SYSTEM AND METHOD FOR PERFORMING REAL-TIME CONTEXT-AWARE RECOMMENDATION

TECHNICAL FIELD

[0001] Various aspects of this disclosure relate to a system for performing real-time context-aware recommendation. Various aspects of this disclosure relate to a method for performing real-time context-aware recommendation. Various aspects of this disclosure relate to a non-transitory computer-readable medium storing computer executable code comprising instructions for performing real-time context-aware recommendation. Various aspects of this disclosure relate to a computer executable code comprising instructions for performing real-time context-aware recommendation.

BACKGROUND

[0002] Recommendation systems can be modeled as sequences of user behaviors from clicks, or baskets to purchase. Nowadays, many recommendation systems focus on session based recommendations. Many recommendation systems use sequence models to predict the next item that a user is likely to visit. The sequence model can be modeled as: H={(x1, b1), (x2, b2), . . . , (xT-1, bT-1)} H ∈ S, where u represents target user, xj represents the j th item that the target user u interacts with and bj indicates the type of behavior (for example, click, collect, and purchase), to to predict the item xT that the target user u is most likely to visit next time.

[0003] Traditional sequence models focus on the scenario that users have various activities (clicks, purchases, etc.) or leverage the users' previous behaviours to learn the Hierarchical RNN representation (HRNN) to predict the next item.

[0004] In e-commerce business to customers (b2c) websites, the users usually have a clear target to purchase during a certain period and thus the traditional models will perform well to recommend the single target based on the users' session activities.

[0005] However, in a food delivery platform, the situation may be quite different, as a user may have multiple interests for different food delivery options. Furthermore, the user's interest may change in different contexts like locations (for example, home or offices) and date time slots (morning, noon, evening).

[0006] Traditional session based sequence model does not consider the user's real-time context features and multi-interest of both short term and long-term data. Further while some of the models may use user's geohashing, the system may not have a fine granularity as a user may be around a geohash boundary.

SUMMARY

[0007] Therefore, there is a need for an improved system for performing real-time context-aware recommendation.

[0008] An advantage of the present disclosure may include an improved system for performing real-time context-aware recommendation resulting in higher user satisfaction due to more accurate recommendations.

[0009] These and other aforementioned advantages and features of the aspects herein disclosed will be apparent through reference to the following description and the accompanying drawings. Furthermore, it is to be understood that the features of the various aspects described herein are not mutually exclusive and can exist in various combinations and permutations.

[0010] The present disclosure generally relates to a system for performing real-time context-aware recommendation. The system may include one or more processor(s); and a

memory having instructions stored therein, the instructions, when executed by the one or more processor(s), may cause the one or more processor(s) to: determine long-term data of a user; determine short-term data of the user; determine context data for the user; and perform the real-time context-aware recommendation based on the long-term data, the short-term data and the context data.

[0011] According to an embodiment, the long-term data may include previous purchase history of the user.

[0012] According to an embodiment, the short-term data may include current real-time clickstream data of the user.

[0013] According to an embodiment, the context data may include location data and time data of the user.

[0014] According to an embodiment, the one or more processor(s) may be configured to determine the long-term data using a first predetermined weight, determine the short-term data using a second predetermined weight and determine the context data using a third predetermined weight.

[0015] According to an embodiment, the one or more processor(s) may be configured to perform the real-time context-aware recommendation using a recommendation model. The recommendation model may include one or more long-term layers, one or more short-term layers, one or more context layers and one or more ranking layers. The one or more ranking layers may receive inputs from the one or more long-term layers, the one or more short-term layers and the one or more context layers.

[0016] According to an embodiment, a long-term output of the one or more long-term layers may be used as an encoded initial state input of the one or more short-term layers.

[0017] According to an embodiment, the context data of the one or more context layers may be used as a context input of the one or more short-term layers.

[0018] According to an embodiment, the system may be configured for performing the real-time context-aware recommendation for food deliveries.

[0019] The present disclosure generally relates to a method for performing real-time context-aware recommendation. The method may include using one or more processor(s) to: determine long-term data of a user; determine short-term data of the user; determine context data for the user; and perform the real-time context-aware recommendation based on the long-term data, the short-term data and the context data.

[0020] According to an embodiment, the long-term data may include previous purchase history of the user.

[0021] According to an embodiment, the short-term data may include current real-time clickstream data of the user.

[0022] According to an embodiment, the context data may include location data and time data of the user.

[0023] According to an embodiment, the one or more processor(s) may be configured to determine the long-term data using a first predetermined weight, determine the short-term data using a second predetermined weight and determine the context data using a third predetermined weight.

[0024] According to an embodiment, the one or more processor(s) may be configured to perform the real-time context-aware recommendation using a recommendation model. The recommendation model may include one or more long-term layers, one or more short-term layers, one or more context layers and one or more ranking layers. The one or more ranking layers may receive inputs from the one or more long-term layers, the one or more short-term layers and the one or more context layers.

[0025] According to an embodiment, a long-term output of the one or more long-term layers may be used as an encoded initial state input of the one or more short-term layers.

[0026] According to an embodiment, the context data of the one or more context layers may be used as a context input of the one or more short-term layers.

[0027] According to an embodiment, the system may be configured for performing the real-time context-aware recommendation for food deliveries.

[0028] The present disclosure generally relates to a non-transitory computer-readable medium storing computer executable code comprising instructions for performing real-time context-aware recommendation according to the present disclosure.

[0029] The present disclosure generally relates to a computer executable code comprising instructions for performing real-time context-aware recommendation according to the present disclosure.

[0030] To the accomplishment of the foregoing and related ends, the one or more embodiments include the features hereinafter fully described and particularly pointed out in the claims. The following description and the associated drawings set forth in detail certain illustrative features of the one or more aspects. These features are indicative, however, of but a few of the various ways in which the principles of various aspects may be employed, and this description is intended to include all such aspects and their equivalents.

BRIEF DESCRIPTION OF THE DRAWINGS

[0031] In the drawings, like reference characters generally refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead generally being placed upon illustrating the principles of the present disclosure. The dimensions of the various features or elements may be arbitrarily expanded or reduced for clarity. In the following description, various aspects of the present disclosure are described with reference to the following drawings, in which:

[0032] FIG. 1 illustrates a system according to various embodiments.

[0033] FIG. 2 shows a flowchart of a method according to various embodiments.

[0034] FIG. 3 illustrates an exemplary illustration of a recommendation model according to various embodiments.

[0035] FIG. 4 illustrates a table with model performances of various known models with the disclosed model according to various embodiments

DETAILED DESCRIPTION

[0036] The following detailed description refers to the accompanying drawings that show, by way of illustration, specific details and embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention. Other embodiments may be utilized and structural, and logical changes may be made without departing from the scope of the invention. The various embodiments are not necessarily mutually exclusive, as some embodiments can be combined with one or more other embodiments to form new embodiments.

[0037] Embodiments described in the context of one of the systems or server or methods or computer program are analogously valid for the other systems or server or methods or computer program and vice-versa.

[0038] Features that are described in the context of an embodiment may correspondingly be applicable to the same or similar features in the other embodiments. Features that are described in the context of an embodiment may correspondingly be applicable to the other embodiments, even if not explicitly described in these other embodiments. Furthermore, additions and/or combinations and/or alternatives as described for a feature in the context of an embodiment may correspondingly be applicable to the same or similar feature in the other embodiments.

[0039] The word “exemplary” is used herein to mean “serving as an example, instance, or illustration.” Any embodiment or design described herein as “exemplary” is not necessarily to be construed as preferred or advantageous over other embodiments or designs

[0040] In the context of various embodiments, the articles “a”, “an”, and “the” as used with regard to a feature or element include a reference to one or more of the features or elements.

[0041] As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0042] The terms “at least one” and “one or more” may be understood to include a numerical quantity greater than or equal to one (e.g., one, two, three, four, […], etc.). The term “a plurality” may be understood to include a numerical quantity greater than or equal to two (e.g., two, three, four, five, […], etc.).

[0043] The words “plural” and “multiple” in the description and the claims expressly refer to a quantity greater than one. Accordingly, any phrases explicitly invoking the aforementioned words (e.g. “a plurality of [objects]”, “multiple [objects]”) referring to a quantity of objects expressly refers more than one of the said objects. The terms “group (of)”, “set [of]”, “collection (of)”, “series (of)”, “sequence (of)”, “grouping (of)”, etc., and the like in the description and in the claims, if any, refer to a quantity equal to or greater than one, i.e. one or more. The terms “proper subset”, “reduced subset”, and “lesser subset” refer to a subset of a set that is not equal to the set, i.e. a subset of a set that contains less elements than the set.

[0044] The term “data” as used herein may be understood to include information in any suitable analog or digital form, e.g., provided as a file, a portion of a file, a set of files, a signal or stream, a portion of a signal or stream, a set of signals or streams, and the like. Further, the term “data” may also be used to mean a reference to information, e.g., in form of a pointer.

The term data, however, is not limited to the aforementioned examples and may take various forms and represent any information as understood in the art.

[0045] The term "processor" or "controller" as, for example, used herein may be understood as any kind of entity that allows handling data, signals, etc. The data, signals, etc. may be handled according to one or more specific functions executed by the processor or controller.

[0046] A processor or a controller may thus be or include an analog circuit, digital circuit, mixed-signal circuit, logic circuit, processor, microprocessor, Central Processing Unit (CPU), Graphics Processing Unit (GPU), Digital Signal Processor (DSP), Field Programmable Gate Array (FPGA), integrated circuit, Application Specific Integrated Circuit (ASIC), etc., or any combination thereof. Any other kind of implementation of the respective functions, which will be described below in further detail, may also be understood as a processor, controller, or logic circuit. It is understood that any two (or more) of the processors, controllers, or logic circuits detailed herein may be realized as a single entity with equivalent functionality or the like, and conversely that any single processor, controller, or logic circuit detailed herein may be realized as two (or more) separate entities with equivalent functionality or the like.

[0047] The term "system" (e.g., a drive system, a position detection system, etc.) detailed herein may be understood as a set of interacting elements, the elements may be, by way of example and not of limitation, one or more mechanical components, one or more electrical components, one or more instructions (e.g., encoded in storage media), one or more controllers, etc.

[0048] A "circuit" as user herein is understood as any kind of logic-implementing entity, which may include special-purpose hardware or a processor executing software. A circuit may thus be an analog circuit, digital circuit, mixed-signal circuit, logic circuit, processor, microprocessor, Central Processing Unit ("CPU"), Graphics Processing Unit ("GPU"),

Digital Signal Processor ("DSP"), Field Programmable Gate Array ("FPGA"), integrated circuit, Application Specific Integrated Circuit ("ASIC"), etc., or any combination thereof. Any other kind of implementation of the respective functions which will be described below in further detail may also be understood as a "circuit." It is understood that any two (or more) of the circuits detailed herein may be realized as a single circuit with substantially equivalent functionality, and conversely that any single circuit detailed herein may be realized as two (or more) separate circuits with substantially equivalent functionality. Additionally, references to a "circuit" may refer to two or more circuits that collectively form a single circuit.

[0049] As used herein, "memory" may be understood as a non-transitory computer-readable medium in which data or information can be stored for retrieval. References to "memory" included herein may thus be understood as referring to volatile or non-volatile memory, including random access memory ("RAM"), read-only memory ("ROM"), flash memory, solid-state storage, magnetic tape, hard disk drive, optical drive, etc., or any combination thereof. Furthermore, it is appreciated that registers, shift registers, processor registers, data buffers, etc., are also embraced herein by the term memory. It is appreciated that a single component referred to as "memory" or "a memory" may be composed of more than one different type of memory, and thus may refer to a collective component including one or more types of memory. It is readily understood that any single memory component may be separated into multiple collectively equivalent memory components, and vice versa. Furthermore, while memory may be depicted as separate from one or more other components (such as in the drawings), it is understood that memory may be integrated within another component, such as on a common integrated chip.

[0050] The following detailed description refers to the accompanying drawings that show, by way of illustration, specific details and aspects in which the present disclosure may

be practiced. These aspects are described in sufficient detail to enable those skilled in the art to practice the present disclosure. Various aspects are provided for the present system, and various aspects are provided for the methods. It will be understood that the basic properties of the system also hold for the methods and vice versa. Other aspects may be utilized and structural, and logical changes may be made without departing from the scope of the present disclosure. The various aspects are not necessarily mutually exclusive, as some aspects can be combined with one or more other aspects to form new aspects.

[0051] To more readily understand and put into practical effect, the present system, method, and other particular aspects will now be described by way of examples and not limitations, and with reference to the figures. For the sake of brevity, duplicate descriptions of features and properties may be omitted.

[0052] It will be understood that any property described herein for a specific system or device may also hold for any system or device described herein. It will also be understood that any property described herein for a specific method may hold for any of the methods described herein. Furthermore, it will be understood that for any device, system, or method described herein, not necessarily all the components or operations described will be enclosed in the device, system, or method, but only some (but not all) components or operations may be enclosed.

[0053] The term “comprising” shall be understood to have a broad meaning similar to the term “including” and will be understood to imply the inclusion of a stated integer or operation or group of integers or operations but not the exclusion of any other integer or operation or group of integers or operations. This definition also applies to variations on the term “comprising” such as “comprise” and “comprises”.

[0054] The term “coupled” (or “connected”) herein may be understood as electrically coupled or as mechanically coupled, e.g., attached or fixed or attached, or just in contact

without any fixation, and it will be understood that both direct coupling or indirect coupling (in other words: coupling without direct contact) may be provided.

[0055] FIG. 1 illustrates a system 100 according to various embodiments.

[0056] According to various embodiments, the system 100 may include a server 110, and/or a user device 120.

[0057] In various embodiments, the server 110 and the user device 120 may be in communication with each other through communication network 130. In an embodiment, even though FIG. 1 shows a line connecting the server 110 to the communication network 130, a line connecting the user device 120 to the communication network 130, the server 110, and the user device 120 may not be physically connected to each other, for example through a cable. In an embodiment, the server 110, and the user device 120 may be able to communicate wirelessly through communication network 130 by internet communication protocols or through a mobile cellular communication network.

[0058] In various embodiments, the server 110 may be a single server as illustrated schematically in FIG. 1, or have the functionality performed by the server 110 distributed across multiple server components. In an embodiment, the server 110 may include one or more server processor(s) 112. In an embodiment, the various functions performed by the server 110 may be carried out across the one or more server processor(s). In an embodiment, each specific function of the various functions performed by the server 110 may be carried out by specific server processor(s) of the one or more server processor(s).

[0059] In an embodiment, the server 110 may include a memory 114. In an embodiment, the server 110 may also include a database. In an embodiment, the memory 114 and the database may be one component or may be separate components. In an embodiment, the memory 114 of the server may include computer executable code defining the functionality that the server 110 carries out under control of the one or more server processor 112. In an

embodiment, the database and/or memory 114 may include historical data such as historical long-term data, e.g., previous purchase history data, and/or location data and/or time data. In an embodiment, the memory 114 may include or may be a computer program product such as a non-transitory computer-readable medium.

[0060] According to various embodiments, a computer program product may store the computer executable code including instructions for performing real-time context-aware recommendation according to the various embodiments. In an embodiment, the computer executable code may be a computer program. In an embodiment, the computer program product may be a non-transitory computer-readable medium. In an embodiment, the computer program product may be in the system 100 and/or the server 110.

[0061] In some embodiments, the server 110 may also include an input and/or output module allowing the server 110 to communicate over the communication network 130. In an embodiment, the server 110 may also include a user interface for user control of the server 110. In an embodiment, the user interface may include, for example, computing peripheral devices such as display monitors, user input devices, for example, touchscreen devices and computer keyboards.

[0062] In an embodiment, the user device 120 may include a user device memory 122. In an embodiment, the user device 120 may include a user device processor 124. In an embodiment, the user device memory 122 may include computer executable code defining the functionality the user device 120 carries out under control of the user device processor 124. In an embodiment, the user device memory 122 may include or may be a computer program product such as a non-transitory computer-readable medium.

[0063] In an embodiment, the user device 120 may also include an input and/or output module allowing the user device 120 to communicate over the communication network 130. In an embodiment, the user device 120 may also include a user interface for the user to control

the user device 120. In an embodiment, the user interface may be a touch panel display. In an embodiment, the user interface may include a display monitor, a keyboard or buttons.

[0064] In an embodiment, the system 100 may be used for performing real-time context-aware recommendation. In an embodiment, the memory 114 may have instructions stored therein. In an embodiment, the processor 112 may be configured to determine long-term data of a user. The long-term data may include previous purchase history of the user. The long term data may also include location data and/or time data. The location data and/or time data may be associated with the previous purchase history. For example, there may be historical data of the user purchasing from merchant A at location 1 during lunch time during a weekday. There may also be historical data of the user purchasing from merchant B at location 2 during dinner time during a weekend. The long term data may be in the memory or database of system 100.

[0065] Adventageously, the system 100 may be able to perform real-time context-aware recommendation more accurately and the recommendations may be more personalised for the user.

[0066] In an embodiment, the processor 112 may be configured to determine short-term data of a user. The short-term data may include current real-time clickstream data of the user. The current real-time clickstream data may include a type of merchants the user is looking at (i.e., clicking) and/or items which the user added to a virtual basket of an e-commerce platform. For example, there may be current real-time clickstream data of the user clicking to look at merchant A. The system 100 may determine that merchant A is a fast-food merchant, and may recommend other fast-food merchants to the user. In an embodiment, the system 100 may determine that the user has clicked to add an item from merchant A into the virtual basket of the e-commerce platform, and the system 100 may recommend other merchants which are in the same location or a vicinity of merchant A to add to the order. This may be known as

batch orders, where a single delivery driver may deliver one or more items from one or more merchants to a single user.

[0067] Adventageously, there may be higher user satisfication at being given more alternates to choose from or being given more merchants to purchase from in a single order. There may also be higher driver satisfaction if an order is being batched.

[0068] In an embodiment, the processor 112 may be configured to determine context data of the user. The context data data may include location data and/or time data of previous purchases. The location data and/or time data may be associated with the previous purchase history. For example, if the user opens the ecommerce platform during time A at location B, the system may determine that based on historical data, the users may want to order from merchant C and may recommend merchant C to the user.

[0069] Adventageously, the system 100 may be able to perform real-time context-aware recommendation more accurately and the recommendations may be more personalised for the user.

[0070] In an embodiment, the processor 112 may be configured to determine the long-term data using a first predetermined weight. The processor 112 may be configured to determine the short-term data using a second predetermined weight. The processor 112 may be configured to determine the context data using a third predetermined weight. The processor 112 may be configured to determine the first predetermined weight, the second predetermined weight and the third predetermined weight based on an importance of each of the long-term data, the short-term data and the context data.

[0071] In an embodiment, the processor 112 may be configured to perform the real-time context-aware recommendation using a recommendation model. The recommendation model may include one or more long-term layers, one or more short-term layers, one or more context layers and one or more ranking layers. The one or more ranking layers may receive inputs

from the one or more long-term layers, the one or more short-term layers and the one or more context layers. The recommendation model may use neural networks such as a recurrent neural network (RNN) which is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. Any suitable neural networks may be used.

[0072] In an embodiment, a long-term output of the one or more long-term layers may be used as an encoded initial state input of the one or more short-term layers.

[0073] In an embodiment, the context data of the one or more context layers may be used as a context input of the one or more short-term layers.

[0074] In an embodiment, the system 100 may be configured for performing the real-time context-aware recommendation for food deliveries.

[0075] In an embodiment, the one or more context layers may include at least one context-aware feature encoding layer. The at least one context-aware feature encoding layer may transform user and/or items location information into encoded user items distances and/or date time slots.

[0076] In an embodiment, the one or more long-term layers and/or the one or more short-term layers may include at least one attention encoder to model users' multi-interest of both the long term and short term.

[0077] In an embodiment, a real-time context-aware recommendation system based on users' multi-interest with attention may be disclosed. This may also be known as a context-aware attention recommendation neural network (CANN) system. The CANN system may include at least one of: an attention encoder, context-aware feature encoding, RNN layers, and ranking layers to model users' real-time context features, and multi-interests of both long-term and short-term.

[0078] In an embodiment, the CANN system may include a context-aware encoding layer to transform real-time context features, such as user items distances, date time slots information into encoded features.

[0079] In an embodiment, the CANN system may model the users' multi-interests in both short-term data and long-term data with an attention encoder to be applied to the long term data. The long term data may be combined with users' short term data as inputs.

[0080] In an embodiment, the long-term data and/or the short-term data may be fed into a pre-trained embedding (e.g., trained with item2vec) to improve model generalization ability.

[0081] In an embodiment, the CANN system may integrates with linear and/or DNN ranking layers to rerank the recall results based on the RNN score, and the context, user and item features.

[0082] In an embodiment, the CANN system may be an end-to-end model that may be able to be trained. The CANN system may serve as one self-contained service and may easily integrated with any serving system.

[0083] In an embodiment, the CANN system may be designed to provide personalized context-aware real-time recommendation for food delivery services on an e-commerce platform e.g., on a GrabFood platform.

[0084] In an embodiment, the CANN system may take into consideration both the users' short-term data and also long-term data by an attention encoder and a RNN (e.g., GRU cells) and may based on the real-time context, to make the best recommendations. GRU cells may be chosen for better performance in recommendation use case.

[0085] In an embodiment, instead of taking the users' click and purchases with equal importance, the attention layer may be designed to learn the multi-interests of the users' long term data (e.g., purchases), by giving different weights to the purchase histories and the

current session clicks. In an embodiment, the attention output may be combined with the encoded initial state inputs as the input to the RNN layers.

[0086] In an embodiment, recall items, with the item features, may be reranked with a combination of a linear layer and deep layers. This makes it possible to leverage the item features in the model to further improve the model performance. In an embodiment, by combining the recall with the rerank layer, the computer system may be improved. The computer system may offer faster processing and less memory because it may be an end-to-end model offering recall and rerank in single service.

[0087] FIG. 2 shows a flowchart of a method 200 according to various embodiments.

[0088] According to various embodiments, the method 200 for performing real-time context-aware recommendation may be provided. In some embodiments, the method 200 may include a step 202 of using one or more processor(s) of a system (e.g., the system 100) to determine long-term data of a user. The long-term data may include previous purchase history of the user.

[0089] In an embodiment, the method 200 may include a step 204 of using the one or more processor(s) to determine short-term data of the user. The short-term data may include current real-time clickstream data of the user.

[0090] In an embodiment, the method 200 may include a step 206 of using the one or more processor(s) to determine context data for the user. The context data may include location data and time data of the user

[0091] In an embodiment, the method 200 may include a step 208 of using the one or more processor(s) to perform the real-time context-aware recommendation based on the long-term data, the short-term data and the context data.

[0092] Steps 202 to 208 are shown in a specific order, however other arrangements are possible, for example, in some embodiments, step 202 may be carried out after step 204. Steps may also be combined in some cases. Any suitable order of steps 202 to 208 may be used.

[0093] FIG. 3 illustrates an exemplary illustration of a recommendation model 300 according to various embodiments.

[0094] In an embodiment, the exemplary recommendation model 300 may include one or more long-term layers 302, one or more short-term layers 304, one or more context layers 306, one or more ranking layers 308 and one or more prediction layer 310.

[0095] In an embodiment, the one or more long-term layers 302 may take in historical purchases as inputs 312 to an embedding layer 314. Outputs of the embedding layer 314 may be used as inputs of an attention encoder 316 (or attention layers). The attention encoder 316 may be designed to learn the multi-interests of the users' long term preferences (purchases), by giving different weights to the purchase histories and the current session clicks. The encoded vector 318 may be passed to the short-term layers 304.

[0096] In an embodiment, attention may be a mechanism that provides a richer encoding of the source sequence from which to construct a context vector that can then be used by the decoder. Attention may allow the model to learn what encoded items in the source sequence to pay attention to and to what degree during the prediction of the next item.

[0097] In an embodiment, the one or more short-term layers 304 may take in current clickstream data as inputs 320 to an embedding layer 322. Outputs of the embedding layer 322 may be used as inputs of RNN layers 324 (e.g., GRU cells). The encoded vector 318 from the one or more long-term layers 302 may be also be used as inputs to the passed to the RNN layers 324.

[0098] In an embodiment, the exemplary model 300 may be modeled as a long-term and a short term (data) sequence problem with context features, user features and item features, to predict the next item based on previous historical purchases and the current session clicks.

[0099] In an embodiment, a personalized next item recommendation may be defined as follows: since user interaction naturally forms a sequence over time, the user-item interaction in the recommendation system is a set of sequence interactions. Given target user u and its interaction sequence, long term stable interactions (like purchase histories) $\{(z_1, c_1), \dots, (z_l, c_l)\}$, and short term real-time interactions (like clicks and add baskets):$\{(x_1, b_1), \dots, (x_{t-1}, b_{t-1})\}$, x_{t-1} represents the t-1 th item that user u interacts and b indicates the type of item interaction (for example, click, and purchase).

[00100] In an embodiment, the personalized recommendation task may predict the next item y_t that user u is most likely to interact.

[00101] In an embodiment, the personalized recommendation task may be modeled with the following equation:

$$y_t \sim p(Y_t|x_{1:t-1}, z_{1:l}, a_t) = softmax(\varphi(h(x_1, \dots x_{t-1}, z_1, \dots z_l, a_t))).$$

[00102] That is, the model 300 may be trained to learn to predict the next item y_t based on the current interactions with short term inputs $x_{1:t-1}$ such as session clicks, and long term inputs $z_{1:l}$ such as purchase histories, and along with the real-time context, user and item features a_t.

[00103] In an embodiment, the users historical purchase may be modelled as the source sequence. The attention may be applied with the users historical purchase to obtain the encoded information as a context vector 318. The context vector 318 may be used as one of the RNN input.

[00104] In an embodiment, the context layers 306 may take in context features 330 and/or user features 332 and item features 334.

[00105] In an embodiment, the context features include the real-time location and date time slots information. The context features may be encoded into the user-item distances 336 and/or date time slot features. In an embodiment, the user features may include the user features that are being feature-transformed and provided as RNN input. These features may be concatenated along with the embeddings and then to be fed into a RNN of a current timestamp.

[00106] In an embodiment, the item features may be static features that may be integrated into the model 300. This alleviates the need of an external feature store and thus making the model 300 more flexible to be integrated into the serving infrastructure. Dynamic features may still be supported by being passed as an input to the model serving.

[00107] In an embodiment, the item embeddings may be trained with the Item2Vec with skip-gram and negative sampling. The pretrained embeddings are then being loaded by the embedding layers. The pretrained embedding can improve the generalization for the models and also can improve the training speed.

[00108] In an embodiment, the RNN 324 (from short-term layer 304) with attention results (from long-term layer 302) may be fed into the ranking layer 308 to rerank the recall results with the RNN score and the encoded user-item features 336 and the item features 334.

[00109] In an embodiment, the ranking layer 308 may include a linear layer 340 and deep layers 342, 344, 346 (DNN). The ranking layer 308 may be designed to learn both the linear representation and also the high order non-linear features. The ranking layer 308 may further improve the model performance by considering the encoded user-item features 336, item features 334 along with the RNN score from RNN 324.

[00110] FIG. 4 illustrates a table 400 with model performances of various known models with the disclosed model according to various embodiments

[00111] In an embodiment, an industrial dataset is constructed by click and order logs from an online restaurant recommendation system. For the training set, the click sequence of each user in the last 30 days is determined, and the next item will be used as the prediction target. For example, for a sequence [A, B, C, D], [A, B] may be used as inputs, then [C] will be the target. For the testing set, the click and order sequence of each user on the last day is used.

[00112] In an embodiment, the models performances of known models such as SIMS, and HRNN-Meta is compared with disclosed CANN model and CANN with trainable embedding.

[00113] In an embodiment, the SIMS model may be a model built with embedding of Item2Vec with skip-gram and negative sampling, and cosmul algorithms to predict the next items based on item to item similarities. In an embodiment, the HRNN-Meta may be similar to the disclosed CANN model without attention and ranking layers.

[00114] In an embodiment, evaluation metrics used may be recall@k and mrr@k which may be defined in the following equations：

$$recall@k = \frac{number\ of\ relevant\ items\ in\ top\ k}{number\ of\ relevant\ items}$$

$mrr@k = \frac{1}{|R|}\sum_{i \in R}\frac{1}{rank_i}$, R is the set which includes all relevant items in top k. $rank_i$is the rank of the relevant item.

[00115] In an embodiment, the table 400 shows the names 402, where *_click or *_order means evaluation with click or order as the prediction target. In an embodiment, the table 400 also shows results for a known model SIMS 404, a known model HRNN-Meta 406, disclosed CANN model 408 and disclosed CANN model with trainable embeddings 410. Based on the results, the CANN models 408, 410 have a clear advantage over the traditional known models 404, 406.

[00116] In an embodiment, the CANN models 408, 410 may significantly improve the top-k recall for the recommendation in comparison to the traditional known models 404, 406.

[00117] While the present disclosure has been particularly shown and described with reference to specific aspects, it should be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the scope of the present disclosure as defined by the appended claims. The scope of the present disclosure is thus indicated by the appended claims and all changes which come within the meaning and range of equivalency of the claims are therefore intended to be embraced.

[00118] Various examples are provided.

[00119] Example 1 is a system for performing real-time context-aware recommendation, the system comprising: one or more processors; and a memory having instructions stored therein, the instructions, when executed by the one or more processors, cause the one or more processors to: determine long-term data of a user; determine short-term data of the user; determine context data for the user; and perform the real-time context-aware recommendation based on the long-term data, the short-term data and the context data.

[00120] In Example 2, the system of Example 1, the long-term data comprises previous purchase history of the user.

[00121] In Example 3, the system of Example 1 or Example 2, the short-term data comprises current real-time clickstream data of the user.

[00122] In Example 4, the system of any one of Examples 1-3, the context data comprises location data and time data of the user.

[00123] In Example 5, the system of any one of Examples 1-4, the one or more processors are configured to determine the long-term data using a first predetermined weight, determine

the short-term data using a second predetermined weight and determine the context data using a third predetermined weight.

[00124] In Example 6, the system of any one of Examples 1-5, the one or more processors are configured to perform the real-time context-aware recommendation using a recommendation model; the recommendation model comprises one or more long-term layers, one or more short-term layers, one or more context layers and one or more ranking layers; and the one or more ranking layers receive inputs from the one or more long-term layers, the one or more short-term layers and the one or more context layers.

[00125] In Example 7, the system of Example 6, a long-term output of the one or more long-term layers is used as an encoded initial state input of the one or more short-term layers.

[00126] In Example 8, the system of Example 6 or Example 7, the context data of the one or more context layers is used as a context input of the one or more short-term layers.

[00127] In Example 9, the system of any one of Examples 1-8, the system is configured for performing the real-time context-aware recommendation for food deliveries.

[00128] Example 10 is a method for performing real-time context-aware recommendation, the method comprising using one or more processors to: determine long-term data of a user; determine short-term data of the user; determine context data for the user; perform the real-time context-aware recommendation based on the long-term data, the short-term data and the context data.

[00129] In Example 11, the method of Example 10, the long-term data comprises previous purchase history of the user.

[00130] In Example 12, the method of Example 10 or Example 11, the short-term data comprises current real-time clickstream data of the user.

[00131] In Example 13, the method of any one of Examples 10-12, the context data comprises location data and time data of the user.

[00132] In Example 14, the method of any one of Examples 10-13, the method comprising: using the one or more processors to determine the long-term data using a first predetermined weight, determine the short-term data using a second predetermined weight and determine the context data using a third predetermined weight.

[00133] In Example 15, the method of any one of Examples10-14, the method comprising: using the one or more processors to perform the real-time context-aware recommendation using a recommendation model; wherein the recommendation model comprises one or more long-term layers, one or more short-term layers, one or more context layers and one or more ranking layers; and wherein the one or more ranking layers receive inputs from the one or more long-term layers, the one or more short-term layers and the one or more context layers.

[00134] In Example 16, the method of Example 15, a long-term output of the one or more long-term layers is used as an encoded initial state of the one or more short-term layers.

[00135] In Example 17, the method of Example 15 or Example 16, the context data of the one or more context layers is used as a context input of the one or more short-term layers.

[00136] In Example 18, the method of any one of Examples 10-17, wherein the method is used for performing the real-time context-aware recommendation for food deliveries.

[00137] Example 19 is a non-transitory computer-readable medium storing computer executable code comprising instructions for performing real-time context-aware recommendation according to any one of Examples 10 to 18.

[00138] Example 20 is a computer executable code comprising instructions for performing real-time context-aware recommendation according to any one of Examples 10 to 18.

CLAIMS

What is claimed is:

1. A system for performing real-time context-aware recommendation, the system comprising:

one or more processors; and

a memory having instructions stored therein, the instructions, when executed by the one or more processors, cause the one or more processors to:

determine long-term data of a user using a first predetermined weight;

determine short-term data of the user using a second predetermined weight;

determine context data for the user using a third predetermined weight; and

perform the real-time context-aware recommendation based on the long-term data, the short-term data and the context data,

wherein the one or more processors are configured to determine the first predetermined weight, the second predetermined weight, and the third predetermined weight based on an importance of each of the long-term data, the short-term data, and the context data.

2. The system of claim 1, wherein the long-term data comprises previous purchase history of the user.

3. The system of claim 1 or claim 2, wherein the short-term data comprises current real-time clickstream data of the user.

4. The system of any one of claims 1-3, wherein the context data comprises location data and time data of the user.

5. The system of any one of claims 1-4,

wherein the one or more processor(s) is configured to perform the real-time context-aware recommendation using a recommendation model;

wherein the recommendation model comprises one or more long-term layers, one or more short-term layers, one or more context layers and one or more ranking layers; and

wherein the one or more ranking layers receive inputs from the one or more long-term layers, the one or more short-term layers and the one or more context layers.

6. The system of claim 5, wherein a long-term output of the one or more long-term layers is used as an encoded initial state input of the one or more short-term layers.

7. The system of claim 5 or claim 6, wherein the context data of the one or more context layers is used as a context input of the one or more short-term layers.

8. The system of any one of claims 1-7, wherein the system is configured for performing the real-time context-aware recommendation for food deliveries.

9. A method for performing real-time context-aware recommendation, the method comprising using one or more processor(s) to:

determine long-term data of a user using a first predetermined weight;

determine short-term data of the user using a second predetermined weight;

determine context data for the user using a third predetermined weight; and

perform the real-time context-aware recommendation based on the long-term data, the short-term data and the context data,

wherein the method further comprises determining the first predetermined weight, the second predetermined weight, and the third predetermined weight based on an importance of each of the long-term data, the short-term data, and the context data.

10. The method of claim 9, wherein the long-term data comprises previous purchase history of the user.

11. The method of claim 9 or claim 10, wherein the short-term data comprises current real-time clickstream data of the user.

12. The method of any one of claims 9-11, wherein the context data comprises location data and time data of the user.

13. The method of any of claims 9-12, the method comprising:

using the one or more processors to perform the real-time context-aware recommendation using a recommendation model;

wherein the recommendation model comprises one or more long-term layers, one or more short-term layers, one or more context layers and one or more ranking layers; and

wherein the one or more ranking layers receive inputs from the one or more long-term layers, the one or more short-term layers and the one or more context layers.

14. The method of claim 13, wherein a long-term output of the one or more long-term layers is used as an encoded initial state of the one or more short-term layers.

15. The method of claim 13 or claim 14, wherein the context data of the one or more context layers is used as a context input of the one or more short-term layers.

16. The method of any one of claims 9-15, wherein the method is used for performing the real-time context-aware recommendation for food deliveries.

17. A non-transitory computer-readable medium storing computer executable code comprising instructions for performing real-time context-aware recommendation according to any one of claims 9 to 16.

18. A computer executable code comprising instructions for performing real-time context-aware recommendation according to any one of claims 9 to 16.