



(51) International Patent Classification:

G06Q 10/06 (2012.01)

G06Q 30/02 (2012.01)

(21) International Application Number:

PCT/US2021/028529

(22) International Filing Date:

22 April 2021 (22.04.2021)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

202041021829

25 May 2020 (25.05.2020)

IN

16/930,279

15 July 2020 (15.07.2020)

US

(71) Applicant: MICROSOFT TECHNOLOGY LICENSING, LLC [US/US]; One Microsoft Way, Redmond, Washington 98052-6399 (US).

(72) Inventors: SHRIVASTAVA, Mayank; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). GOYAL, Sagar; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). BHATNAGAR, Sahil; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). CHEN, Pin-Jung;

MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). SHUKLA, Pushpraj; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). MUKHERJEE, Arko Provo; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US).

(74) Agent: SWAIN, Cassandra T. et al.; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, IT, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ,

(54) Title: SELF-SUPERVISED SYSTEM GENERATING EMBEDDINGS REPRESENTING SEQUENCED ACTIVITY

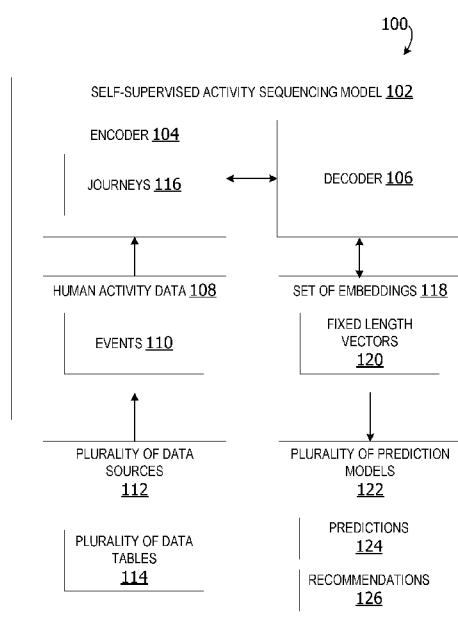


FIG. 1

(57) Abstract: The disclosure herein describes a system for generating embeddings representing sequential human activity by self-supervised, deep learning models capable of being utilized by a variety of machine learning prediction models to create predictions and recommendations. An encoder-decoder is provided to create user-specific journeys, including sequenced events, based on human activity data from a plurality of tables, a customer data platform, or other sources. Events are represented by sequential feature vectors. A user-specific embedding representing user activities in relationship to activities of one or more other users is created for each user in a plurality of users. The embeddings are updated in real-time as new activity data is received. The embeddings can be fine-tuned using labeled data to customize the embeddings for a specific predictive model. The embeddings are utilized by predictive models to create product recommendations and predictions, such as customer churn, next steps in a customer journey, etc.

UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
- *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

- *with international search report (Art. 21(3))*

SELF-SUPERVISED SYSTEM GENERATING EMBEDDINGS REPRESENTING SEQUENCED ACTIVITY

BACKGROUND

5 **[0001]** Many businesses utilize large scale enterprise data, most of which is tabular/structured data containing customers profiles and customer activities in the form of visit log, sale log and other transactions. This data frequently comes together in a Customer Data Platform (CDP) from various sources. The data can be analyzed by predictive models to generate predictions, which can include, for example, churn prediction, lifetime value
10 prediction, product recommendations, purchase propensity etc. Currently, most of these predictive scenarios are done on customized models that are built by working directly with customers. These models can be time-consuming and costly to develop. Moreover, custom models are not scalable across customers and do not work out of the box (OOB).

SUMMARY

15 **[0002]** This Summary is provided to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used as an aid in determining the scope of the claimed subject matter.

20 **[0003]** Some examples provide a system for generating embeddings representing sequential human activity. A data storage device stores unlabeled activity data associated with a plurality of users and a set of time indicators associated with the unlabeled activity data. The activity data describes events having order over time. A sequencing component creates a plurality of user-specific journeys based on the unlabeled activity data. A user-specific journey includes a set of sequential feature vectors corresponding to a set of events
25 associated with selected user placed into a sequence in accordance with the set of time indicators. An embedding component generates a plurality of embeddings based on sequential feature vectors in the plurality of user-specific journeys and non-sequential feature vectors associated with the plurality of users. An embedding includes a set of fixed length vectors representing sequential human activity of the selected user. The plurality of
30 embeddings configured for utilization by a plurality of prediction models for generating user-specific activity predictions and recommendations.

[0004] Other examples provide a method of generating embeddings representing sequential human activity. A sequencing component creates a plurality of user-specific journeys based on unlabeled activity data obtained from a plurality of data sources. An

embedding component generates a plurality of embeddings based on sequential feature vectors in the plurality of user-specific journeys and non-sequential feature vectors associated with the plurality of users. The plurality of embeddings comprising one embedding for each user in the plurality of users. The plurality of embeddings is output to
5 a set of machine learning (ML) prediction models for generating user-specific activity predictions and recommendations based on the activity data associated with the plurality of users.

[0005] Still other examples provide a computer storage device having computer-executable instructions stored thereon for generating embeddings representing sequential
10 human activity, which, on execution by a computer, cause the computer to perform operations including generating at least one user-specific journey based on unlabeled activity data describing human activities associated with at least one user; creating at least one embedding based on sequential feature vectors in the user-specific journey. An embedding comprising a set of fixed length vectors representing sequential human activity
15 of the user for generating user-specific activity predictions and recommendations by a plurality of prediction models. Analyzing the at least one embedding by a set of prediction models to generate a set of predictions associated with the at least one user. The set of predictions including at least one prediction or recommendation.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The disclosed examples are described in detail below with reference to the
20 accompanying drawing figures listed below:

[0007] FIG. 1 is an exemplary block diagram illustrating a system for generating fixed-length embeddings representing sequenced human activity for use in generating predictions and product recommendations.

[0008] FIG. 2 is an exemplary block diagram illustrating an encoder-decoder framework
25 for training a self-supervised activity sequencing model for generating embeddings.

[0009] FIG. 3 is an exemplary block diagram illustrating a self-supervised activity sequencing model including an encoder for generating embeddings.

[0010] FIG. 4 is an exemplary block diagram illustrating a decoder associated with a self-
30 supervised activity sequencing model.

[0011] FIG. 5 is an exemplary block diagram illustrating a self-supervised activity sequencing model generating user-specific journeys based on activity data from a plurality of data sources.

[0012] FIG. 6 is an exemplary block diagram illustrating an embedding representing a set

of events in a user-specific journey.

[0013] FIG. 7 is an exemplary block diagram illustrating a self-supervised activity sequencing model generating embeddings for use by a set of prediction models based on data obtained from a customer data platform.

5 [0014] FIG. 8 is an exemplary block diagram illustrating a plurality of prediction models generating predictions based on embeddings.

[0015] FIG. 9 is an exemplary block diagram illustrating a set of prediction models generating a set of predictions based on at least one embedding.

10 [0016] FIG. 10 is an exemplary block diagram illustrating fine-tuning a self-supervised activity sequencing model for generating fine-tuned predictions.

[0017] FIG. 11 is an exemplary block diagram illustrating a data storage device storing a plurality of tables of data.

[0018] FIG. 12 is an exemplary graph 1200 illustrating explainability data.

15 [0019] FIG. 13 is an exemplary flow chart illustrating operation of the computing device to generate embeddings representing user-specific journeys.

[0020] FIG. 14 is an exemplary flow chart illustrating operation of the computing device to generate embeddings representing human activity for use by prediction models to generate predictions and recommendations.

20 [0021] FIG. 15 is an exemplary flow chart illustrating operation of the computing device to train a neural network deep learning model for generating embeddings representing human activity.

[0022] FIG. 16 is a block diagram of an example computing device 1600 for implementing aspects disclosed herein and is designated generally as computing device 1600.

25 [0023] Corresponding reference characters indicate corresponding parts throughout the drawings.

DETAILED DESCRIPTION

30 [0024] The various examples will be described in detail with reference to the accompanying drawings. Wherever possible, the same reference numbers will be used throughout the drawings to refer to the same or like parts. References made throughout this disclosure relating to specific examples and implementations are provided solely for illustrative purposes but, unless indicated to the contrary, are not meant to limit all examples.

[0025] Some aspects of the disclosure describe a system for generating customer activity sequence prediction and recommendation (CASPR) embeddings representing sequenced user activity. The system utilizes a self-supervised neural network which does not need

much human input. This reduces or eliminates user time spent building and training customized models.

[0026] The system in other examples generates self-supervised deep embeddings based on customer activity sequences for business prediction problems, such as predicting churn or generating product recommendations. The deep learning approach enables the system to encapsulate semantic information about user and their activity patterns and behaviors as embeddings which are task independent, and common across prediction problems. The embeddings are compatible for consumption by various prediction models. The embeddings generated by the system may be used to solve business prediction problems without the need to customize features for every problem.

[0027] In other examples, the examples provide a self-supervised activity sequencing model that generates embeddings which are standardized such that the embeddings can be generated with minimal input from an external customer. The generated embeddings are self-supervised, scalable and can serve as the common base layer across different end applications, such as predicting customer churn, recommending products for customers, lifetime business value of customers, etc. The resulting predictions made using such embeddings are much better than standardized machine learning model predictions, and the embeddings compare favorably even against custom-made specific models by data scientists.

[0028] FIG. 1 is an exemplary block diagram illustrating a system 100 for generating fixed-length embeddings representing sequenced human activity for use in generating predictions and product recommendations. In some examples, a self-supervised activity sequencing model 102 is a self-supervised, neural network deep learning model having an encoder 104 and decoder 106 framework. The encoder 104 analyzes human activity data 108 describing events 110 having an order in time obtained from a plurality of data sources 112.

[0029] In some examples, the human activity is described as customer data. However, in other examples, the human activity is patient activity data, education/student activity data, travel activity data, sports data, or any other type of data. In one example, patient history data, including hospitalization, immunization data, diet, exercise, and other health related data, could be used by the system to generate health related predictions and/or generate recommendations associated with diet, exercise, check-ups, etc.

[0030] The plurality of data sources 112 includes any sources of human activity data 108. The plurality of data sources 112 in some examples includes a plurality of data tables 114

storing or organizing different types of activity data for one or more users. Some non-limiting examples of activity data stored on the plurality of data tables 114 includes, for example but without limitation, visit logs, customer support calls, subscription data, transaction data, product return data, web search data, or any other data describing human activities or events.

[0031] The encoder 104 in some examples is a neural network algorithm that analyzes the human activity data from the plurality of data sources to create unified customer journeys 116. A journey in the journeys 116 places the events 110 described in one or more data sources into a single, unified timeline or sequence of activities. In other words, the journeys 116 provide a set of one or more sequenced events for each user in a plurality of users based on data from the plurality of data sources 112.

[0032] The encoder 104 performs a sequence to vector conversion to generate a set of one or more embeddings based on the one or more customer journeys 116. An embedding in the set of embeddings 118 is a set of one or more fixed-length vectors representing the set of events in a given user-specific journey in the journeys 116. In some examples, an embedding includes a set of ten (10) vectors representing the set of events in the user journey. However, the examples are not limited to a ten-vector embedding. In other examples, an embedding can include eight vectors, twelve vectors or any other number of vectors to represent the user's journey. The number of vectors included within the embedding is a function of the different types of data sources available and the amount of data provided (history).

[0033] The set of embeddings 118 for one or more users is compatible for utilization by a plurality of prediction models 122. In other words, the same embedding may be used by multiple different prediction models to generate predictions 124 and/or recommendations 126 customized for one or more users based on the set of embeddings.

[0034] The self-supervised activity sequencing model 102 in this example is a self-supervised, deep learning model. Self-supervised models are trained using unlabeled training data. In this example, the unlabeled training data includes unlabeled, historical activity data. The activity data is analyzed by the encoder to generate the set of embeddings 118. The decoder 106 re-creates the original input. Any errors in the recreated input activity data are backpropagated to adjust the weights of the network. The regenerated activity data is compared with the originally input training data (historical activity data). Errors in the regenerated activity data (portions of regenerated activity data that does not match the input training data) are identified and used to adjust weights applied to the inputs and/or

algorithms used by the encoder to generate the set of embeddings. This enables training of the self-supervised activity sequencing model 102 without labeled training data.

[0035] Thus, in some examples, the encoder 104 takes in sequential feature inputs like transaction data, subscription data, support calls, etc. It combines the sequential feature inputs with non-sequential features, such as demographic data (age, gender, residence—city, state). The demographic (non-sequential) data gets combined with sequential data to generate an embedding. The embedding is a representation of a specific human user in a fixed number of decimal values. The output of the encoder is dimensional sequential feature vectors and non-sequential feature vectors into a compressed fixed length embedding.

[0036] In some examples, the self-supervised activity sequencing model 102 operates with little to no input from human users. The self-supervised activity sequencing model 102 uses auto-encoder frameworks to generate parts of the input and do multi-task optimizations on different parts of the input. The encoder-decoder frameworks utilize sequential models with attention and combined with non-sequential data (e.g. profile data) in linear layers in the encoder. The generated embeddings can be used as input to other models and fine-tuned using DL techniques.

[0037] Deep learning is an attractive option to build out of the box ML models, because it removes the need for extensive feature engineering and puts the burden on a good pre-trained neural architecture underneath. Moreover, there is a natural overlap in prediction problems – the same customer activities and their order over time defines the outcome of most of these tasks, be it customer churn, lifetime value or purchase propensity.

[0038] The self-supervised activity sequencing model 102 encapsulates semantic information about customers and their activity patterns and behaviors, in a way that does not require much human input and is common across prediction problems. In some examples, the embeddings provide encoding used for multiple customer activity-based prediction problems.

[0039] The embeddings can be used for a variety of different applications. They can simply be used as inputs to other models, they can be fine-tuned using deep learning techniques. The embeddings can serve as the base horizontal layer for a lot of different types of models. In some examples, the self-supervised activity sequencing model 102 can be used on scenarios like predicting customer churn, lifetime value, recommending products for customers and for segmenting customers based on activity and behavior. These scalable embeddings when used for such applications can match or, in some cases, exceed custom and non-scalable machine learning models.

[0040] FIG. 2 is an exemplary block diagram illustrating an encoder-decoder framework for training a self-supervised activity sequencing model 102 for generating embeddings. In some examples, the encoder includes a sequencing component 204 which generates a user-specific journey 206 based on a set of sequenced events 208. The set of sequenced events 208 are represented in a set of sequential feature vectors 210.

[0041] An embedding component 212 in some examples combines the sequential feature vectors 210 with non-sequential feature vectors 214 to create a user-specific embedding 216 based on the journey 206.

[0042] In this manner the system translates the user-specific journey into a set of numbers or values. The embeddings generated for multiple users have properties. If two human users (customers) have similar journeys and similar demographics, they will have similar embeddings (closer). The non-sequential feature vectors 214 includes a set of vectors representing non-sequential, user-specific data, such as, but not limited to, demographic data or other user profile data.

[0043] The user-specific embedding 216 is an embedding including a set of one or more vectors 218 representing a set of events for a selected user, such as, but not limited to, an embedding in the set of embeddings 118 in FIG. 1. In this example, the user-specific embedding is generated using unlabeled, input data 202. The input data 202 includes historical activity data as unlabeled training data for the model. After the embedding is generated by the model, the decoder creates regenerated data 224. The regenerated data is a re-creation or attempted recreation of the original input data 202.

[0044] A comparison component 220 performs back propagation 222 on the user-specific embedding. During back propagation, the comparison component 220 identifies any errors 226 in the regenerated data 224. The errors 226 include any mistakes or mismatches between the input data 202 and the regenerated data 224. The errors 226 are used to makes updates 228 to the model algorithms.

[0045] In some examples, the errors 226 are used to makes updates or weight adjustment 230 to the set of one or more weights 232. The set of weights 232 are applied to the inputs and/or connections within the neural network used to generate the embeddings. This process is continued iteratively until the encoder/decoder network is able to re-generate both sequential and non-sequential features with a certain degree of accuracy. The set of weights 232 are initialized using pre-set or default values prior to training or fine-tuning.

[0046] FIG. 3 is an exemplary block diagram illustrating a self-supervised activity sequencing model 102 including an encoder 104 for generating embeddings. The encoder

104 combines sequential feature vectors 302 and non-sequential vectors 304 data at a dense embedding layer 306.

[0047] In some non-limiting examples, the encoder 104 includes a long short-term memory 308 artificial neural network architecture and an attention layer 310. However, the
5 examples are not limited to an LSTM 308 or attention layer 310 for generating the embeddings. In another example, the encoder includes a dense layer 312 for modeling mathematical functions to generate the embedding 314.

[0048] FIG. 4 is an exemplary block diagram illustrating a decoder 106 associated with a self-supervised activity sequencing model 102. The decoder 106 performs back propagation
10 to create regenerated nonsequential feature vectors 402 and regenerated sequential feature vectors 404 for comparison with the original input data. Any errors in the regenerated data is used to adjust the weights used to train the model.

[0049] FIG. 5 is an exemplary block diagram illustrating a self-supervised activity sequencing model 102 generating user-specific journeys based on activity data from a
15 plurality of data sources. The plurality of data sources 112 includes user-related data which is ordered in time or which can be ordered in time. In some examples, the activity data includes a time stamp. A time stamp is a type of time indicator. In some examples, the activity data in the plurality of data sources 112 includes data such as, but not limited to, web activity 506, subscriptions 508, support requests 510, transactions 512 and/or returns
20 514.

[0050] In some examples, the web activity 506 includes web search activity (web browsing), e-commerce activity, online purchases, etc. The subscriptions 508 include subscription data associated with memberships, subscriptions to streaming services, etc. The support requests 510 includes customer support requests associated with goods or
25 services. Customer support requests can include phone calls to a support line, emails, as well chat support lines or other online customer support lines. Returns 514 in other examples refers to returning of a purchased product. A return can include an in-store return and/or an online return.

[0051] A user-specific journey 502 for a first user includes a set of sequenced events 516.
30 The set of sequenced events 516 is obtained from the activity data in the plurality of data sources 112. The events taken from different tables are placed into a unified sequence. In this non-limiting example, the set of sequenced events 516 in the user-specific journey 502 includes a website visit 518 which is followed by an item purchase 520, a support call 522 and then an item return 524. This set of sequenced events represents the activities performed

by a user sequenced in a chronological order.

[0052] Another user-specific journey 504 for a second user includes a different set of sequenced events 526. In this non-limiting example, the user-specific journey 504 includes a website visit 528, an in-store visit 530 followed by an in-store item purchase 532, another website visit 534 and then an online accessory purchase 536. The online accessory purchase can include the purchase of an accessory for the item purchased at the item in-store purchase 532.

[0053] FIG. 6 is an exemplary block diagram illustrating an embedding representing a set of events in a user-specific journey. In this non-limiting example, the user-specific journey 602 includes a set of events placed into a chronological order. In this example, a first event 604 which occurred first in time is placed first in the journey. The second event 606 is placed second. A third event 608 which occurred after the second event 606 and before the fourth event 610. The fifth event 612 which occurred after the other four events is placed last in the journey 602.

[0054] Sequences exist naturally in Enterprise tabular data as activities and events. For example, web activity data can include the following event sequence: home page -> search -> form-fill. Another non-limiting example of a journey including the following e-commerce related events: item visit -> add to cart -> buy. A user-specific journey generated at least in part based on subscription data associated with a user includes the following sequence of events: subscription start -> usage -> subscription end. Another example of event data can include the following:

Event = [Type of Web Visit (View, AddToCart, Buy), Type of Product (Product ID), Timestamp].

[0055] The system generates an embedding 614 based on the sequence of events in the journey 602. The embedding 614 includes a set of vectors 616 representing the set of events in the user-specific journey. In some examples, the system performs customer activity sequence prediction and recommendation (CASPR) embedding via self-supervised learning enabling automatically learnt compressed representation of sequence with LSTM Auto-encoders. The system generating CASPR embeddings can improve many CDP prediction problems where human activities and the sequence of those activities matter.

[0056] The embedding 614 vector values can vary within a range from negative one (-1.0) to one (1.0). For example, a vector can have a value of 0.8, 0.1, 0.5 or 0.3. Other exemplary vector values as shown, can include any fraction between one and negative one, such as, but not limited to, -0.5, -0.9, -0.1.

[0057] FIG. 7 is an exemplary block diagram illustrating a self-supervised activity sequencing model 102 generating embeddings for use by a set of prediction models based on data obtained from a customer data platform. In some examples, the self-supervised activity sequencing model 102 has an encoder-decoder framework 702.

5 [0058] A CDP 706 in some examples unifies activity data from the plurality of data sources into aggregated activity data 704 including one or more-time indicators 708. A CDP combines data stored across different platforms. A CDP 706 can bring together data about human users are fragmented in different data tables. The role of CDP 706 brings together fragmented data sources into a unified activity timeline of customer's journey (table
10 or timeline).

[0059] Time indicators 708 can include, for example but without limitation, one or more timestamps 710. A time stamp can include a date 712 and/or a time 714 at which a given event occurs. The time indicators 708 can be used to place events into an order.

[0060] The self-supervised activity sequencing model 102 analyzes the aggregated
15 activity data 704 to generate a plurality of user-specific journeys 716 for a plurality of users. Each user-specific journey includes a set of sequenced events 718.

[0061] The self-supervised activity sequencing model 102 performs sequence to vector embedding 720 on each user-specific journey to generate a plurality of user-specific embeddings 722. Each user-specific embedding is a fixed-length vector which can be used
20 by one or more models in the set of prediction models 726 for generating one or more predictions 728 and/or one or more recommendations 730.

[0062] FIG. 8 is an exemplary block diagram illustrating a plurality of prediction models 800 generating predictions based on embeddings. In some examples, the model utilizes sequence learning, including a generic deep learning formulation for events and activities
25 that have an order over time.

[0063] In some examples, a sequence prediction model 802 generates a prediction 804 based on a plurality of embeddings representing sequenced human activities for a plurality of users. The prediction can include time series prediction and/or product recommendations. In other examples, a sequence classification model 806 in the plurality of prediction models
30 generates a prediction 808. The prediction 808 can include, without limitation, sentiment analysis prediction, lead/opportunity scoring prediction, buying propensity prediction, and/or churn prediction.

[0064] A sequence to sequence prediction model 810 in some examples generates a prediction 812. The prediction 812 can include, for example but without limitation, a text

summarization, language translation and/or journey next steps prediction.

[0065] FIG. 9 is an exemplary block diagram illustrating a set of prediction models 726 generating a set of predictions 902 based on at least one embedding. In some examples, the set of prediction models 726 includes a machine learning prediction (MLP) model for predicting churn 904. The MLP model for churn 904 generates a prediction 906 using the set of embeddings 905. The prediction 906 predicts whether a user is likely to churn.

[0066] In other examples, the set of prediction models 726 includes an MLP model for recommendation 908. The MLP model for recommendation 908 generates a recommendation 910 based on the same input set of embeddings 905. The recommendation 910 can include a product recommendation, a discount/promotional recommendation, etc.

[0067] FIG. 10 is an exemplary block diagram illustrating fine-tuning a self-supervised activity sequencing model for generating fine-tuned predictions. In some examples, a trained model 1002 is fine-tuned using labeled data 1004. The trained model 1002 is a trained self-supervised activity sequencing model. The fine-tuning 1006 permits the trained model weights to be adjusted to fine-tune embeddings generated by the trained model 1002 for a specific prediction model 1008, rather than generating embeddings which can be used by a variety of models. In other words, once the model is fine-tuned using the labeled data, the fine-tuned model generates embeddings which are only suitable for use by a specific prediction model. The self-supervised activity sequencing models which are trained using unlabeled data generate embeddings which are suitable for use by a plurality of different prediction models.

[0068] In one example, the model is fine-tuned to generate embeddings customized for a churn prediction model. In these examples, the input activity data for users is weighted to indicate which types of activity data are more valuable/accurate for generating embeddings for predicting churn.

[0069] In an example scenario, labeled sequence data is fed into the trained neural network to generate embeddings and generate predictions for churn. The generated embeddings are compared with the labeled data to back propagate errors for predicting churn and fine tune the network to generate more accurate/useful embeddings for churn. Likewise, if the system has labeled data associated with embeddings used for generating product recommendation, the system can create a separate fine-tuned neural network which work very well for generating product recommendations. Both of these fine-tuned neural networks can be created by fine-tuning the same base model.

[0070] In other examples, the model is fine-tuned to generate customized embeddings for

a product recommendation prediction model. In these examples, the input activity data and ML model algorithms are weighted to indicate which activity data for users is more relevant for generating product recommendations. In other words, there are weights associated with connections in the neural network tuned to generate embeddings which are optimized for specific prediction and recommendation tasks.

[0071] FIG. 11 is an exemplary block diagram illustrating a data storage device 1100 storing a plurality of tables of data. The plurality of tables can be implemented as any type of tables, such as, but not limited to, SQL tables, relational tables, or any other type of data tables.

[0072] The data storage device 1100 can include one or more different types of data storage devices, such as, for example, one or more rotating disks drives, one or more solid state drives (SSDs), and/or any other type of data storage device. The data storage device 1100 in some non-limiting examples includes a redundant array of independent disks (RAID) array. In other examples, the data storage device 1100 includes a database.

[0073] The plurality of tables provides activity data from a plurality of sources. The plurality of tables can include, in some examples, a transaction table 1102 including transaction data 1104 describing transactions. A support table 1106 stores support data 1108 describing customer support requests associated with products or services. A returns table 1110 includes product returns 1112 data describing product returns and/or exchanges.

[0074] In other examples, web activity table 1114 stores web data 1116 describing online events, such as, but not limited to, online purchases, visits to websites, etc. A loyalty account 1118 includes customer rewards data 1120 in some examples. In other examples, a subscription table 1122 includes subscription data 1124 describing user subscriptions to services.

[0075] FIG. 12 is an exemplary graph 1200 illustrating explainability data. To add wider transparency and build trust in the models, the system can optionally generate explanations behind the different dimensions of the embeddings, which can be used to explain the predictions and applications that the embeddings are used to generate. This sheds light on what is traditionally a black-box setup with deep learning. In some examples, to generate these explanations, explainers such as DeepLift and Integrated Gradients are provided.

[0076] In this example, explainability is a separate output from the embedding, but output alongside embeddings. The explainability data is provided as the graph 1200. However, in other examples, the explainability data may be output in other formats, such as, but not limited to, charts, tables, line graphs, bar graphs, diagrams, or any other data format

indicating vectors in embeddings contributing to predictions and recommendations using those embeddings.

[0077] FIG. 13 is an exemplary flow chart illustrating operation of the computing device to generate embeddings representing user-specific journeys. The process shown in FIG. 13 is performed by a self-supervised activity sequencing model, executing on a computing device, such as the computing device computing device 1600 of FIG. 16.

[0078] The process begins by creating a plurality of user-specific journeys at 1302. A plurality of embeddings is generated based on the user-specific journeys at 1304. The embeddings are output to a set of predictive models at 1306. A determination is made whether new activity data is received at 1308. If yes, one or more embeddings are updated at 1310 based on the new activity data. The process continues iteratively executing operations 1302 through 1308 until no new activity data is received. The process terminates thereafter.

[0079] While the operations illustrated in FIG. 13 are performed by a computing device, aspects of the disclosure contemplate performance of the operations by other entities. In a non-limiting example, a cloud service performs one or more of the operations. In another example, one or more computer-readable storage media storing computer-readable instructions may execute to cause at least one processor to implement the operations illustrated in FIG. 13. Computer-readable storage media may also be referred to as computer-readable medium, computer-readable media and/or computer-readable storage medium.

[0080] FIG. 14 is an exemplary flow chart illustrating operation of the computing device to generate embeddings representing human activity for use by prediction models to generate predictions and recommendations. The process shown in FIG. 14 is performed by a self-supervised activity sequencing model, executing on a computing device, such as the computing device computing device 1600 of FIG. 16.

[0081] The process begins by determining whether dynamic activity data is received at 1402. If yes, events described in the activity data are sequenced at 1404. In some examples, the sequenced data is referred to as a user-specific journey. Embeddings are generated at 1405 based on the sequenced events. Embeddings are output to a set of prediction models at 1406. A determination is made to determine whether to continue at 1408. If yes, the process iteratively executes operations 1402 through 1408. The process terminates thereafter.

[0082] While the operations illustrated in FIG. 14 are performed by a computing device,

aspects of the disclosure contemplate performance of the operations by other entities. In a non-limiting example, a cloud service performs one or more of the operations. In another example, one or more computer-readable storage media storing computer-readable instructions may execute to cause at least one processor to implement the operations illustrated in FIG. 14.

[0083] FIG. 15 is an exemplary flow chart illustrating operation of the computing device to train a neural network deep learning model for generating embeddings representing human activity. The process shown in FIG. 15 is performed by a self-supervised activity sequencing model, executing on a computing device, such as the computing device 1600 of FIG. 16.

[0084] The process begins by receiving unlabeled training data at 1502. Journey(s) are generated at 1504. Each journey is a user-specific journey including sequenced events. A sequence to vector embedding is performed at 1506 to generate embedding(s). The model back propagates errors at 1508. During back propagation, the regenerated data is compared with the input training data at 1510. Any error(s) in the regenerated data are identified at 1512. A set of weights are adjusted at 1514 based on the weights. A determination is made whether to continue at 1516. If yes, the process iteratively executes operations 1502 through 1515 until a determination is made not to continue. A decision may be made to not continue when the encoder/decoder network is able to re-generate both sequential and non-sequential features with a certain degree of accuracy. The process terminates thereafter.

[0085] While the operations illustrated in FIG. 15 are performed by a computing device, aspects of the disclosure contemplate performance of the operations by other entities. In a non-limiting example, a cloud service performs one or more of the operations. In another example, one or more computer-readable storage media storing computer-readable instructions may execute to cause at least one processor to implement the operations illustrated in FIG. 15.

Additional Examples

[0086] Some aspects and examples disclosed herein are directed to a system, method and/or computer executable instructions for generating customer activity sequence predictions and recommendations comprising: a processor; and a computer-readable medium storing instructions that are operative upon execution by the processor to: generate at least one user-specific journey based on unlabeled activity data describing human activities associated with at least one user; create at least one embedding based on sequential feature vectors in the user-specific journey, an embedding comprising a set of fixed length

vectors representing sequential human activity of the user for generating user-specific activity predictions and recommendations by a plurality of prediction models; and analyze the at least one embedding to generate a set of predictions associated with the at least one user. The set of predictions comprising at least one prediction or recommendation.

5 **[0087]** Additional aspects and examples disclosed herein are directed to a system, method, or computer executable instructions for generating embeddings representing sequential human activity. A plurality of data sources associated with at least one data storage device store unlabeled activity data associated with a plurality of users and a set of time indicators associated with the unlabeled activity data, the activity data describing events having order
10 over time. A time indicator in the set of time indicators may be implemented as any device or component for indicating time, such as, but not limited to, a time stamp. A sequencing component creates a plurality of user-specific journeys based on the unlabeled activity data. A user-specific journey in the plurality of user-specific journeys comprising a set of sequential feature vectors corresponding to a set of events associated with selected user
15 placed into a sequence in accordance with the set of time indicators. An embedding component generates a plurality of embeddings based on sequential feature vectors in the plurality of user-specific journeys and non-sequential feature vectors associated with the plurality of users. An embedding in the plurality of embeddings comprising a set of fixed length vectors representing sequential human activity of the selected user. The plurality of
20 embeddings is output to a set of ML prediction models for generating user-specific activity predictions and recommendations based on the activity data associated with the plurality of users.

[0088] Additional aspects and examples disclosed herein are directed to a system, method and/or one or more computer storage devices having computer-executable instructions
25 stored thereon for creating a plurality of user-specific journeys based on unlabeled activity data obtained from a plurality of data sources, a user-specific journey in the plurality of user-specific journeys comprising a set of sequential feature vectors corresponding to a set of events associated with selected user placed into a sequence in accordance with the set of time indicators; generating a plurality of embeddings based on sequential feature vectors in
30 the plurality of user-specific journeys and non-sequential feature vectors associated with the plurality of users, an embedding in the plurality of embeddings comprising a set of fixed length vectors representing sequential human activity of the selected user; and outputting the plurality of embeddings to a set of ML prediction models for generating user-specific activity predictions and recommendations based on the activity data associated with the

plurality of users.

[0089] In an exemplary scenario, the system generates self-supervised embeddings for customer segmentation tasks. Customers can be segmented into various groups/segments based on embeddings, such as, but not limited to, new customers, returning customers, customers likely to return products, etc. Segmentation may be done using customer metrics to segment customer base into different populations which are not activity dependent. In one example, unsupervised clustering of all customer embeddings is performed to identify users close to each other on the dimensional space. This clustering enables identification of segments of users having similar features/characteristics, properties, etc.

[0090] In other examples, an embedding is generated for each human user (customer) in a group or population. If a group includes 100 customers, the system generates 100 sets of embeddings for the 100 customers. In another example if there are 100,000 customers, the system creates 100,000 embeddings. A unique embedding is generated for each embedding. Each embedding in the plurality of 100 embeddings is informed/influenced by every other embedding in the group such that human users having similar activities and similar demographics have similar embeddings.

[0091] In some examples, embeddings represent human user features and activities. Embeddings can include a numerical value indicating when the customer last visited a store or made a purchase. The embeddings remove need to manually generate customer features.

The decimal numbers within the embeddings incorporate all the available information about the human user available from the plurality of sources to remove the need for featurization.

[0092] In an example scenario, some dimensions of embeddings capture the fact that this customer has recently not purchased anything from a service. When a model uses the embedding to predict churn, the dimensions indicating lack of purchase informs the prediction that the customer is likely to churn.

[0093] In still other examples, explainability provides explanations as to which dimensions/values contributed to which predictions/recommendations. In one example, explainability may indicate that if the embedding generated by the model indicates the first-dimension value is greater than 0.8, the customer is likely to churn. This helps build user trust in the model. The system uses embeddings to generate predictions. The system provides explainability to explain the predictions.

[0094] The embeddings are applicable to wide variety of end applications. In some examples, the embeddings are generated using Enterprise data. In other examples, the model can be trained and/or embeddings can be generated using at least a portion of publicly

available data. In this manner, the system does not require large initial amounts of data to begin training.

[0095] In still other examples, the system models user-specific journeys using encoder and decoder framework. Self-supervised embeddings on customer journeys can be applied to end applications. The activity data is sequenced into events of a journey the human user took to generate embeddings. The system treats activities as sequence itself. Seeing the connected events in time that create a sequence enables the system to convert human activities into sequences of interconnected events (chain of events) that can be converted into a numeric form that machine can consume rather than treating the activities as isolated instances. This leads to improved predictions.

[0096] The system, in other examples, creates the journey from data obtained from multiple sources and puts the journey into sequence as an encoding. The system summarizes the journey as a mathematical form in the embedding. Thus, the system generates embeddings that is representing customer journey sequence from various data sources. CDP is meant to bring data sources together.

[0097] In another example scenario, the system can use data from health-care related sources. The system converts the data into sequenced activity data converted into numbers. The sequenced activities can include symptoms experienced on different days, meals eaten on each day, exercise on each day, etc. The resulting embeddings can be used to predict possible diagnosis, recommend treatments, predict treatment successes or side-effects, etc.

[0098] The system uses ML and weights to improve embeddings and predictions generated using the embeddings over time. In some examples, the system uses autoencoder frameworks during training using self-supervision to regenerate parts of the input and perform multi-task optimizations on different parts of the input without obtaining any input from customers or users. The encoder-decoder frameworks use sequential models with attention, which are combined with non-sequential data such as profile data, in linear layers in the encoder. This enables utilization of the self-supervised activity sequencing model without customization (OOB), because users do not have to provide training labels for their data.

[0099] In another example scenario, if data indicates a customer subscription canceled led to loss of the customer, the system learns to use subscription cancellation data as an indication of churn. The subscription canceled database can then be used to predict which current customers may cancel in future months.

[00100] Alternatively, or in addition to the other examples described herein, examples

include any combination of the following:

- an encoder-decoder framework of the neural network providing a self-supervised activity sequencing model for generating the plurality of embeddings based on the human activity data obtained from the plurality of data sources, wherein the sequencing component and the embedding component are part of the self-supervised activity sequencing model;
- analyze, by an encoder component, unlabeled input data for training the self-supervised activity sequencing model, wherein the unlabeled input data comprises historical human activity data;
- generate, by the encoder component, an embedding representing sequenced human activity;
- generate, by a decoder component, regenerated data;
- perform back propagation by a comparison component, the back propagation including a comparison of the regenerated data with the input data to identify a set of errors;
- update a set of weights used to generate the embeddings based on the identified set of errors;
- receive, from a customer data platform, updated human activity data representing a set of new activities by a second user in the plurality of users in real-time;
- generate, by the sequencing component, an updated user-specific journey for the second user; and
- create, by the embedding component, an updated user-specific embedding for the second user, wherein the updated user-specific embedding comprises a set of fixed length embeddings representing human activities associated with the second user, including the set of new activities described in the updated activity data;
- generate explainability data associated with at least one embedding in the plurality of embeddings;
- wherein the embedding component comprises at least one long short-term memory (LSTM) artificial recurrent neural network architecture for generating the plurality of embeddings;
- fine-tune the embedding component using labeled input data to generate embeddings for a predictive model selected from the set of ML prediction models;
- analyzing, by an encoder component, input data for training the self-supervised

activity sequencing model, wherein the input data comprises unlabeled historical human activity data;

-generating, by the encoder component, an embedding representing sequenced human activity;

5 -creating, by a decoder component, regenerated data based on the embedding(s);

-comparing, by a comparison component, the regenerated data with the input data to identify a set of errors;

-update a set of weights used to generate the embeddings based on the identified set of errors;

10 -receiving, from a customer data platform, updated human activity data representing a set of new activities by a second user in the plurality of users in real-time;

-generating, by the sequencing component, an updated user-specific journey for the second user;

-creating, by the embedding component, an updated user-specific embedding for the second user, wherein the updated user-specific embedding comprises a set of fixed length embeddings representing human activities associated with the second user, including the set of new activities described in the updated activity data;

15 -generating explainability data associated with at least one embedding in the plurality of embeddings;

20 -fine-tuning the embedding component using labeled input data to generate embeddings for a predictive model selected from the set of machine learning prediction models;

-analyzing at least one embedding by a predictive model to generate a customized product recommendation for a user;

25 -analyzing at least one embedding by a predictive model to generate a customer churn prediction;

-analyzing at least one embedding by a predictive model to generate a user-specific next steps journey prediction.

30 [00101] While the aspects of the disclosure have been described in terms of various examples with their associated operations, a person skilled in the art would appreciate that a combination of operations from any number of different examples is also within scope of the aspects of the disclosure.

Example Operating Environment

[00102] FIG. 16 is a block diagram of an example computing device 1600 for

implementing aspects disclosed herein and is designated generally as computing device 1600. Computing device 1600 is an example of a suitable computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the examples disclosed herein. Neither should computing device 1600 be interpreted as having any
5 dependency or requirement relating to any one or combination of components/modules illustrated. The examples disclosed herein may be described in the general context of computer code or machine-useable instructions, including computer-executable instructions such as program components, being executed by a computer or other machine, such as a personal data assistant or other handheld device. Generally, program components including
10 routines, programs, objects, components, data structures, and the like, refer to code that performs particular tasks, or implement particular abstract data types. The disclosed examples may be practiced in a variety of system configurations, including personal computers, laptops, smart phones, mobile tablets, hand-held devices, consumer electronics, specialty computing devices, etc. The disclosed examples may also be practiced in
15 distributed computing environments when tasks are performed by remote-processing devices that are linked through a communications network.

[00103] Computing device 1600 includes a bus 1610 that directly or indirectly couples the following devices: computer-storage memory 1612, one or more processors 1614, one or more presentation components 1616, I/O ports 1618, I/O components 1620, a power
20 supply 1622, and a network component 1624. While computing device 1600 is depicted as a seemingly single device, multiple computing devices 1600 may work together and share the depicted device resources. For example, memory 1612 may be distributed across multiple devices, and processor(s) 1614 may be housed with different devices.

[00104] Bus 1610 represents what may be one or more busses (such as an address bus, data bus, or a combination thereof). Although the various blocks of FIG. 16 are shown with
25 lines for the sake of clarity, delineating various components may be accomplished with alternative representations. For example, a presentation component such as a display device is an I/O component in some examples, and some examples of processors have their own memory. Distinction is not made between such categories as “workstation,” “server,”
30 “laptop,” “hand-held device,” etc., as all are contemplated within the scope of FIG. 16 and the references herein to a “computing device.” Memory 1612 may take the form of the computer-storage media references below and operatively provide storage of computer-readable instructions, data structures, program modules and other data for computing device 1600. In some examples, memory 1612 stores one or more of an operating system, a

universal application platform, or other program modules and program data. Memory 1612 is thus able to store and access data 1612a and instructions 1612b that are executable by processor 1614 and configured to carry out the various operations disclosed herein.

[00105] In some examples, memory 1612 includes computer-storage media in the form of volatile and/or nonvolatile memory, removable or non-removable memory, data disks in virtual environments, or a combination thereof. Memory 1612 may include any quantity of memory associated with or accessible by computing device 1600. Memory 1612 may be internal to computing device 1600 (as shown in FIG. 16), external to computing device 1600 (not shown), or both (not shown). Examples of memory 1612 include, without limitation, random access memory (RAM); read only memory (ROM); electronically erasable programmable read only memory (EEPROM); flash memory or other memory technologies; CD-ROM, digital versatile disks (DVDs) or other optical or holographic media; magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices; memory wired into an analog computing device; or any other medium for encoding desired information and for access by computing device 1600. Additionally, or alternatively, memory 1612 may be distributed across multiple computing devices 1600, for example, in a virtualized environment in which instruction processing is carried out on multiple computing devices 1600. For the purposes of this disclosure, “computer storage media,” “computer-storage memory,” “memory,” and “memory devices” are synonymous terms for computer-storage memory 1612, and none of these terms include carrier waves or propagating signaling.

[00106] Processor(s) 1614 may include any quantity of processing units that read data from various entities, such as memory 1612 or I/O components 1620 and may include CPUs and/or GPUs. Specifically, processor(s) 1614 are programmed to execute computer-executable instructions for implementing aspects of the disclosure. The instructions may be performed by the processor, by multiple processors within computing device 1600, or by a processor external to client computing device 1600. In some examples, processor(s) 1614 are programmed to execute instructions such as those illustrated in the accompanying drawings. Moreover, in some examples, processor(s) 1614 represent an implementation of analog techniques to perform the operations described herein. For example, the operations may be performed by an analog client computing device 1600 and/or a digital client computing device 1600. Presentation component(s) 1616 present data indications to a user or other device. Exemplary presentation components include a display device, speaker, printing component, vibrating component, etc. One skilled in the art will understand and

appreciate that computer data may be presented in a number of ways, such as visually in a graphical user interface (GUI), audibly through speakers, wirelessly between computing devices 1600, across a wired connection, or in other ways. I/O ports 1618 allow computing device 1600 to be logically coupled to other devices including I/O components 1620, some of which may be built in. Example I/O components 1620 include, for example but without limitation, a microphone, joystick, game pad, satellite dish, scanner, printer, wireless device, etc.

[00107] Computing device 1600 may operate in a networked environment via network component 1624 using logical connections to one or more remote computers. In some examples, network component 1624 includes a network interface card and/or computer-executable instructions (e.g., a driver) for operating the network interface card. Communication between computing device 1600 and other devices may occur using any protocol or mechanism over any wired or wireless connection. In some examples, network component 1624 is operable to communicate data over public, private, or hybrid (public and private) using a transfer protocol, between devices wirelessly using short range communication technologies (e.g., near-field communication (NFC), Bluetooth™ branded communications, or the like), or a combination thereof. Network component 1624 communicates over wireless communication link 1626 and/or a wired communication link 1626a to a cloud resource 1628 across network 1630. Various different examples of communication links 1626 and 1626a include a wireless connection, a wired connection, and/or a dedicated link, and in some examples, at least a portion is routed through the internet.

[00108] Although described in connection with an example computing device 1600, examples of the disclosure are capable of implementation with numerous other general-purpose or special-purpose computing system environments, configurations, or devices. Examples of well-known computing systems, environments, and/or configurations that may be suitable for use with aspects of the disclosure include, but are not limited to, smart phones, mobile tablets, mobile computing devices, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, gaming consoles, microprocessor-based systems, set top boxes, programmable consumer electronics, mobile telephones, mobile computing and/or communication devices in wearable or accessory form factors (e.g., watches, glasses, headsets, or earphones), network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, virtual reality (VR) devices, augmented reality (AR) devices, mixed reality (MR)

devices, holographic device, and the like. Such systems or devices may accept input from the user in any way, including from input devices such as a keyboard or pointing device, via gesture input, proximity input (such as by hovering), and/or via voice input.

[00109] Examples of the disclosure may be described in the general context of computer-

5 executable instructions, such as program modules, executed by one or more computers or other devices in software, firmware, hardware, or a combination thereof. The computer-executable instructions may be organized into one or more computer-executable components or modules. Generally, program modules include, but are not limited to, routines, programs, objects, components, and data structures that perform particular tasks
10 or implement particular abstract data types. Aspects of the disclosure may be implemented with any number and organization of such components or modules. For example, aspects of the disclosure are not limited to the specific computer-executable instructions or the specific components or modules illustrated in the figures and described herein. Other examples of the disclosure may include different computer-executable instructions or
15 components having more or less functionality than illustrated and described herein. In examples involving a general-purpose computer, aspects of the disclosure transform the general-purpose computer into a special-purpose computing device when configured to execute the instructions described herein.

[00110] By way of example and not limitation, computer readable media comprise
20 computer storage media and communication media. Computer storage media include volatile and nonvolatile, removable, and non-removable memory implemented in any method or technology for storage of information such as computer readable instructions, data structures, program modules, or the like. Computer storage media are tangible and mutually exclusive to communication media. Computer storage media are implemented in
25 hardware and exclude carrier waves and propagated signals. Computer storage media for purposes of this disclosure are not signals per se. Exemplary computer storage media include hard disks, flash drives, solid-state memory, phase change random-access memory (PRAM), static random-access memory (SRAM), dynamic random-access memory (DRAM), other types of random-access memory (RAM), read-only memory (ROM),
30 electrically erasable programmable read-only memory (EEPROM), flash memory or other memory technology, compact disk read-only memory (CD-ROM), digital versatile disks (DVD) or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other non-transmission medium that can be used to store information for access by a computing device. In contrast, communication media

typically embody computer readable instructions, data structures, program modules, or the like in a modulated data signal such as a carrier wave or other transport mechanism and include any information delivery media.

[00111] The order of execution or performance of the operations in examples of the disclosure illustrated and described herein is not essential and may be performed in different sequential manners in various examples. For example, it is contemplated that executing or performing a particular operation before, contemporaneously with, or after another operation is within the scope of aspects of the disclosure. When introducing elements of aspects of the disclosure or the examples thereof, the articles “a,” “an,” “the,” and “said” are intended to mean that there are one or more of the elements. The terms “comprising,” “including,” and “having” are intended to be inclusive and mean that there may be additional elements other than the listed elements. The term “exemplary” is intended to mean “an example of.” The phrase “one or more of the following: A, B, and C” means “at least one of A and/or at least one of B and/or at least one of C.”

[00112] Having described aspects of the disclosure in detail, it will be apparent that modifications and variations are possible without departing from the scope of aspects of the disclosure as defined in the appended claims. As various changes could be made in the above constructions, products, and methods without departing from the scope of aspects of the disclosure, it is intended that all matter contained in the above description and shown in the accompanying drawings shall be interpreted as illustrative and not in a limiting sense.

CLAIMS

1. A system for generating embeddings representing sequential human activity, the system comprising:

a plurality of data sources associated with at least one data storage device storing unlabeled activity data associated with a plurality of users and a set of time indicators, the unlabeled activity data describing human activity-related events having order over time;

a computer-readable medium storing instructions that are operative upon execution by a processor to:

create, by a sequencing component, a plurality of user-specific journeys based on the unlabeled activity data, a user-specific journey in the plurality of user-specific journeys comprising a set of sequential feature vectors corresponding to a set of events associated with a selected user placed into a sequence in accordance with the set of time indicators; and

generate, by an embedding component, a plurality of embeddings based on sequential feature vectors in the plurality of user-specific journeys and non-sequential feature vector associated with the plurality of users, an embedding in the plurality of embeddings comprising a set of fixed-length vectors representing sequential human activity of the selected user, wherein the plurality of embeddings are suitable for utilization by a plurality of prediction models generate user-specific activity predictions and recommendations.

2. The system of claim 1, further comprises:

an encoder-decoder framework of the neural network providing a self-supervised activity sequencing model configured to generate the plurality of embeddings based on the unlabeled activity data obtained from the plurality of data sources, wherein the sequencing component and the embedding component are part of the self-supervised activity sequencing model.

3. The system of claim 2, wherein the instructions are further operative to:

analyze, by an encoder component, unlabeled input data for training the self-supervised activity sequencing model, wherein the unlabeled input data comprises historical human activity data;

generate, by the encoder component, an embedding representing sequenced human activity;

generate, by a decoder component, regenerated data based on the embedding;

perform, by a comparison component, back propagation, including comparing the

regenerated data with the input data to identify a set of errors; and

update a set of weights used to generate the embeddings based on the identified set of errors.

4. The system of claim 1, wherein the selected user is a first user and wherein instructions are further operative to:

receive, from a customer data platform, updated activity data representing a set of new activities by a second user in the plurality of users in real-time;

generate, by the sequencing component, an updated user-specific journey for the second user; and

create, by the embedding component, an updated user-specific embedding for the second user, wherein the updated user-specific embedding comprises a set of fixed length embeddings representing human activities associated with the second user, including the set of new activities described in the updated activity data.

5. The system of claim 1, wherein the instructions are further operative to:

generate explainability data associated with at least one embedding in the plurality of embeddings.

6. The system of claim 1, wherein the embedding component comprises at least one long short-term memory (LSTM) artificial recurrent neural network architecture for generating the plurality of embeddings.

7. The system of claim 1, wherein the instructions are further operative to:

fine-tune the embedding component using labeled input data to generate embeddings for a predictive model selected from a set of machine learning (ML) prediction models.

8. A method of generating embeddings representing sequential human activity, the method comprising:

creating, by a sequencing component, a plurality of user-specific journeys based on unlabeled activity data obtained from a plurality of data sources, a user-specific journey in the plurality of user-specific journeys comprising a set of sequential feature vectors corresponding to a set of events associated with selected user placed into a sequence in accordance with a set of time indicators;

generating, by an embedding component, a plurality of embeddings based on the set of sequential feature vectors in the plurality of user-specific journeys and non-sequential feature vectors associated with the plurality of users, an embedding in the plurality of embeddings comprising a set of fixed-length vectors representing sequential

human activity of the selected user; and

outputting the plurality of embeddings to a set of ML prediction models, wherein the set of ML prediction models generate user-specific activity predictions and recommendations based on the unlabeled activity data associated with a plurality of users.

9. The method of claim 8, further comprising:

analyzing, by an encoder component, input data for training a self-supervised activity sequencing model, wherein the input data comprises unlabeled historical human activity data;

generating, by the encoder component, an embedding representing sequenced human activity;

creating, by a decoder component, regenerated data based on the embedding;

comparing, by a comparison component, the regenerated data with the input data to identify a set of errors; and

update a set of weights used to generate the embeddings based on the identified set of errors.

10. The method of claim 8, further comprising:

receiving, from a customer data platform, updated activity data representing a set of new activities by a second user in the plurality of users in real-time;

generating, by the sequencing component, an updated user-specific journey for the second user; and

creating, by the embedding component, an updated user-specific embedding for the second user, wherein the updated user-specific embedding comprises a set of fixed length embeddings representing human activities associated with the second user, including the set of new activities described in the updated activity data.

11. The method of claim 8, further comprising:

generating explainability data associated with at least one embedding in the plurality of embeddings.

12. The method of claim 8, further comprising:

fine-tuning the embedding component using labeled input data to generate embeddings for a predictive model selected from the set of prediction models.

13. The method of claim 8, further comprising:

analyzing at least one embedding by a predictive model to generate a customized product recommendation for a user.

14. The method of claim 8, further comprising:

analyzing at least one embedding by a predictive model to generate a customer churn prediction.

15. One or more computer storage media having computer-executable instructions for generating embeddings representing sequential human activity that, upon execution by a processor, cause the processor to at least:

generate, by a sequencing component, at least one user-specific journey based on unlabeled activity data describing human activities associated with at least one user, the unlabeled activity data obtained from a plurality of data sources associated with at least one data storage device, the unlabeled activity data comprising a set of sequential feature vectors corresponding to a set of events associated with a user placed into a sequence in accordance with the set of time indicators;

create, by an embedding component, at least one embedding based on the set of sequential feature vectors in the user-specific journey, an embedding comprising a set of fixed-length vectors representing sequential human activity of the user for generating user-specific activity predictions and recommendations by a plurality of prediction models; and

generate, by a set of prediction models a set of predictions associated with the at least one user based on the at least one embedding, the set of predictions comprising at least one prediction or recommendation.

1/16

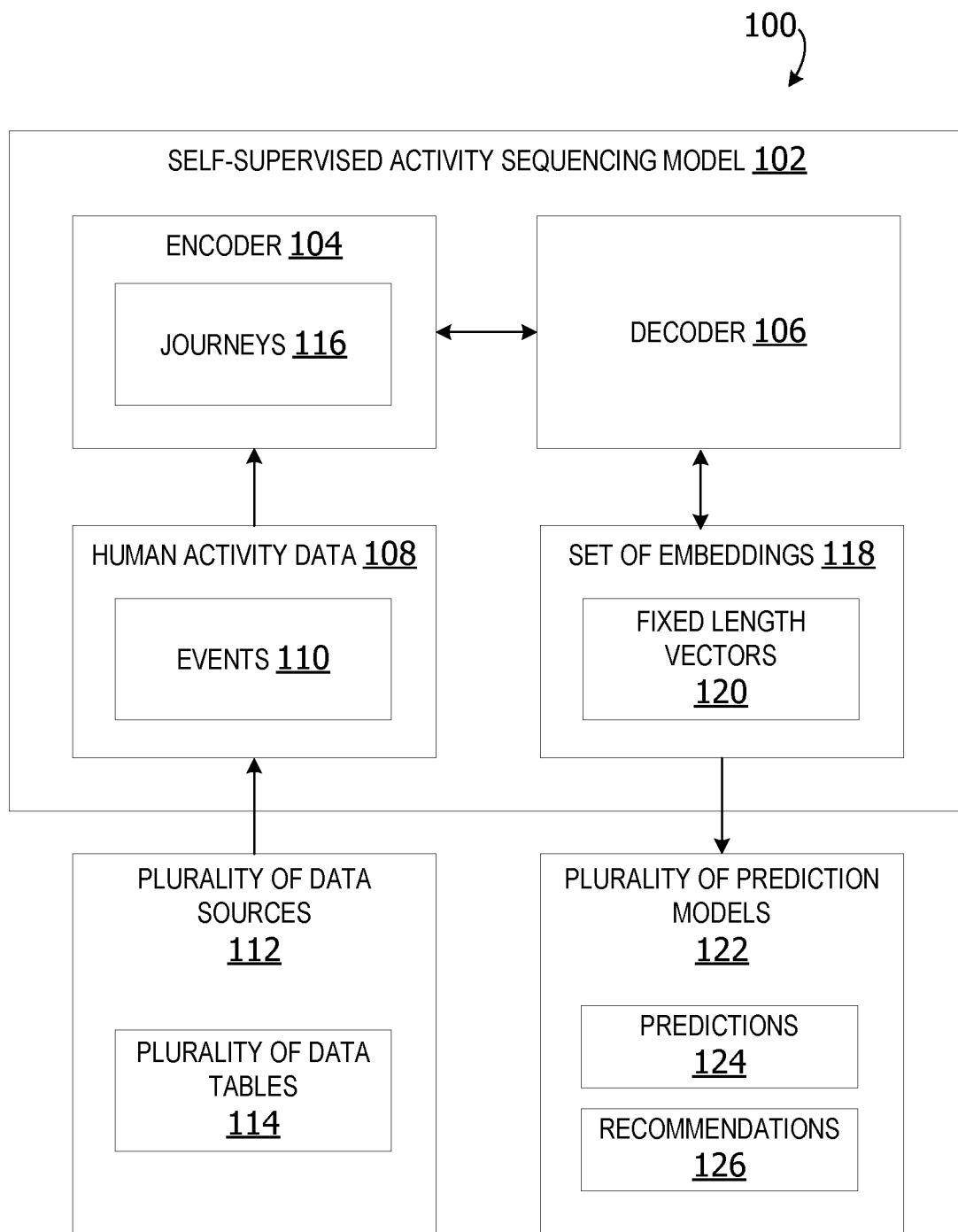


FIG. 1

2/16

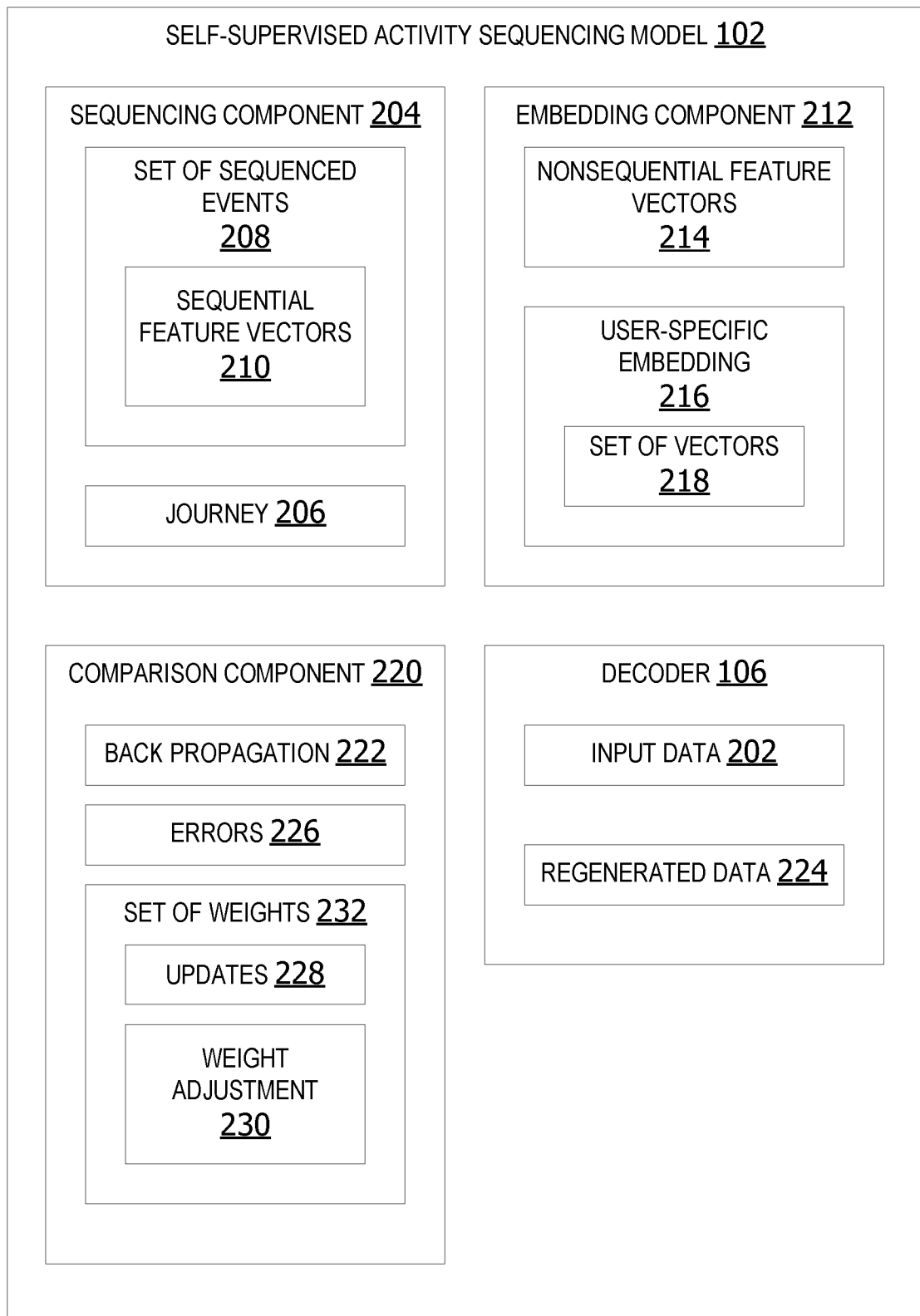


FIG. 2

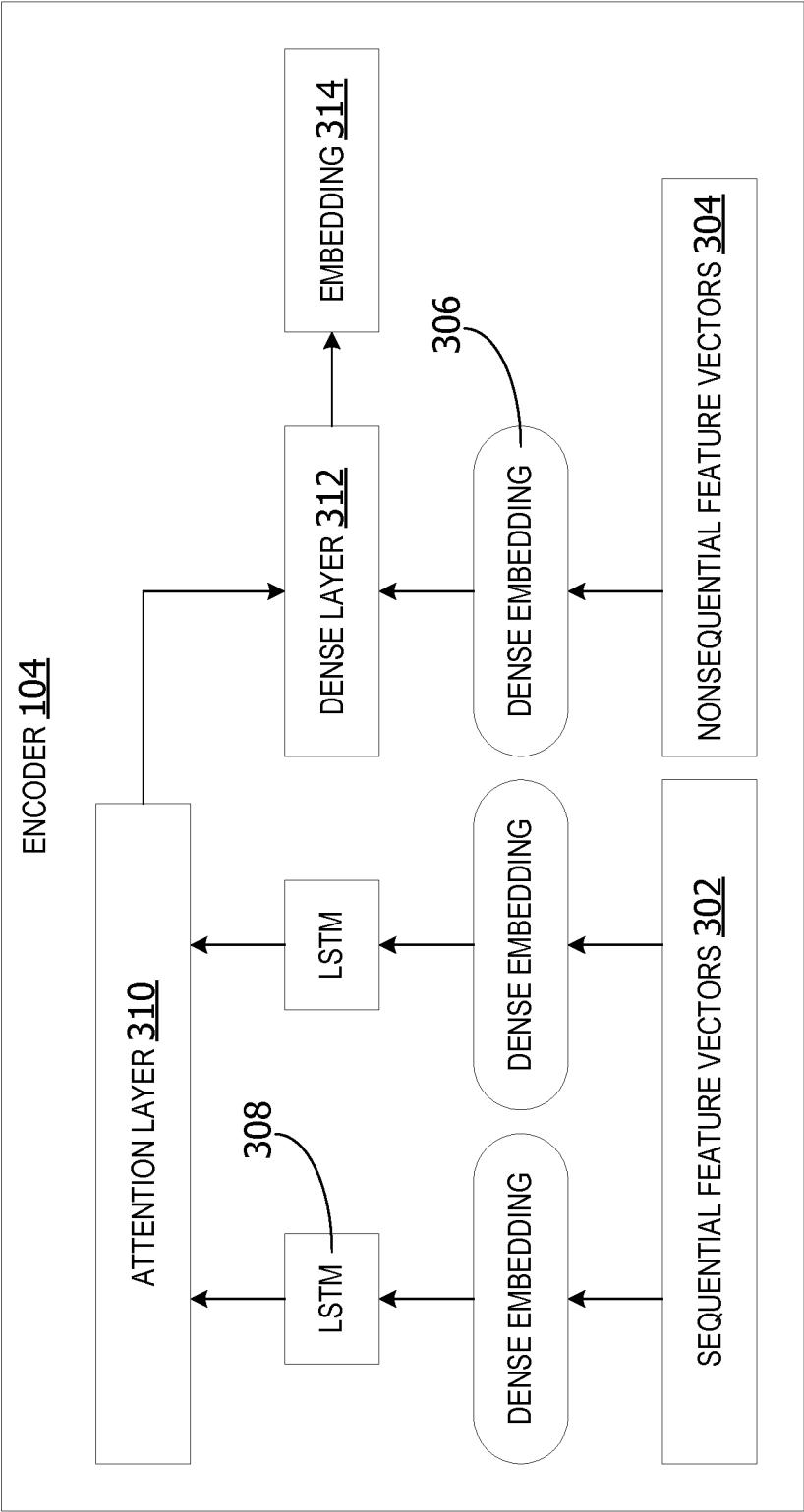


FIG. 3

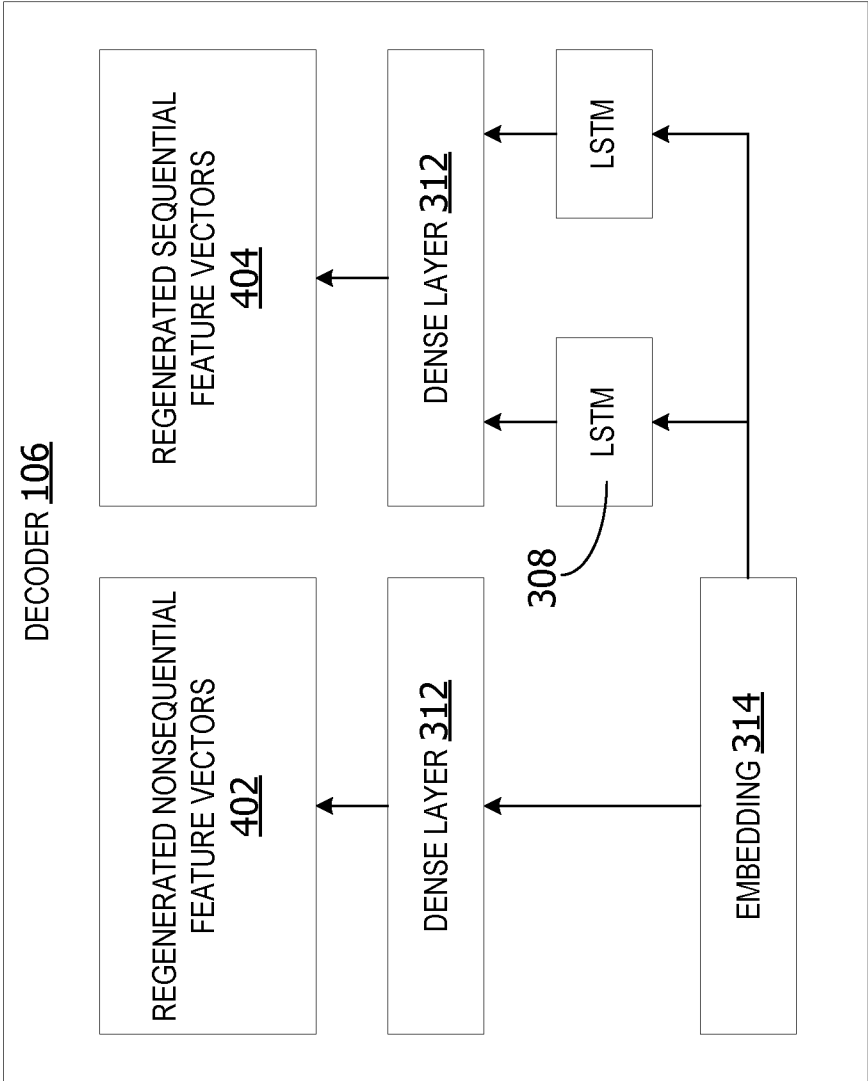


FIG. 4

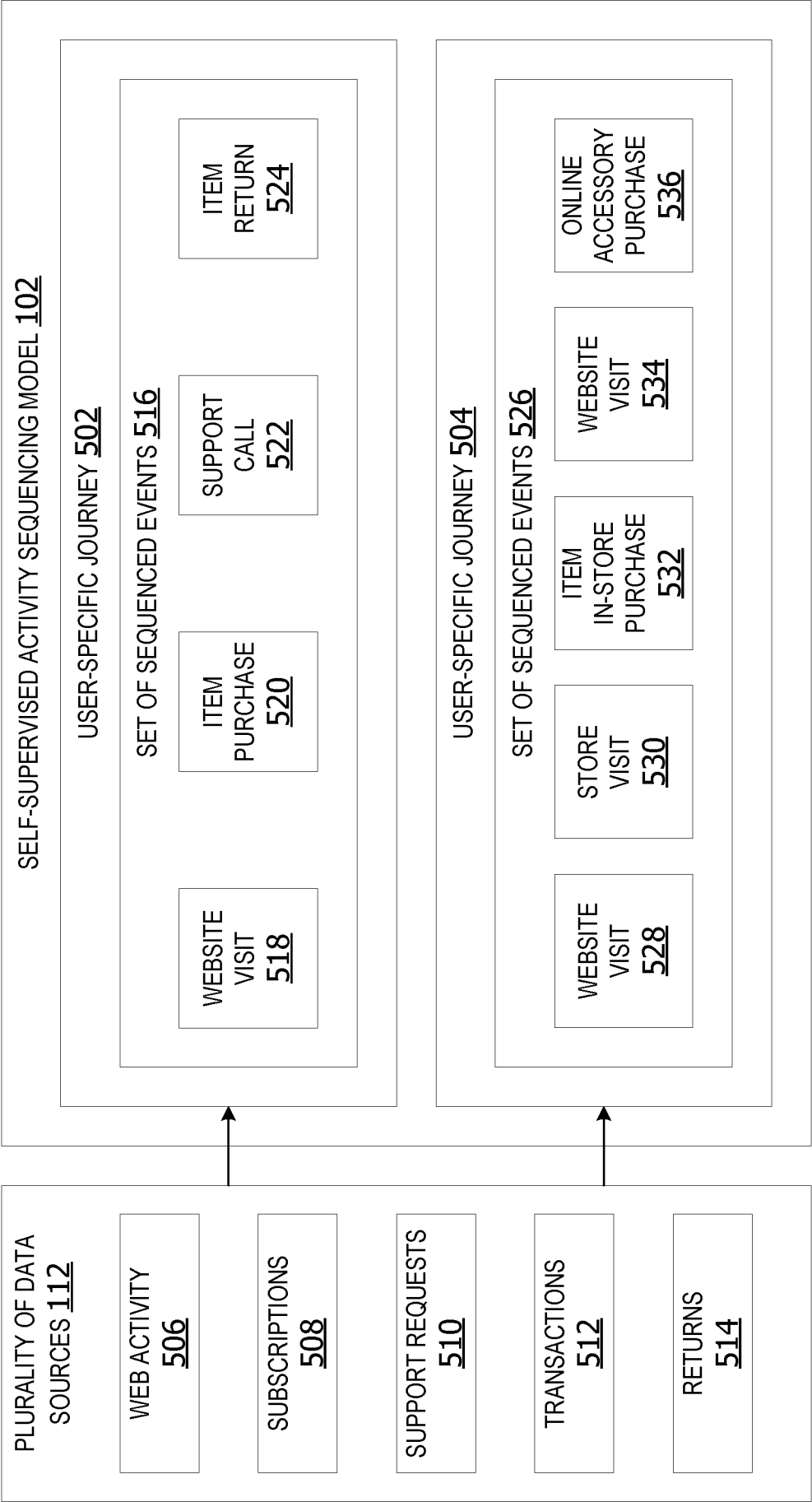


FIG. 5

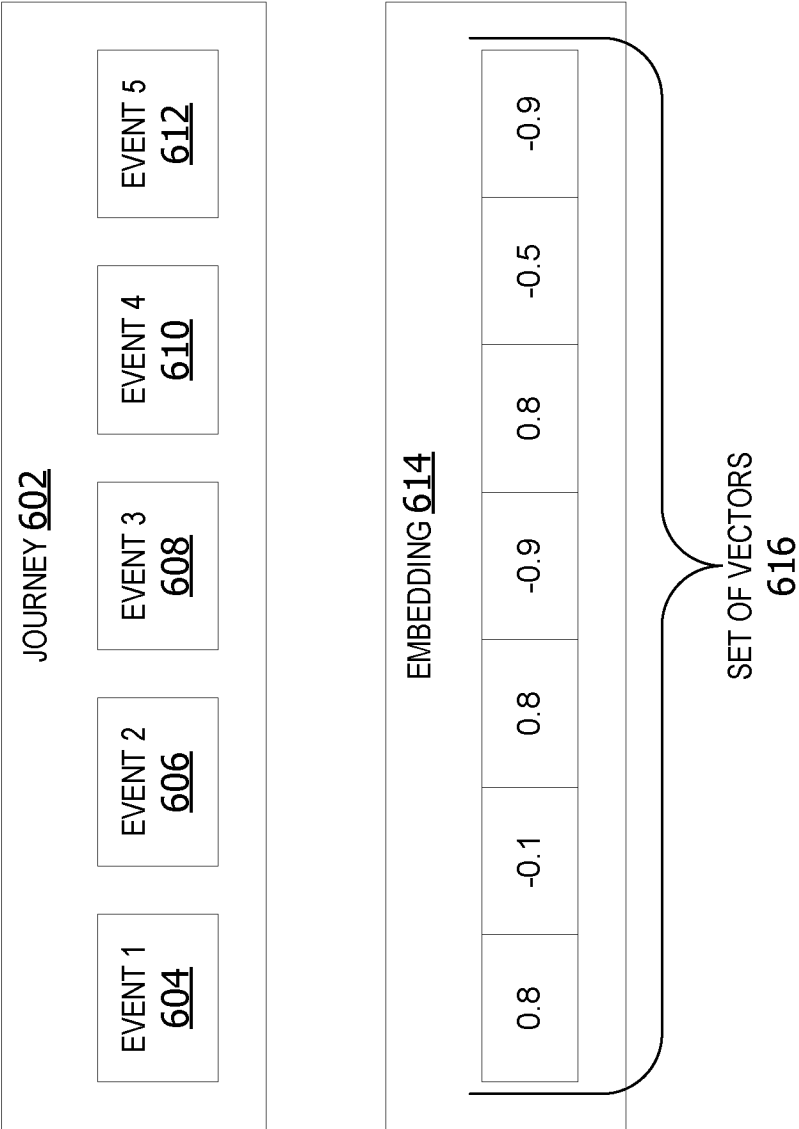


FIG. 6

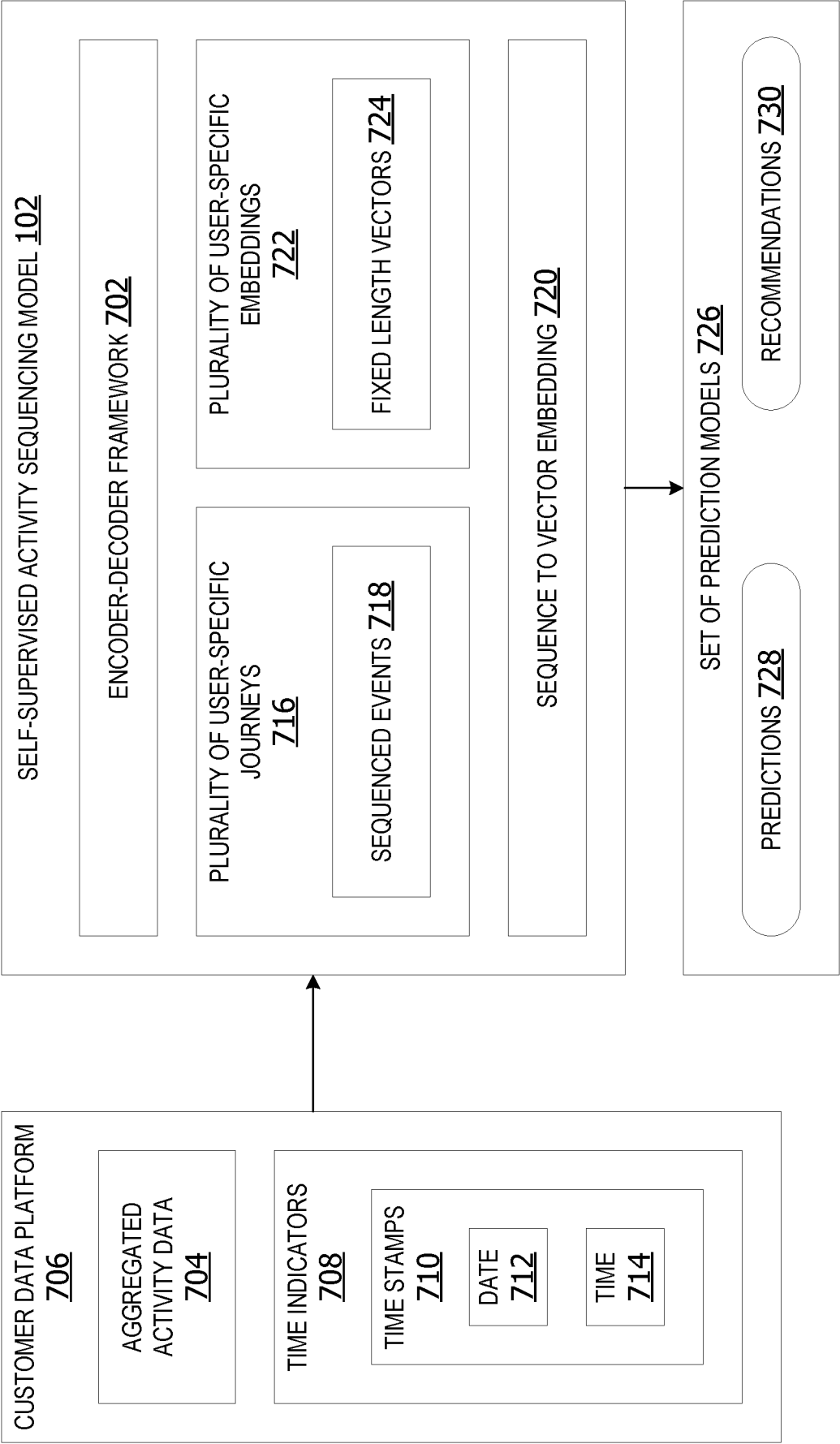


FIG. 7

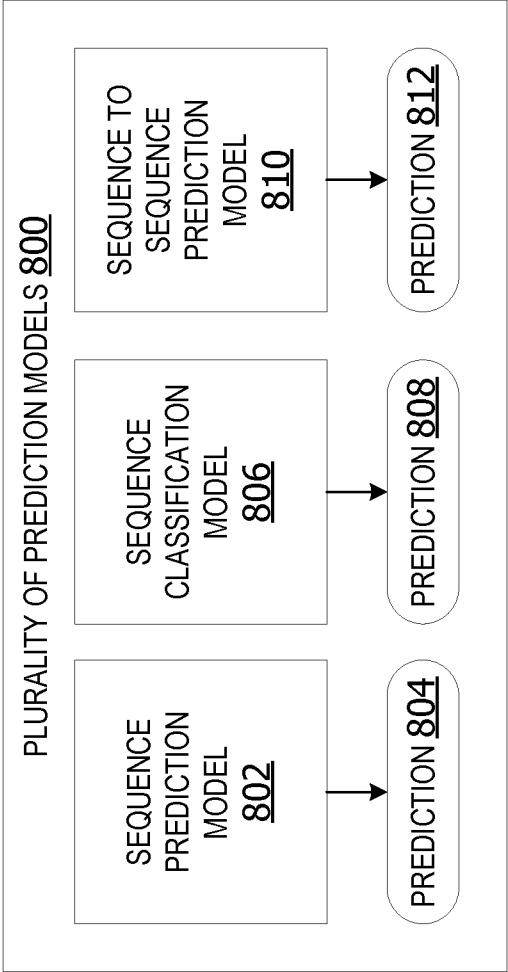


FIG. 8

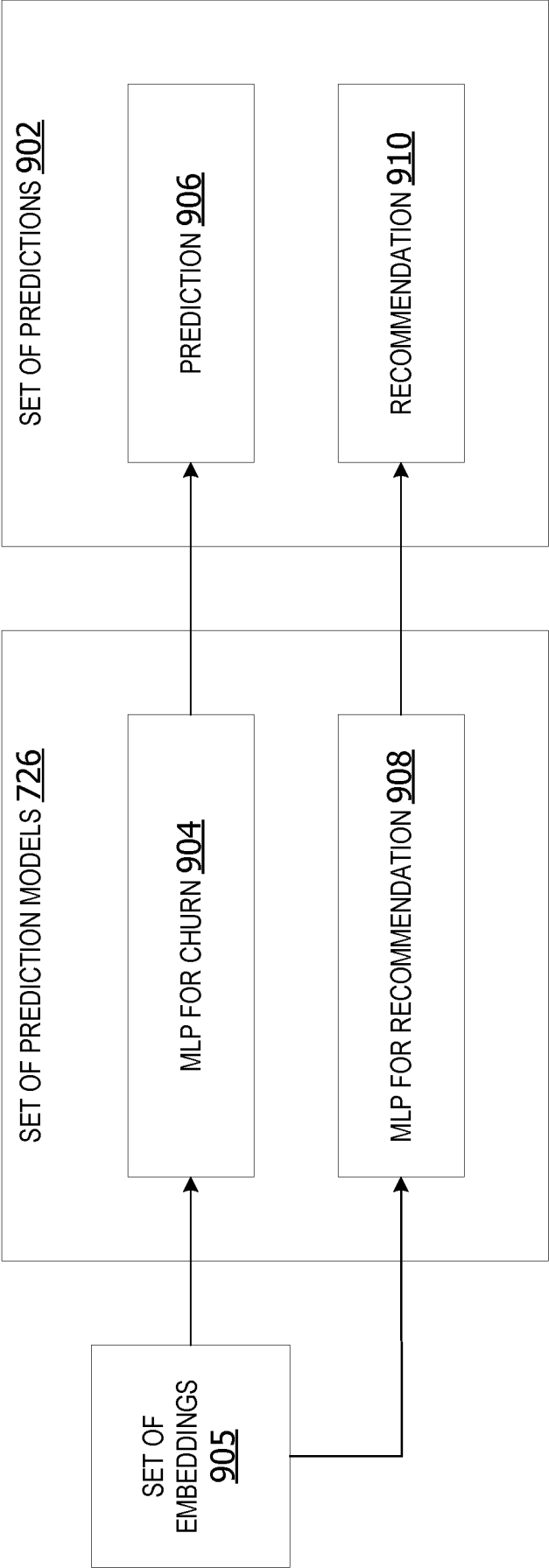
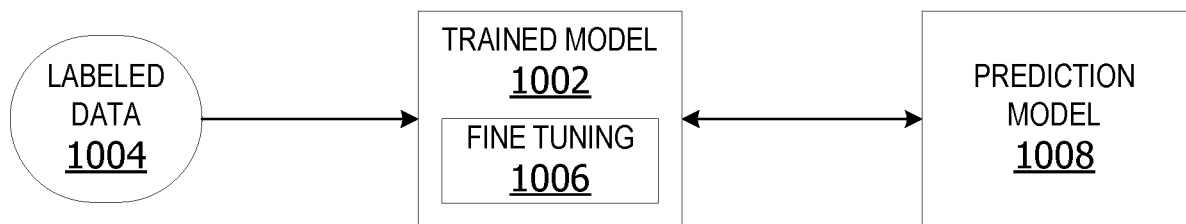


FIG. 9

10/16

*FIG. 10*

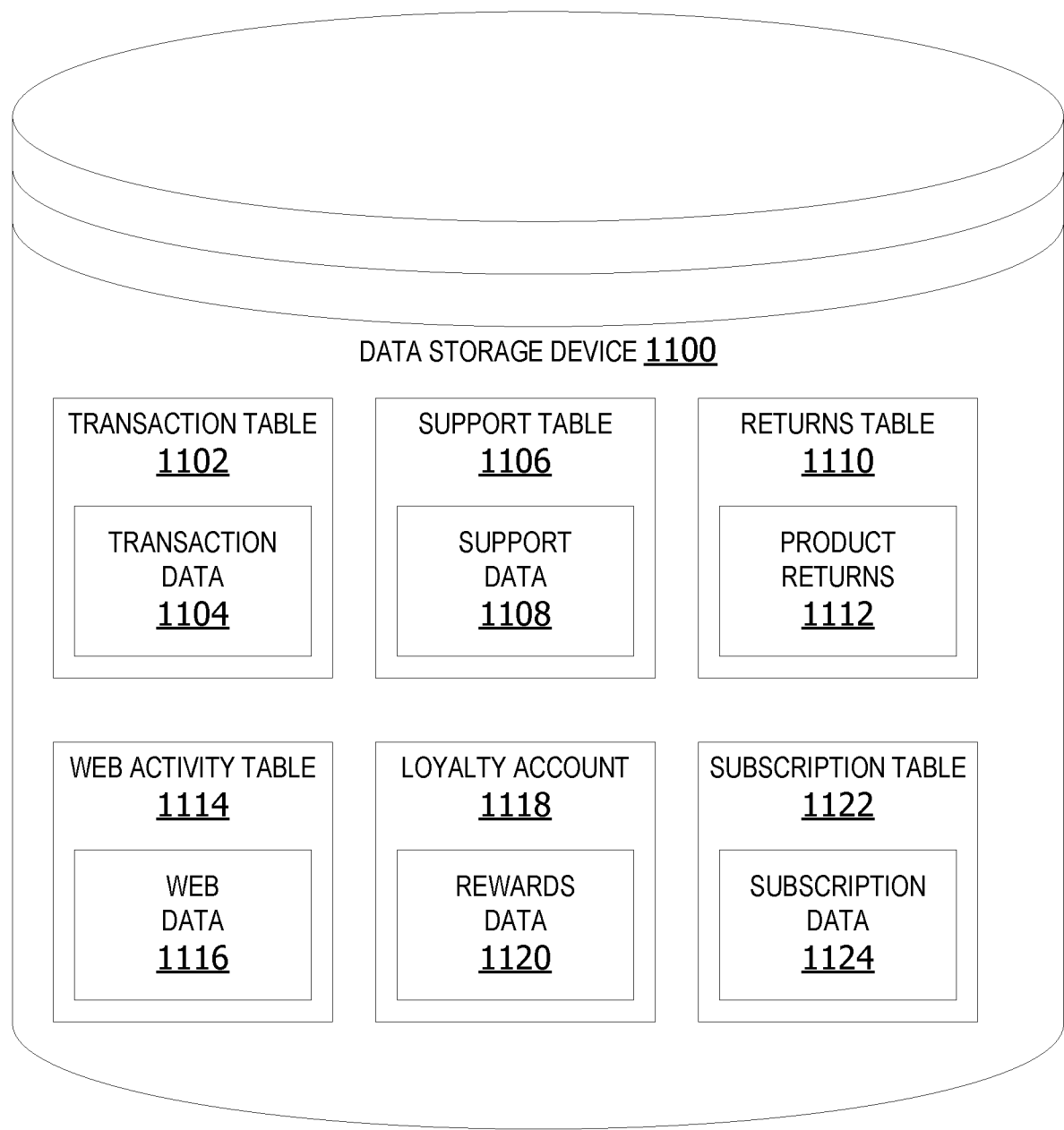


FIG. 11

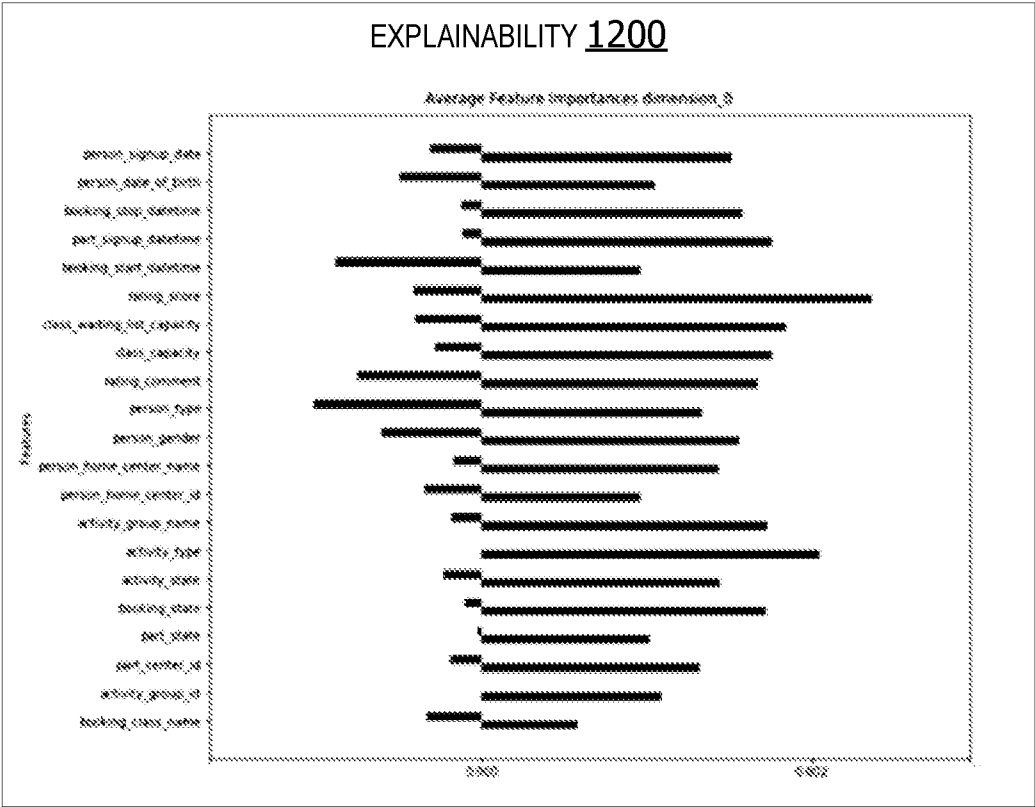
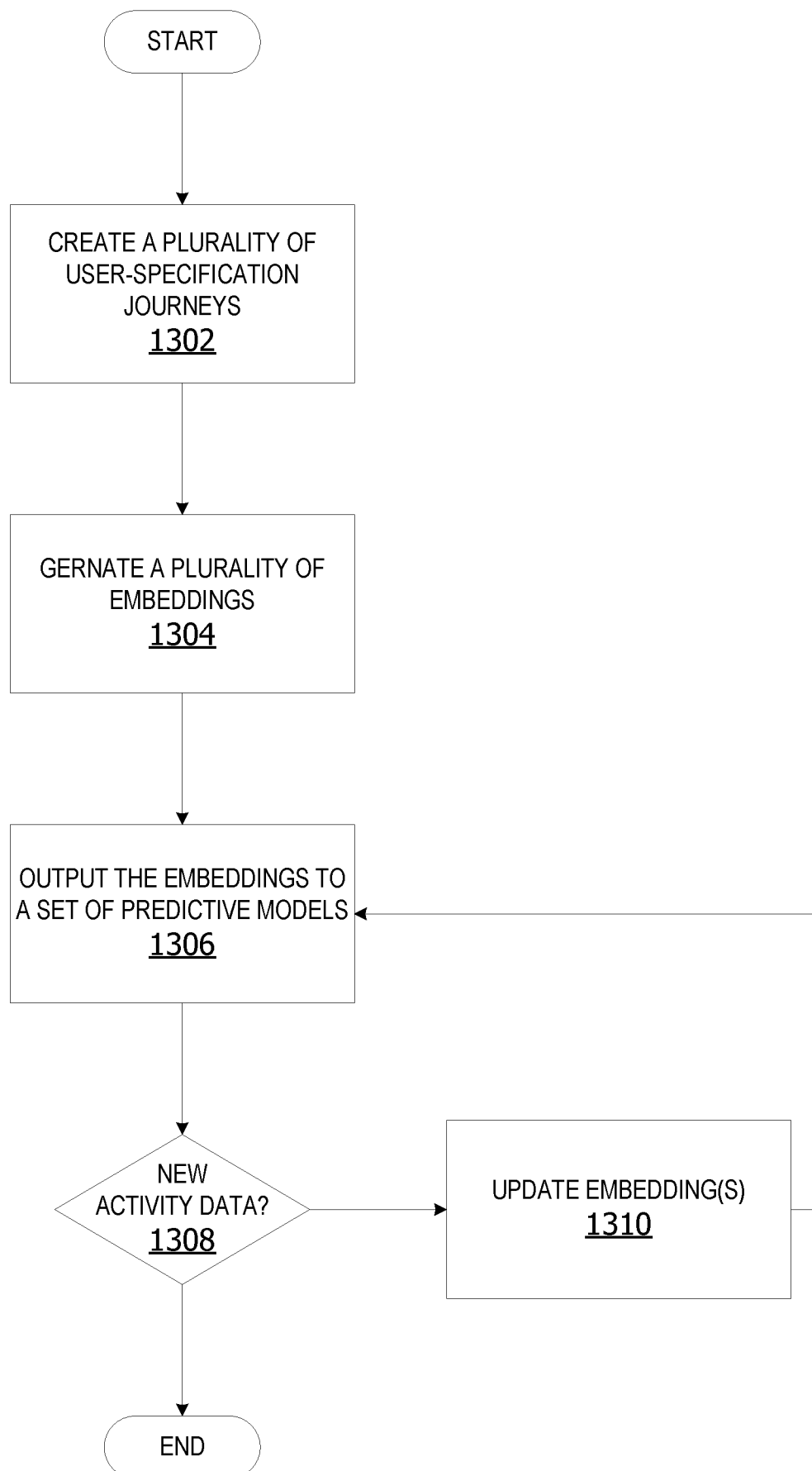
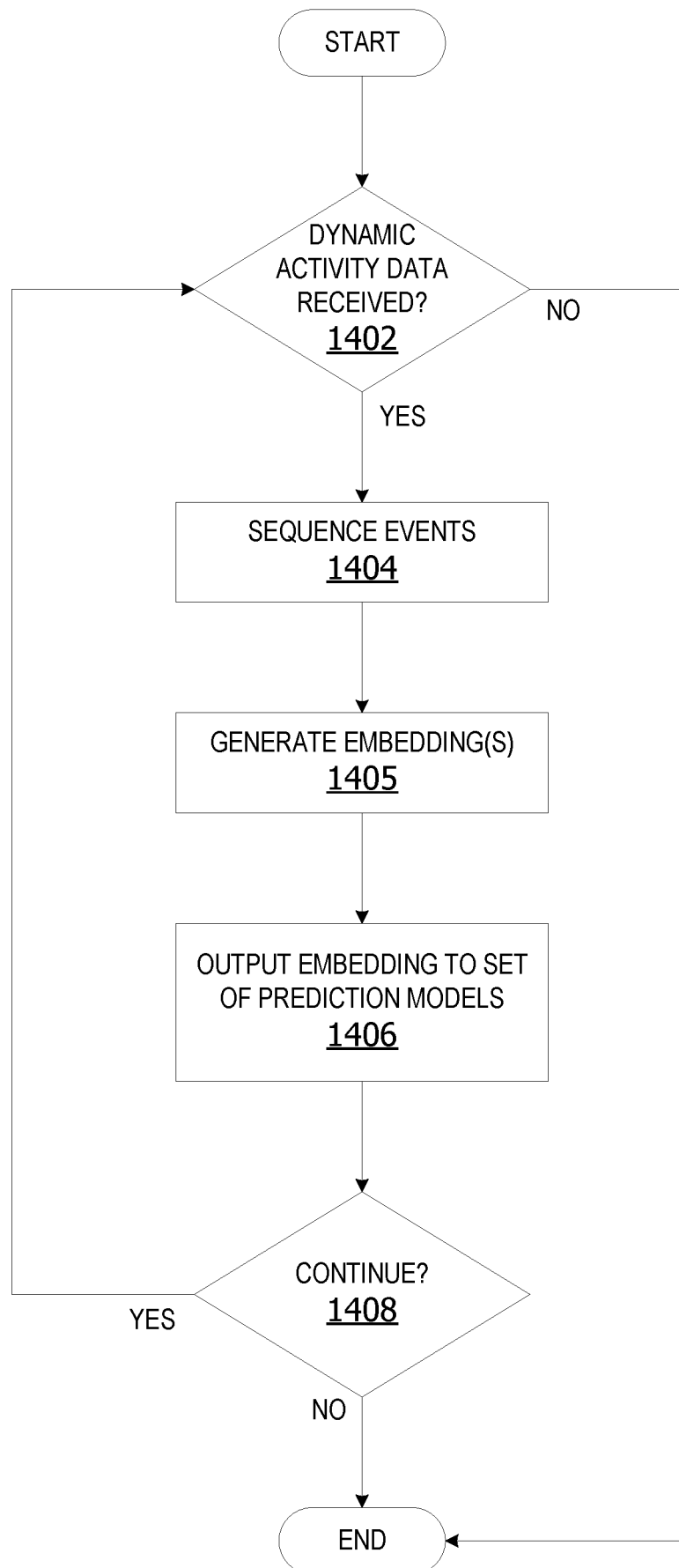


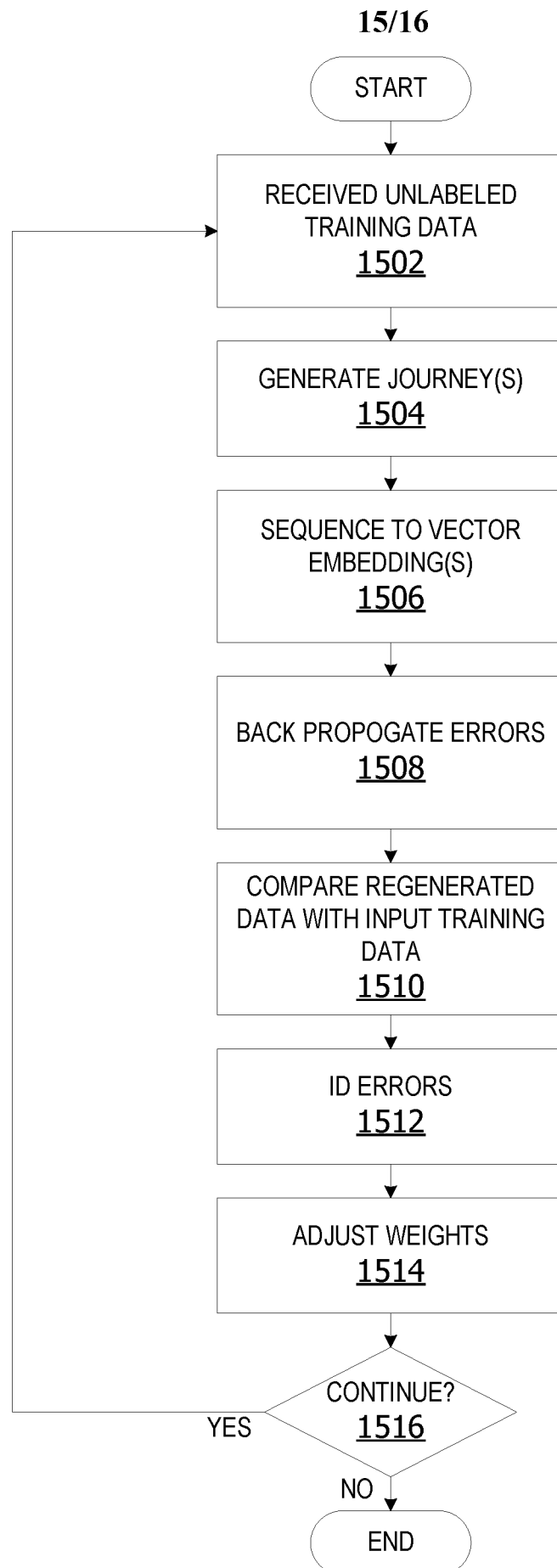
FIG. 12

13/16

**FIG. 13**

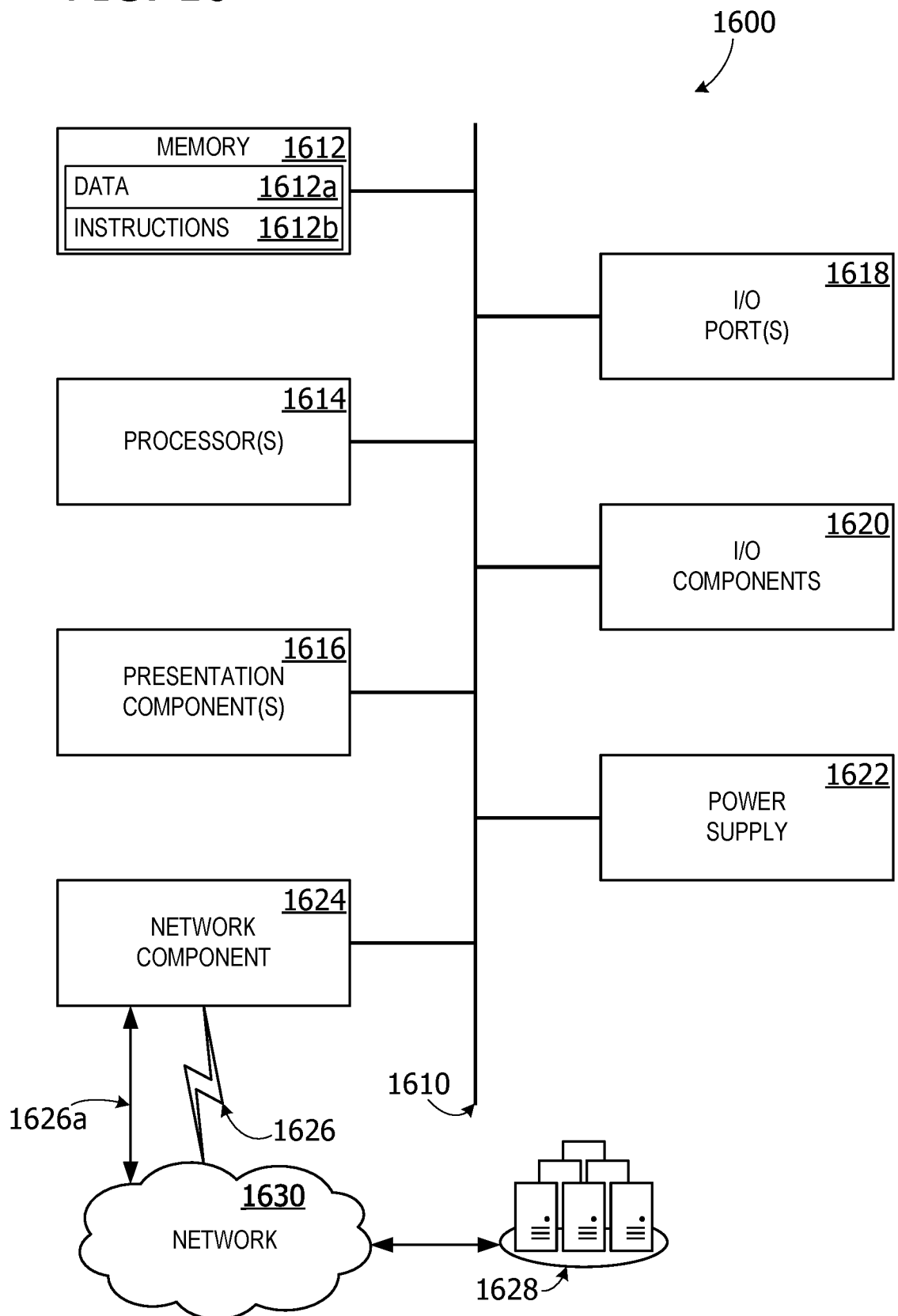
14/16

**FIG. 14**



16/16

FIG. 16



INTERNATIONAL SEARCH REPORT

International application No
PCT/US2021/028529

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06Q10/06 G06Q30/02
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2020/107072 A1 (LOMADA VIJETH [US] ET AL) 2 April 2020 (2020-04-02) paragraphs [0007], [0036], [0041], [0061] - [0064] paragraphs [0086] - [0127] paragraphs [0132], [0151] - [0154], [0157] - [0162] -----	1-15
X	US 2019/347523 A1 (ROTHBERG JONATHAN M [US] ET AL) 14 November 2019 (2019-11-14) paragraphs [0005] - [0035], [0096] - [0107], [0129] - [0133], [0175] - [0177] -----	1-15
A	US 2019/139092 A1 (NOMULA JAGADESHWAR [US]) 9 May 2019 (2019-05-09) paragraphs [0053] - [0058], [0081] - [0087], [0097], [0098] ----- -/--	1-15



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

5 July 2021

Date of mailing of the international search report

14/07/2021

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Breidenich, Markus

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2021/028529

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2017/372206 A1 (KAVADIKI RAJESH [IN] ET AL) 28 December 2017 (2017-12-28) paragraphs [0031] - [0038] -----	1-15

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2021/028529

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2020107072	A1	02-04-2020	NONE
US 2019347523	A1	14-11-2019	NONE
		AU 2019269312 A1	19-11-2020
		BR 112020022270 A2	23-02-2021
		CA 3098447 A1	21-11-2019
		CN 112119411 A	22-12-2020
		EP 3794512 A1	24-03-2021
		KR 20210010505 A	27-01-2021
		US 2019347523 A1	14-11-2019
		US 2021192290 A1	24-06-2021
		WO 2019221985 A1	21-11-2019
US 2019139092	A1	09-05-2019	NONE
US 2017372206	A1	28-12-2017	NONE