

SYSTEM AND METHOD FOR PREDICTING DESTINATION LOCATION

TECHNICAL FIELD

[0001] Various aspects of this disclosure relate to a system for predicting a destination location. Various aspects of this disclosure relate to a method for predicting a destination location. Various aspects of this disclosure relate to a non-transitory computer-readable medium storing computer executable code comprising instructions for predicting a destination location. Various aspects of this disclosure relate to a computer executable code comprising instructions for predicting a destination location.

BACKGROUND

[0002] Predicting the destination of a trip is a task in human mobility which finds several applications in real-world scenarios, from optimizing the efficiency of electronic dispatching systems to predicting and reducing traffic jams. In particular, it is of interest in the context of e-hailing, which, thanks to the advance of smartphone technology, has become popular globally and enables customers to hail taxis using their smartphones.

[0003] Next destination recommendations are important in the transportation domain of taxi and ride-hailing services, where users are recommended with personalized destinations given their current origin location.

[0004] For predicting a user's next destination, models such as frequency, sequential learning, matrix factorization may be used to predict the user's next destination based on a user's visiting sequence. Existing works for next destination recommendation task are not designed to learn from both origin and destination sequences. They are designed to only learn from destination sequences. Therefore, the existing solutions may not be accurate and may provide less contextualized and impractical recommendations (e.g. very far away destinations).

[0005] More importantly, existing works does not consider the current inputted origin location by the user to perform the recommendation, this makes it highly impractical in real world applications as very far destinations may be recommended even when the user tends to take a ride to nearby places, leading to sub-optimal performance.

SUMMARY

[0006] Therefore, it is desirable to increase the accuracy of predicting a destination location and to achieve an accurate and reliable prediction of a vehicle's (or, equivalently, a user's) destination.

[0007] An advantage of the present disclosure may include an accurate and reliable prediction of a destination location by learning the sequential transitions from both origin and destination sequences.

[0008] An advantage of the present disclosure may include a reliable prediction system which may be origin-aware, meaning that it performs prediction based on knowledge of where the user is currently at or their inputted origin location.

[0009] An advantage of the present disclosure may include personalization of prediction of a destination location which may be achieved through user embedding to learn and compute a hidden representation that best represents the user's current preference to best predict the next destination. This may be interpreted as a system being able to understand user preferences.

[0010] 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.

[0011] The present disclosure generally relates to a system for predicting a destination location. The system may include one or more processors. The system may also include a memory having instructions stored there, in the instructions, when executed by the one or more processors,

may cause the one or more processors to use at least one recurrent neural network to: process spatial data which may include a first set of information about origin locations and destination locations; process temporal data which may include a second set of information about times at the origin locations and the destination locations; determine hidden state data based on the spatial data and the temporal data, wherein the hidden state data may include data on origin-destination relationships; receive a current input data from a user, wherein the current input data may include an identity of the user and the current origin location of the user; and predict the destination location based on the hidden state data and the current input data.

[0012] According to an embodiment, the current input data further may include a previous destination of the user.

[0013] According to an embodiment, the first set of information may include local origin locations and local destination locations. The second set of information may include times at the local origin locations and the local destination locations. The local origin locations and the local destination locations may be within a geohash.

[0014] According to an embodiment, the first set of information may include global origin locations and/or global destination locations. The second set of information may include times at the global origin locations and/or the global destination locations. The global origin locations and/or the global destination locations may be outside of the geohash.

[0015] According to an embodiment, the system may include an encoder. The encoder may be configured to process the spatial data and the temporal data. The encoder may be configured to determine the hidden state data.

[0016] According to an embodiment, the system may include a decoder. The decoder may be configured to receive the current input data from the user. The decoder may be configured to receive the hidden state data from the encoder. The decoder may be configured to predict the destination location based on the hidden state data and the current input data.

[0017] According to an embodiment, the decoder may be configured to determine personalized preference data of the user based on the hidden state data, the current input data and a first predetermined weight.

[0018] According to an embodiment, the decoder may be configured to predict the destination location based on the personalized preference data and the hidden state data with a second predetermined weight.

[0019] According to an embodiment, the decoder may be configured to determine a probability that the destination location predicted is a correct destination location.

[0020] The present disclosure generally relates to a method for predicting a destination location. The method may include: using at least one recurrent neural network to: process spatial data comprising information about origin locations and destination locations; process temporal data comprising information about times at the origin locations and the destination locations; determine origin-destination relationships based on the spatial data and the temporal data; receive a current input data from a user, wherein the current input data comprises an identity of the user and the current origin location of the user; and predict the destination location based on the origin-destination relationships and the current input data.

[0021] According to an embodiment, the current input data further may include a previous destination of the user.

[0022] According to an embodiment, the first set of information may include local origin locations and local destination locations. The second set of information may include times at the local origin locations and the local destination locations. The local origin locations and the local destination locations may be within a geohash.

[0023] According to an embodiment, the first set of information may include global origin locations and/or global destination locations. The second set of information may include times at the global origin locations and/or the global destination locations. The global origin locations and/or the global destination locations may be outside of the geohash.

[0024] According to an embodiment, the method may include using an encoder to: process the spatial data and the temporal data; and determine the hidden state data.

[0025] According to an embodiment, the method may include using a decoder to: receive the current input data from the user; receive the hidden state data from the encoder; and predict the destination location based on the hidden state data and the current input data.

[0026] According to an embodiment, the method may include determining personalized preference data of the user based on the hidden state data, the current input data and a first predetermined weight using the decoder.

[0027] According to an embodiment, the method may include predicting the destination location based on the personalized preference data and the hidden state data with a second predetermined weight using the decoder.

[0028] According to an embodiment, the method may include determining a probability that the destination location predicted is a correct destination location using the decoder.

[0029] The present disclosure generally relates to a non-transitory computer-readable medium storing computer executable code comprising instructions for predicting a destination location according to the present disclosure.

[0030] The present disclosure generally relates to a computer executable code comprising instructions for predicting a destination location according to the present disclosure.

[0031] 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

[0032] 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:

[0033] FIG. 1 illustrates a schematic diagram of a system 100 according to an embodiment of the present disclosure.

[0034] FIG. 2 shows a flowchart of a method 200 according to various embodiments.

[0035] FIG. 3 illustrates an exemplary destination recommendation interface according to various embodiments.

[0036] FIG. 4 illustrates exemplary relationships between origins and destinations according to various embodiments.

[0037] FIG. 5A illustrates exemplary relationships between origins and destinations in a local view according to various embodiments.

[0038] FIG. 5B illustrates exemplary relationships between origins and destinations in a global view according to various embodiments.

[0039] FIG. 6 illustrates a schematic diagram of a recurrent neural network according to an embodiment of the present disclosure.

[0040] FIG. 7 illustrates a schematic diagram of an encoder and a decoder system according to an embodiment of the present disclosure.

[0041] FIG. 8A illustrates exemplary statistics of datasets according to various embodiments.

[0042] FIG. 8B illustrates exemplary performance of the datasets SE-1 to SE-4 of FIG. 8A according to various embodiments.

[0043] FIG. 8C illustrates exemplary performance of the datasets SE-5 to SE-7 of FIG. 8A according to various embodiments.

[0044] FIG. 9 illustrates a schematic diagram of an encoder and a decoder system according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0045] 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.

[0046] 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.

[0047] 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.

[0048] 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.

[0049] 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.

[0050] As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0051] 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.).

[0052] 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.

[0053] 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.

[0054] 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.

[0055] 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.

[0056] 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.

[0057] A “circuit” as used 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.

[0058] 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.

[0059] As used herein, the term “geohash” may be predefined geocoded cells of partitioned areas of a city or country.

[0060] 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.

[0061] 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.

[0062] 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.

[0063] 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”.

[0064] 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.

[0065] The term “entity” herein may be understood as a human user, a business, a group of users or an organization.

[0066] FIG. 1 illustrates a schematic diagram of a system 100 according to an embodiment of the present disclosure.

[0067] According to various embodiments, the system 100 may include a server 110, and/or a user device 120.

[0068] 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.

[0069] 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).

[0070] 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 of past transportation service, e.g., a origin location and/or destination location, and/or time at origin location, and/or time at destination location and/or user profile, e.g., user identity and/or user preference. In an embodiment, the memory 114 may include or may be a computer program product such as a non-transitory computer-readable medium.

[0071] According to various embodiments, a computer program product may store the computer executable code including instructions for predicting a destination location 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.

[0072] 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.

[0073] 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.

[0074] 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.

[0075] In an embodiment, the system 100 may be used for predicting a destination location. In an embodiment, the memory 114 may have instructions stored therein. In an embodiment, the instructions, when executed by the one or more processors may cause the processor 112 to use at least one recurrent neural network to predict a destination location.

[0076] In an embodiment, the processor 112 may use at least one recurrent neural network to process spatial data which may include a first set of information about origin locations and destination locations.

[0077] In an embodiment, the processor 112 may use at least one recurrent neural network to process temporal data which may include a second set of information about times at the origin locations and the destination locations.

[0078] In an embodiment, the processor 112 may use at least one recurrent neural network to determine hidden state data based on the spatial data and the temporal data. In an embodiment, the hidden state data may include data on origin-destination relationships.

[0079] In an embodiment, the processor 112 may use at least one recurrent neural network to receive a current input data from a user. In an embodiment, the current input data may include an identity of the user and the current origin location of the user.

[0080] In an embodiment, the processor 112 may use at least one recurrent neural network to predict the destination location based on the hidden state data and the current input data.

[0081] In an embodiment, the current input data received from the user may include a previous destination of the user. In an embodiment, the previous destination of the user may be used to predict the destination location.

[0082] In an embodiment, the first set of information may include local origin locations and/or local destination locations. In an embodiment, the second set of information may include times at the local origin locations and/or the local destination locations. In an embodiment, the local origin locations and/or the local destination locations may be within a geohash. The term “geohash” may be predefined geocoded cells of partitioned areas of a city or country.

[0083] In an embodiment, the first set of information may include global origin locations and/or global destination locations. In an embodiment, the second set of information may include times at the global origin locations and/or the global destination locations. In an embodiment, the global origin locations and/or the global destination locations may be outside of the geohash.

[0084] In an embodiment, the system 100 may include an encoder. In an embodiment, the encoder may be configured to process the spatial data and the temporal data. In an embodiment, the encoder may be configured to determine the hidden state data.

[0085] In an embodiment, the system 100 may include a decoder. In an embodiment, the decoder may be configured to receive the current input data from the user. In an embodiment, the decoder may be configured to receive the hidden state data from the encoder. In an embodiment, the decoder may be configured to predict the destination location based on the hidden state data and the current input data.

[0086] In an embodiment, the decoder may be configured to determine personalized preference data of the user based on the hidden state data, the current input data and a first predetermined weight.

[0087] In an embodiment, the decoder may be configured to predict the destination location based on the personalized preference data and the hidden state data with a second predetermined weight.

[0088] In an embodiment, the decoder may be configured to determine a probability that the destination location predicted is a correct destination location.

[0089] FIG. 2 shows a flowchart of a method 200 according to various embodiments.

[0090] According to various embodiments, the method 200 for predicting a destination location may be provided. In an embodiment, the method 200 may include a step 202 of using at least one recurrent neural network to process spatial data. The spatial data may include information about origin locations and destination locations.

[0091] In an embodiment, the method 200 may include a step 204 of using at least one recurrent neural network to process temporal data. The temporal data may include information about times at the origin locations and the destination locations.

[0092] In an embodiment, the method 200 may include a step 206 of using at least one recurrent neural network to determine origin-destination relationships based on the spatial data and the temporal data.

[0093] In an embodiment, the method 200 may include a step 208 of using at least one recurrent neural network to receive a current input data from a user. The current input data may include an identity of the user and the current origin location of the user.

[0094] In an embodiment, the method 200 may include a step 210 of using at least one recurrent neural network to predict the destination location based on the origin-destination relationships and the current input data.

[0095] In an embodiment, steps 202 to 210 are shown in a specific order, however other arrangements are possible. Steps may also be combined in some cases. Any suitable order of steps 202 to 210 may be used.

[0096] FIG. 3 illustrates an exemplary destination recommendation interface according to various embodiments.

[0097] In an embodiment, the exemplary destination recommendation interface 300 may include a current origin location 302. In an embodiment, there may include at least one recommended destination for the current origin location 302. In an embodiment, the at least one recommended destination may be predicted by a system for predicting a destination location. In an embodiment, the system may predict the top X destination locations that a user may travel to from the current origin location 302, where X may be a predetermined value. In an embodiment, the system may predict the top X destination locations based on a user identity. That is, the top X destinations locations from the current origin location 302 may differ from different users.

[0098] In an embodiment, at least one recommended destination may include a first recommended destination 304A. In an embodiment, at least one recommended destination may include a second recommended destination 304B. In an embodiment, at least one recommended destination may include a third recommended destination 304C. In an embodiment, the first recommended destination 304A may be a destination which the system predicts to be the most likely destination. In an embodiment, the third recommended destination 304C may be a

destination which the system predicts to be the most unlikely destination out of all the recommended locations.

[0099] In an embodiment, the following problem formulation may be used:

[00100] Let $U = \{u_1, u_2, \dots, u_M\}$ be the set of M users and $L = \{l_1, l_2, \dots, l_N\}$ be the set of N locations for the users in U to visit, where each location $l_n \in L$ may either have the role of an origin o or a destination d , or both among the dataset of user trajectories. Each user u_m may have an OD sequence of pick-up (origin) and drop-off (destination) tuples $s_{u_m} = \{(o_{t_1}, d_{t_1}), (o_{t_2}, d_{t_2}), \dots, (o_{t_i}, d_{t_i})\}$ that may record their taxi trips, and $S = \{s_{u_1}, s_{u_2}, \dots, s_{u_M}\}$ may be the set of all users' OD sequences. All location visits in S , regardless of origin or destination may have their own location coordinates LC and timestamp *time*.

[00101] In an embodiment, the objective of the origin-aware next destination recommendation task may be to consider at least one of: the user u_m , the current origin o_{t_i} , and the historical sequence of OD tuples $\{(o_{t_1}, d_{t_1}), (o_{t_2}, d_{t_2}), \dots, (o_{t_{i-1}}, d_{t_{i-1}})\}$ to recommend an ordered set of destinations from L . In an embodiment, the next destination d_{t_i} may be highly ranked in the recommendation set. In an embodiment, $s_{u_m}^{train}$ and S^{train} may be sets from the training partition with the superscript *train* for clarity.

[00102] FIG. 4 illustrates exemplary relationships between origins and destinations according to various embodiments.

[00103] In a map 400 of FIG. 4, at least one origin location may be in the map 400. The at least one origin location may include a first origin location 402A, a second origin location 402B, and a third origin location 402C.

[00104] In the map 400 of FIG. 4, at least one destination location may be in the map 400. The at least one destination location may include a first destination location 402A, a second destination location 402B, and a third destination location 402C.

[00105] In an embodiment, the at least one destination location may be used to learn and/or predict the next destination by learning destination-destination (DD) relationships.

[00106] In an embodiment, the inclusion of origin location such as an origin sequence or information may help to learn origin-origin (OO) and/or origin-destination (OD) relationships.

[00107] Origin-origin (OO) and/or origin-destination (OD) relationships and/or how to include origin information is not studied by existing works and methods. An advantage of learning origin-origin (OO) and/or origin-destination (OD) relationships may result in a model which may optimally learn from both sequences can best perform the task.

[00108] FIG. 5A illustrates exemplary relationships between origins and destinations in a local view according to various embodiments.

[00109] In a map of FIG. 5A, at least one location may be in the map 500. The at least one destination location may include a first location 502A, a second location 502B, a third location 502C, a fourth location 502D, a fifth location 502E. In an embodiment, for the at least one location, spatial and/or temporal factor may be obtained for each location of the at least one location. In an embodiment, for spatial factors, the relationships between origins and destinations may be obtained. In an embodiment, for temporal factors, time slot embeddings of each location may be obtained. Temporal intervals may also be obtained.

[00110] In the map of FIG. 5A, the spatial and the temporal factors may be in a local view. In an embodiment, local views may be considered as locations within a geohash.

[00111] FIG. 5B illustrates exemplary relationships between origins and destinations in a global view according to various embodiments.

[00112] In a map of FIG. 5B, at least one location may be in the map 510. The at least one destination location may include a first location 512A, a second location 512B, a third location 512C, a fourth location 512D, a fifth location 512E. In an embodiment, for the at least one location, spatial and/or temporal factor may be obtained for each location of the at least one location. In an embodiment, for spatial factors, the relationships between origins and destinations may be

obtained. In an embodiment, for temporal factors, time slot embeddings of each location may be obtained. Temporal intervals may also be obtained.

[00113] In the map of FIG. 5B, the spatial and the temporal factors may be in a global view. In an embodiment, global views may be considered as locations not within a geohash. In an embodiment, the spatial and/or the temporal factors may be calculated by computing pairwise intervals from a location to all other locations to understand “how far” and “how near” they are.

[00114] FIG. 6 illustrates a schematic diagram of a recurrent neural network according to an embodiment of the present disclosure.

[00115] In an embodiment, the recurrent neural network 600 may be a Spatial-Temporal LSTM (ST-LSTM) model. In an embodiment, an Adam optimizer with cross entropy loss for multi-classification problem may be used to train the model. In an embodiment, the Adam optimizer may have a batch size of 1. The adam optimizer may use 15 epochs and/or a learning rate of 0.0001 for training.

[00116] In the example of FIG. 6, the spatial and the temporal factors may be incorporated into new spatial and temporal cell states in an LSTM. In an embodiment, the ST-LSTM model may be an extension of a LSTM model with spatial and temporal cell states. The ST-LSTM model may be used to learn origin-destination relationships based on the spatial and temporal factors in both local and global views. In an embodiment, the extension from a LSTM model to a ST-LSTM model may seek to allow origin-destination relationships to be learned as the LSTM is capable of learning origin-origin or destination-destination relationships given an origin or a destination sequence respectively, but is unable to learn origin-destination relationships.

[00117] In an embodiment, given an input location $\vec{l}_{t_i}$ for timestep t_i , W_i^s, W_f^s, W_c^s and V_i^s, V_f^s, V_c^s may be the corresponding weight matrices for the spatial cell state’s input and forget gates (i.e. $i_{t_i}^s$ and $f_{t_i}^s$), and cell input $\widetilde{c}_{t_i}^s$. These weight matrices may learn representations for the location’s geohash embedding $\vec{l}_{t_i}^{geo}$ and spatial interval vector $\Delta s_{l_{t_i}}$. U_i^s, U_f^s, U_c^s weight matrices.

Corresponding biases b_i^s, b_f^s, b_c^s may learn a representation for the previous hidden state $h_{t_{i-1}}$, which may enforce a recurrent structure and may learn sequential dependencies for the spatial cell state. After computing the representations, activation functions of sigmoid σ and hyperbolic tangent $\tanh$ may be applied. Then, the spatial cell state $c_{t_i}^s$ may be computed from the gates and cell input where $\odot$ is the Hadamard product. The following equations may be used to compute the spatial cell state $c_{t_i}^s$:

$$\begin{aligned} i_{t_i}^s &= \sigma \left(W_i^s \overrightarrow{l_{t_i}^{geo}} + V_i^s \Delta \overrightarrow{s_{l_{t_i}}} + U_i^s h_{t_{i-1}} + b_i^s \right) \\ f_{t_i}^s &= \sigma \left(W_f^s \overrightarrow{l_{t_i}^{geo}} + V_f^s \Delta \overrightarrow{s_{l_{t_i}}} + U_f^s h_{t_{i-1}} + b_f^s \right) \\ \widetilde{c_{t_i}^s} &= \tanh \left(W_c^s \overrightarrow{l_{t_i}^{geo}} + V_c^s \Delta \overrightarrow{s_{l_{t_i}}} + U_c^s h_{t_{i-1}} + b_c^s \right) \\ c_{t_i}^s &= f_{t_i}^s \odot c_{t_{i-1}}^s + i_{t_i}^s \odot \widetilde{c_{t_i}^s}. \end{aligned}$$

[00118] In an embodiment, the spatial cell state equations may be incorporate values from the local view such as geohash embeddings and values from the global view such as spatial intervals into the spatial cell state $c_{t_i}^s$.

[00119] In an embodiment, given an input location $\overrightarrow{l_{t_i}}$ for timestep t_i , W_i^t, W_f^t, W_c^t and V_i^t, V_f^t, V_c^t may be the corresponding weight matrices for the temporal cell state's input and forget gates (i.e. $i_{t_i}^t$ and $f_{t_i}^t$), and cell input $\widetilde{c_{t_i}^t}$. These weight metrics may learn representations for the location's visit timeslot embedding $\overrightarrow{l_{t_i}^{slot}}$ and temporal interval vector $\Delta \overrightarrow{t_{l_{t_i}}}$. U_i^t, U_f^t, U_c^t weight matrices. Corresponding biases b_i^s, b_f^s, b_c^s may learn a representation for the previous hidden state $h_{t_{i-1}}$, which may enforce a recurrent structure and learn sequential dependencies for the temporal cell state. After computing the representations, activation functions of sigmoid σ and hyperbolic tangent $\tanh$ may be applied. Then, the temporal cell state $c_{t_i}^t$ may be computed from the gates and cell input where $\odot$ may be the Hadamard product. The following equations may be used to compute the temporal cell state $c_{t_i}^t$:

$$\begin{aligned}
i_{t_i}^t &= \sigma \left(W_i^t \overrightarrow{l_{t_i}^{slot}} + V_i^t \overrightarrow{\Delta t_{t_i}} + U_i^t h_{t_{i-1}} + b_i^t \right) \\
f_{t_i}^t &= \sigma \left(W_f^t \overrightarrow{l_{t_i}^{slot}} + V_f^t \overrightarrow{\Delta t_{t_i}} + U_f^t h_{t_{i-1}} + b_f^t \right) \\
\widetilde{c}_{t_i}^t &= \tanh \left(W_c^t \overrightarrow{l_{t_i}^{slot}} + V_c^t \overrightarrow{\Delta t_{t_i}} + U_c^t h_{t_{i-1}} + b_c^t \right) \\
c_{t_i}^t &= f_{t_i}^t \odot c_{t_{i-1}}^t + i_{t_i}^t \odot \widetilde{c}_{t_i}^t.
\end{aligned}$$

[00120] In an embodiment, the temporal cell state equations may be incorporate values from the local view such as timeslot embeddings and values from the global view such as temporal intervals into the spatial cell state $c_{t_i}^t$.

[00121] In an embodiment, the three hidden states may be fused as the output hidden state for the current timestep. In an embodiment, to compute the hidden state h_{t_i} for the ST-LSTM of timestep t_i , a representation may be learned with the weight matrix W_h from the concatenation $||$ of the cell state c_{t_i} , spatial cell state $c_{t_i}^s$ and temporal cell state $c_{t_i}^t$. The representation may be subjected to the hyperbolic tangent function $\tanh$ and Hadamard product $\odot$ with the LSTM's existing output gate o_{t_i} . The following equation may be used to compute the hidden state h_{t_i} :

$$h_{t_i} = o_{t_i} \odot \tanh \left(W_h (c_{t_i} || c_{t_i}^s || c_{t_i}^t) \right).$$

[00122] In an embodiment, the spatial and temporal cell states in addition to the LSTM's cell state may enable OD relationships to be learned as the LSTM alone may not be able to learn OD relationships.

[00123] FIG. 7 illustrates a schematic diagram of an encoder and a decoder system according to an embodiment of the present disclosure.

[00124] In an embodiment, with the recurrent neural network model (e.g., the ST-LSTM model), a Personalized Preference Attention (PPA) model may be disclosed. The PPA may be a Spatial-Temporal Origin-Destination Personalized Preference Attention (STOD-PPA). The PPA model may be or may use an encoder-decoder framework.

[00125] In an embodiment, the system may include an encoder. In an embodiment, the encoder may encode the historical origin and destination sequences of the user to capture their preferences.

In an embodiment, the encoder may use the recurrent neural network model to learn OO, DD and OD relationships. In an embodiment, as each user's sequence of OD tuples s_{u_m} is partitioned into training and testing partitions, the training partition $s_{u_m}^{train} = \{(o_{t_1}, d_{t_1}), (o_{t_2}, d_{t_2}), \dots, (o_{t_i}, d_{t_i})\}$ may be used and may be split into separate origin and destination sequences of $s_{u_m}^{train^O} = \{o_{t_2}, o_{t_3}, \dots, o_{t_i}\}$ and $s_{u_m}^{train^D} = \{d_{t_1}, d_{t_2}, \dots, d_{t_{i-1}}\}$ respectively. In an embodiment, for efficiency, the first origin o_{t_1} from $s_{u_m}^{train^O}$ and the last destination d_{t_i} from $s_{u_m}^{train^D}$ may be omitted so that both the encoder and decoder will use the same set of input sequences, which may allow batch training to be performed for each user. In an embodiment, both $s_{u_m}^{train^O}$ and $s_{u_m}^{train^D}$ may be encoded separately, with the ST-LSTMs of ϕ^O and ϕ^D respectively:

$$h_{u_m}^O = \phi^O(s_{u_m}^{train^O})$$

$$h_{u_m}^D = \phi^D(s_{u_m}^{train^D}).$$

[00126] In an embodiment, this may allow OO and DD relationships to be learned in their own ST-LSTM, as well as OD relationships from the newly proposed spatial and temporal cell states. In an embodiment, both $h_{u_m}^O$ and $h_{u_m}^D$ may be concatenated for a final set of all hidden states $h_{u_m}^{OD}$ for user $u_m \in U$ as the output of the encoder, for use by the decoder in training and testing to predict the next destination.

[00127] In an embodiment, the system may include a decoder. In an embodiment, the decoder may encode the encoded hidden states to perform the predictive task of the next destination or drop-off point given the current origin, and/or previous destination and/or the user ID. In an embodiment, after encoding the OD sequences $h_{u_m}^{OD}$, the Personalized Preference Attention (PPA) decoder module may be applied to attend to all the encoded OD hidden states and may compute an origin-aware personalized hidden representation based on the users' dynamic preferences.

[00128] In an embodiment, this may allow OO and DD relationships to be learned in their own ST-LSTM, as well as OD relationships from the newly proposed spatial and temporal cell states.

In an embodiment, both $h_{u_m}^O$ and $h_{u_m}^D$ may be concatenated for a final set of all hidden states $h_{u_m}^{OD}$ for user $u_m \in U$ as the output of the encoder, for use by the decoder in training and testing to predict the next destination.

[00129] In an embodiment, the decoder may compute an attention score for each encoded hidden state in $h_{u_m}^{OD}$. In an embodiment, the attention score may be computed by taking the embedding inputs of the previous destination $\overrightarrow{d_{t_{i-1}}}$, current origin $\overrightarrow{o_{t_i}}$ and current user $\overrightarrow{u_m}$ and each hidden state $\overrightarrow{h_i} \in h_{u_m}^{OD}$ after encoding both origin and destination sequences. In an embodiment, the attention score $\overrightarrow{\alpha_i}$ may be computed using the equation:

$$\overrightarrow{\alpha_i} = \frac{\exp\left(\sigma_{LR}\left(W_A(\overrightarrow{u_m} \parallel \overrightarrow{o_{t_i}} \parallel \overrightarrow{d_{t_{i-1}}}) \parallel \overrightarrow{h_i}\right)\right)}{\sum_{\overrightarrow{p_i} \in h_{u_m}^{OD}} \exp\left(\sigma_{LR}\left(W_A(\overrightarrow{u_m} \parallel \overrightarrow{o_{t_i}} \parallel \overrightarrow{d_{t_{i-1}}}) \parallel \overrightarrow{p_i}\right)\right)}.$$

where W_A may be a weight matrix to learn a representation for the concatenated inputs, followed by the Leaky ReLU activation function σ_{LR} , then applying a softmax normalization across $h_{u_m}^{OD}$.

[00130] In an embodiment, a weighted sum of the attention scores $\overrightarrow{\alpha_i} \in \alpha_{t_i}$ may be applied to each encoded hidden state $\overrightarrow{h_i} \in h_{u_m}^{OD}$ to compute a hidden representation that best represents the user's preferences. In an embodiment, the output hidden representation $\overrightarrow{y_{t_i}}$ for timestep t_i may be calculated using the following equation:

$$\overrightarrow{y_{t_i}} = \sum_{\overrightarrow{\alpha_i} \in \alpha_{t_i}, \overrightarrow{h_i} \in h_{u_m}^{OD}} \overrightarrow{\alpha_i} \odot \overrightarrow{h_i}.$$

[00131] In an embodiment, the probability distribution of the next destination may be computed using the equation:

$$P(d_{t_i} | o_{t_i}, d_{t_{i-1}}, u_m) = \text{softmax}\left(W_{loc}(\overrightarrow{y_{t_i}})\right).$$

where $\overrightarrow{y_{t_i}}$ may be projected to the number of locations or $|L|$ using the weight matrix W_{loc} , followed by a softmax function to derive a probability distribution of all locations by learning $P(d_{t_i} | o_{t_i}, d_{t_{i-1}}, u_m)$ as a multi-classification problem. Accordingly, we may sort the distribution

in descending order to achieve the final ranked recommendation set where the next destination location d_{t_i} should be highly ranked.

[00132] FIG. 8A illustrates exemplary statistics of datasets according to various embodiments.

[00133] In FIG. 8A, seven datasets SE1-SE7 are shown. In an embodiment, each dataset may include data indicating at least one of: a number of users, and/or number of locations, and/or number of origin locations, and/or number of destination locations, and/or number of trips. In an embodiment, the number of locations may be equal to the number of origin locations, and the number of destination locations.

[00134] FIG. 8B illustrates exemplary performance of the datasets SE-1 to SE-4 of FIG. 8A according to various embodiments. FIG. 8C illustrates exemplary performance of the datasets SE-5 to SE-7 of FIG. 8A according to various embodiments.

[00135] In FIGS. 8B and 8C, the evaluation results where the STOD-PPA model surpassed all existing methods is shown. In FIGS. 8B and 8C, the STOD-PPA along with their standard deviations after 10 runs on different random seeds, on surpassing the LSTPM, the LSTPM-OD extension which also considers both origin and destination information for a fair comparison, as well as the other existing methods and baselines, for all datasets and all metrics. Acc@K may evaluate the quality of the ranked list up to K and MAP evaluates the quality of the entire ranked list.

[00136] In an embodiment, standard metrics of Acc@ K may be used, in which K may be $K \in \{1, 5, 10\}$. K and Mean Average Precision (MAP) may be used for evaluation. Acc@ K may measure the performance of the recommendation set up to K , where the smaller K is, the more challenging it is to perform well. In an embodiment, in Acc@1, a score of 1 may be awarded if the ground truth next destination is in the first position ($K = 1$) of the predicted ranked set, *i.e.*, given the highest probability. In an embodiment, in Acc@1, a score of 0 may be awarded if the ground truth next destination is given the lowest probability. In an embodiment, Acc@ K focuses on top K .

In an embodiment, MAP may evaluate the quality of the entire recommendation set and/or may measure the overall performance of the model.

[00137] FIG. 9 illustrates a schematic diagram of an encoder and a decoder system according to an embodiment of the present disclosure.

[00138] FIG. 9 shows a sample test case from a dataset where the model may be interpreted to understand the user preferences. In an embodiment, a test input tuple of a user ID 3250, previous destination ID 1321 and current origin ID 6, may be used as inputs to the PPA decoder. The PPA decoder may apply the personalized preference attention on the encoded OD hidden states from the user's historical OD sequences. In an embodiment, in the encoder, the corresponding origin and destination ID sequences, as well as attention weights computed for each hidden state (in percentages for clarity), may be done by the PPA decoder. In an embodiment a notable difference of weights computed may be used to best perform the predictive task and may support interpretability. For example, transition from 1671 to 1331 has the highest weight and while origin ID 79 has the lowest weight). In the example, with the ground truth destination ID of 1671, the STOD-PPA approach was able to correctly predict the destination ID 1671 with the highest probability score of 0.93.

[00139] 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.

CLAIMS

What is claimed is:

1. A system for predicting a destination location, comprising:
one or more processors; and
a memory having instructions stored therein, the instructions, when executed by the one or more processors, causing the one or more processors to use at least one recurrent neural network to:
process spatial data comprising a first set of information about origin locations and destination locations;
process temporal data comprising a second set of information about times at the origin locations and the destination locations;
determine hidden state data based on the spatial data and the temporal data, wherein the hidden state data comprises data on origin-destination relationships;
receive a current input data from a user, wherein the current input data comprises an identity of the user and the current origin location of the user; and
predict the destination location based on the hidden state data and the current input data.
2. The system of claim 1, wherein the current input data further comprises a previous destination of the user.
3. The system of any of claims 1 to 2, wherein the first set of information comprises local origin locations and local destination locations, wherein the second set of information comprises times at the local origin locations and the local destination locations, and wherein the local origin locations and the local destination locations are within a geohash.
4. The system of claim 3, wherein the first set of information comprises global origin locations and global destination locations, wherein the second set of information comprises times at the global origin locations and the global destination locations, and wherein the global origin locations and the global destination locations are outside of the geohash.
5. The system of any of claims 1 to 4, further comprising:

an encoder configured to process the spatial data and the temporal data, and configured to determine the hidden state data.

6. The system of claim 5, further comprising:

a decoder configured to receive the current input data from the user, to receive the hidden state data from the encoder, and to predict the destination location based on the hidden state data and the current input data.

7. The system of claim 6, wherein the decoder is configured to determine personalized preference data of the user based on the hidden state data, the current input data and a first predetermined weight.

8. The system of claim 7, wherein the decoder is configured to predict the destination location based on the personalized preference data and the hidden state data with a second predetermined weight.

9. The system of any of claims 6 to 8, wherein the decoder is configured to determine a probability that the destination location predicted is a correct destination location.

10. A method for predicting a destination location, comprising:

using at least one recurrent neural network to:

process spatial data comprising information about origin locations and destination locations;

process temporal data comprising information about times at the origin locations and the destination locations;

determine origin-destination relationships based on the spatial data and the temporal data;

receive a current input data from a user, wherein the current input data comprises an identity of the user and the current origin location of the user; and

predict the destination location based on the origin-destination relationships and the current input data.

11. The method of claim 10, wherein the current input data further comprises a previous destination of the user.

12. The method of any of claims 10 to 11, wherein the first set of information comprises local origin locations and local destination locations, wherein the second set of information comprises times at the local origin locations and the local destination locations, and wherein the local origin locations and the local destination locations are within a geohash.
13. The method of claim 12, wherein the first set of information comprises global origin locations and global destination locations, wherein the second set of information comprises times at the global origin locations and the global destination locations, and wherein the global origin locations and the global destination locations are outside of the geohash.
14. The method of any of claims 10 to 13, further comprising:
 - using an encoder to:
 - process the spatial data and the temporal data; and
 - determine the hidden state data.
15. The method of claim 14, further comprising:
 - using a decoder to:
 - receive the current input data from the user;
 - receive the hidden state data from the encoder; and
 - predict the destination location based on the hidden state data and the current input data.
16. The method of claim 15, further comprising:
 - determining personalized preference data of the user based on the hidden state data, the current input data and a first predetermined weight using the decoder.
17. The method of claim 16, further comprising:
 - predicting the destination location based on the personalized preference data and the hidden state data with a second predetermined weight using the decoder.
18. The method of any of claims 15 to 17, further comprising:
 - determining a probability that the destination location predicted is a correct destination location using the decoder.

19. A non-transitory computer-readable medium storing computer executable code comprising instructions for predicting a destination location according to any one of claims 1 to 18.
20. A computer executable code comprising instructions for predicting a destination location according to any one of claims 1 to 19.