US 20150200822A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2015/0200822 A1**

Zelenko et al. (43) **Pub. Date: Jul. 16, 2015**

(54) **ANALYTICS DRIVEN ENGAGEMENT**

(71) Applicant: **LivePerson, Inc.**, New York, NY (US)

(72) Inventors: **Avinoam Zelenko**, Ra'anana (IL); **Vitaly Gordon**, Bnei Zion (IL); **Arnon-Naftally Kehat**, Zoran (IL); **Matan Barak**, Ra'anana (IL)

(21) Appl. No.: **14/570,963**

(22) Filed: **Dec. 15, 2014**

Related U.S. Application Data

(63) Continuation of application No. 13/413,158, filed on Mar. 6, 2012, now Pat. No. 8,943,002.

(60) Provisional application No. 61/632,815, filed on Feb. 10, 2012.

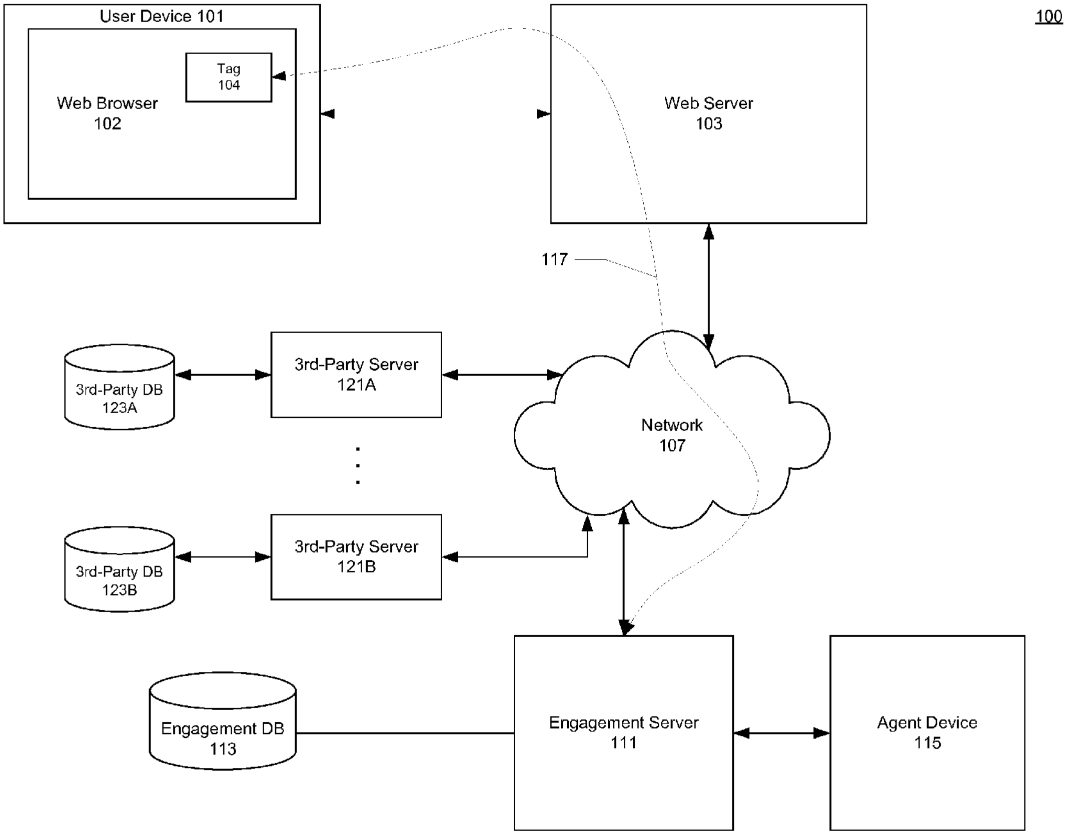
Publication Classification

(51) **Int. Cl.**
H04L 12/26 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 43/04*** (2013.01); ***H04L 43/08*** (2013.01)

(57) **ABSTRACT**

Various embodiments relate to an engagement server and a related method of initiating an engagement between an agent device and a user device based on formed and modified engagement rules. In one embodiment, the engagement server can receive one or more third-party analytic reports and use them to form engagement rules that determine when the engagement server initiates an engagement between the agent device and user device. In some embodiments, the engagement server can also receive user interaction data from the user device to modify engagement rules to fit a particular user. In some embodiments, the engagement server can use the third-party analytic reports and/or user interaction data from a plurality of users to form a statistical model. The engagement server may then use the statistical model to form and/or modify the engagement rules.



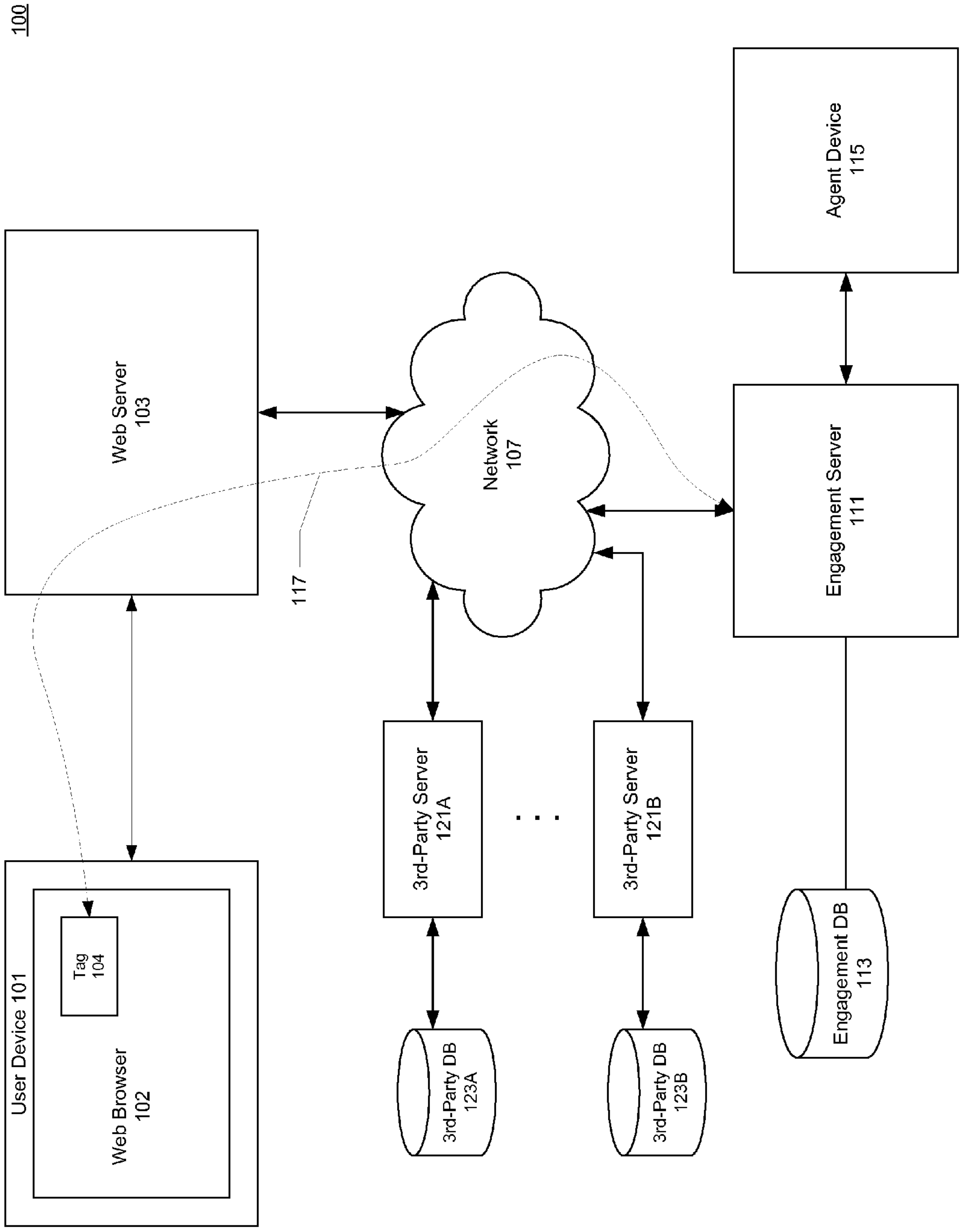

100
Web Server
103
Network
107
117
Agent Device
115
Engagement Server
111
User Device 101
Web Browser
102
Tag
104
3rd-Party Server
121A
3rd-Party Server
121B
. . .
3rd-Party DB
123A
3rd-Party DB
123B
Engagement DB
113

Fig. 1

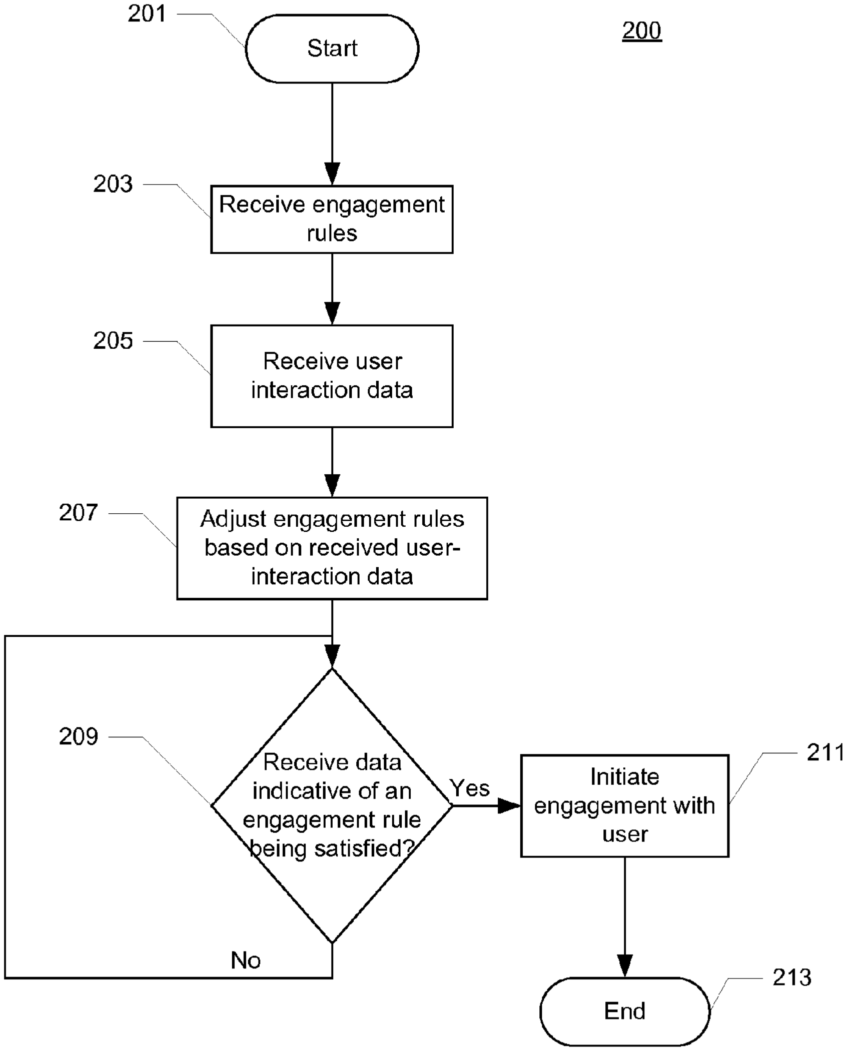

200
201
Start
203
Receive engagement rules
205
Receive user interaction data
207
Adjust engagement rules based on received user-interaction data
209
Receive data indicative of an engagement rule being satisfied?
Yes
No
211
Initiate engagement with user
213
End

Fig. 2

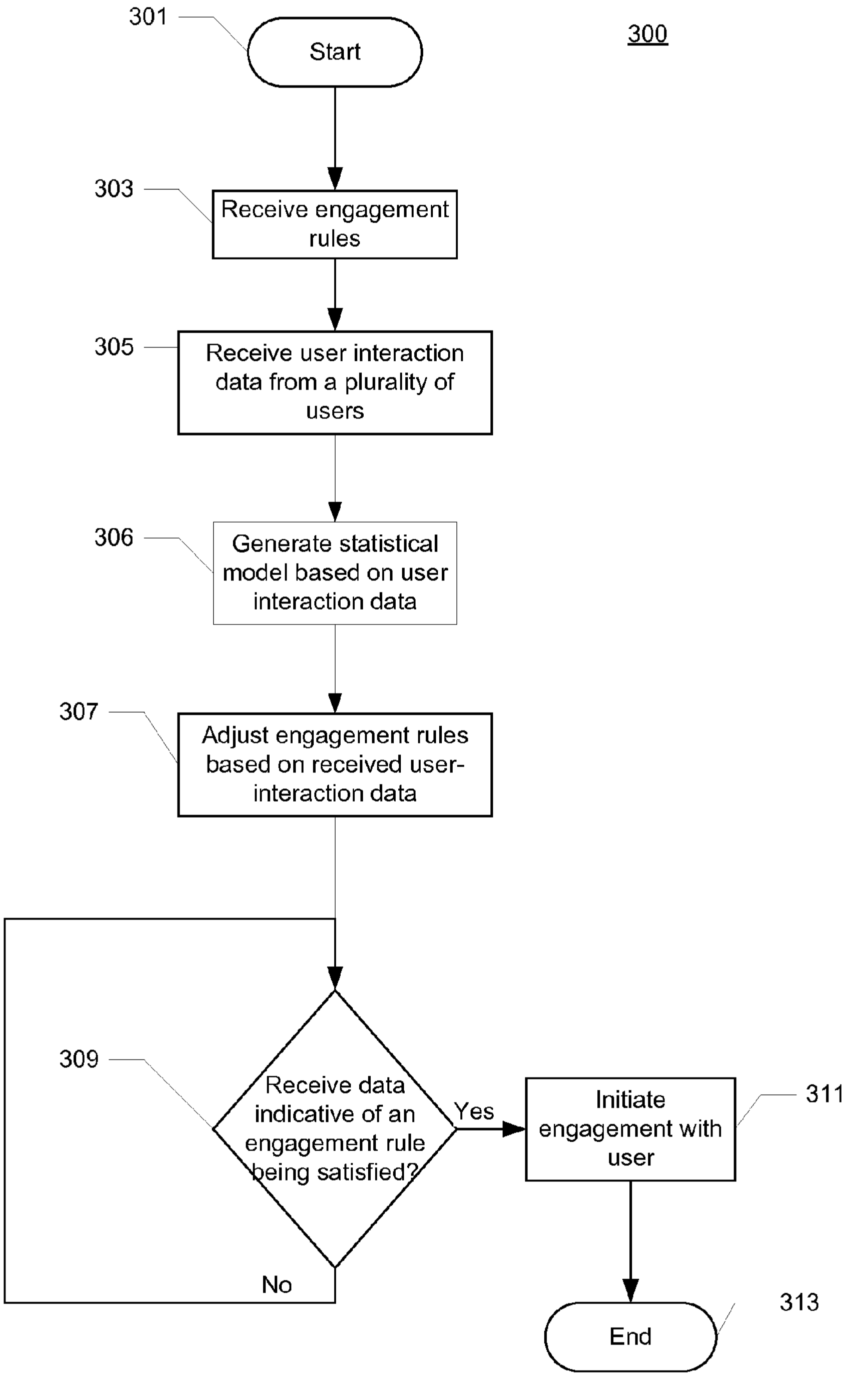

301
Start
300
303
Receive engagement rules
305
Receive user interaction data from a plurality of users
306
Generate statistical model based on user interaction data
307
Adjust engagement rules based on received user-interaction data
309
Receive data indicative of an engagement rule being satisfied?
Yes
No
Initiate engagement with user
311
End
313

Fig. 3

ANALYTICS DRIVEN ENGAGEMENT

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of U.S. patent application Ser. No. 13/413,158, filed on Mar. 6, 2012, entitled "Analytic Driven Engagement," which claims the benefit of an priority under 35 U.S.C. §119(e) to U.S. Provisional Application No. 61/632,815, filed on Feb. 10, 2012, entitled "Analytic Driven Engagement," the entire disclosures of which are incorporated herein by reference.

TECHNICAL FIELD

[0002] Various exemplary embodiments disclosed herein relate generally to methods executed on a computer and computer-based apparatus, including computer program products, for an analytic driven engagement.

BACKGROUND

[0003] A company can use its website in a variety of ways to communicate information to its customers. For example, a company can configure its website to provide information to a customer before the customer makes a purchasing decision. In other instances, the company can directly provide its products and/or services to its customers through its website, provide customer service through a support network, and/or gather specific information about its potential customers.

[0004] In order to implement such a wide array of services (which customers in this day and age have come to expect from Internet companies), many companies must employ large, robust websites and maintain large web presences. One consequence of creating such large and complex websites is that customers many not readily find the information or service they desire. For example, a customer may navigate to a desired location (e.g., web page) on a company's website, yet many not know the appropriate next steps to take once he or she reaches the desired location (e.g., how to complete a purchase transaction, how to compare a displayed product with other products, etc.). It is often difficult to determine when to engage a user of a website (e.g., to provide online chat help, to send coupons, to send advertisements, etc.).

SUMMARY

[0005] The methods and apparatus disclosed herein allow a website to engage website users (or to offer engagement to users) when the users may want or need help (e.g., to offer online chat with customer service agents, to provide coupons, etc.). Actions or other properties associated with the user's interaction with the website can be used to decide when the website should engage with a customer (e.g., offered help).

[0006] A brief summary of various exemplary embodiments is presented. Some simplifications and omissions may be made in the following summary, which is intended to highlight and introduce some aspects of the various exemplary embodiments, but not limit the scope of the invention. Detailed descriptions of a preferred exemplary embodiment adequate to allow those of ordinary skill in the art to make and use the inventive concepts will follow in the later sections.

[0007] In one embodiment, a computerized method for automatically adjusting engagement rules based on user interaction with a website is featured. The method includes a server computer storing a set of engagement rules for a website, wherein each engagement rule defines criteria that, if met, causes the website to engage a user of the website. The method includes the server computer receiving data indicative of a set of users' interactions with the website, wherein the data is generated by a third-party data provider. The method also includes automatically the server computer adjusting one or more generated rules from the set of engagement rules based on the received data.

[0008] In another embodiment, a computer program product, tangibly embodied in a non-transitory computer readable medium is featured. The computer program product includes instructions being configured to cause a data processing apparatus to store a set of engagement rules for a website, wherein each engagement rule defines criteria that, if met, causes the website to engage a user of the website. The computer program product includes instructions being configured to cause a data processing apparatus to receive data indicative of a set of users' interactions with the website, wherein the data is generated by a third-party data provider. The computer program product includes instructions being configured to cause a data processing apparatus to automatically adjust one or more generated rules from the set of engagement rules based on the received data.

[0009] In another embodiment, an apparatus for automatically adjusting engagement rules based on user interactions with a website is featured. The apparatus comprises a processor and memory. The apparatus is configured to store a set of engagement rules for a website, wherein each engagement rule defines criteria that, if met, causes the website to engage a user of the website. The apparatus is configured receive data indicative of a set of users' interactions with the website, wherein the data is generated by a third-party data provider. The apparatus is configured to automatically adjust one or more generated rules from the set of engagement rules based on the received data.

[0010] In other examples, any of the aspects above include one or more of the following features. In some examples, the server computer receives second data indicative of a second set of users' interactions with the website, wherein the second data is generated by a second third-party data provider that is different than the third-party data provider. In some examples, the server computer generates a statistical model based on the received data and the second received data. The received data and the second received data can each comprise one or more parameters. Generating the statistical model can comprise correlating a first parameter from the received data and a second parameter from the second received data using a principal component analysis (PCA).

[0011] In some examples, the statistical model can be indicative of user interaction with the website. The received data can comprise a website analytic report. Engaging a user can comprise transmitting a pop-up window to the user. Automatically adjusting can comprise creating a new engagement rule in the set of engagement rules. Automatically adjusting can comprise changing an existing engagement rule in the set of engagement rules.

[0012] It should be apparent that, in this manner, various exemplary embodiments enable dynamic engagement between an agent and a user on a website (or other user engagements, such as providing targeted advertisements, providing coupons, etc.). Particularly, by using analytical reports and modeling an agent can engage a user under predefined conditions, which can result in a more meaningful engagement between the agent and user.

[0013] The techniques, which include both methods and apparatuses, described herein can provide one or more of the following advantages. The incorporation of one or more third-party analytic tools and analytic reports allows the system to determine when to engage a user (e.g., based on robust statistics) that will more likely result in a successful engagement (e.g., a helpful engagement to the user, an engagement that promotes the sale of products or services, etc.) between agent and user. Automatically integrating such analytic tools with the rest of the engagement system eliminates the need for manual creation and maintenance of engagement rules that define the criterion(s) for when the system engages a user. Other aspects and advantages of the embodiments will become apparent from the following detailed description, taken in conjunction with the accompanying drawings, illustrating the principles of the invention by way of example only.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] In order to better understand various exemplary embodiments, reference is made to the accompanying drawings wherein:

[0015] FIG. **1** illustrates an exemplary analytic-driven engagement system;

[0016] FIG. **2** illustrates a flowchart of an exemplary method of initiating engagement; and

[0017] FIG. **3** illustrates a flowchart of another exemplary method of initiating engagement.

DETAILED DESCRIPTION

[0018] Referring now to the drawings, in which like numerals refer to like components or steps, there are disclosed broad aspects of various exemplary embodiments.

[0019] In general overview, computerized systems and methods are provided for an analytic-driven engagement system that initiates engagement with a user (e.g., engagement between a customer service agent and the user) based on, at least, analytic reports and/or analytic tools. The analytic reports and tools can be produced both internally and can be received from a third party. An engagement can include, for example, direct communications between an agent and a user (e.g., online text chat, voice chat, video chat, etc.). An engagement can also include other forms of action based on a user's interaction with the website. For example, a user can be presented with instructional text or video (e.g., that is automatically-generated based on the user's actions taken at the website), coupons (e.g., for products offered on the website and/or for products related to the website content), articles, community-created content, social media content (e.g., Facebook, Twitter), etc. An engagement server can monitor a user's activities when connected to a website or webpage. The engagement server, using engagement rules dynamically formed from at least the analytic tools, waits until the engagement rules are satisfied before engaging the user. In some embodiments, the engagement server can form the engagement rules from other users' activity data. In some embodiments, the engagement server can initiate the engagement between an agent and the user through the creation of a chat session on both the user device and agent device that enable direct communication between the user and agent. In some embodiments, the engagement server can initiate another form of initiation, such as providing specific content like an advertisement.

[0020] FIG. **1** illustrates an exemplary analytic-driven engagement system **100**, which includes a user device **101**, a web server **103**, a network **107**, an engagement server **111**, an engagement database (DB) **113**, an agent device **115**, third-party servers **121**A-**121**B, and third-party databases **123**A-**123**B. The user device **101** includes a web browser **102** that connects to the web server **103**. The web browser **102** can also include a tag **104**. In the illustrative embodiment, the user device **101** is in communication with the web server **103**, while the engagement server **111** is in communication with the engagement DB **113** and the agent device **115**. In some embodiments, the engagement server **111** can enable communication between the user device **101** and the agent device **115** through the tag **104** in the web browser **102** (e.g., the tag **104** determines an engagement rule is satisfied, and sets up a chat session between the user of the user device **101** and the agent device **115** through the engagement server **111**). In some embodiments, the engagement server **111** can receive information from the tag **104** via the web server **103** and the network **107**, where the web server **103** and the network **107** relay the information between the engagement server **111** and the tag **104** (e.g., as illustrated by connection **117**).

[0021] Network **107** can be, for example, a packet-switching network that is able to forward packets to other devices based on information included in the packet. The network **107** can provide, for example, phone and/or Internet service to various devices like the user device **101** and the agent device **115** in communication with the network **107**.

[0022] The engagement server **111** can be, for example, a single web server with a processor and memory. In some embodiments, the engagement server **111** can include multiple web servers connected directly or through the network **107**. For example, the agent or agent company can improve brand management and/or customer service by connecting its agents with its customers (or users) at what it deems opportune times (e.g., as defined by the engagement rules). Advantageously, the engagement server **111** can receive user interaction data from a particular user device **101** (e.g., from tag **104**) and use various engagement rules to determine when to engage with user device **101** (e.g., to initiate an engagement between an agent and a user via the agent device **115** and the user device **101**). In some embodiments, the engagement server **111** can retrieve other data, such as user analytics stored in engagement database (DB) **113** to form and/or modify its engagement rules. The engagement server **111** can also store the engagement rules in the engagement database **113**. In some embodiments, the engagement server **111** can retrieve one or more third-party analytics, such as third-party analytic reports, from third party databases **123**A-**123**B via third-party servers **121**A-**121**B when forming and/or modifying the stored engagement rules. The engagement server **111** can also provide other software to both the user device **101** and the agent device **115**. For example, the engagement server **111** can provide chat software to the user device **101** and the agent device **115** to facilitate a chat session between the user device **101** and the agent device **115**.

[0023] Engagement database (DB) **113** can be a database in communication with the engagement server **111** that can provide user analytics to the engagement server **111**. In some embodiments, the engagement DB **113** can be in communication with the engagement server **111** through a direct connection. In other embodiments, the engagement DB **113** can be in communication with the engagement server **111** through the network **107**. In some embodiments, the engagement database **113** stores the engagement rules. In some embodiments, the engagement DB **113** can store web analytics received by the engagement server **111**. In some embodiments, the engagement DB **113** can accumulate and store user interaction data the engagement server **111** received from a plurality of user devices **101** over time. In some embodiments, the engagement DB **113** can store statistical models form by the engagement server **111**. The engagement server **111** can access the engagement DB **113** when forming and updating engagement rules, statistical models, and accumulated user interaction data. The engagement server **111** can use information stored in the engagement DB **113** with other information, such as third-party web analytics retrieved from other sources when forming or modifying engagement rules and/or statistical models.

[0024] Third-Party databases (DBs) **123**A-**123**B can be one or more databases in communication with the engagement server **111** through the network **107** and third-party servers **121**A-**121**B. Each of the third-party servers **121**A-**121**B can be, for example, a single web server with a processor and memory that are controlled by a party other than the user or agent. Third-party DBs **123**A-**123**B can accumulate and store web analytics, which the engagement server **111** can access when forming and modifying its engagement rules and/or statistical models. In some embodiments, the web analytics stored in the third-party DBs **123**A-**123**B can be static web analytic reports. In other embodiments, the third-party DBs can include web analytic reports that are consistently updated by the third-party. In some embodiments, the engagement server **111** can access the static and/or updated web analytic reports stored in one or more of the third-party DBs when forming and modifying its engagement rules, or when forming and modifying its statistical models. For example, the engagement server **111** can connect to the first third-party DB **123**A to retrieve a static third-party report, connect to the second third-party DB **123**B to retrieve an updated third-party report, and/or connect to the engagement DB **113** to retrieve accumulated user interaction data to form a statistical model for the user.

[0025] In some embodiments, the statistical model can be generated using partial-least-squares projections to latent structures, (PLS), principal component analysis (PCA), or a combination of PLS and PCA. Further details of PCA and PLS analysis can be found in "Multi- and Megavariate Data Analysis, Part I, Basic Principles and Applications", Eriksson et al, Umetrics Academy, January 2006, and "Multi- and Megavariate Data Analysis, Part II, Advanced Applications and Method Extensions", Eriksson et al., Umetrics Academy, March 2006, the entirety of which are herein incorporated by reference.

[0026] PLS is a multivariate analytical tool commonly used to analyze data. PLS is a method for relating two data matrices, X and Y, to each other by a linear multivariate model. In its simplest form, a linear model specifies the relationship between a dependent or response variable y, or a set of response variables Y, and a set of predictor variables X's. For example, the response variable y is a user-specified quality for irregular objects, and the predictor variables X are the measured data for the irregular objects.

[0027] A PLS component includes a vector of X-scores t, Y-scores u, weights w and c, and loadings p. PLS components of a PLS model are traditionally calculated using the nonlinear iterative partial least squares (NIPALS) algorithm. There are many variations on the NIPALS algorithm, which consist of a matrix-vector multiplication (e.g., X'y) to generate the weight vector w. The matrix-vector multiplication is computed through a set of vector-vector multiplications xk' x y, to result in scalar results w_k.

[0028] PLS can be represented graphically. For example, consider a regression application with N observations, 3 X-variables (factors/predictors), and 1 y-variable (response). Here, the dimensions of the X matrix are 3 columns by N observations, and the dimensions of the Y matrix is 1 column by N observations. Because there are two matrices, each row (or each observation of the N observations) corresponds to two points; one point on the X-space for the X matrix and one in the Y-space for the Y matrix. When the data table is graphed for all N observations, there is a cluster of N points in the X-space and a cluster of N points in the Y-space. The first PLS component is a line, or vector, in the X-space which approximates the cluster of points and provides a good correlation with the y-vector. Upon computing the first PLS component, the co-ordinate of an observation i along the first component vector can be obtained by projecting the sample onto the line to achieve the score t_{i1} of observation i. The scores of all the observations form the first X-score vector t_1. A model estimate of y can be determined by multiplying t_1 by the weight of the y-vector c_1:

$$\hat{y}_{(1)}=c_1t_1 \qquad \text{Equation 1}$$

where:

$\hat{y}_{(1)}$ is an estimate of y after one PLS model component;
c_1 is the weight of the y-vector; and
t_1 is the first X-score vector.

[0029] Usually, one PLS component is insufficient to adequately model the variation in the y-data. A second PLS component is used to expand on the PLS model. The second PLS component is also a line in the X-space, which passes through the origin and is orthogonal to the first PLS component. After computing the second PLS component, a second score vector t_2 is achieved, as well as weights c_2 and w_2:

$$\hat{y}_{(2)}=c_1t_1+c_2t_2 \qquad \text{Equation 2}$$

where:

$\hat{y}_{(2)}$ is an estimate of y after two PLS model components;
c_1 is the weight of the first y-vector;
t_1 is the first X-score vector;
c_2 is the weight of the second y-vector; and
t_2 is the second X-score vector.

[0030] This PLS component generation process is repeated until reaching the desired number of components for the PLS model. Additionally, this can be performed with a single response y or multiple responses Y.

[0031] The vector-vector multiplications of the NIPALS algorithm produce similar results to a least squares estimation of a slope of a line through the origin b=(x'y)/(x'x), where x'x is a constant. Further, partial averages of the sorted data set divided into three portions gives a good estimate of $b=(yy_3-yy_1)/(xx_3-xx_1)$, where the average of the highest third of y is yy_3, the average of the lowest third of y is yy_1, the average of the highest third of x is xx_3, and the average of the lowest third of x is xx_1. While partial averages can be a good estimate of b, partial averages can be affected by outside tails of the distribution. Further, computation of PLS components is limited based on the amount of memory available in the processing system. With large amounts of data storage becoming available at lower prices, the size of data sets are becoming increasingly larger than the computer memory available in the processing system. Consequently PLS processing of large data sets becomes time consuming. Further, standard PLS deflates both the X and Y matrices to speed up the computation with constant vector subtractions.

[0032] PCA is a multivariate projection method designed to extract and display the systematic variation in a data matrix X, revealing groups of observations, trends, and outliers. Data matrix X is a matrix of data with N rows (observations) and K columns (variables). The observations can be, for example, analytical samples, chemical compounds or reactions, process time points of a continuous process, batches from a batch process, biological individuals, trials of a DOE-protocol, and other measurements. To characterize the properties of the observations, variables are measured. The variables can be, for example, of spectral origin, of chromatographic origin, or measurements from sensors in a process (e.g., temperatures, flows, pressures, curves, etc.).

[0033] In some embodiments, the data are pre-processed (e.g., through scaling and mean-centering, described in further detail below). Once pre-processed (if at all), the first principal component (PC**1**) is computed, which is the line in the K-dimensional space that best approximates the data in the least squares sense. The line goes through the average point, and each observation can be projected onto the line to calculate the observation's score. The model can be extended with additional principal components. Usually, one principal component is insufficient to model the systematic variation of a data set. The second principal component, for example, is also represented by a line in the K-dimensional space which is orthogonal to the first PC. The line also passes through the average point, and improves the approximation of the X-data as much as possible.

[0034] To determine which variables are responsible for the patterns seen among the observations, the principal component loadings are analyzed, which are vectors called p_1 and p_2. Geometrically, the principal component loadings express the orientation of the model plane in the K-dimensional variable space. The direction of PC**1** in relation to the original variables is given by the cosine of the angles for each variable. For example, for three variables, the direction is given by a_1, a_2, and a_3. These values indicate how the original variables (e.g., x_1, x_2, and x_3 for a three variable matrix) load, or contribute to, PC**1**. A second set of loading coefficients expresses the direction of PC**2** in relation to the original variables.

[0035] By using PCA a data table X is modeled as:

$$X=1*\bar{x}'+T*P'+E \qquad \text{Equation 3}$$

where:

$1*\bar{x}'$ represents the variable averages after pre-processing the data, if any;

$T*P'$ models the structure; and

E contains noise.

[0036] The principal component scores are the columns of the score matrix T (e.g., t_1, t_2, and t_3 for first, second, and third components). These scores are the coordinates of the observations in the model. The scores can be sorted in descending importance (e.g., t_1 explains more variation than t_2). As discussed above, the meaning of the scores is given by the loadings, which build up the loading matrix P (e.g., loadings of the first, second, and third components are p_1, p_2, and p_3. The loadings demonstrate the magnitude (e.g., large or small correlation) and the manner (e.g., positive or negative correlation) in which the measured variables contribute to the scores.

[0037] In some embodiments, the data matrices are preprocessed before performing PCA and/or PLS. For example, when using PLS, one or more of the matrices (e.g., matrix X and/or matrix Y) are transformed, centered, and/or scaled by a preprocessing method. Similarly, prior to PCA, the data can be pre-treated into a form suitable for analysis (e.g., to reshape the data such that important assumptions are better fulfilled). PLS modeling works best when the data are generally symmetrically distributed and have a generally constant error variance. Variables that vary more than ten-fold can be logarithmically transformed before the analysis to remove undesired behavior. Transforming variables can improve the predictive power and interpretability of a multivariate model. For example, a dataset which includes measurements that are outliers may unduly influence model building. Manipulating such measurements in some way prior to data analysis prevents the measurements from exerting a large influence on the model, causing the measurements to dominate over the other measurements. For example, outliers can be removed to minimize the effect that the measurements would have on the model.

[0038] To give variables (i.e., columns) of a matrix relatively or approximately equal weight in the subsequent analysis, the data can be column-wise transformed, scaled, and/or centered. Transformations of variables are often used to give them a more symmetrical distribution. For example, logarithmic transformations, negative logarithm scaling, logit scaling, square root scaling, fourth root scaling, inverse scaling, or power transformation scaling can be used.

[0039] For many types of data, centering and scaling are intertwined. Centering corresponds to a subtraction of a reference vector, where scaling the variables involves multiplying the variables by a scaling vector. The choice of scaling vector is crucial. In situations where variables of different origin and numerical range are encountered the scaling vector is usually chosen as the inverse spread of the variables. In other situations, such as with process data, the scaling vector may be defined relative to a tolerable spread in the variables.

[0040] A scaling vector is a representation of a number of observations on a line such that the positioning of the points is related in some mathematical sense to one or more criteria relevant to the observations. The scaling process involves regulating the length of a coordinate axis in variable space according to a predetermined criteria (e.g., that the length of each coordinate axis be set to the same variance). A common technique for scaling data is referred to as "unit variance," "UV" scaling, or "auto-scaling." Unit variance scaling involves calculating a standard deviation for a particular variable from a data set. A scaling weight is calculated as the inverse of the standard deviation. Each value of the variable is multiplied by the scaling weight to determine the scaled variable. After all of the variables in the data matrices have been scaled, each of the variables (i.e., coordinate axes) have unit variance.

[0041] In FIG. **1**, The web server **103** can be, for example, a single web server with a processor and memory. In other embodiments, the web server **103** can be a plurality of web servers configured to provide web services to a user device **101**. The website can provide tag **104** along with website code (e.g., HTML code), comprising code (e.g., JavaScript code) that, when executed by the web browser **102** on the user device **101**, monitors user activity of an associated website displayed in the web browser **102**. In some embodiments, instead of or in addition to the tag **104**, the website provided by the web server **103** can provide an application programming interface (API) that facilitates monitoring user activity, such as the monitoring of keywords in messages, or other data used within the website (e.g., which can be viewed by the user device **101**). The API can be an interface implemented by the website that enables other software programs to interact with the website. In some embodiments, the web server **103** can provide a same website viewed on both the user device **101** via the web browser **102** and the agent device **115** (e.g., via a web browser on the agent device **115**, not shown). In such instances, the website viewed on the agent device **115** and user device **101** may not be identical.

[0042] The user device **101** can be computing devices with a processor and memory that can interact with the website through its web browser **102**. For example, the user device **101** can include desktop computers, laptop computer, tablet computers, and/or mobile phones connected to the network **107**. A user of the user device **101** can use the user device **101** to connect to the website provided by the web server **103** and can be engaged with an agent device (e.g., agent device **115**) through the engagement server **111**.

[0043] The web browser **102** can be software used by the user device to connect to other devices through the network **107**. In some embodiments, the user device can be connected to the web server **103** before connecting to the network **107**. In other embodiments, the web server **103** can be connected to the user device **101** through the network **107**.

[0044] Tag **104** can be a coded tag, such as an HTML tag, that can be included in a web page the web server **103** provides to the user device **101** through the web browser **102**. In some embodiments, the tag **104** can be a tag module stored in the memory of the user device **101** that communicates with the web server **103** and/or the engagement server **111** via the connection **117**. For example, when the user causes the web browser **102** to load the web page, the web browser **102** processes the tag **104**, which causes the web browser **102** to download default tag code (e.g., a JavaScript code file) from the web server **103**. In some embodiments, the engagement server **111** can provide the default tag code to the web browser **102**. The web browser **102** can receive, process, and execute the default tag code to generate the tag. The default tag code can contain code/instructions that monitor and transmit information indicative of the user's interaction with the web page (e.g., user interaction data) to the web server **103**, the engagement server **111**, or both. For example, the user interaction data can include mouse clicks, form entries, and Uniform Resource Locator (URL) history. Combinations of one or more user interactions can trigger actions by the engagement server **111**, which is described below with reference to FIGS. **2** and **3**.

[0045] Agent device **115** can be computing devices with a processor and memory that can connect to the engagement server **111** and can be in communication with the user device **101**. For example, the agent device **115** can include desktop computers, laptop computers, tablet computers, and/or mobile phones connected to the network **107**. The agent using the agent device **115** can use the agent device **115** to connect to the user device through an engagement (e.g., chat, video chat, phone chat, etc.) initiated by the engagement server **111**. Once engagement is initiated, the agent device **115** and user device **101** can be in communication with each other through the network **107**. This can allow, for example, engagement between the agent and user through interactions such as a live chat session.

[0046] FIG. **2** illustrates a flowchart of an exemplary method **200** for initiating engagement with a user (e.g., a user browsing a website). For example, the engagement server **111** can run method **200** to determine whether to connect the agent device **115** to the user device **101**. Method **200** starts at step **201** and proceed to step **203**, where the engagement server **111** receives and stores engagement rules (e.g., receives and stores them in engagement DB **113**). These engagement rules can be rules for determining when (e.g., based on certain user interactions with a website, based on the passage of a preconfigured amount of time, etc.) to connect the agent device **115** to the user device **101** for direct engagement (or mediated engagement), or for the web server **103** or the engagement server **111** to provide specific content to the user device **101** (e.g., a coupon, a targeted advertisement, etc.). For example, an engagement rule can be configured such that the engagement server **111** initiates a chat session between the agent device **115** and the user device **101** when the user device **101** has been idle on the same web page within a website for a specified period. In some embodiments, the engagement rule can be generated based on previous third-party analytics. For example, the engagement server **111** can retrieve one or more third-party analytic reports from third-party databases **123**A-**123**B, which can indicate that users have a higher bounce rate once they reach a specific web page (e.g., a "FAQ" or "Help" web page). The engagement server **111** can then create and store an engagement rule that includes a shortened timer when a user device **101** is visiting one of these specific, "high-bounce" web pages.

[0047] The engagement server **111** can then proceed to step **205**, where it receives user interaction data from the user device **101**. For example, the tag **104** included in the user device **101** (e.g., downloaded from web server **103**) can track and record user interaction data within the web browser **102**. In some embodiments, the tag **104** sends the user interaction data to the web server **103**, which can then send the user interaction data through the network **107** to the engagement server **111**. In some embodiments, the tag **104** sends the user interaction data directly to the engagement server via connection **117**. In some embodiments, the tag **104** can send the user interaction data to the engagement server **111** in packets at standard intervals, where the engagement server **111** can accrue the packets of data. In other embodiments, the tag **104** can wait for triggers to send user interaction data to the engagement server **111**. This can, for example, help reduce the volume of user interaction data sent to the engagement server **111**.

[0048] After receiving the user interaction data, the engagement server **111** can then proceed to step **207**, where it can adjust the engagement rules. In some embodiments, the engagement server **111** can adjust the previously-stored engagement rules to, for example, trigger based on different conditions. For example, the engagement server **111** in step **205** may receive user interaction data that indicates that the user is spending longer-than-average times on each webpage before traversing to another page within the website. The engagement server **111** can then adjust engagement rule to have a longer trigger time so that the engagement server **111** initiates an engagement between the agent device **115** and the user device **101** after a longer period (e.g., to prevent initiating an engagement before a user may need help).

[0049] In step **209**, the engagement server **111** determines whether it has received data indicative of an engagement rule being satisfied. In some embodiments, the engagement server

111 can monitor the user interaction data received in step **205** and compare the user interaction data against a plurality of engagement rules, initiating an engagement between the agent device **115** and the user device **101** when the conditions of any of the plurality of engagement rules is satisfied. For example, the engagement server **111** can be using an engagement rule that waits until a user has been on a specific web page (e.g., an FAQ page) for at least a specified duration. When the engagement server **111** reviews the received user interaction data to determine that the user has been on the specific web page for longer than the specified duration, the engagement server may then proceed to step **211** and initiate an engagement between with user device **101** (e.g., present the user device **101** with a popup, set up a service session between the user device **101** and the agent device **115** (e.g., a chat session, a video session, etc.), etc.). Otherwise, the engagement server **111** can loop back to step **209** (e.g., to analyze newly received user interaction data) to determine whether any of its engagement rules have been satisfied.

[0050] In step **211**, the engagement server **111** can initiate an engagement with the user. For example, the engagement server **111** can initiate a chat session between the agent and the user via the agent device **115** and the user device **101**. In some embodiments, the engagement server **111** can initiate an engagement with the user through other means, such as providing additional text or video instructions based on the engagement rule satisfied. For example, the engagement server **111** can monitor an engagement rule regarding a user comparing two or more similar products. Once the engagement server **111** determines that the comparison engagement rule has been satisfied, the engagement server **111** can provide a stored video comparison of the two or more products. In some embodiments, the engagement server can use other means through JavaScript and HTML to engage with the user device **101**. This can include, for example, providing to the web browser **102** a banner, a tool-tip, hover-over text, and similar methods to provide more information to the user. Once the engagement has been initiated, the engagement server **111** can stop method **200** at step **213**.

[0051] FIG. **3** illustrates a flowchart of another exemplary method of initiating engagement. Similar to the method **200** (e.g., implemented by the engagement server **111**), the engagement server **111** can implement method **300** to incorporate web analytics from various sources along with received user interaction data to determine whether to initiate an engagement with the user device **101**. Method **300** begins at step **301** and proceeds to step **303**, where the engagement server **111** receives and stores one or more engagement rules. Similar to step **203**, the engagement server **111** can receive one or more engagement rules that may indicate when to initiate an engagement with the user device **101** and the agent device **115**. In some embodiments, the engagement server **111** can incorporate one or more third-party analytics, such as multiple third-party analytic reports, from one or more third-party databases **123**A-**123**B via third-party servers **121**A-**121**B.

[0052] In step **305**, the engagement server **111** can receive user interaction data from a plurality of user devices similar to the user device **101** in FIG. **1**. Similar to step **205** in method **200**, the engagement server **111** can receive data that indicates how a particular user is interacting with the site. In step **305**, the engagement server **111** can receive user interaction data from a plurality of user devices **101**, such as simultaneously receiving data from multiple user devices **101** in parallel. In some embodiments, the engagement server **111** can accumulate the user interaction data from the plurality of users and store this data for later use. The engagement server **111** can track different data for different individual users; this data may still be used together by the engagement server **111** in later use.

[0053] The engagement server **111** can then proceed from step **305** and proceed to step **306**, where the engagement server **111** generates a statistical model based on the accumulated user interaction data. In some embodiments, the engagement server can use various third-party analytics received from the plurality of third-party databases **123**A-**123**B, in addition to stored user interaction data and the user interaction data received in step **305**. The engagement server **111** can form the statistical model to create a "model" user profile, for example, that can represent the expected actions of a typical, default user. In such instances, the engagement server **111** bases its engagement decisions on the default statistical profile that can be constantly updated as the engagement server **111** receives new user interaction data. In some embodiments, the engagement server **111** can use the statistical model to form a plurality of user profiles that it may use to model each of the plurality of user devices for which it is receiving user interaction data.

[0054] After creating the statistical model, the engagement server **111** can then proceed to step **307**, where it can adjust the engagement rules based on the received user interaction data. In some embodiments, the engagement server **111** can use the user interaction data to adjust the engagement rules. In other embodiments, the engagement server **111** can first use the statistical model to adjust the engagement rules and subsequently use the user interaction data to further adjust the engagement rules. The engagement server **111** may act in a similar manner to that of step **207**, as the engagement server **111** can use the user interaction data along with the statistical model to modify the previously received and stored engagement rules so that the engagement server **111** initiates an engagement between the agent device **115** and the user device **101** at more opportune times.

[0055] In order to determine whether it is an opportune time to initiate an engagement between the user device **101** and the agent device **115**, the engagement server **111** may proceed to step **309**, where the engagement server **111** determines whether it has received user interaction data from a user device **101** indicative of any of the engagement rules being satisfied. In some embodiments, the engagement server **111** can constantly monitor multiple user devices **101** and can determine whether one of the plurality of user devices **101** satisfied its conditions for engagement. If not, the engagement server **111** can loop back to step **309** and wait until a condition is satisfied.

[0056] When a condition is satisfied, the engagement server **111** can proceed to step **311**, where the engagement server **111** initiates an engagement with the user device **101**. Similar to step **211**, the engagement server **111** can initiate direct communication, such as, for example, a text or video chat session, between an agent and user via the agent device **115** and the user device **101**. Once the engagement server **111** successfully initiates an engagement with the user device **101**, it may proceed to step **313** and end method **300**.

[0057] The above-described computerized methods and apparatuses can be implemented in digital and/or analog electronic circuitry, or in computer hardware, firmware, software, or in combinations of them. The implementation can be as a computer program product, i.e., a computer program tangibly embodied in a machine-readable storage device, for execution by, or to control the operation of, a data processing apparatus, e.g., a programmable processor, a computer, and/or multiple computers. A computer program can be written in any form of computer or programming language, including source code, compiled code, interpreted code and/or machine code, and the computer program can be deployed in any form, including as a stand-alone program or as a subroutine, element, or other unit suitable for use in a computing environment. A computer program can be deployed to be executed on one computer or on multiple computers at one or more sites.

[0058] The computerized method steps can be performed by one or more processors executing a computer program to perform functions of the invention by operating on input data and/or generating output data. Method steps can also be performed by, and an apparatus can be implemented as, special purpose logic circuitry, e.g., a FPGA (field programmable gate array), a FPAA (field-programmable analog array), a CPLD (complex programmable logic device), a PSoC (Programmable System-on-Chip), ASIP (application-specific instruction-set processor), or an ASIC (application-specific integrated circuit). Subroutines can refer to portions of the computer program and/or the processor/special circuitry that implement one or more functions.

[0059] Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital or analog computer. Generally, a processor receives instructions and data from a read-only memory or a random access memory or both. The essential elements of a computer are a processor for executing instructions and one or more memory devices for storing instructions and/or data. Memory devices, such as a cache, can be used to temporarily store data. Memory devices can also be used for long-term data storage. Generally, a computer also includes, or is operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto-optical disks, or optical disks. A computer can also be operatively coupled to a communications network in order to receive instructions and/or data from the network and/or to transfer instructions and/or data to the network. Computer-readable storage devices suitable for embodying computer program instructions and data include all forms of volatile and non-volatile memory, including by way of example semiconductor memory devices, e.g., DRAM, SRAM, EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto-optical disks; and optical disks, e.g., CD, DVD, HD-DVD, and Blu-ray disks. The processor and the memory can be supplemented by and/or incorporated in special purpose logic circuitry.

[0060] To provide for interaction with a user, the above described techniques can be implemented on a computer in communication with a display device, e.g., a CRT (cathode ray tube), plasma, or LCD (liquid crystal display) monitor, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse, a trackball, a touchpad, or a motion sensor, by which the user can provide input to the computer (e.g., interact with a user interface element). Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, and/or tactile input.

[0061] The above described techniques can be implemented in a distributed computing system that includes a back-end component. The back-end component can, for example, be a data server, a middleware component, and/or an application server. The above described techniques can be implemented in a distributed computing system that includes a front-end component. The front-end component can, for example, be a client computer having a graphical user interface, a Web browser through which a user can interact with an example implementation, and/or other graphical user interfaces for a transmitting device. The above described techniques can be implemented in a distributed computing system that includes any combination of such back-end, middleware, or front-end components.

[0062] The computing system can include clients and servers. A client and a server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

[0063] The components of the computing system can be interconnected by any form or medium of digital or analog data communication (e.g., a communication network). Examples of communication networks include circuit-based and packet-based networks. Packet-based networks can include, for example, the Internet, a carrier internet protocol (IP) network (e.g., local area network (LAN), wide area network (WAN), campus area network (CAN), metropolitan area network (MAN), home area network (HAN)), a private IP network, an IP private branch exchange (IPBX), a wireless network (e.g., radio access network (RAN), 802.11 network, 802.16 network, general packet radio service (GPRS) network, HiperLAN), and/or other packet-based networks. Circuit-based networks can include, for example, the public switched telephone network (PSTN), a private branch exchange (PBX), a wireless network (e.g., RAN, bluetooth, code-division multiple access (CDMA) network, time division multiple access (TDMA) network, global system for mobile communications (GSM) network), and/or other circuit-based networks.

[0064] Devices of the computing system and/or computing devices can include, for example, a computer, a computer with a browser device, a telephone, an IP phone, a mobile device (e.g., cellular phone, personal digital assistant (PDA) device, laptop computer, electronic mail device), a server, a rack with one or more processing cards, special purpose circuitry, and/or other communication devices. The browser device includes, for example, a computer (e.g., desktop computer, laptop computer) with a world wide web browser (e.g., Microsoft® Internet Explorer® available from Microsoft Corporation, Mozilla® Firefox available from Mozilla Corporation). A mobile computing device includes, for example, a Blackberry®. IP phones include, for example, a Cisco® Unified IP Phone 7985G available from Cisco System, Inc, and/or a Cisco® Unified Wireless Phone 7920 available from Cisco System, Inc.

[0065] One skilled in the art will realize the invention may be embodied in other specific forms without departing from the spirit or essential characteristics thereof. The foregoing embodiments are therefore to be considered in all respects illustrative rather than limiting of the invention described herein. Scope of the invention is thus indicated by the appended claims, rather than by the foregoing description, and all changes that come within the meaning and range of equivalency of the claims are therefore intended to be embraced therein.

We claim:

1. A computerized method for automatically adjusting engagement rules based on user interactions with a website, the method comprising:

storing, by a server computer, a set of engagement rules for a website, wherein each engagement rule defines criteria that, if met, causes the website to engage a user of the website;

receiving, by the server computer, data indicative of a set of users' interactions with the website, wherein the data is generated by a third-party data provider; and

automatically adjusting, by the server computer, one or more generated rules from the set of engagement rules based on the received data.

2. The method of claim **1**, further comprising:

receiving, by the server computer, second data indicative of a second set of users' interactions with the website, wherein the second data is generated by a second third-party data provider that is different than the third-party data provider; and

generating, by the server computer, a statistical model based on the received data and the second received data.

3. The method of claim **2**, wherein:

the received data and the second received data each comprise one or more parameters; and

generating the statistical model comprises correlating a first parameter from the received data and a second parameter from the second received data using a principal component analysis (PCA).

4. The method of claim **2**, wherein the statistical model is indicative of user interaction with the website.

5. The method of claim **1**, wherein the received data comprises a website analytic report.

6. The method of claim **1**, wherein engaging a user comprises transmitting a pop-up window to the user.

7. The method of claim **1**, wherein automatically adjusting comprises creating a new engagement rule in the set of engagement rules.

8. The method of claim **1**, wherein automatically adjusting comprises changing an existing engagement rule in the set of engagement rules.

9. A computer program product, tangibly embodied in a non-transitory computer readable medium, the computer program product including instructions being configured to cause a data processing apparatus to:

store a set of engagement rules for a website, wherein each engagement rule defines criteria that, if met, causes the website to engage a user of the website;

receive data indicative of a set of users' interactions with the website, wherein the data is generated by a third-party data provider; and

automatically adjust one or more generated rules from the set of engagement rules based on the received data.

10. The computer program product of claim **9** further comprising instructions being configured to cause a data processing apparatus to:

receive second data indicative of a second set of users' interactions with the website, wherein the second data is generated by a second third-party data provider that is different than the third-party data provider; and

generate a statistical model based on the received data and the second received data.

11. The computer program product of claim **10**, wherein:

the received data and the second received data each comprise one or more parameters; and

instructions to generate the statistical model comprises instructions to correlate a first parameter from the received data and a second parameter from the second received data using a principal component analysis (PCA).

12. The computer program product of claim **10**, wherein the statistical model is indicative of user interaction with the website.

13. The computer program product of claim **9**, wherein the received data comprises a website analytic report.

14. The computer program product of claim **9**, wherein engaging a user comprises transmitting a pop-up window to the user.

15. An apparatus for automatically adjusting engagement rules based on user interactions with a website, the apparatus comprising a processor and memory configured to:

store a set of engagement rules for a website, wherein each engagement rule defines criteria that, if met, causes the website to engage a user of the website;

receive data indicative of a set of users' interactions with the website, wherein the data is generated by a third-party data provider; and

automatically adjust one or more generated rules from the set of engagement rules based on the received data.

16. The apparatus of claim **15**, further configured to:

receive second data indicative of a second set of users' interactions with the website, wherein the second data is generated by a second third-party data provider that is different than the third-party data provider; and

generate a statistical model based on the received data and the second received data.

17. The apparatus of claim **16**, wherein:

the received data and the second received data each comprise one or more parameters; and

generating the statistical model comprises correlating a first parameter from the received data and a second parameter from the second received data using a principal component analysis (PCA).

18. The apparatus of claim **16**, wherein the statistical model is indicative of user interaction with the website.

19. The apparatus of claim **15**, wherein the received data comprises a website analytic report.

20. The apparatus of claim **15**, wherein engaging a user comprises transmitting a pop-up window to the user.

* * * * *