US 20150293899A1

(19) United States

(12) Patent Application Publication

Baughman et al.

(10) Pub. No.: US 2015/0293899 A1

(43) Pub. Date: Oct. 15, 2015

(54) MOBILE BASED LEXICON AND FORECASTING

(71) Applicant: **INTERNATIONAL BUSINESS MACHINES CORPORATION**, Armonk, NY (US)

(72) Inventors: **Aaron K. Baughman**, Silver Spring, MD (US); **Blaine H. Dolph**, Western Springs, IL (US); **Kamran R. Khan**, Lombard, IL (US); **Carlos A. Paez, JR.**, Chicago, IL (US); **Palani Sakthi**, Palatine, IL (US)

(73) Assignee: **INTERNATIONAL BUSINESS MACHINES CORPORATION**, Armonk, NY (US)

(21) Appl. No.: **14/250,748**

(22) Filed: **Apr. 11, 2014**

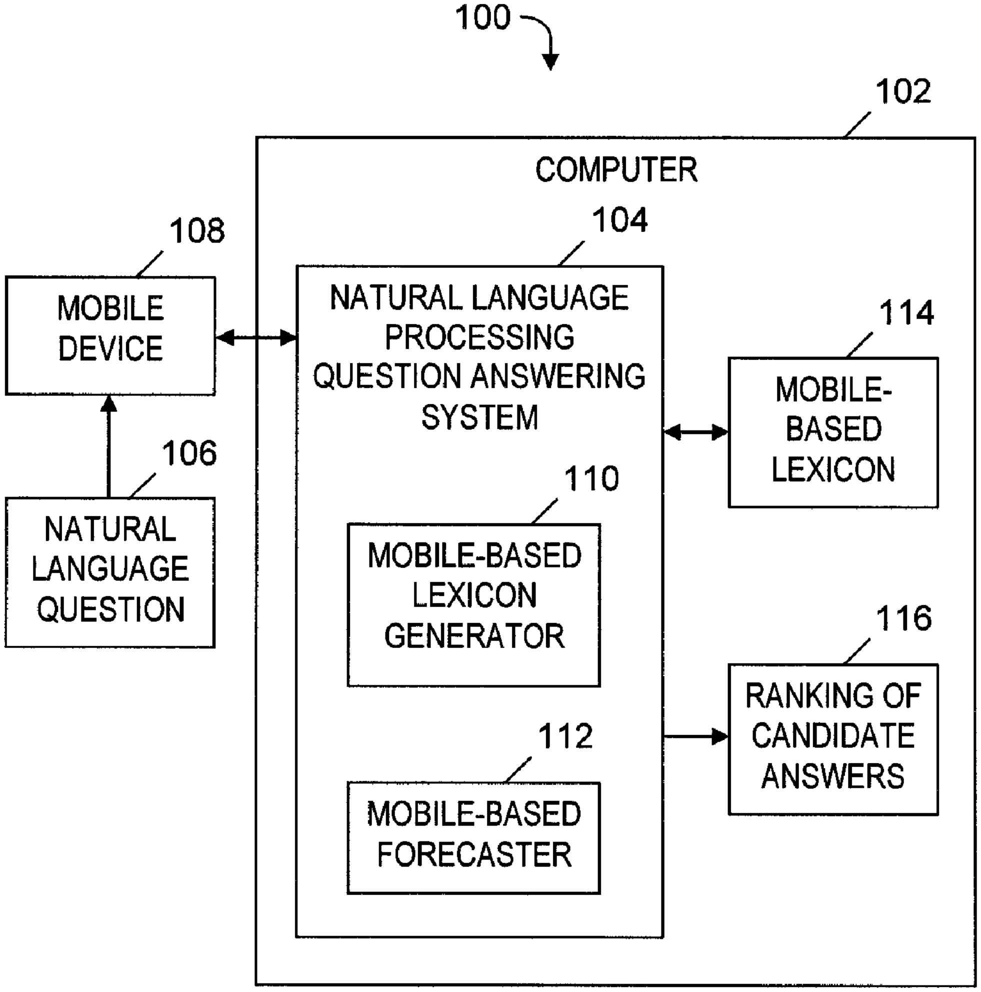
Publication Classification

(51) **Int. Cl.**
G06F 17/27 (2006.01)
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC *G06F 17/277* (2013.01); *G06F 17/30684* (2013.01)

(57) ABSTRACT

An approach is provided for ranking candidate answers to a natural language question. A natural language question is received from a user of a mobile device. Candidate answers to the received natural language question are generated. A lexicon is generated based on contextual information of the user determined by the mobile device. Contextual information of the user is forecasted based on usage of the mobile device. Based in part on the lexicon and the forecasted contextual information, the candidate answers to the natural language question are ranked.


100
102
COMPUTER
104
NATURAL LANGUAGE PROCESSING QUESTION ANSWERING SYSTEM
108
MOBILE DEVICE
106
NATURAL LANGUAGE QUESTION
110
MOBILE-BASED LEXICON GENERATOR
112
MOBILE-BASED FORECASTER
114
MOBILE-BASED LEXICON
116
RANKING OF CANDIDATE ANSWERS

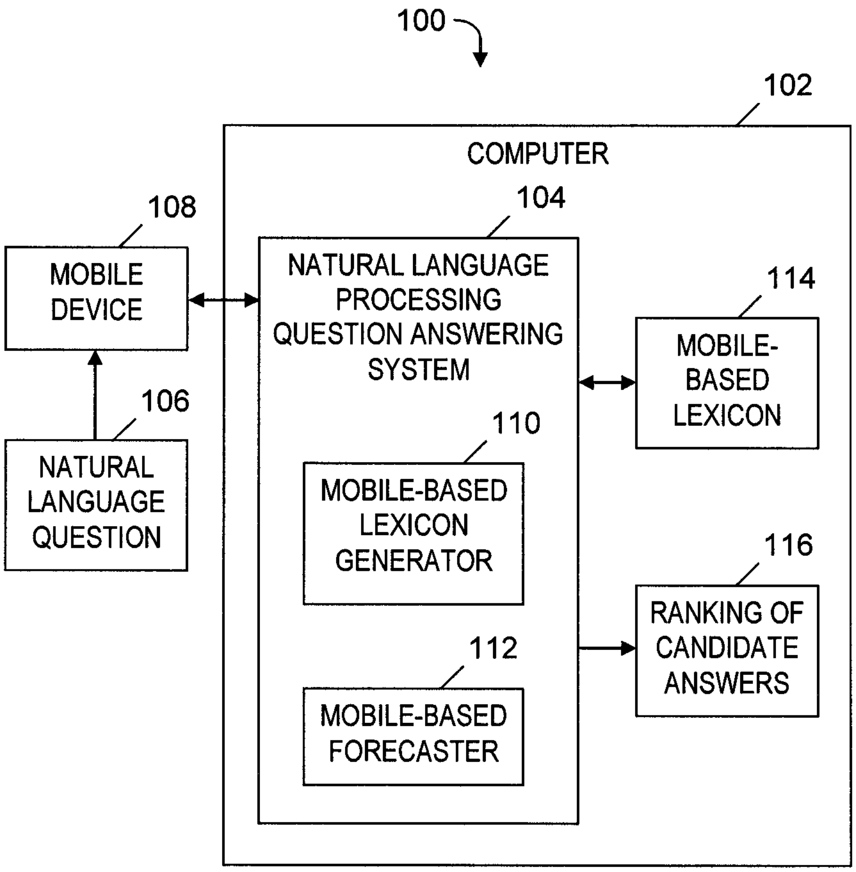

100
102
COMPUTER
104
NATURAL LANGUAGE PROCESSING QUESTION ANSWERING SYSTEM
108
MOBILE DEVICE
106
NATURAL LANGUAGE QUESTION
110
MOBILE-BASED LEXICON GENERATOR
112
MOBILE-BASED FORECASTER
114
MOBILE-BASED LEXICON
116
RANKING OF CANDIDATE ANSWERS

FIG. 1

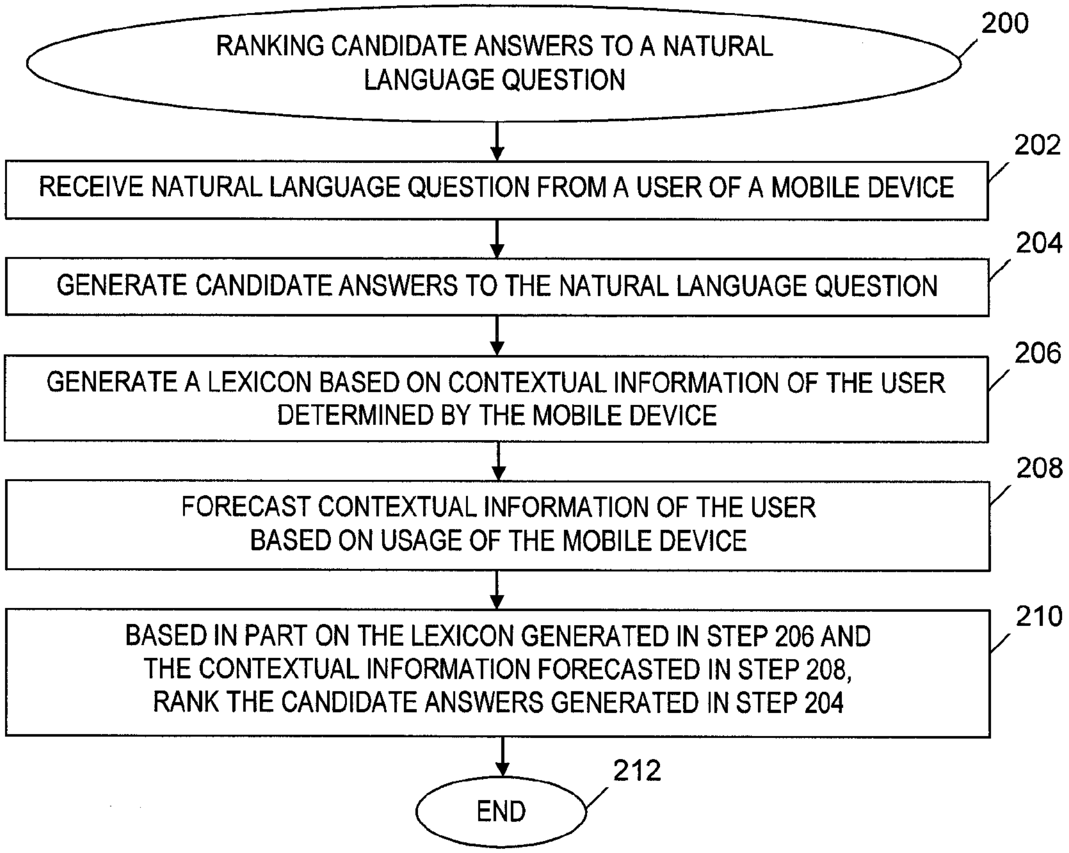

RANKING CANDIDATE ANSWERS TO A NATURAL LANGUAGE QUESTION
200
RECEIVE NATURAL LANGUAGE QUESTION FROM A USER OF A MOBILE DEVICE
202
GENERATE CANDIDATE ANSWERS TO THE NATURAL LANGUAGE QUESTION
204
GENERATE A LEXICON BASED ON CONTEXTUAL INFORMATION OF THE USER DETERMINED BY THE MOBILE DEVICE
206
FORECAST CONTEXTUAL INFORMATION OF THE USER BASED ON USAGE OF THE MOBILE DEVICE
208
BASED IN PART ON THE LEXICON GENERATED IN STEP 206 AND THE CONTEXTUAL INFORMATION FORECASTED IN STEP 208, RANK THE CANDIDATE ANSWERS GENERATED IN STEP 204
210
END
212

FIG. 2

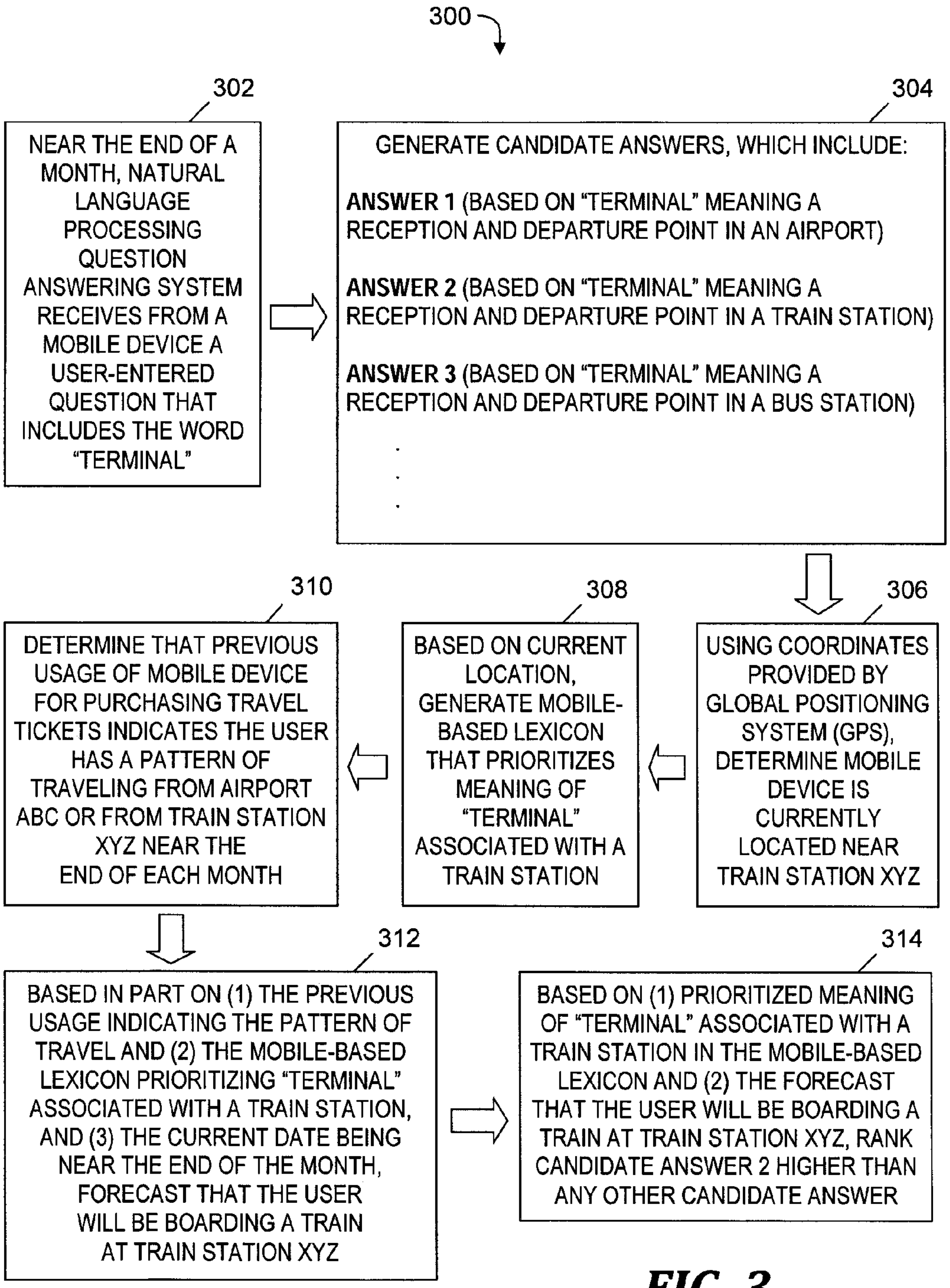
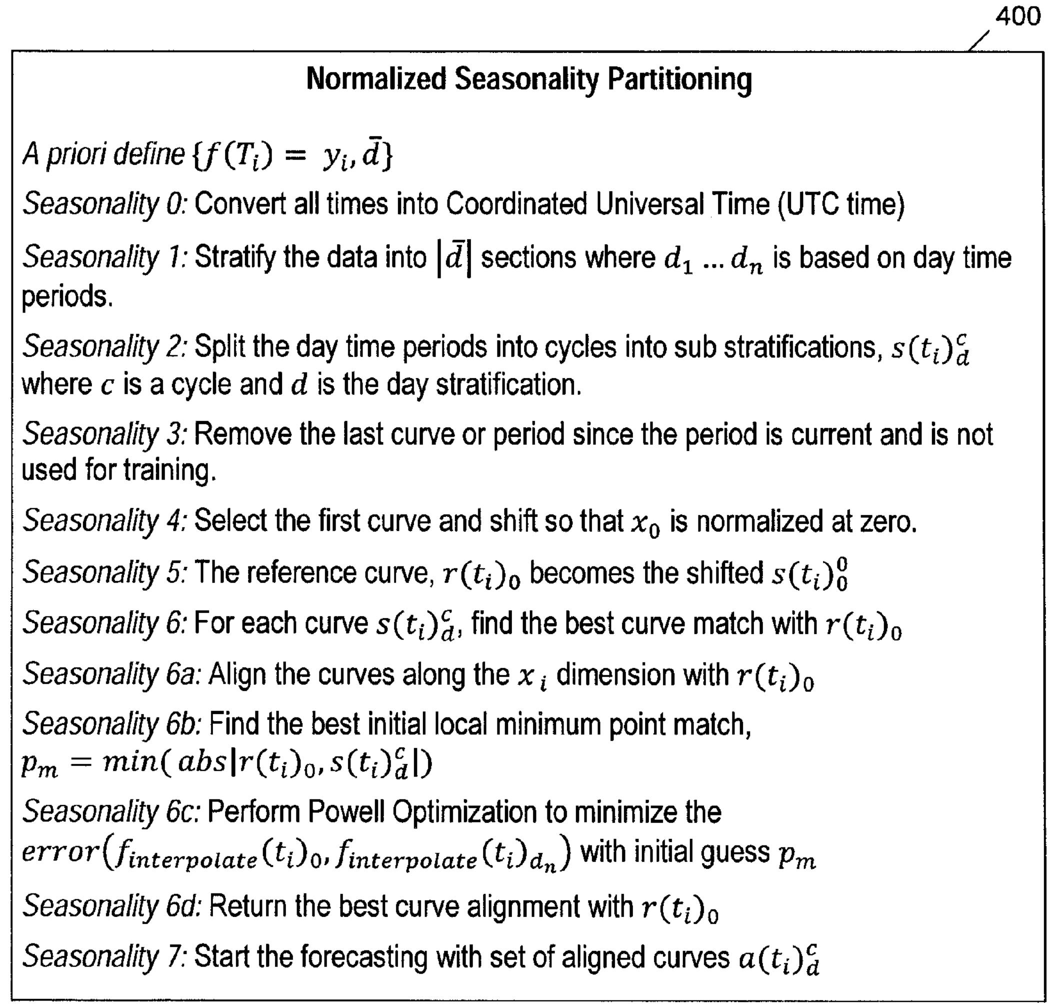

300
302
NEAR THE END OF A MONTH, NATURAL LANGUAGE PROCESSING QUESTION ANSWERING SYSTEM RECEIVES FROM A MOBILE DEVICE A USER-ENTERED QUESTION THAT INCLUDES THE WORD "TERMINAL"
304
GENERATE CANDIDATE ANSWERS, WHICH INCLUDE:
ANSWER 1 (BASED ON "TERMINAL" MEANING A RECEPTION AND DEPARTURE POINT IN AN AIRPORT)
ANSWER 2 (BASED ON "TERMINAL" MEANING A RECEPTION AND DEPARTURE POINT IN A TRAIN STATION)
ANSWER 3 (BASED ON "TERMINAL" MEANING A RECEPTION AND DEPARTURE POINT IN A BUS STATION)
.
.
.
306
USING COORDINATES PROVIDED BY GLOBAL POSITIONING SYSTEM (GPS), DETERMINE MOBILE DEVICE IS CURRENTLY LOCATED NEAR TRAIN STATION XYZ
308
BASED ON CURRENT LOCATION, GENERATE MOBILE-BASED LEXICON THAT PRIORITIZES MEANING OF "TERMINAL" ASSOCIATED WITH A TRAIN STATION
310
DETERMINE THAT PREVIOUS USAGE OF MOBILE DEVICE FOR PURCHASING TRAVEL TICKETS INDICATES THE USER HAS A PATTERN OF TRAVELING FROM AIRPORT ABC OR FROM TRAIN STATION XYZ NEAR THE END OF EACH MONTH
312
BASED IN PART ON (1) THE PREVIOUS USAGE INDICATING THE PATTERN OF TRAVEL AND (2) THE MOBILE-BASED LEXICON PRIORITIZING "TERMINAL" ASSOCIATED WITH A TRAIN STATION, AND (3) THE CURRENT DATE BEING NEAR THE END OF THE MONTH, FORECAST THAT THE USER WILL BE BOARDING A TRAIN AT TRAIN STATION XYZ
314
BASED ON (1) PRIORITIZED MEANING OF "TERMINAL" ASSOCIATED WITH A TRAIN STATION IN THE MOBILE-BASED LEXICON AND (2) THE FORECAST THAT THE USER WILL BE BOARDING A TRAIN AT TRAIN STATION XYZ, RANK CANDIDATE ANSWER 2 HIGHER THAN ANY OTHER CANDIDATE ANSWER

FIG. 3



Normalized Seasonality Partitioning

A priori define $\{f(T_i) = y_i, \bar{d}\}$

Seasonality 0: Convert all times into Coordinated Universal Time (UTC time)

Seasonality 1: Stratify the data into $|\bar{d}|$ sections where $d_1 \ldots d_n$ is based on day time periods.

Seasonality 2: Split the day time periods into cycles into sub stratifications, $s(t_i)_d^c$ where c is a cycle and d is the day stratification.

Seasonality 3: Remove the last curve or period since the period is current and is not used for training.

Seasonality 4: Select the first curve and shift so that x_0 is normalized at zero.

Seasonality 5: The reference curve, $r(t_i)_0$ becomes the shifted $s(t_i)_0^0$

Seasonality 6: For each curve $s(t_i)_d^c$, find the best curve match with $r(t_i)_0$

Seasonality 6a: Align the curves along the x_i dimension with $r(t_i)_0$

Seasonality 6b: Find the best initial local minimum point match, $p_m = min(\, abs|r(t_i)_0, s(t_i)_d^c|)$

Seasonality 6c: Perform Powell Optimization to minimize the $error(f_{interpolate}(t_i)_0, f_{interpolate}(t_i)_{d_n})$ with initial guess p_m

Seasonality 6d: Return the best curve alignment with $r(t_i)_0$

Seasonality 7: Start the forecasting with set of aligned curves $a(t_i)_d^c$

FIG. 4A

420

Residual Forecaster

A priori define $\{a(t_i)_d^c\}$

Residual 0: Calculate polynomial splines for each adjusted curve $a(t_i)_d^c$

Residual 0a: For each point,$t_i \in a(t_i)_d^c$, apply Loess Interpolation and calculate the residual curve $r(t_i)_d^c = a(t_i)_d^c - a_{interpolation}(t_i)_d^c$

Residual 0b: For each point, $t_i^r \in r(t_i)_d^c$, shift around the mean with $\frac{1+\mu}{r_{interpolation}(t_i)_d^c}$ to produce $r_2(t_i)_d^c$

Residual 1: For each $r_2(t_i)_d^c$, determine a polynomial spline function using Loess Interpolation to yield $r_3(t_i)_d^c$

Residual 2: For each $r_3(t_i)_d^c$ on all $t_i's$, create a new data set, $r_f(t_i)_d^c = \mu_t + \alpha\sigma_t$, where α determines the number of standard deviations

Residual 3: Return the final residual curve, $r_f(t_i)_d^c$

FIG. 4B

440

Partial Forecaster

A priori define $\{a(t_i)_d^c\}$

Partial 0: For each $a(t_i)_d^c$, determine a polynomial spline function using Loess Interpolation to yield $pf_1(t_i)_d^c$

Partial 1: For each $pf_1(t_i)_d^c$ on all $t_i's$, create a new data set, $pf_f(t_i)_d^c = \mu_t + \alpha\sigma_t$, where α determines the number of standard deviations and σ is the standard deviation value.

Partial 2: Repeat Steps *Partial 0* and *Partial 1* for only weekend and select the best curve match $wdc(t_i)$ by minimizing the squared error.

Partial 3: Repeat Steps *Partial 0* and *Partial 1* for only week periods and select the best curve match $wec(t_i)$ by minimizing the squared error.

Partial 4: Create a forecaster with $wdc(t_i)$ or $wec(t_i)$, $O(t_i) = \frac{t_i - a(t_i)_{start}}{a(t_i)_{end} - a(t_i)_{start}}$, and error function $E\big(O(t_i)\big) = e$ that determines the penalty for not using a part of curve for forecasting.

Partial 5: Interpolate a forecast value for either a weekend or weekday

Partial 6: Return the $\{f_i, e_i\}$ values for the forecast.

FIG. 4C

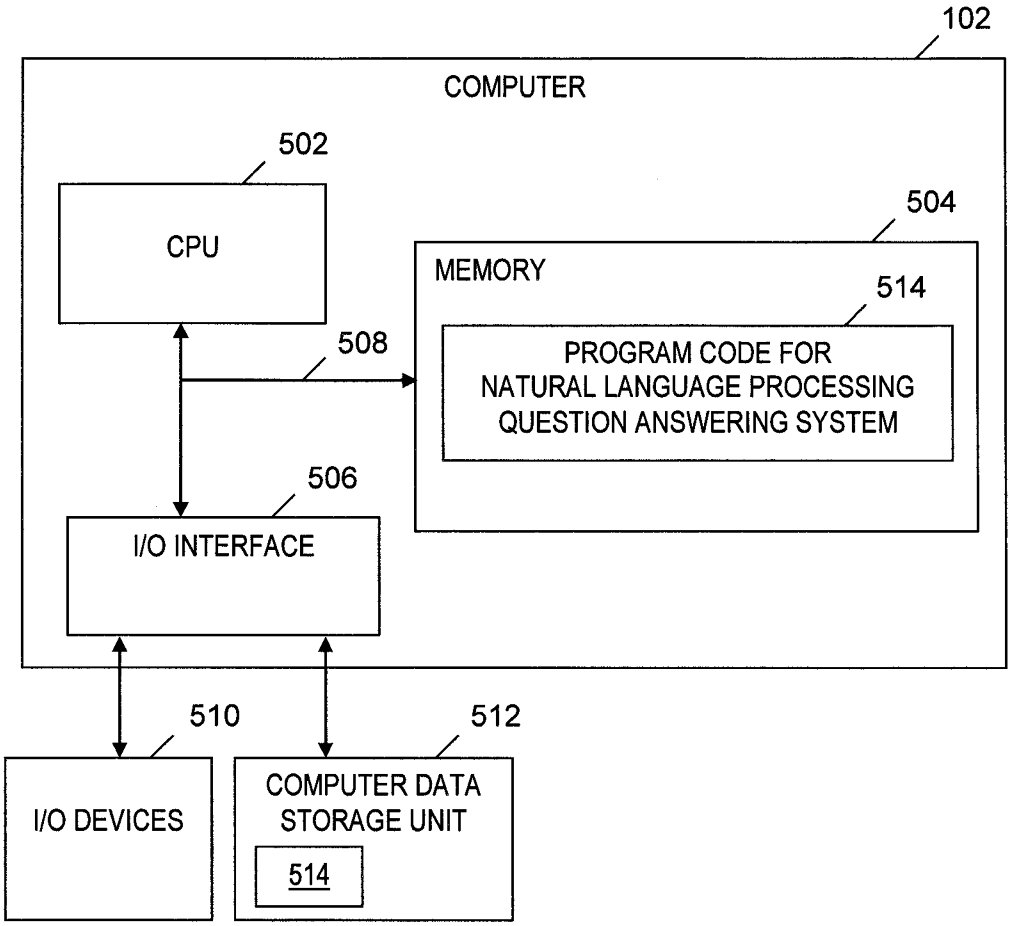

102
COMPUTER
502
CPU
504
MEMORY
514
PROGRAM CODE FOR
NATURAL LANGUAGE PROCESSING
QUESTION ANSWERING SYSTEM
508
506
I/O INTERFACE
510
I/O DEVICES
512
COMPUTER DATA
STORAGE UNIT
514

FIG. 5

MOBILE BASED LEXICON AND FORECASTING

TECHNICAL FIELD

[0001] The present invention relates to information retrieval and natural language processing, and more particularly to managing candidate answers in a natural language query and answer system.

BACKGROUND

[0002] Natural language processing (NLP) systems are attempting to minimize uncertainty from reasoning through the use of known lexicons, ontologies, and corpora ingestion, but overall system accuracy using these known techniques has limitations. A known probabilistic distributed NLP system delivers answers to questions by parsing a natural language query in input text provided by a user, searching for multiple candidate answers to the query in its data set, and assigning confidence scores to the candidate answers. The probabilistic distributed NLP system can, in a limited way, take into account the context of the user who provided the input text if the context is appended to the query. Appending such context to a query, however, does not provide an ideal user experience because the combination of the query and the appended portion is not in the form of a question that would be posed naturally by a user. A known web search engine being utilized via a user's mobile device can take into account context consisting of the user's geographic location by processing search terms based in part on the user's location, which is provided by a global positioning system (GPS) incorporated into the device.

BRIEF SUMMARY

[0003] In first embodiments, the present invention provides a method of ranking candidate answers to a natural language question. The method includes a computer receiving the natural language question from a user of a mobile device. The method further includes the computer generating candidate answers to the received natural language question. The method further includes the computer generating a lexicon based on contextual information of the user determined by the mobile device. The method further includes the computer forecasting contextual information of the user based on usage of the mobile device. The method further includes, based in part on the lexicon and the forecasted contextual information, the computer ranking the candidate answers to the natural language question.

[0004] Embodiments of the present invention enhances the accuracy of NLP question answering systems by allowing the knowledge base of the question answering system to include contextual information about a user, where the contextual information is captured by a mobile device.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] FIG. **1** is a block diagram of a system for ranking candidate answers to a natural language question, in accordance with embodiments of the present invention.

[0006] FIG. **2** is a flowchart of a process of ranking candidate answers to a natural language question, where the process is implemented in the system of FIG. **1**, in accordance with embodiments of the present invention.

[0007] FIG. **3** depicts an example of ranking a candidate answer by steps included in the process of FIG. **2**, in accordance with embodiments of the present invention.

[0008] FIGS. **4A-4C** depict forecasting algorithms that are used in the process of FIG. **2**, in accordance with embodiments of the present invention.

[0009] FIG. **5** is a block diagram of a computer that is included in the system of FIG. **1** and that implements the process of FIG. **2**, in accordance with embodiments of the present invention.

DETAILED DESCRIPTION

Overview

[0010] Embodiments of the present invention provide a ranking of candidate answers generated by a NLP question answering system in response to receiving a natural language question from a user utilizing a mobile device. The ranking of the candidate answers is based on a mobile-based lexicon and forecasted contextual information about the user. Entries in the mobile-based lexicon are based on contextual information of the user, as determined by the mobile device. The forecasted contextual information is a forecast of future behavior or interest(s) of the user based on current and/or historical usage of the mobile device.

[0011] A known NLP question answering system provides an answer to a natural language question by using traditional lexicons, ontologies and corpora ingestion, but the overall system accuracy in terms of precision and recall can be improved, which presents a unique challenge to users of NLP systems who desire minimized uncertainty about the answer provided by the NLP system. Furthermore, a user's mobile device interacting with the NLP question answering system presents a large number of candidate answers, which is a unique challenge to the user's experience with the mobile device based on the device having limited display space and the user having limited time to sift through the large number of candidate answers. At least one of these unique challenges is overcome by one or more embodiments of the present invention.

System for Ranking Candidate Answers to a Natural Language Question

[0012] FIG. **1** is a block diagram of a system **100** for ranking candidate answers to a natural language question, in accordance with embodiments of the present invention. System **100** includes a computer **102**, which runs a software-based natural language processing (NLP) question answering (QA) system **104**. NLP QA system **104** receives a natural language question **106** or other unstructured textual input (not shown) which is entered into a mobile device **108** by a user of the mobile device **108**. Natural language question **106** is text in a natural language that includes a question. As used herein, mobile device **108** is a handheld computing device such as a smartphone or a wearable computer, such as smartglasses. In response to receiving a user entry of natural language question **106**, mobile device **108** sends natural language question **106** to computer **102** for processing by NLP QA system **104** via a computer network (not shown). Mobile device **108** can use a combination of on-board and off-board mobile processing.

[0013] NLP QA system **104** includes the following software modules: a mobile-based lexicon generator **110** and a mobile-based forecaster **112**. Mobile-based lexicon generator **110** generates a lexical analyzer (not shown) and a mobile-based lexicon **114**. The lexical analyzer (also known as (a.k.a.) lexer or tokenizer) generated by mobile-based lexicon generator **110** is a computer program that transforms a stream of characters into a stream of "atomic chunks of meaning," which are tokens (i.e., meaningful character strings). Mobile-based lexicon generator **110** generates the lexical analyzer based on contextual information about the user of mobile device **108**. The generated lexical analyzer transforms a stream of characters based in part on meanings of character strings, where the meanings are based on the contextual information about the user. The mobile-based lexicon **114** is a repository of words and other terms along with their definitions, which are ranked or otherwise prioritized based on contextual information about the user. The prioritization of the words and other terms in mobile-based lexicon **114** can be based on historical usage of mobile device **108**, geographic locations of mobile device **108** provided by a GPS system coupled to the mobile device, or other data about movement of the user or surroundings of the user, as detected by sensor (s) coupled to mobile device **108**. The historical usage of mobile device **108** can include website(s) visited by the user of mobile device **108** and search terms entered by the user into a search engine accessed by mobile device **108**.

[0014] Mobile-based forecaster **112** employs forecasting algorithm(s) to generate one or more predictions about what the contextual information of the user of mobile device **108** will be in a specified time in the future. The contextual information includes interests of the user. The prediction(s) generated by mobile-based forecaster **112** can be based on historical usage of mobile device **108**, including website(s) visited by the user of mobile device **108**, search term(s) entered by the user into a search engine, and geographic position(s) at which mobile device **108** was located as determined by a GPS system coupled to the mobile device.

[0015] NLP QA system **104** generates candidate answers to natural language question **106**. Based on the contextual information prediction(s) generated by mobile-based forecaster **112** and the prioritization of definitions in mobile-based lexicon **114**, NLP QA system **104** generates a ranking **116** of the candidate answers.

[0016] Although not shown in FIG. **1**, other embodiments include multiple mobile devices **108** that work together to determine and provide the contextual information about the user, where the multiple mobile devices **108** can use a combination of on-board and off-board mobile processing.

[0017] As one example, NLP QA system **104** is an enhancement of the Watson® question answering computing system offered by International Business Machines Corporation located in Armonk, N.Y., whereby the enhancement receives the aforementioned contextual information about a user of mobile device **108**, incorporates the contextual information into a base of knowledge, and uses the contextual information in the base of knowledge as at least a partial basis for ranking candidate answers to a natural language question. The Watson® question answering system delivers answers to questions by receiving and parsing input text, looking for multiple candidate answers in its data set, and assigning a confidence score to each of the candidate answers. In its current form, without the enhancement described herein, the Watson® question answering system does not take into account the context of the user asking the question. For example, a user entering "What is the best way to treat my condition of X", where X is a medical condition, will receive the same answer from the known Watson® question answering system if the user is located in Country A, Country B, Country C, or Country D when entering the question. The actual answer or best answer, however, may differ based in part on whether the user is currently located in Country A, Country B, Country C, or Country D (i.e., based in part on the context of the user, where the context is the geographic location of the user). In this example, the enhancement provided by the NLP QA system **104** generates a ranking of candidate answers that is based in part on whether the user is currently located in Country A, Country B, Country C, or Country D, where the country in which the user is located is determined from geographic coordinates provided by a GPS (not shown) coupled to mobile device **108**.

[0018] The functionality of the components shown in FIG. **1** is described in more detail in the discussion of FIG. **2** and FIG. **5** presented below.

Process for Ranking Candidate Answers to a Natural Language Question

[0019] FIG. **2** is a flowchart of a process of ranking candidate answers to a natural language question, where the process is implemented in the system of FIG. **1**, in accordance with embodiments of the present invention. The process of FIG. **2** begins at step **200**. In step **202**, NLP QA system **104** (see FIG. **1**) receives natural language question **106** (see FIG. **1**) as unstructured text from a user of mobile device **108** (see FIG. **1**).

[0020] In step **204**, NLP QA system **104** (see FIG. **1**) analyzes natural language question **106** (see FIG. **1**), generates candidate answers in response to natural language question **106** (see FIG. **1**), gathers evidence in support of the candidate answers, and generates initial scores for the candidate answers based on the gathered evidence. To analyze natural language question **106** (see FIG. **1**), NLP QA system **104** (see FIG. **1**) generates a lexical analyzer, which transforms streams of characters in natural language question **106** into meaningful character strings. In step **204** and subsequent steps of FIG. **2**, NLP QA system **104** (see FIG. **1**) dynamically adjusts the criteria for determining which character string is meaningful or not based in part on contextual information about the user (see the discussion below relative to step **206**) and based in part on forecasted contextual information about the user (see the discussion below relative to step **208**).

[0021] In one embodiment, the candidate answers generated in step **204** are potentially correct answers that are replies or responses to natural language question **106** (see FIG. **1**). In another embodiment, the candidate answers generated in step **204** are questions that are presented to the user of mobile device **108** (see FIG. **1**) to obtain additional contextual information about the user or other additional information (e.g., to assist in a medical diagnosis).

[0022] Prior to step **206**, NLP QA system **104** (see FIG. **1**) identifies contextual information about the user of mobile device **108** (see FIG. **1**), where the contextual information has been determined by mobile device **108** (see FIG. **1**). The aforementioned contextual information about the user can include interest(s) of the user and is determined by any combination of: (1) a list of applications that the user has run on mobile device **108** (see FIG. **1**), (2) the websites visited by the user via a browser provided by mobile device **108** (see FIG. **1**), (3) textual information, articles or a summarization of articles displayed on and/or downloaded to mobile device **108**

(see FIG. **1**) (i.e., textual nugget capture by the mobile device **108** (see FIG. **1**)), (4) terms entered by the user into a search engine via mobile device **108** (see FIG. **1**), (5) multimedia content and/or other types of content captured via mobile device **108** (see FIG. **1**), (6) data specifying characteristic(s) of the environment in proximity to the user as detected and/or measured by sensor(s) coupled to mobile device **108** (see FIG. **1**), (7) data specifying function(s) of the body of the user or movement(s) of the user as detected and/or measured by sensor(s) coupled to mobile device **108** (see FIG. **1**), (8) a geographic location of the user provided by a GPS (not shown) coupled to mobile device **108** (see FIG. **1**), and (9) other information detected by mobile device **108** or other information the user enters into mobile device **108** (see FIG. **1**). In another embodiment, the contextual information consists of the information listed in (1)-(9) presented above.

[0023] The contextual information listed in (1)-(9) presented above may be limited by respective time periods. For example, the applications that the user has run on mobile device **108** (see FIG. **1**) may be the applications run within the last D**1** days, while the summarization of articles read by the user on mobile device **108** (see FIG. **1**) may be the articles read within the last D**2** days, where D**1** can be the same as or different from D**2**.

[0024] In step **206**, NLP QA system **104** (see FIG. **1**) generates mobile-based lexicon **114** (see FIG. **1**) as a repository or words and other terms, along with one or more definitions for each word or other term. The generation of mobile-based lexicon **114** (see FIG. **1**) includes NLP QA system **104** (see FIG. **1**) ranking or otherwise prioritizing definitions corresponding to word(s) or term(s) in lexicon **114** (see FIG. **1**) based on the aforementioned contextual information determined by mobile device **108** (see FIG. **1**). In subsequent steps of the process of FIG. **2** or in a subsequent performance of the entire process of FIG. **2**, NLP QA system **104** (see FIG. **1**) can dynamically update the terms in mobile-based lexicon **114** (see FIG. **1**) and the ranking of the terms. In one embodiment, mobile-based lexicon **114** (see FIG. **1**) is ingested into a natural language processing system included in NLP QA system **104** (see FIG. **1**).

[0025] Subsequent to step **206**, NLP QA system **104** (see FIG. **1**) can extend mobile-based lexicon **114** (see FIG. **1**) by mobile device acronym symbol definition, mobile computing terminology, gesture-based and user emotion summarization, and augmented reality word definitions and word senses.

[0026] In step **208**, NLP QA system **104** (see FIG. **1**) forecasts what the contextual information about the user of mobile device **108** (see FIG. **1**) will be at a specified time in the future. The forecasted contextual information is based on historical usage of mobile device **108** (see FIG. **1**), including any combination of (1) applications that the user has run on mobile device **108** (see FIG. **1**), (2) websites visited by the user via a browser provided by mobile device **108** (see FIG. **1**), (3) textual information, articles or a summarization of articles displayed on or downloaded to mobile device **108** (see FIG. **1**), (4) terms entered by the user into a search engine via mobile device **108** (see FIG. **1**), and (5) multimedia content and/or other types of content captured via mobile device **108** (see FIG. **1**). In one embodiment, the forecasted contextual information and is based in part on historical usage of mobile device **108** (see FIG. **1**) and based in part on a combination of: geographic position(s) at which mobile device **108** was located as determined by a GPS coupled to the mobile device **108** (see FIG. **1**), data specifying characteristic(s) of the environment in proximity to the user as detected and/or measured by sensor(s) coupled to mobile device **108** (see FIG. **1**), and data specifying function(s) of the body of the user or movement(s) of the user as detected and/or measured by sensor(s) coupled to mobile device **108** (see FIG. **1**).

[0027] The forecasting performed in step **208** may use any of the forecasting algorithms shown in FIGS. **4**A-**4**C, or a forecasting algorithm (i.e., forecaster) that includes mobile-based time series manipulation and pattern recognition, autoregressive integrated moving average (ARIMA) based modeling based on the aforementioned contextual information of the user as determined by mobile device **108** (see FIG. **1**), lag-based modeling based on the contextual information of the user as determined by mobile device **108** (see FIG. **1**), residual calculations based on the contextual information of the user as determined by mobile device **108** (see FIG. **1**), or probability density based functions such as logistic regression based on mobile-based event detection. In one embodiment, the forecasting in step **208** includes querying the user of mobile device **108** (see FIG. **1**), whose response provides additional information to the aforementioned forecasting algorithm so that inferences generated by the forecasting algorithm about the contextual information are increased, as compared to inferences generated by the forecasting algorithm operating without the additional information.

[0028] In one embodiment, a mobile context dependent analytical pipeline (not shown) annotates the contextual information forecasted in step **208**. The annotated information creates machine learning features based on the text in natural language question **106** (see FIG. **1**), the candidate answers generated or modified in the process of FIG. **2**, and the context in which each candidate answer appeared, such as passages or evidences.

[0029] In one embodiment, a mobile forecaster pipeline (not shown) annotates the contextual information forecasted in step **208**, thereby creating machine learning features based on the text of natural language question **106** and the candidate answers generated or modified in the process of FIG. **2**, without regard to the context in which the candidate answer appeared.

[0030] In step **210**, based in part on the prioritization of definitions of word(s) and/or other term(s) in mobile-based lexicon **114** (see FIG. **1**) generated in step **206** and the contextual information about the user forecasted in step **208**, NLP QA system **104** (see FIG. **1**) ranks the candidate answers generated in step **204**. The candidate answers ranked from the lowest ranked candidate answer to the highest ranked candidate answer provides candidate answers ordered by how likely each answer is a correct answer to natural language question **106** (see FIG. **1**) (i.e., ordered from the candidate answer that is least likely to be the correct answer to natural language question **106** (see FIG. **1**) to the candidate answer that is most likely to be the correct answer to natural language question **106** (see FIG. **1**)). After step **210**, NLP QA system **104** (see FIG. **1**) initiates a display on mobile device **108** (see FIG. **1**) of the highest ranked candidate answer as the final answer to natural language question **106** (see FIG. **1**). The process of FIG. **2** ends at step **212**.

[0031] In one embodiment, NLP QA system **104** (see FIG. **1**) uses the contextual information forecasted in step **208** to perform word sense disambiguation and adjust rankings of terms in mobile-based lexicon **112** (see FIG. **1**). Prior to step **210**, NLP QA system **104** (see FIG. **1**) uses the results of the disambiguation and adjusted rankings to add to, delete from, or modify the candidate answers generated in step **204**.

[0032] In one embodiment, a machine learning ensemble performs the steps of FIG. **2** and uses the contextual information forecasted in step **208** to alter or adjust machine learning ensemble weights. The machine learning ensemble can include, for example, (1) logistic regression phases and routes, (2) dimensions of evidence refinement, (3) probability density functions, classification techniques, and regression techniques, (4) natural language processing systems, and (5) any data retrieval system.

Example

[0033] FIG. **3** depicts an example **300** of ranking a candidate answer by steps included in the process of FIG. **2**, in accordance with embodiments of the present invention. In step **302**, near the end of a month, NLP QA system **104** (see FIG. **1**) receives from mobile device **108** (see FIG. **1**) a question entered by a user of mobile device **108** (see FIG. **1**), where the question includes the word "terminal." Step **302** is an example of an action included in step **202** (see FIG. **2**).

[0034] In step **304**, NLP QA system **104** (see FIG. **1**) generates multiple candidate answers to the question received in step **302**, including first, second and third candidate answers (i.e., Answer 1, Answer 2, and Answer 3, respectively, as shown in FIG. **3**). The first candidate answer is based on "terminal" in the question meaning a reception and departure point in an airport. The second candidate answer is based on "terminal" in the question meaning a reception and departure point in a train station. The third candidate answer is based on "terminal" in the question meaning a reception and departure point in a bus station. Although not shown in FIG. **3**, the candidate answers may include another answers that are based on "terminal" meaning (1) a mechanical device by means of which an electric connection to an apparatus is established, (2) a point at which current enters or leaves an electrical device, and (3) an electronic or electromechanical device for entering data into a computer or communications system and displaying data received. Step **304** is an example of an action included in step **204** (see FIG. **2**).

[0035] In step **306**, using coordinates provided by a GPS coupled to mobile device **108** (see FIG. **1**), NLP QA system **104** (see FIG. **1**) determines the current geographic location of mobile device **108** (see FIG. **1**) and determines that mobile device **108** (see FIG. **1**) is currently located near train station XYZ. Step **306** is an example of an action taken prior to step **206** (see FIG. **2**).

[0036] In step **308**, based on the current location of mobile device **108** (see FIG. **1**) determined in step **306**, NLP QA system **104** (see FIG. **1**) generates mobile-based lexicon **114** (see FIG. **1**) so that the definitions of the word "terminal" are ranked or otherwise prioritized with the definition indicating "terminal" means a reception and departure point in a train station receiving the highest rank or priority. Step **308** is an example of an action included in step **206** (see FIG. **2**).

[0037] In step **310**, NLP QA system **104** (see FIG. **1**) determines previous usage of mobile device **108** (see FIG. **1**) by which travel tickets were purchased by the user of mobile device **108** (see FIG. **1**), which indicate the user has established a pattern of traveling from either airport ABC or train station XYZ near the end of each month for the last n months, where n is a predetermined number of months.

[0038] In a step that not shown in FIG. **3**, but is prior to step **312**, NLP QA system **104** (see FIG. **1**) determines the current date.

[0039] In step **312**, based in part on (1) the previous usage of mobile device **108** (see FIG. **1**) indicating the user's pattern of traveling from either airport ABC or train station XYZ near the end of each month, (2) the mobile-based lexicon **114** (see FIG. **1**), which prioritizes the definition of "terminal" associated with a train station, and (3) the current date being near the end of the current month, NLP QA system **104** (see FIG. **1**) forecasts that the user will be boarding a train at train station XYZ. Step **312** is an example of an action included in step **208** (see FIG. **2**).

[0040] In step **314**, based on (1) the definition of "terminal" associated with a train station being prioritized in step **308** over all other definitions of "terminal" in mobile-based lexicon **114** (see FIG. **1**), and (2) the forecast made in step **312** that the user will be boarding a train at train station XYZ, NLP QA system **104** (see FIG. **1**) ranks Answer 2 higher than any other candidate answer generated in step **304** because Answer 2 is based on "terminal" meaning a reception and departure point in a train station. Step **314** is an example of an action included in step **210** (see FIG. **2**).

[0041] FIGS. **4A-4C** depict forecasting algorithms that are used in the process of FIG. **2**, in accordance with embodiments of the present invention. In one embodiment, step **208** in FIG. **2** includes mobile-based forecaster **112** (see FIG. **1**) employing a normalized seasonality partitioning algorithm **400** in FIG. **4**A. Normalized seasonality partitioning algorithm **400** includes partitioning a time periodfinto seasonality components or multiple curves using both days and a Powell Optimization process. Normalized seasonality partitioning algorithm **400** includes steps of a process of time series stratification and normalization into $a(t_i)_d^c$ aligned curves that are used forecast the contextual information about the user of mobile device **108** (see FIG. **1**).

[0042] In another embodiment, step **208** in FIG. **2** includes mobile-based forecaster **112** (see FIG. **1**) employing a residual forecaster algorithm **420** in FIG. **4**B. Residual forecaster algorithm **420** includes matching amplitudes of historical demand events to future events over a twenty-four hour period. The residual curve generated by residual forecaster algorithm **420** is the maximum curve determined by the differences between actual hit value and the interpolated value.

[0043] In yet another embodiment, step **208** in FIG. **2** includes mobile-based forecaster **112** (see FIG. **1**) employing a Historical Forecaster ensemble that uses historical information and the current time of day to develop a regression line. The Historical Forecaster ensemble is comprised of five forecasters: Partial Forecaster, Adjusted Average Forecaster, Past Forecaster, Vector Forecaster and Quadratic Expansion Forecaster.

[0044] The Partial Forecaster included in the historical forecaster employs a partial forecaster algorithm **440** in FIG. **4**C. Partial forecaster algorithm **440** includes forecasting several hours into the future only if the current time is at night with an average curve plus one standard deviation. Half of the data is regressed from the past while the other data is within the future. If the current time to predict is within a weekend, only the weekend partial forecaster is applied; otherwise, the entire partial forecaster is implemented. The error of the forecaster is D(b)=v or a predefined constant equal across all forecasts.

[0045] The Adjusted Average Forecaster is identical to the partial forecaster with the exception of its error function and it being applied only during the morning times. The error function of the Adjusted Average Forecaster is equation (1) presented below, which accounts for whether the current is before or after the curve.

$$E(O(t_i)) = \begin{cases} O(t_i) > 1{:}1 \\ O(t_i) < 1{:}0 \\ O(t_i)^2 \end{cases} \tag{1}$$

[0046] The Past Forecaster retrieves the previous 24-hour forecast curve. The error function for the Past Forecaster is defined as $E(O(t_i))=0$ since the forecast retrieves actual past forecasts that have been corrected.

[0047] The Vector Forecaster has the error function $E(b)=((t_{current}-t_{start})/(t_{end}-t_{current}))$.

[0048] The Quadratic Expansion Forecaster is uses quadratic curve regression that regresses two hours into the past and projects the curve two hours into the future. The error function for the Quadratic Expansion Forecaster is $E(b)=((t_{current}-t_{start})/(t_{end}-t_{current}))$.

Computer System

[0049] FIG. **5** is a block diagram of a computer that is included in the system of FIG. **1** and that implements the process of FIG. **2**, in accordance with embodiments of the present invention.

[0050] Computer **102** is a computer system that generally includes a central processing unit (CPU) **502**, a memory **504**, an input/output (I/O) interface **506**, and a bus **508**. Further, computer **102** is coupled to I/O devices **510** and a computer data storage unit **512**. CPU **502** performs computation and control functions of computer **102**, including carrying out instructions included in program code **514** to perform a method of ranking candidate answers to a natural language question, where the instructions are carried out by CPU **502** via memory **504**. CPU **502** may include a single processing unit, or be distributed across one or more processing units in one or more locations (e.g., on a client and server). Program code **514** includes program code for mobile-based lexicon generator **110** (see FIG. **1**) and mobile-based forecaster **112** (see FIG. **1**).

[0051] Memory **504** includes a known computer readable storage medium, which is described below. In one embodiment, cache memory elements of memory **504** provide temporary storage of at least some program code (e.g., program code **514**) in order to reduce the number of times code must be retrieved from bulk storage while instructions of the program code are carried out. Moreover, similar to CPU **502**, memory **504** may reside at a single physical location, including one or more types of data storage, or be distributed across a plurality of physical systems in various forms. Further, memory **504** can include data distributed across, for example, a local area network (LAN) or a wide area network (WAN).

[0052] I/O interface **506** includes any system for exchanging information to or from an external source. I/O devices **510** include any known type of external device, including a display device, keyboard, etc. Bus **508** provides a communication link between each of the components in computer **102**, and may include any type of transmission link, including electrical, optical, wireless, etc.

[0053] I/O interface **506** also allows computer **102** to store information (e.g., data or program instructions such as program code **514**) on and retrieve the information from computer data storage unit **512** or another computer data storage unit (not shown). Computer data storage unit **512** includes a known computer-readable storage medium, which is described below. In one embodiment, computer data storage unit **512** is a non-volatile data storage device, such as a magnetic disk drive (i.e., hard disk drive) or an optical disc drive (e.g., a CD-ROM drive which receives a CD-ROM disk).

[0054] Memory **504** and/or storage unit **512** may store computer program code **514** that includes instructions that are carried out by CPU **502** via memory **504** to rank candidate answers to a natural language question. Although FIG. **5** depicts memory **504** as including program code **514**, the present invention contemplates embodiments in which memory **504** does not include all of code **514** simultaneously, but instead at one time includes only a portion of code **514**.

[0055] Further, memory **504** includes an operating system (not shown) and may include other systems not shown in FIG. **5**. The operating system may be a Linux®-based operating system that runs on CPU **502** and provides control of various components within and/or connected to computer **102**. Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

[0056] Storage unit **512** and/or one or more other computer data storage units (not shown) that are coupled to computer **102** may store the content of mobile-based lexicon **114** (see FIG. **1**).

[0057] As will be appreciated by one skilled in the art, in a first embodiment, the present invention may be a system; in a second embodiment, the present invention may be a method; and in a third embodiment, the present invention may be a computer program product.

[0058] Any of the components of an embodiment of the present invention can be deployed, managed, serviced, etc. by a service provider that offers to deploy or integrate computing infrastructure with respect to ranking candidate answers to a natural language question. Thus, an embodiment of the present invention discloses a process for supporting computer infrastructure, where the process includes providing at least one support service for at least one of integrating, hosting, maintaining and deploying computer-readable code (e.g., program code **514**) in a computer system (e.g., computer **102**) including one or more processors (e.g., CPU **502**), wherein the processor(s) carry out instructions contained in the code causing the computer system to rank candidate answers to a natural language question. Another embodiment discloses a process for supporting computer infrastructure, where the process includes integrating computer-readable program code into a computer system including a processor. The step of integrating includes storing the program code in a computer-readable storage device of the computer system through use of the processor. The program code, upon being executed by the processor, implements a method of ranking candidate answers to a natural language question.

[0059] While it is understood that program code **514** for ranking candidate answers to a natural language question may be deployed by manually loading directly in client, server and proxy computers (not shown) via loading a computer-readable storage medium (e.g., computer data storage unit **512**), program code **514** may also be automatically or semi-automatically deployed into computer **102** by sending program code **514** to a central server or a group of central servers. Program code **514** is then downloaded into client computers (e.g., computer **102**) that will execute program code **514**. Alternatively, program code **514** is sent directly to the client computer via e-mail. Program code **514** is then either detached to a directory on the client computer or loaded into a directory on the client computer by a button on the e-mail that executes a program that detaches program code **514** into a directory. Another alternative is to send program code **514** directly to a directory on the client computer hard drive. In a case in which there are proxy servers, the process selects the proxy server code, determines on which computers to place the proxy servers' code, transmits the proxy server code, and then installs the proxy server code on the proxy computer. Program code **514** is transmitted to the proxy server and then it is stored on the proxy server.

[0060] Another embodiment of the invention provides a method that performs the process steps on a subscription, advertising and/or fee basis. That is, a service provider, such as a Solution Integrator, can offer to create, maintain, support, etc. a process of ranking candidate answers to a natural language question. In this case, the service provider can create, maintain, support, etc. a computer infrastructure that performs the process steps for one or more customers. In return, the service provider can receive payment from the customer (s) under a subscription and/or fee agreement, and/or the service provider can receive payment from the sale of advertising content to one or more third parties.

[0061] The present invention may be a system, a method, and/or a computer program product. The computer program product may include a computer readable storage medium (or media) (memory **504** and computer data storage unit **512**) having computer readable program instructions **514** thereon for causing a processor (e.g., CPU **502**) to carry out aspects of the present invention.

[0062] The computer readable storage medium can be a tangible device that can retain and store instructions (e.g., program code **514**) for use by an instruction execution device. The computer readable storage medium may be, for example, but is not limited to, an electronic storage device, a magnetic storage device, an optical storage device, an electromagnetic storage device, a semiconductor storage device, or any suitable combination of the foregoing. A non-exhaustive list of more specific examples of the computer readable storage medium includes the following: a portable computer diskette, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), a static random access memory (SRAM), a portable compact disc read-only memory (CD-ROM), a digital versatile disk (DVD), a memory stick, a floppy disk, a mechanically encoded device such as punch-cards or raised structures in a groove having instructions recorded thereon, and any suitable combination of the foregoing. A computer readable storage medium, as used herein, is not to be construed as being transitory signals per se, such as radio waves or other freely propagating electromagnetic waves, electromagnetic waves propagating through a waveguide or other transmission media (e.g., light pulses passing through a fiber-optic cable), or electrical signals transmitted through a wire.

[0063] Computer readable program instructions (e.g., program code **514**) described herein can be downloaded to respective computing/processing devices (e.g., computer **102**) from a computer readable storage medium or to an external computer or external storage device (e.g., computer data storage unit **512**) via a network (not shown), for example, the Internet, a local area network, a wide area network and/or a wireless network. The network may comprise copper transmission cables, optical transmission fibers, wireless transmission, routers, firewalls, switches, gateway computers and/or edge servers. A network adapter card (not shown) or network interface (not shown) in each computing/processing device receives computer readable program instructions from the network and forwards the computer readable program instructions for storage in a computer readable storage medium within the respective computing/processing device.

[0064] Computer readable program instructions (e.g., program code **514**) for carrying out operations of the present invention may be assembler instructions, instruction-set-architecture (ISA) instructions, machine instructions, machine dependent instructions, microcode, firmware instructions, state-setting data, or either source code or object code written in any combination of one or more programming languages, including an object oriented programming language such as Smalltalk, C++ or the like, and conventional procedural programming languages, such as the "C" programming language or similar programming languages. The computer readable program instructions may execute entirely on the user's computer, partly on the user's computer, as a stand-alone software package, partly on the user's computer and partly on a remote computer or entirely on the remote computer or server. In the latter scenario, the remote computer may be connected to the user's computer through any type of network, including a local area network (LAN) or a wide area network (WAN), or the connection may be made to an external computer (for example, through the Internet using an Internet Service Provider). In some embodiments, electronic circuitry including, for example, programmable logic circuitry, field-programmable gate arrays (FPGA), or programmable logic arrays (PLA) may execute the computer readable program instructions by utilizing state information of the computer readable program instructions to personalize the electronic circuitry, in order to perform aspects of the present invention.

[0065] Aspects of the present invention are described herein with reference to flowchart illustrations (e.g., FIG. **2**) and/or block diagrams (e.g., FIG. **1** and FIG. **5**) of methods, apparatus (systems), and computer program products according to embodiments of the invention. It will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, can be implemented by computer readable program instructions (e.g., program code **514**).

[0066] These computer readable program instructions may be provided to a processor (e.g., CPU **502**) of a general purpose computer, special purpose computer, or other programmable data processing apparatus (e.g., computer **102**) to produce a machine, such that the instructions, which execute via the processor of the computer or other programmable data processing apparatus, create means for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks. These computer readable program instructions may also be stored in a computer readable storage medium (e.g., computer data storage unit **512**) that can direct a computer, a programmable data processing apparatus, and/or other devices to function in a particular manner, such that the computer readable storage medium having instructions stored therein comprises an article of manufacture including instructions which implement aspects of the function/act specified in the flowchart and/or block diagram block or blocks.

[0067] The computer readable program instructions (e.g., program code **514**) may also be loaded onto a computer (e.g. computer **102**), other programmable data processing apparatus, or other device to cause a series of operational steps to be performed on the computer, other programmable apparatus or other device to produce a computer implemented process, such that the instructions which execute on the computer, other programmable apparatus, or other device implement the functions/acts specified in the flowchart and/or block diagram block or blocks.

[0068] The flowchart and block diagrams in the Figures illustrate the architecture, functionality, and operation of possible implementations of systems, methods, and computer program products according to various embodiments of the present invention. In this regard, each block in the flowchart or block diagrams may represent a module, segment, or portion of instructions, which comprises one or more executable instructions for implementing the specified logical function (s). In some alternative implementations, the functions noted in the block may occur out of the order noted in the figures. For example, two blocks shown in succession may, in fact, be executed substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved. It will also be noted that each block of the block diagrams and/or flowchart illustration, and combinations of blocks in the block diagrams and/or flowchart illustration, can be implemented by special purpose hardware-based systems that perform the specified functions or acts or carry out combinations of special purpose hardware and computer instructions.

[0069] While embodiments of the present invention have been described herein for purposes of illustration, many modifications and changes will become apparent to those skilled in the art. Accordingly, the appended claims are intended to encompass all such modifications and changes as fall within the true spirit and scope of this invention.

What is claimed is:

1. A method of ranking candidate answers to a natural language question, the method comprising the steps of:

a computer receiving the natural language question from a user of a mobile device;

the computer generating candidate answers to the received natural language question;

the computer generating a lexicon based on contextual information of the user determined by the mobile device;

the computer forecasting contextual information of the user based on usage of the mobile device; and

based in part on the lexicon and the forecasted contextual information, the computer ranking the candidate answers to the natural language question.

2. The method of claim **1**, wherein the step of forecasting contextual information includes forecasting contextual information that includes a combination of:

applications that the user has run on the mobile device;

websites visited by the user via a browser provided by the mobile device;

textual articles displayed on the mobile device that were read by the user;

terms entered by the user into a search engine via the mobile device;

multimedia content captured by the mobile device and viewed by the user;

a geographic location of the user;

data specifying functions of the body of the user; and

data specifying movements of the user.

3. The method of claim **1**, further comprising the computer adjusting weights of a machine learning ensemble based on the forecasted contextual information, wherein the machine learning ensemble includes (1) logistic regression phases and routes, (2) dimensions of evidence refinement, (3) probability density functions, classification techniques, and regression techniques, (4) natural language processing systems, or (5) a data retrieval system.

4. The method of claim **1**, wherein the step of forecasting the contextual information includes utilizing a software-based forecaster that employs mobile-based time series manipulation and pattern recognition, autoregressive integrated moving average (ARIMA) based modeling, lag-based modeling, residual calculations, or probability density based functions.

5. The method of claim **4**, wherein the step of forecasting the contextual information includes querying the user whose response provides additional information to the forecaster so that inferences by the forecaster about the contextual information are increased as compared to inferences by the forecaster without the additional information.

6. The method of claim **1**, wherein the step of generating the lexicon includes generating the lexicon based on the contextual information determined by the mobile device, which includes at least one of: (1) textual information displayed by the mobile device to the user, (2) multimedia information displayed by the mobile device to the user, and (3) applications run on the mobile device by the user.

7. The method of claim **1**, further comprising the computer disambiguating senses of words in the lexicon based on the forecasted contextual information.

8. The method of claim **1**, further comprising providing at least one support service for at least one of creating, integrating, hosting, maintaining, and deploying computer-readable program code in the computer, the program code being executed by a processor of the computer to implement the steps of receiving, generating candidate answers, generating the lexicon, forecasting, and ranking.

9. A computer program product, comprising:

a computer-readable, storage device; and

a computer-readable program code stored in the computer-readable, storage device, the computer-readable program code containing instructions that are carried out by a central processing unit (CPU) of a computer system to implement a method of ranking candidate answers to a natural language question, the method comprising the steps of:

the computer system receiving the natural language question from a user of a mobile device;

the computer system generating candidate answers to the received natural language question;

the computer system generating a lexicon based on contextual information of the user determined by the mobile device;

the computer system forecasting contextual information of the user based on usage of the mobile device; and

based in part on the lexicon and the forecasted contextual information, the computer system ranking the candidate answers to the natural language question.

10. The program product of claim **9**, wherein the step of forecasting contextual information includes forecasting contextual information that includes a combination of:

applications that the user has run on the mobile device;

websites visited by the user via a browser provided by the mobile device;

textual articles displayed on the mobile device that were read by the user;

terms entered by the user into a search engine via the mobile device;

multimedia content captured by the mobile device and viewed by the user;

data specifying characteristics of the environment in proximity to the user, the characteristics measured by one or more sensors coupled to the mobile device;

a geographic location of the user;

data specifying functions of the body of the user; and

data specifying movements of the user.

11. The program product of claim **9**, wherein the method further comprises the computer system adjusting weights of a machine learning ensemble based on the forecasted contextual information, wherein the machine learning ensemble includes (1) logistic regression phases and routes, (2) dimensions of evidence refinement, (3) probability density functions, classification techniques, and regression techniques, (4) natural language processing systems, or (5) a data retrieval system.

12. The program product of claim **9**, wherein the step of forecasting the contextual information includes utilizing a software-based forecaster that employs mobile-based time series manipulation and pattern recognition, autoregressive integrated moving average (ARIMA) based modeling, lag-based modeling, residual calculations, or probability density based functions.

13. The program product of claim **12**, wherein the step of forecasting the contextual information includes querying the user whose response provides additional information to the forecaster so that inferences by the forecaster about the contextual information are increased as compared to inferences by the forecaster without the additional information.

14. The program product of claim **9**, wherein the step of generating the lexicon includes generating the lexicon based on the contextual information determined by the mobile device, which includes at least one of: (1) textual information displayed by the mobile device to the user, (2) multimedia information displayed by the mobile device to the user, and (3) applications run on the mobile device by the user.

15. A computer system comprising:

a central processing unit (CPU);

a memory coupled to the CPU; and

a computer readable storage device coupled to the CPU, the storage device containing instructions that are executed by the CPU via the memory to implement a method of ranking candidate answers to a natural language question, the method comprising the steps of:

the computer system receiving the natural language question from a user of a mobile device;

the computer system generating candidate answers to the received natural language question;

the computer system generating a lexicon based on contextual information of the user determined by the mobile device;

the computer system forecasting contextual information of the user based on usage of the mobile device; and

based in part on the lexicon and the forecasted contextual information, the computer system ranking the candidate answers to the natural language question.

16. The computer system of claim **15**, wherein the step of forecasting contextual information includes forecasting contextual information that includes a combination of:

applications that the user has run on the mobile device;

websites visited by the user via a browser provided by the mobile device;

textual articles displayed on the mobile device that were read by the user;

terms entered by the user into a search engine via the mobile device;

multimedia content captured by the mobile device and viewed by the user;

data specifying characteristics of the environment in proximity to the user, the characteristics measured by one or more sensors coupled to the mobile device;

a geographic location of the user;

data specifying functions of the body of the user; and

data specifying movements of the user.

17. The computer system of claim **15**, wherein the method further comprises the computer system adjusting weights of a machine learning ensemble based on the forecasted contextual information, wherein the machine learning ensemble includes (1) logistic regression phases and routes, (2) dimensions of evidence refinement, (3) probability density functions, classification techniques, and regression techniques, (4) natural language processing systems, or (5) a data retrieval system.

18. The computer system of claim **15**, wherein the step of forecasting the contextual information includes utilizing a software-based forecaster that employs mobile-based time series manipulation and pattern recognition, autoregressive integrated moving average (ARIMA) based modeling, lag-based modeling, residual calculations, or probability density based functions.

19. The computer system of claim **18**, wherein the step of forecasting the contextual information includes querying the user whose response provides additional information to the forecaster so that inferences by the forecaster about the contextual information are increased as compared to inferences by the forecaster without the additional information.

20. The computer system of claim **15**, wherein the step of generating the lexicon includes generating the lexicon based on the contextual information determined by the mobile device, which includes at least one of: (1) textual information displayed by the mobile device to the user, (2) multimedia information displayed by the mobile device to the user, and (3) applications run on the mobile device by the user.

* * * * *