful starting point in our analysis in the development and improvement of intelligent interfaces for call center agents and other workers dealing with large volumes of structured data in a time-critical environment. Perhaps unsurprisingly the machine learning based selection strategy outperformed the closed form based solution. Despite this we do see the closed form based approach as being worthy of further development by specializing the method to take account of the particular types of distributions seen in the data.

With respect to the machine learning based selection methods we believe that type of labeling required does not place a great burden on organizations. That said, the policy of pre-labeling a large data set is only a starting point for our deployment method. We are currently developing an active learning based solution which allows training to be performed on a periodic basis, and also provide a more robust framework which can account for potential drift in KPI measures over time.

### ACKNOWLEDGMENTS

xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx  
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx  
xxxx

### REFERENCES

1. Alotaibi, M., and Rigas, D. Fostering the user interface acceptance in customer relationship management: A multimedia-aided approach. In *Information Technology: New Generations (ITNG), 2012 Ninth International Conference on* (2012), 796–801.
2. Baroudi, R. *KPI Mega Library: 17,000 Key Performance Indicators*. CreateSpace, 2010.
3. Chaudhuri, S., Dayal, U., and Narasayya, V. An overview of business intelligence technology. *Communications of the ACM* 54, 8 (2011), 88–98.
4. Dyche, J. *The CRM Handbook: A Business Guide to Customer Relationship Management*. Addison Wesley, 2001.
5. Findlater, L., and Gajos, K. Z. Design space and evaluation challenges of adaptive graphical user interfaces. *AI Magazine* 30 (4) (2009), 68–73.
6. Hearst, M. A., Dumais, S., Osman, E., Platt, J., and Scholkopf, B. Support vector machines. *Intelligent Systems and their Applications, IEEE* 13, 4 (1998), 18–28.
7. Kang, J. G., and Han, K. H. A business activity monitoring system supporting real-time business performance management. In *Third International Conference on Convergence and Hybrid Information Technology* (2008).
8. Markou, M., and Singh, S. Novelty detection: a reviewpart 1: statistical approaches. *Signal processing* 83, 12 (2003), 2481–2497.
9. Olsson, D. M., and Nelson, L. S. The nelder-mead simplex procedure for function minimization. *Technometrics* 17, 1 (1975), 45–51.
10. Parmenter, D. *Key Performance Indicators: Developing, Implementing, and Using Winning KPIs*. John Wiley & Sons, Inc., 2010.
11. Reiter, E., and Dale, R. *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge, U.K., 2000.
12. Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B., Eds. *Recommender Systems Handbook*. Springer, 2011.
13. Singh, A. *Designing Adaptive User Interfaces for Enterprise Resource Planning Systems for Small Enterprises*. PhD thesis, Nelson Mandela Metropolitan University, 2013.
14. Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., and Feuston, B. P. Random forest: a classification and regression tool for compound classification and qsar modeling. *Journal of chemical information and computer sciences* 43, 6 (2003), 1947–1958.
15. Tsoumakas, G., and Katakis, I. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)* 3, 3 (2007), 1–13.