

Description

Title of Invention: Method and System for Recommending an Item to a User

5 FIELD OF INVENTION

1. The present disclosure relates broadly, but not exclusively, to methods and systems for recommending an item to a user.

10 BACKGROUND

2. Platforms that offer delivery services for items such as food, groceries, and other similar items deliver the purchased items from restaurants and marts to millions of users everyday. One crucial component in these platforms is to recommend further items to these users.

3. However, various recommendation techniques that are used in the platforms today generally only find and recommend similar candidate items based on a single item that a candidate has previously purchased. Further, traditional recommendation systems used by the platforms utilize offline batch recommendation method which typically provides recommendations based on historical and outdated data.

4. A need therefore exists to provide methods and systems that seek to overcome or at least minimize the above mentioned challenges.

SUMMARY

- 5 5. According to a first aspect of the present disclosure, there is provided a method for
recommending an item to a user, the method comprising: identifying, by a server,
a first plurality of items, the first plurality of items being a plurality of items that are
selected or ordered by a user, the identification based on data relating to a plurality
of orders or selections made by the user; computing, by the server, a score for
each item of a second plurality of items, the second plurality of items being a
10 plurality of items that are selected or ordered by a plurality of users, the score
indicating a measure of similarity between the first plurality of items and each item
of the second plurality of items; and recommending, by the server, one or more
items of the second plurality of items to the user based on the computed score, the
recommendation being made in real-time and in response to the plurality of orders
15 or selections made by the user.
6. According to a second aspect of the present disclosure, there is provided a system
for recommending an item to a user, comprising: at least one processor; and at
least one memory including computer program code; the at least one memory and
20 the computer program code configured to, with the at least one processor, cause
the system at least to: identify a first plurality of items, the first plurality of items
being a plurality of items that are selected or ordered by a user, the identification
based on data relating to a plurality of orders or selections made by the user;
compute a score for each item of a second plurality of items, the second plurality
25 of items being a plurality of items that are selected or ordered by a plurality of
users, the score indicating a measure of similarity between the first plurality of
items and each item of the second plurality of items; and recommend one or more
items of the second plurality of items to the user based on the computed score,
the recommendation being made in real-time and in response to the plurality of
30 orders or selections made by the user.

BRIEF DESCRIPTION OF THE DRAWINGS

7. Embodiments and implementations are provided by way of example only, and will be better understood and readily apparent to one of ordinary skill in the art from the following written description, read in conjunction with the drawings, in which:

8. Fig. 1 illustrates a system for recommending an item to a user according to various embodiments of the present disclosure.

9. Fig. 2 is a schematic diagram of a recommendation server, according to various embodiments of the present disclosure.

10. Fig. 3A depicts an overview of how a typical recommendation system works according to an example.

11. Fig. 3B depicts an illustration of an item sequence according to an example.

12. Fig. 4 depicts an overview of how an item may be recommended to a user according to various embodiments.

13. Fig. 5 depicts an illustration of an item recommendation system according to various embodiments.

14. Fig. 6 depicts an illustration of a cluster model according to various embodiments.

15. Fig. 7 illustrates an example flow diagram for recommending an item to a user according to various embodiments.

16. Fig. 8A is a schematic block diagram of a general purpose computer system upon which the recommendation server shown in Figs. 1 and 2 can be practiced.

17. Fig. 8B is a schematic block diagram of a general purpose computer system upon which a combined transaction processing and recommendation server of Fig. 1 can be practiced.

18. Fig. 9 shows an example of a computing device to realize the transaction processing server shown in Fig. 1.

19. Fig. 10 shows an example of a computing device to realize the recommendation server shown in Figs. 1 and 2.

20. Fig. 11 shows an example of a computing device to realize a combined transaction processing and recommendation server shown in Fig. 1.

21. Skilled artisans will appreciate that elements in the figures are illustrated for simplicity and clarity and have not necessarily been depicted to scale. For example, the dimensions of some of the elements in the illustrations, block diagrams or flowcharts may be exaggerated in respect to other elements to help to improve understanding of the present embodiments.

DETAILED DESCRIPTION

Terms Description

22. A platform refers to a set of technologies that is used as a base for facilitating exchanges between two or more interdependent servers, entities and/or devices, for example between a requestor device (of a product or service) and a provider device (of the product or service). For example, a platform may offer a service offered by a provider such as a ride, delivery, online shopping, insurance, and other similar services to a requestor. The requestor can typically access the platform via a website, an application, or other similar methods.

23. An item may refer to any product, dish, food, drink, grocery, stall, shop, brand, and other similar entity that is available for selection and/or purchasing by a user on a platform. A user browsing an interface of the platform may select an item that interests him (e.g., by clicking a mouse button, pressing a keyboard button, pressing a touch screen, etc over an icon, picture, button or other similar representation of the item that is shown to the user on the interface of the platform) to check out further items (e.g., clicking on a brand icon to open up a new menu showing all available items on sale relating to the brand) or options (e.g., customizations relating to an item before purchasing the item) related to the item. The user may also order the item (e.g., purchase the item by a payment method).

An item may be selected or ordered during a session (e.g., a period of time in which a user is interacting with the platform). Data relating to one or more items that are ordered or selected by a user during a session may be transmitted in real-time (e.g., transmitted in a real-time stream of data as the user is interacting with the platform during the session) and stored in, for example, a database. These data may be processed to provide recommendation of items to the user during a new session. Each session may be set to a certain duration (e.g., 30 minutes or other possible value), so that a new session starts after the set duration. The data relating to the one or more items may indicate an item identifier that identifies each of the one or more items. The data may also indicate a session identifier (ID) that identifies the session at which each of the one or more items is selected or ordered, time stamp that indicates a date and time at which an item is selected or ordered, a country/city ID that indicates a location at which the session takes place, a user ID identifying the user, and other similar data. It will be appreciated that the database also stores data relating a plurality of items that are selected or ordered by a plurality of users, and the data may be sorted based on the various IDs and timestamps for data retrieval purposes.

24. Clustering may be performed on the data. Cluster analysis or clustering is a task of grouping a set of objects (e.g., in this case, a set of items) in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). Techniques for performing cluster analysis include DBSCAN, K-means, and other similar techniques. In the present disclosure, a cluster model may be utilized for pre-training an item embedding (e.g., embedding a feature or vector relating to an item in a feature space) and then classifying the item embedding into different clusters, so that similar items (e.g., similar to items that are ordered or selected by a user in a previous or current session) can be identified and recommended to the user in the next session.

25. In at least some embodiments, a user may be any suitable type of entity, which may include a person, a consumer looking to purchase or order a product or service via a transaction processing server, a seller or merchant looking to sell a product or service via the transaction processing server, a motorcycle driver or pillion rider in a case of the user looking to book or provide a motorcycle ride via

the transaction processing server, a car driver or passenger in a case of the user looking to book or provide a car ride via the transaction processing server, and other similar entity that selects or orders an item on the platform. A user who is registered to the transaction processing server will be called a registered user. A user who is not registered to the transaction processing server will be called a non-registered user. The term user will be used to collectively refer to both registered and non-registered users. A user may interchangeably be referred to as a requestor (e.g. a person who requests for a product or service) or a provider (e.g. a person who provides the requested product or service to the requestor).

26. In at least some embodiments, a recommendation server is a server that hosts software application programs for recommending an item to a user. The recommendation server may be implemented as shown in the schematic diagram of Fig. 2 for recommending an item to a user.

27. In at least some embodiments, a transaction processing server is a server that hosts software application programs for processing payment transactions for, for example, a travel-ordination request, purchasing of a good or service by a user, and other similar services. The transaction processing server communicates with any other servers (e.g., a recommendation server) concerning processing payment transactions relating to the purchasing of the good or service, such as ordering an item. For example, data relating to a purchase of an item (e.g. time stamp indicating date and time of purchase, session ID, item ID, user ID, country/city ID, and other similar data) may be provided to the recommendation server for processing and/or storing. The transaction processing server may use a variety of different protocols and procedures in order to process the payment and/or travel co-ordination requests. It will be appreciated that purchasing and ordering may be used interchangeably.

28. Transactions that may be performed via a transaction processing server include product or service purchases, credit purchases, debit transactions, fund transfers, account withdrawals, etc. Transaction processing servers may be configured to process transactions via cash-substitutes, which may include payment cards, letters of credit, checks, payment accounts, etc.

29. In at least some embodiments, the transaction processing server is usually managed by a service provider that may be an entity (e.g. a company or organization) which operates to process transaction requests (e.g., for ordering an item) and/or travel co-ordination requests. The transaction processing server may include one or more computing devices that are used for processing transaction requests and/or travel co-ordination requests.

30. In at least some embodiments, a transaction account is an account of a user who is registered at a transaction processing server. The user can be a customer, a merchant providing a product for sale on a platform and/or for onboarding the platform, a hail provider (e.g., a driver), or any third parties (e.g., a courier) who want to use the transaction processing server. In certain circumstances, the transaction account is not required to use the transaction processing server. A transaction account includes details (e.g., name, address, vehicle, face image, user ID, etc.) of a user. The transaction processing server manages the transaction.

31. Embodiments will be described, by way of example only, with reference to the drawings. Like reference numerals and characters in the drawings refer to like elements or equivalents.

32. Some portions of the description which follows are explicitly or implicitly presented in terms of algorithms and functional or symbolic representations of operations on data within a computer memory. These algorithmic descriptions and functional or symbolic representations are the means used by those skilled in the data processing arts to convey most effectively the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities, such as electrical, magnetic or optical signals capable of being stored, transferred, combined, compared, and otherwise manipulated.

33. Unless specifically stated otherwise, and as apparent from the following, it will be appreciated that throughout the present specification, discussions utilizing terms such as “identifying”, “detecting”, “grouping”, “determining”, “associating”,

“selecting”, “calculating”, “processing”, “storing”, “indicating”, “discerning”, or the like, refer to the action and processes of a computer system, or similar electronic device, that manipulates and transforms data represented as physical quantities within the computer system into other data similarly represented as physical quantities within the computer system or other information storage, transmission or display devices.

34. In addition, the present specification also implicitly discloses a computer program, in that it would be apparent to the person skilled in the art that the individual steps of the method described herein may be put into effect by computer code. The computer program is not intended to be limited to any particular programming language and implementation thereof. It will be appreciated that a variety of programming languages and coding thereof may be used to implement the teachings of the disclosure contained herein. Moreover, the computer program is not intended to be limited to any particular control flow. There are many other variants of the computer program, which can use different control flows without departing from the scope of the specification.

35. Furthermore, one or more of the steps of the computer program may be performed in parallel rather than sequentially. Such a computer program may be stored on any computer readable medium. The computer readable medium may include storage devices such as magnetic or optical disks, memory chips, or other storage devices suitable for interfacing with a computer. The computer readable medium may also include a hard-wired medium such as exemplified in the Internet system, or wireless medium such as exemplified in the GSM mobile telephone system. The computer program when loaded and executed on such a computer effectively results in an apparatus that implements the steps of the preferred method.

36. Fig. 3A shows a typical recommendation system 300 including two stages: recall and ranking. At the recall stage 302, a popular approach is that relevant candidates are retrieved based on matching (e.g., by a matching module 306) between embedding of all candidates 304 (a matrix with dimension $N \times D$; N is number of all items; D is the feature dimension for an item) and target candidates (a vector with dimension $1 \times D$). For example, if a user ordered a chicken burger previously, a beef burger, fries et al may be recommended to the user because these items

are frequently ordered together. Even though a large number (e.g., 800) of candidates are retrieved, only top K (e.g., 100) candidates are selected to reduce the latency and cost of the recommendation system 300. After that, the top K recalled candidates are then passed to the ranking stage 308 where the candidates will be ranked based on the preference of each user (e.g., based on user's preferred cuisine, price, location, time) in order to improve the user's experience.

37. The present disclosure focuses on the recall stage of a recommendation system. For this stage, the traditional method for platforms in the industry is to use item2vec (e.g. a neural-network-based item-to-item recommendation approach which maps items into a common "semantic" high-dimensional space where those can be compared using associated vectors) to create embeddings of all candidates and use Facebook AI Similarity Search (Faiss) to do a similarity search to find relevant candidates based on Euclidean distance.

38. The item2vec learns the item embedding by capturing the relation between an item and its surrounding items in an item sequence. Item embedding is a learned representation for items where items that have the same meaning have a similar representation. It is usually in a high dimensional vector form, the distance between two item embeddings can be calculated by the distance by their corresponding vectors. An example of an item sequence 312 used in item2vec is shown in Fig. 3B. The item sequence may comprise ordered or clicked item lists of different users and the sequence order may be determined by the order or click time.

39. With the item embedding learnt by item2vec, frameworks like Faiss were used to perform a similarity search on a large scale. Faiss contains multiple algorithms for similarity search. The distance between item embedding can either be Euclidean distance or dot product. In the recall stage of a recommendation system, the relevant candidates are retrieved by the similar items of the items in the users' order or click history. The items in users' order or click history may also be called target items or target candidates.

40. A drawback of this method is that it only finds similar candidates of a single target candidate. For example, if a user ordered item A and item B, these two items are used as target items. The Faiss may find a similar candidate list List_A for target

candidate A and a similar candidate list List_B for target candidate B. However, it is not able to find a similar candidate list List_AB for target candidates [item A, item B]. In some platforms, users are most likely to interact with more than one item. By comparing the list List_AB and merged list List_A and list List_B with additional logic, list List_AB may be able to provide more accurate candidates and also save computation cost in both the recall stage and following stage.

41. Another drawback of the traditional recommendation system is that it uses the offline batch recommendation method which generates the similar candidate list based on history data, e.g., data from yesterday or last month, and it does not consider real time interactions of a user, for example users' clicked items in the last 30 minutes. Compared to traditional offline batch recommendation methods, a real time item recommendation is trying to predict which items the consumer will click or interact in the next session based on the user's interactions in the previous session. User experience can advantageously be improved if recommended items are refreshed as the user interacts with Grab App by using real time item recommendation.

42. In the present disclosure, a real time recommendation system which recommends similar items based on real time user behavior (e.g. a user's selected items and ordered items) is proposed to address the above-mentioned issues. In this system, a similarity search (SIMS) algorithm to find similar items based on interacted items of multiple users is proposed. For example, similar items may be recommended based on multiple target items instead of a single target item. This algorithm may be developed by using, for example, Tensorflow, and can advantageously be configured to serve millions of item embedding with low latency, scalability and cost efficiency. If a traditional similarity search framework is used to find similar items for each item in a target item list, an additional merging logic is required. Advantageously, the proposed system is an end-to-end recommendation model which can serve as a self-contained service and does not need merging logic of different item lists. Furthermore, it is possible to use the proposed system for other business cases, for example, recommending items based on a similar merchant, brand or other similar entity, instead of (or on top of) being based on similar items.

5 **43.** Fig. 1 illustrates a block diagram of an example system 100 for recommending an item to a user. In some embodiments, the system 100 enables a payment transaction for an item, a good or service, and/or a request for a ride or delivery of a physical item (e.g. one or more items) by a user (e.g., between a requestor and a provider).

10 **44.** The system 100 comprises a requestor device 102, a provider device 104, an acquirer server 106, a transaction processing server 108, an issuer server 110, an recommendation server 140 and a reference database 150.

15 **45.** The requestor device 102 is in communication with a provider device 104 via a connection 112, and may be associated with a user. The connection 112 may be wireless (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet). The requestor device 102 is also in communication with the recommendation server 140 via a connection 121, wherein the recommendation server 140 may be configured to receive information relating to one or more items that are selected or ordered from the requestor device 102. The connection 121 may be via a network (e.g., the Internet). The requestor device 102 may also be connected to a cloud that facilitates the system 100 for recommending an item to a user. For example, the requestor device 102 can send a signal or data to the cloud directly via a wireless connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

20 **46.** The provider device 104 is in communication with the requestor device 102 as described above, usually via the transaction processing server 108, and may be associated with a provider of a ride or delivery (e.g. a driver responding to the request for the ride or delivery). The provider device 104 is, in turn, in communication with an acquirer server 106 via a connection 114. The provider device 104 is also in communication with the recommendation server 140 via a connection 123, wherein the recommendation server 140 may be configured to receive information relating to one or more items that are selected or ordered from the provider device 104. The connections 114 and 123 may be via a network (e.g., the Internet). The provider device 104 may also be connected to a cloud that facilitates the system 100 for recommending an item to a user. For example, the provider device 104 can send a signal or data to the cloud directly via a wireless

connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

47. The acquirer server 106, in turn, is in communication with the transaction processing server 108 via a connection 116. The transaction processing server 108, in turn, is in communication with an issuer server 110 via a connection 118. The connections 116 and 118 may be via a network (e.g., the Internet).

48. The transaction processing server 108 is further in communication with the recommendation server 140 via a connection 120. The connection 120 may be over a network (e.g., a local area network, a wide area network, the Internet, etc.). In one arrangement, the transaction processing server 108 and the recommendation server 140 are combined and the connection 120 may be an interconnected bus.

49. The recommendation server 140, in turn, is in communication with the reference database 150 via respective connection 122. The connection 122 may be over a network (e.g., the Internet). The recommendation server 140 may also be connected to a cloud that facilitates the system 100 for recommending an item to a user. For example, the recommendation server 140 can send a signal or data to the cloud directly via a wireless connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

50. The reference database 150 may comprise data that is utilized by the recommendation server 140 for recommending an item to a user. For example, data relating to a plurality of items that are selected or ordered by a plurality of users (e.g., referred to herein as a second plurality of items) may be transmitted in real time (e.g., transmitted when each item is selected or ordered) and stored in the reference database 150, the data indicating a respective time at which each item of the second plurality of items is selected or ordered, and indicating a respective user identifier (e.g., for identifying the user who ordered or selected the item) associated with each item. The data relating to the second plurality of items may be sorted based on the indicated time and user identifier, and may referred to herein as historical data. Further, the data relating to a plurality of orders or selections made by the user may also be transmitted in real time (e.g., transmitted

when each item is selected or ordered) and stored in the reference database 150, the data indicating a respective time at which each item of a plurality of items is selected or ordered by the user (e.g., herein referred to as a first plurality of items), and indicating a respective user identifier (e.g., for identifying the user who
 5 ordered or selected the item) associated with each item, and the data may be sorted based on the indicated time and user identifier. In an implementation, the reference database 150 may be combined with the recommendation server 140. In an example, the reference database 150 may be managed by an external entity.

10 **51.** The recommendation server 140 may be configured to identify the first plurality of items based on the data relating to the plurality of orders or selections made by the user, for example by checking the user identifier and the time indicated for each item in the data. The recommendation server 140 may then be configured to
 15 compute a score for each item of the second plurality of items, the score indicating a measure of similarity between the first plurality of items and each item of the second plurality of items. The recommendation server 140 may then be configured to recommend one or more items of the second plurality of items to the user based on the computed score, the recommendation being made in real-time and in response to the plurality of orders or selections made by the user.

20 **52.** Computing the score may comprise: generating, for each respective item of the first plurality of items, a respective first cluster comprising a mapping of the respective item with one or more other items from the second plurality of items that are similar to the respective item; generating, by the recommendation server 140,
 25 a second cluster comprising a mapping of the first plurality of items; and computing the score for each of the one or more other items, the computation based on a processing of the plurality of first clusters and the second cluster. The processing may comprise: generating a first matrix based on the plurality of first clusters, and a second matrix based on the second cluster; computing a dot product between
 30 the first and second matrix to generate a similarity matrix; and performing row product operation on the similarity matrix to obtain the computed scores.

53. In an implementation, there may be more than one reference databases, in which the recommendation server 140 may be configured to determine which database
 35 to use for each step during the process of recommending an item to a user.

Alternatively, one or more modules may store the above-mentioned data instead of the reference database 150, wherein the module may be integrated as part of the recommendation server 140 or external from the recommendation server 140.

- 5 **54.** In the illustrative embodiment, each of the devices 102, 104, and the servers 106, 108, 110, 140, and/or reference database 150 provides an interface to enable communication with other connected devices 102, 104 and/or servers 106, 108, 110, 140, and/or reference database 150. Such communication is facilitated by an application programming interface ("API"). Such APIs may be part of a user interface that may include graphical user interfaces (GUIs), Web-based interfaces, programmatic interfaces such as application programming interfaces (APIs) and/or sets of remote procedure calls (RPCs) corresponding to interface elements, messaging interfaces in which the interface elements correspond to messages of a communication protocol, and/or suitable combinations thereof. For example, it is possible for the requestor device 102 to send data relating to a requested ride or delivery, and for the provider device 104 to send data relating to acceptance or rejection of a ride or delivery request, in response to an enquiry shown on the GUI running on the respective API. It is also possible for the requestor device 102 and/or the provider device 104 to send data in real-time relating to a selection or order of an item by a user, for example in a real-time stream.
- 15
- 20
- 25 **55.** Use of the term 'server' herein can mean a single computing device or a plurality of interconnected computing devices which operate together to perform a particular function. That is, the server may be contained within a single hardware unit or be distributed among several or many different hardware units.
- 30 **56.** The recommendation server 140 is associated with an entity (e.g. a company or organization or moderator of the service). In one arrangement, the recommendation server 140 is owned and operated by the entity operating the transaction processing server 108. In such an arrangement, the recommendation server 140 may be implemented as a part (e.g., a computer program module, a computing device, etc.) of the transaction processing server 108.

57. The transaction processing server 108 may also be configured to manage the registration of users. A registered user has a transaction account (see the discussion above) which includes details of the user. The registration step is called on-boarding. A user may use either the requestor device 102 or the provider device 104 to perform on-boarding to the transaction processing server 108.

58. It may not be necessary to have a transaction account at the transaction processing server 108 to access the functionalities of the transaction processing server 108. However, there are functions that are available to a registered user. These additional functions will be discussed below.

59. The on-boarding process for a user is performed by the user through one of the requestor device 102 or the provider device 104. In one arrangement, the user downloads an app (which includes the API to interact with the transaction processing server 108) to the requestor device 102 or the provider device 104. In another arrangement, the user accesses a website (which includes the API to interact with the transaction processing server 108) on the requestor device 102 or the provider device 104. The user is then able to interact with the recommendation server 140. The user may be a requestor or a provider associated with the requestor device 102 or the provider device 104, respectively.

60. Details of the registration may include, for example, name of the user, address of the user, emergency contact, blood type or other healthcare information, next-of-kin contact, permissions to retrieve data and information from the requestor device 102 and/or the provider device 104 for recommending an item to a user, such as permission to retrieve location information and data relating to a selection or order of an item from the requestor device 102 and/or the provider device 104. Alternatively, another mobile device may be selected instead of the requestor device 102 and/or the provider device 104 for retrieving the data. Once on-boarded, the user would have a transaction account that stores all the details.

61. The requestor device 102 is associated with a customer (or requestor) who is a party to a transaction that occurs between the requestor device 102 and the provider device 104, or between the requestor device 102 and the recommendation server 140. The requestor device 102 may be a computing

device such as a desktop computer, an interactive voice response (IVR) system, a smartphone, a laptop computer, a personal digital assistant computer (PDA), a mobile computer, a tablet computer, and the like. The requestor device 102 may be associated with a user who initiates a request for a ride or delivery, or a user who selects or orders an item.

62. The requestor device 102 includes transaction credentials (e.g., a payment account) of a requestor to enable the requestor device 102 to be a party to a payment transaction. If the requestor has a transaction account, the transaction account may also be included (i.e., stored) in the requestor device 102. For example, a mobile device (which is a requestor device 102) may have the transaction account of the customer stored in the mobile device.

63. In one example arrangement, the requestor device 102 is a computing device in a watch or similar wearable and is fitted with a wireless communications interface (e.g., a NFC interface). The requestor device 102 can then electronically communicate with the provider device 104 regarding a transaction request. The customer uses the watch or similar wearable can make a request regarding the transaction request, or make a selection or order of an item, by pressing a button on the watch or wearable.

64. The provider device 104 is associated with a provider who is also a party to the transaction request that occurs between the requestor device 102 and the provider device 104. The provider device 104 may be a computing device such as a desktop computer, an interactive voice response (IVR) system, a smartphone, a laptop computer, a personal digital assistant computer (PDA), a mobile computer, a tablet computer, and the like. The provider device 104 may be associated with a provider of a ride or delivery (e.g. a driver or deliverer responding to the request for the ride or delivery), or a user who selects or orders an item.

65. Hereinafter, the term “provider” refers to a service provider and any third party associated with providing a product or service for purchase, or a travel or ride or delivery service via the provider device 104. Therefore, the transaction account of a provider refers to both the transaction account of a provider and the transaction

account of a third party (e.g., a travel co-ordinator or merchant) associated with the provider.

66. If the provider has a transaction account, the transaction account may also be included (i.e., stored) in the provider device 104. For example, a mobile device (which is a provider device 104) may have the transaction account of the provider stored in the mobile device.

67. In one example arrangement, the provider device 104 is a computing device in a watch or similar wearable and is fitted with a wireless communications interface (e.g., a NFC interface). The provider device 104 can then electronically communicate with the requestor to make request regarding the transaction request, or make a selection or order of an item, by pressing a button on the watch or wearable.

68. The acquirer server 106 is associated with an acquirer who may be an entity (e.g. a company or organization) which issues (e.g. establishes, manages, administers) a payment account (e.g. a financial bank account) of a merchant. Examples of the acquirer include a bank and/or other financial institution. As discussed above, the acquirer server 106 may include one or more computing devices that are used to establish communication with another server (e.g., the transaction processing server 108) by exchanging messages with and/or passing information to the other server. The acquirer server 106 forwards the payment transaction relating to a transaction request (e.g., in relation to an order of an item) or a request for a ride to the transaction processing server 108.

69. The transaction processing server 108 is configured to process processes relating to a transaction account by, for example, forwarding data and information associated with the transaction to the other servers in the system 100 such as the recommendation server 140. In an example, the transaction processing server 108 may, instead of the requestor device 102 or provider device 104, transmit data relating to a request message such as a transaction request for ordering an item (e.g. item ID, contextual information such as location of the user based on a city ID, user ID, date of order, time of order, and other similar data) to the recommendation server 104. The transaction processing server 108 may use a

variety of different protocols and procedures in order to process the payment and/or travel co-ordination requests. It will be appreciated that payment for a transaction may be made via a variety of methods such as credit cards, debit cards, digital wallets, buy-first pay-later schemes, and other similar payment methods.

70. The issuer server 110 is associated with an issuer and may include one or more computing devices that are used to perform a payment transaction. The issuer may be an entity (e.g. a company or organization) which issues (e.g. establishes, manages, administers) a transaction credential or a payment account (e.g. a financial bank account) associated with the owner of the requestor device 102. As discussed above, the issuer server 110 may include one or more computing devices that are used to establish communication with another server (e.g., the transaction processing server 108) by exchanging messages with and/or passing information to the other server.

71. The reference database 150 is a database or server associated with an entity (e.g. a company or organization) which manages (e.g. establishes, administers) data relating to users, transactions, products, services, and other similar data, for example relating to the entity. In an arrangement, the reference database 150 may comprise data that is utilized by the recommendation server 140 for recommending an item to a user. For example, data relating to a second plurality of items that are selected or ordered by a plurality of users may be transmitted in real time when each item is selected or ordered, and stored in the reference database 150, the data indicating a respective time at which each item of the second plurality of items is selected or ordered, and indicating a respective user identifier associated with each item. The data relating to the second plurality of items may be sorted based on the indicated time and user identifier, and may referred to herein as historical data. Further, the data relating to a plurality of orders or selections made by the user may also be transmitted in real time when each item is selected or ordered, and stored in the reference database 150, the data indicating a respective time at which each item of a plurality of items is selected or ordered by the user (e.g., herein referred to as a first plurality of items), and indicating a respective user identifier associated with each item, and the data may be sorted based on the indicated time and user identifier. In an implementation, the reference database

150 may be combined with the recommendation server 140. In an example, the reference database 150 may be managed by an external entity.

5 **72.** Advantageously, the system 100 enables recommendation of items based on real-time behaviour of a user, and based on multiple items that are selected or ordered by the user to improve accuracy and efficiency.

10 **73.** Fig. 2 illustrates a schematic diagram of an example recommendation server 140 according to various embodiments. The recommendation server 140 may comprise a data module 260 configured to receive data and information from the requestor device 102, provider device 104, transaction processing server 108, reference database 150, a cloud and other sources of information to recommend an item to a user. For example, the data module 260 may be configured to receive data and information required for processing real-time data stream, processing
15 historical data, generate and train item embeddings, perform clustering, perform similar search, and other similar processes from the requestor device 102, the provider device 104, transaction processing server 108, reference database 150, and/or other sources of information. The data module 260 may be further configured to send information relating to data retrieved in response to selections
20 or orders made by a user to the requestor device 102, the provider device 104, the transaction processing server 108, or other destinations where the information is required.

25 **74.** The recommendation server 140 may comprise an Injection module 262 that is configured for injecting events (e.g., a plurality of selections or orders made by a user) incrementally to the database 150. The injection process is further explained in Figs. 4-7.

30 **75.** The recommendation server 140 may also comprise a backend module 264 that is configured for retrieving user ID, session ID (e.g., an identifier associated with a session) and context information (e.g., a user's latitude, longitude, city ID, and other similar information) based on a current session in which a user is interacting with the platform. This process is further explained in Figs. 4-7.

76. The recommendation server 140 may also comprise a clustering module 266 that is configured for generating an item embedding based on the historical data and training the item embedding based on an item2vec model. The clustering module 266 may be referred to as a SIMS cluster module, and may be further configured for computing a score for each item of the second plurality of items that are selected or ordered by a plurality of users, the score indicating a measure of similarity between the first plurality of items and each item of the second plurality of items. Computing the score may further comprise: generating, for each respective item of the first plurality of items, a respective first cluster comprising a mapping of the respective item with one or more other items from the second plurality of items that are similar to the respective item; generating a second cluster comprising a mapping of the first plurality of items; and computing the score for each of the one or more other items, the computation based on a processing of the plurality of first clusters and the second cluster. The processing may further comprise: generating a first matrix based on the plurality of first clusters, and a second matrix based on the second cluster; computing a dot product between the first and second matrix to generate a similarity matrix; and performing row product operation on the similarity matrix to obtain the computed scores. The clustering process is further explained in Figs. 4-7.

77. The recommendation server 140 may also comprise a recommendation module 268 that is configured for identifying the first plurality of items that are selected or ordered by a user, the identification based on the data relating to the plurality of orders or selections made by the user. The recommendation module 268 may be further configured for recommending one or more items of the second plurality of items to the user based on the computed score, the recommendation being made in real-time and in response to the plurality of orders or selections made by the user. This process is further explained in Figs. 4-7.

78. Each of the data module 260, injection module 262, backend module 264, clustering module 266 and recommendation module 268 may further be in communication with a processing module (not shown) of the recommendation server 140, for example for coordination of respective tasks and functions during the process. The data module 260 may be further configured to communicate with and store data and information for each of the processing module, injection module

262, backend module 264, clustering module 266 and recommendation module 268. Alternatively, all the tasks and functions required for recommending an item to a user may be performed by a single processor of the recommendation server 140.

5

79. Fig. 4 shows an illustration 400 depicting an overview of how an item may be recommended to a user according to various embodiments. A top block 402 illustrates a typical user action flow in an item recommendation scenario. An item recommendation service 422 provides item recommendation for a user 404 browsing the platform. The user 404 may have data relating to the user's history (e.g., past interactions with the platform) or previous session events, e.g., browsing item lists, clicking one of the items, browsing the content, and performing checkouts. These events 406 are streamed (e.g., as a real-time data stream) through a user events streaming module 410, which is usually implemented using Kafka streams. When the user 404 browses the platform in a new session 408, a new session request 412 is sent from the user 404 (e.g., from a device of the user 404 for browsing the platform, such as the requestor device 102 or the provider device 104) to an item recommendation system 414.

10

15

20

80. The item recommendation system 414 leverages on the user's history or previous session events to derive an item recommendation for the user 404. Illustration 500 of Fig. 5 depicts the item recommendation system 414 in more detail to explain how data (e.g., relating to the user's history or previous session events) and the new session request 412 are stored and served.

25

81. To obtain real time data relating to clicked or ordered items of users utilizing the platform (e.g., data relating to the second plurality of items that are selected or ordered by a plurality of users), an event injection service 416 (e.g., a service provided by the injection module 262) is utilized to inject user events (e.g. selections and orders of the plurality of users, including selections and orders of the user 404) from real time streams (e.g. kafka streams) into a database 418 (e.g., the reference database 150). For real time item recommendations, selected or ordered items in a previous session (e.g., the first plurality of items that are selected or ordered by the user 404 during the previous session of interaction between the user 404 and the platform) is utilized to predict what the user 404 may

30

35

order in a next session. Therefore, the event injection service 416 needs to inject the events incrementally in a streaming manner to the database 418 instead of overwriting the data. The data structure for storing and sorting the selected or ordered items that are transmitted to the database 418 (e.g., data structure of the historical data that is stored in the database 418) may be, for example, in Redis format as shown in Table 1 below.

Redis Sorted Set	Clicked Items	Ordered Items
Key	/click/{consumer_id}/{city_id}	/order/{consumer_id}/{city_id}
Member	{item_id}&&{timestamp}	{item_id}&&{timestamp}
Value	{timestamp}	{timestamp}

Table 1: Data structure of selected and ordered items in Redis

82. The database 418 may be configured to update the historical data by appending new data points (e.g., adding new entries for the first plurality of items that selected or ordered by the user 404 in the previous session into the historical data stored in the database 418) and querying the values by timestamp (see last row of Table 1). Due to the above requirements, a time series database may be used, such as KairosDB, Redis and other similar database structures. The items in the database 418 may be sorted by, for example, using a timestamp in the sorted set as shown in Table 1, and/or using other similar factors such as consumer ID (e.g., herein referred to as user ID), city ID, item ID, etc. To insert a new item into a corresponding key (e.g., second row of Table 1), the Redis ZADD command may be used. To query an item from the corresponding key, the ZRANGEBYSCORE command of Redis may be utilized.

83. The item recommendation service 422 may operate based a recommendation logic as follows. Upstream Backend Service 420 (e.g., a service provided by the backend module 264) may be configured to transmit user ID (e.g., ID associated with the user 404), session ID and context information (e.g. consumer's latitude, longitude, city ID) relating to the new session 408 to the Item Recommendation Service 422. The Item Recommendation Service 422 may be configured to send the user's key, which is based on user ID and context information, to the database

418. The database 418 may be configured to identify (e.g., from the data relating to the plurality of orders or selections made by the user 404 that is stored in the database 418) and return the first plurality of items that are selected or ordered by the user 404 by to the Item Recommendation Service 422 based on the user's key.

5 The Item Recommendation Service 422 may then send received information of the first plurality of items to a Similarity Search (SIMS) Cluster Model 424 (e.g., the clustering module 266) which can be deployed with various serving infrastructure such as Tensorflow Serving or other similar infrastructures. Based on the received information, the SIMS Cluster Model 424 may be configured to return a list of top
10 k recommended items to the Item Recommendation Service 422, which may then forward the information to the upstream Backend Service 420. These top k recommended items may then be recommended in real-time to the user during the new session 408.

15 **84.** Fig. 6 depicts an illustration 600 of the SIMS cluster model 424 in more detail according to various embodiments. The SIMS Cluster Model 424 may be developed by utilizing Tensorflow Framework. The steps of generating the cluster model is as follows. Item embeddings (e.g., item embedding 602) may be generated based on the second plurality of items that are collected in each session
20 (e.g., wherein a time duration of each session is 30 mins or other suitable time duration). The item embeddings may be trained based on, for example, the item2vec model with a word2vec function from a gensim package. The dimension of the item embedding 602 may be $N \times D$, where N is the number of items and, D is the length of each embedding. A k-means cluster algorithm may be used to
25 classify the item embedding 602 into different clusters. The number of items in a cluster i is denoted as N_i , e.g. N_1 for a cluster 1. During model serving, the first plurality of items are passed to the SIMS Cluster Model 424 as target items. The number of target items is denoted as T . Based on the item IDs associated with the target items, an item embedding 606 with dimension $T \times D$ for the target items can
30 be constructed.

85. In addition, corresponding clusters of each target item (e.g., item embedding clusters 608) may be found by using an index mapping from item to cluster. For example, there may be two target items (i.e. target items 1 and 2), wherein target
35 item 1 maps to a cluster 1 with N_1 items, and target item 2 maps to cluster 2 with

N2 items. After the mapping is done, the clusters of target items (i.e. clusters N1 and N2 for target items 1 and 2) may be constructed to form a cluster item embedding matrix. The dimension of the cluster item embedding matrix is $((N1 + N2) \times D)$. A cosmul method may be used to compute a score indicative of a measure of similarity for each cluster item (e.g., each of the one or more items of the second plurality of items that are mapped to each target item in a cluster). This method considers cosine similarity between cluster items and target items, then aggregates cosine similarity of target items by using product operation. At first, a dot product between cluster item embedding matrix $((N1 + N2) \times D)$ and target item embedding matrix $(T \times D)$ is computed to generate a similarity matrix 610 $((N1 + N2) \times T)$. Then, it does row product operation on the similarity matrix 610 $((N1 + N2) \times T)$ to generate similar item scores 612 $((N1 + N2) \times 1)$. Finally, a top k of the cluster items are selected by ranking the similar item scores 612.

86. As the item embeddings are split into different clusters, the number of items in a cluster is the key parameter to do tradeoff between model latency and model accuracy. Therefore, two experiments have been conducted to investigate the relation between number of items in a cluster and accuracy, and the relation between number of items in a cluster and latency, such as shown in Table 2 below.

Avg number of items in a cluster	Number of clusters	recall@10	recall@20	recall@50
100	5009	0.57601	0.54594	0.49035
1,000	501	0.87859	0.86799	0.84670
10,000	51	0.97919	0.97937	0.97715
100,000	6	0.98401	0.98723	0.98877

Table 2: Relation between number of items in a cluster and accuracy

87. To evaluate the model accuracy of the SIMS cluster model 424 with a different number of clusters, an experiment is set up with a total item number as 500879 ($N=500879$). In the SIMS Cluster Model 424, if the total item number (N) is fixed, as the number of items in a cluster increases, the number of clusters will decrease. With the average number of items in a cluster set to 100, 1000, 10000 and 100000, four SIMS cluster models with 5009, 501, 51 and 6 clusters are obtained. By using

the model with 1 cluster (500879 items in one cluster) as ground truth, recall@10, recall@20, recall@50 for the model with 6, 51, 501, 5009 clusters are computed as shown in the table 2 above. Accordingly, the higher the recall@k metrics, the better the algorithm performance. Based on Table 2, it can be concluded that it is still possible to get high accuracy even if the number of items in a cluster is 10,000.

- 88.** In the SIMS Cluster Model 242 with 1 CPU pod and with 200 Queries-per-second (QPS), as the number of items in a cluster is increased, the latency will also increase as shown in Table 3 below.

Number of items in a cluster	Number of clusters	P99 latency	P95 latency	P50 latency
100	5009	3	3	2
1,000	501	4	3	2
10,000	51	1190	793	276

Table 3: Relation between number of items in a cluster and latency

- 89.** According to the above table 3, it can be concluded that if the number of items in a cluster is increased to 1000, the model can achieve low latency and cost effectiveness.

- 90.** Fig. 7 illustrates an example flow diagram for a method 700 for recommending an item to a user according to various embodiments. In a step 702, a first plurality of items is identified, the first plurality of items being a plurality of items that are selected or ordered by a user, the identification based on data relating to a plurality of orders or selections made by the user. In a step 704, a score for each item of a second plurality of items is computed, the second plurality of items being a plurality of items that are selected or ordered by a plurality of users, the score indicating a measure of similarity between the first plurality of items and each item of the second plurality of items. In a step 706, one or more items of the second plurality of items are recommended to the user based on the computed score, the recommendation being made in real-time and in response to the plurality of orders or selections made by the user.

91. Fig. 8A depict an example computer system 1400, in accordance with which the recommendation server 140 described can be practiced. The computer system 1400 includes a computer module 1401. An external Modulator-Demodulator (Modem) transceiver device 1416 may be used by the computer module 1401 for communicating to and from a communications network 1420 via a connection 1421. The communications network 1420 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 1421 is a telephone line, the modem 1416 may be a traditional "dial-up" modem. Alternatively, where the connection 1421 is a high capacity (e.g., cable) connection, the modem 1416 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 1420.

92. The computer module 1401 typically includes at least one processor unit 1405, and a memory unit 1406. For example, the memory unit 1406 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 1401 also includes an interface 1408 for the external modem 1416. In some implementations, the modem 1416 may be incorporated within the computer module 1401, for example within the interface 1408. The computer module 1401 also has a local network interface 1411, which permits coupling of the computer system 1400 via a connection 1423 to a local-area communications network 1422, known as a Local Area Network (LAN). As illustrated in Fig. 8A, the local communications network 1422 may also couple to the wide network 1420 via a connection 1424, which would typically include a so-called "firewall" device or device of similar functionality. The local network interface 1411 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 1411.

93. The I/O interfaces 1408 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 1409 are provided and typically include a hard disk drive (HDD) 1410. Other storage devices such as a floppy disk drive and a

magnetic tape drive (not illustrated) may also be used. An optical disk drive 1412 is typically provided to act as a non-volatile source of data. Portable memory devices, such as optical disks, USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the system 1400.

94. The components 1405 to 1412 of the computer module 1401 typically communicate via an interconnected bus 1304 and in a manner that results in a conventional mode of operation of the computer system 1400 known to those in the relevant art. For example, the processor 1405 is coupled to the system bus 1404 using a connection 1418. Likewise, the memory 1406 and optical disk drive 1412 are coupled to the system bus 1404 by connections 1419. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple or like computer systems.

95. The method 700, where performed by the recommendation server 140 may be implemented using the computer system 1400. The processes may be implemented as one or more software application programs 1433 executable within the computer system 1400. In particular, the method 700 are effected by instructions in the software 1433 that are carried out within the computer system 1400. The software instructions may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the method 700 and a second part and the corresponding code modules manage a user interface between the first part and the user.

96. The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the computer system 1400 from the computer readable medium, and then executed by the computer system 1400. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product. The use of the computer program product in the computer system 1400 preferably effects an advantageous apparatus for a recommendation server 140.

5 **97.** The software 1433 is typically stored in the HDD 1410 or the memory 1406. The software is loaded into the computer system 1400 from a computer readable medium, and executed by the computer system 1400. Thus, for example, the software 1433 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 1425 that is read by the optical disk drive 1412. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the computer system 1400 preferably effects an apparatus for a recommendation server 140.

10 **98.** In some instances, the application programs 1433 may be supplied to the user encoded on one or more CD-ROMs 1425 and read via the corresponding drive 1412, or alternatively may be read by the user from the networks 1420 or 1422. Still further, the software can also be loaded into the computer system 1400 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the computer system 1400 for execution and/or processing. Examples of such storage media include floppy disks, magnetic tape, optical disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 1401. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 1401 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

30 **99.** The second part of the application programs 1433 and the corresponding code modules mentioned above may be executed to implement one or more graphical user interfaces (GUIs) to be rendered or otherwise represented upon a display. Through manipulation of typically a keyboard and a mouse, a user of the computer system 1400 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech

prompts output via loudspeakers and user voice commands input via a microphone.

100. It is to be understood that the structural context of the computer system 1400 (i.e., the recommendation server 140) is presented merely by way of example. Therefore, in some arrangements, one or more features of the computer system 1400 may be omitted. Also, in some arrangements, one or more features of the computer system 1400 may be combined together. Additionally, in some arrangements, one or more features of the computer system 1400 may be split into one or more component parts.

101. Fig. 9 shows an implementation of the transaction processing server 108 (i.e., the computer system 1300). In this implementation, the transaction processing server 108 may be generally described as a physical device comprising at least one processor 802 and at least one memory 804 including computer program codes. The at least one memory 804 and the computer program codes are configured to, with the at least one processor 802, cause the transaction processing server 108 to facilitate the operations described in method 700. The transaction processing server 108 may also include a transaction processing module 806. The memory 804 stores computer program code that the processor 802 compiles to have transaction processing module 806 perform the respective functions.

102. With reference to Fig. 1, the transaction processing module 806 performs the function of communicating with the requestor device 102 and the provider device 104; and the acquirer server 106 and the issuer server 110 to respectively receive and transmit a transaction for ordering an item, a request for a ride or delivery, or other similar messages. The transaction processing module 806 may be configured to process processes relating to a transaction account by, for example, forwarding data and information associated with the transaction to the other servers in the system 100 such as the recommendation server 140. In an example, the transaction processing server 108 may, instead of the requestor device 102 or provider device 104, transmit data relating to a request message such as a transaction request for ordering an item (e.g. item ID, contextual information such as location of the user based on a city ID, user ID, date of order, time of order, and other similar data) to the recommendation server 104. The transaction

processing server 108 may use a variety of different protocols and procedures in order to process the payment and/or travel co-ordination requests. The transaction processing server may use a variety of different protocols and procedures in order to process the payment and/or travel co-ordination requests.

5

103. Fig. 10 shows an alternative implementation of the recommendation server 140 (i.e., the computer system 1400). In the alternative implementation, recommendation server 140 may be generally described as a physical device comprising at least one processor 902 and at least one memory 904 including computer program codes. The at least one memory 904 and the computer program codes are configured to, with the at least one processor 902, cause the recommendation server 140 to perform the operations described in the method 700. The recommendation server 140 may also include a data module 906, an injection module 908, a backend module 910, a clustering module 912 and a recommendation module 914. The memory 904 stores computer program code that the processor 902 compiles to have each of the modules 906 to 914 perform their respective functions.

10

15

20

104. With reference to Figs. 1 to 7, the injection module 908 performs the function of injecting events (e.g., a plurality of selections or orders made by a user) incrementally to the database 150.

25

105. With reference to Figs. 1 to 7, the backend module 910 performs the function of retrieving user ID, session ID (e.g., an identifier associated with a session) and context information (e.g., a user's latitude, longitude, city ID, and other similar information) based on a current session in which a user is interacting with the platform.

30

106. With reference to Figs. 1 to 7, the clustering module 912 performs the function of generating an item embedding based on the historical data and training the item embedding based on an item2vec model. The clustering module 266 may be further configured for computing a score for each item of the second plurality of items that are selected or ordered by a plurality of users, the score indicating a measure of similarity between the first plurality of items and each item of the second plurality of items. Computing the score may further comprise: generating,

35

for each respective item of the first plurality of items, a respective first cluster comprising a mapping of the respective item with one or more other items from the second plurality of items that are similar to the respective item; generating a second cluster comprising a mapping of the first plurality of items; and computing the score for each of the one or more other items, the computation based on a processing of the plurality of first clusters and the second cluster. The processing may further comprise: generating a first matrix based on the plurality of first clusters, and a second matrix based on the second cluster; computing a dot product between the first and second matrix to generate a similarity matrix; and performing row product operation on the similarity matrix to obtain the computed scores.

107. With reference to Figs. 1 to 7, the recommendation module 914 performs the function of identifying the first plurality of items that are selected or ordered by a user, the identification based on the data relating to the plurality of orders or selections made by the user. The recommendation module 268 may be further configured for recommending one or more items of the second plurality of items to the user based on the computed score, the recommendation being made in real-time and in response to the plurality of orders or selections made by the user.

108. With reference to Figs. 1 to 7, the data module 906 performs the functions of receiving data and information from the requestor device 102, provider device 104, transaction processing server 108, reference database 150, a cloud and other sources of information to recommend an item to a user. For example, the data module 260 may be configured to receive data and information required for processing real-time data stream, processing historical data, generate and train item embeddings, perform clustering, perform similar search, and other similar processes from the requestor device 102, the provider device 104, transaction processing server 108, reference database 150, and/or other sources of information. The data module 260 may be further configured to send information relating to data retrieved in response to selections or orders made by a user to the requestor device 102, the provider device 104, the transaction processing server 108, or other destinations where the information is required.

109. Fig. 8B depicts a general-purpose computer system 1500, upon which a combined transaction processing server 108 and recommendation server 140 described can be practiced. The computer system 1500 includes a computer module 1501. An external Modulator-Demodulator (Modem) transceiver device 1516 may be used by the computer module 1501 for communicating to and from a communications network 1520 via a connection 1521. The communications network 1520 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 1521 is a telephone line, the modem 1516 may be a traditional “dial-up” modem. Alternatively, where the connection 1521 is a high capacity (e.g., cable) connection, the modem 1516 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 1520.
110. The computer module 1501 typically includes at least one processor unit 1505, and a memory unit 1506. For example, the memory unit 1506 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 1501 also includes an interface 1508 for the external modem 1516. In some implementations, the modem 1516 may be incorporated within the computer module 1501, for example within the interface 1508. The computer module 1501 also has a local network interface 1511, which permits coupling of the computer system 1500 via a connection 1523 to a local-area communications network 1522, known as a Local Area Network (LAN). As illustrated in Fig. 8D, the local communications network 1522 may also couple to the wide network 1520 via a connection 1524, which would typically include a so-called “firewall” device or device of similar functionality. The local network interface 1511 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 1511.
111. The I/O interfaces 1508 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 1509 are provided and typically include a hard disk

drive (HDD) 1510. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used. An optical disk drive 1512 is typically provided to act as a non-volatile source of data. Portable memory devices, such optical disks, USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the system 1500.

112. The components 1505 to 1512 of the computer module 1501 typically communicate via an interconnected bus 1504 and in a manner that results in a conventional mode of operation of the computer system 1500 known to those in the relevant art. For example, the processor 1505 is coupled to the system bus 1504 using a connection 1518. Likewise, the memory 1506 and optical disk drive 1512 are coupled to the system bus 1504 by connections 1519. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple or like computer systems.

113. The steps of the method 700 performed by the recommendation server 140 and facilitated by the transaction processing server 108 may be implemented using the computer system 1500. For example, the steps of the method 700 as performed by the recommendation server 140 may be implemented as one or more software application programs 1533 executable within the computer system 1500. In particular, the steps of the method 700 are effected by instructions in the software 1533 that are carried out within the computer system 1500. The software instructions may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the steps of the method 700 and a second part and the corresponding code modules manage a user interface between the first part and the user.

114. The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the computer system 1500 from the computer readable medium, and then executed by the computer system 1500. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer

program product. The use of the computer program product in the computer system 1500 preferably effects an advantageous apparatus for a combined transaction processing and recommendation server.

- 5 **115.** The software 1533 is typically stored in the HDD 1510 or the memory 1506. The software is loaded into the computer system 1500 from a computer readable medium, and executed by the computer system 1500. Thus, for example, the software 1533 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 1525 that is read by the optical disk drive 1512. A computer readable
10 medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the computer system 1500 preferably effects an apparatus for a combined transaction processing and recommendation server.
- 15 **116.** In some instances, the application programs 1533 may be supplied to the user encoded on one or more CD-ROMs 1525 and read via the corresponding drive 1512, or alternatively may be read by the user from the networks 1520 or 1522. Still further, the software can also be loaded into the computer system 1500 from other computer readable media. Computer readable storage
20 media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the computer system 1500 for execution and/or processing. Examples of such storage media include floppy disks, magnetic tape, optical disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and
25 the like, whether or not such devices are internal or external of the computer module 1501. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 1501 include radio or infra-red transmission channels as well as a network connection
30 to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.
- 35 **117.** The second part of the application programs 1533 and the corresponding code modules mentioned above may be executed to implement one or more graphical user interfaces (GUIs) to be rendered or otherwise represented upon a display.

Through manipulation of typically a keyboard and a mouse, a user of the computer system 1500 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech prompts output via loudspeakers and user voice commands input via a microphone.

118. It is to be understood that the structural context of the computer system 1500 (i.e., combined transaction processing and recommendation server 1500) is presented merely by way of example. Therefore, in some arrangements, one or more features of the server 1500 may be omitted. Also, in some arrangements, one or more features of the server 1500 may be combined together. Additionally, in some arrangements, one or more features of the server 1500 may be split into one or more component parts.

119. Fig. 11 shows an alternative implementation of combined transaction processing and recommendation server (i.e., the computer system 1500). In the alternative implementation, the combined transaction processing and recommendation server may be generally described as a physical device comprising at least one processor 1002 and at least one memory 904 including computer program codes. The at least one memory 1004 and the computer program codes are configured to, with the at least one processor 1002, cause the combined transaction processing and recommendation server to perform the operations described in the steps of the method 700. The combined transaction processing and recommendation server may also include a transaction processing module 806, a data module 906, an injection module 908, a backend module 910, a clustering module 912 and a recommendation module 914. The memory 1004 stores computer program code that the processor 1002 compiles to have each of the modules 806 to 912 performs their respective functions. The transaction processing module 806 performs the same functions as described for the same transaction processing module in Fig. 9. The data module 906, injection module 908, backend module 910, clustering module 912 and recommendation module 914 perform the same functions as described for the same corresponding modules in Fig. 10.

- 5 **120.** It will be appreciated by a person skilled in the art that numerous variations and/or modifications may be made to the present disclosure as shown in the specific embodiments without departing from the scope of the specification as broadly described. The present embodiments are, therefore, to be considered in all respects to be illustrative and not restrictive.

CLAIMS

What is claimed is:

1. A method for recommending an item to a user, comprising:
 - 5 identifying, by a server, a first plurality of items, the first plurality of items being a plurality of items that are selected or ordered by a user, the identification based on data relating to a plurality of orders or selections made by the user;
 - computing, by the server, a score for each item of a second plurality of items, the second plurality of items being a plurality of items that are selected or
10 ordered by a plurality of users, the score indicating a measure of similarity between the first plurality of items and each item of the second plurality of items; and
 - recommending, by the server, one or more items of the second plurality of items to the user based on the computed score, the recommendation being made in real-time and in response to the plurality of orders or selections made by the
15 user.
2. The method of claim 1, further comprising:
 - receiving, in real time when the second plurality of items are ordered or selected, data relating to the second plurality of items, the data indicating a
20 respective time at which each item of the second plurality of items is selected or ordered, and indicating a respective user identifier associated with each item; and
 - storing the received data in a database as historical data, the historical data being sorted based on the indicated time and user identifier.
- 25 3. The method of claim 2, further comprising:
 - generating an item embedding based on the historical data; and
 - training the item embedding based on an item2vec model.
4. The method of any one of the preceding claims, further comprising:
 - 30 receiving, in real time when the first plurality of items are ordered or selected, the data in a real time stream when the plurality of orders or selections are made by the user, the data indicating a respective time at which each item of the first plurality of items is selected or ordered, and indicating a respective user identifier associated with each item; and

storing the received data in the database, the stored data being sorted based on the indicated time and user identifier.

5. The method of claim 4, wherein identifying the first plurality of items further comprises:

identifying, from the stored data, the first plurality of items that are selected or ordered by the user based on a user identifier associated with the user.

6. The method of claim 4, further comprising:

updating the historical data based on the stored data;

generating an updated item embedding based on the updated historical data; and

training the updated item embedding based on an item2vec model.

7. The method of claim 2, wherein computing the score further comprises:

generating, for each respective item of the first plurality of items, a respective first cluster comprising a mapping of the respective item with one or more other items from the second plurality of items that are similar to the respective item;

generating, by the server, a second cluster comprising a mapping of the first plurality of items; and

computing the score for each of the one or more other items, the computation based on a processing of the plurality of first clusters and the second cluster.

8. The method of claim 7, wherein the processing further comprises:

generating a first matrix based on the plurality of first clusters, and a second matrix based on the second cluster;

computing a dot product between the first and second matrix to generate a similarity matrix; and

performing row product operation on the similarity matrix to obtain the computed scores.

9. The method of any one of the preceding claims, wherein recommending the one or more items further comprises ranking each item of the one or more other items based on the computed score, and selecting the one or more items based on the ranking.

- 5 10. A system for recommending an item to a user, comprising:
at least one processor; and
at least one memory including computer program code;
the at least one memory and the computer program code configured to,
with the at least one processor, cause the system at least to:
10 identify a first plurality of items, the first plurality of items being a plurality of items that are selected or ordered by a user, the identification based on data relating to a plurality of orders or selections made by the user;
compute a score for each item of a second plurality of items, the second plurality of items being a plurality of items that are selected or ordered by a plurality
15 of users, the score indicating a measure of similarity between the first plurality of items and each item of the second plurality of items; and
recommend one or more items of the second plurality of items to the user based on the computed score, the recommendation being made in real-time and in response to the plurality of orders or selections made by the user.
20
11. The system of claim 10, further configured to:
receive, in real time when the second plurality of items are ordered or selected, data relating to the second plurality of items, the data indicating a
25 respective time at which each item of the second plurality of items is selected or ordered, and indicate a respective user identifier associated with each item; and
store the received data in a database as historical data, the historical data being sorted based on the indicated time and user identifier.
12. The system of claim 11, further configured to:
30 generate an item embedding based on the historical data; and
train the item embedding based on an item2vec model.
13. The system of any one of claims 10 - 12, further configured to:
35 receive, in real time when the first plurality of items are ordered or selected, the data in a real time stream when the plurality of orders or selections are made

by the user, the data indicating a respective time at which each item of the first plurality of items is selected or ordered, and indicating a respective user identifier associated with each item; and

store the received data in the database, the stored data being sorted based on the indicated time and user identifier.

14. The system of claim 13, wherein identifying the first plurality of items further comprises:

identifying, from the stored data, the first plurality of items that are selected or ordered by the user based on a user identifier associated with the user.

15. The system of claim 13, further configured to:

update the historical data based on the stored data;

generate an updated item embedding based on the updated historical data;

and

train the updated item embedding based on an item2vec model.

16. The system of claim 11, wherein computing the score further comprises:

generating, for each respective item of the first plurality of items, a respective first cluster comprising a mapping of the respective item with one or more other items from the second plurality of items that are similar to the respective item;

generating a second cluster comprising a mapping of the first plurality of items; and

computing the score for each of the one or more other items, the computation based on a processing of the plurality of first clusters and the second cluster.

17. The system of claim 16, wherein the processing further comprises:

generating a first matrix based on the plurality of first clusters, and a second matrix based on the second cluster;

computing a dot product between the first and second matrix to generate a similarity matrix; and

performing row product operation on the similarity matrix to obtain the computed scores.

18. The system of any one of claims 10 - 17, wherein recommending the one or more items further comprises ranking each item of the one or more other items based on the computed score, and selecting the one or more items based on the ranking.