

**SERVER AND METHOD FOR DETERMINING FARE FOR ON-DEMAND
SERVICE**

TECHNICAL FIELD

[0001] Various embodiments relate to a server and a method for determining a fare for an on-demand service.

BACKGROUND

[0002] Due to development of information technology, a user (who in some contexts herein may also be referred to as a “requester”, a “consumer”, a “passenger”, a “Pax” or a “customer”) may request an on-demand service using a computing device. The on-demand service may allow the user to fulfil the user’s demand via an immediate access to goods and/or services. The user may request the on-demand service, for example, a delivery service or a transport service (also referred to as an “e-hailing service”), using a user interface presented on the computing device. To request the on-demand service, the user may make an order (also referred to as a “booking”) for the on-demand service. On or before making the order, the user may send a request for a fare-check for the order, to check an estimated fare (for example, a delivery fare) for the on-demand service and determine whether to proceed to make the order considering the estimated fare.

[0003] FIG. 1 is an exemplary diagram showing a conventional user interface 160a of the fare-check for the order for the on-demand service. There may be a plurality of delivery options, for example, three delivery options 162, based on a speed of the delivery. As shown

in FIG. 1, after the request for the fare-check for the order, the three delivery options 162 may be presented on a booking page, as follows:

- a) An option of a priority delivery at a priority delivery fare 163 (also referred to as a “priority option”);
- b) An option of a standard delivery at a standard delivery fare 164 (which may be a generic product of an on-demand service platform provider) (also referred to as a “standard option”); and
- c) An option of a saver delivery at a saver delivery fare 165 (which may be a lower fare for a slower delivery on the user’s side and a higher batch rate on the on-demand service platform provider’s side) (also referred to as a “saver option”).

[0004] As shown in FIG. 1, the saver option may set a static delay (for example, 20 minutes) and a static discount (for example, \$1 off standard delivery fare). Since the saver option may be targeted at potentially highly elastic users, it may be beneficial to determine the saver delivery fare dynamically as the users’ willingness to pay may change more given different contexts (for example, peak hour or tea time, different delivery delay, etc.).

[0005] The dynamic discount may leverage changing market dynamics as well as potentially a variable delay component to the saver option. For example, a lunch peak discount may need to be different from an afternoon discount, as overall time efficiency (OTE) of a lunch time order is likely to be higher than that of a tea time order. However, there may be a risk that users who have preferred the standard option may change their preference to the saver option, if the saver delivery fare is discounted too much.

[0006] Therefore, there is a need to provide a solution for facilitating determining a fare for the saver option, without sacrificing book-through rates (BTR) and gross merchandise values (GMV) of the other options, for example, the priority option and the standard option, too much, and in an auto-adaptive way.

SUMMARY

[0007] According to various embodiments, there is a server for facilitating determining a fare for an on-demand service, the server comprising: a memory configured to store instructions; a communication interface configured to receive a request for a fare-check for an order for the on-demand service from a computing device; and a processor for executing the stored instructions and configured to: predict a first book-through rate (BTR) for a first delivery option and a second BTR for a second delivery option for the order, for each of a plurality of discount options for the fare for the second delivery option, wherein the second delivery option offers a slower delivery at a lower fare than the first delivery option; compute at least one of an expected revenue and an expected profit for the order for each of the plurality of discount options, based on the first BTR and the second BTR; select a discount option from the plurality of discount options, based on the at least one of the expected revenue and the expected profit; and determine the fare for the second delivery option based on the selected discount option.

[0008] In some embodiments, the processor is further configured to, in response to the request, provide the computing device with information about the fare for the second delivery option based on the selected discount option.

[0009] In some embodiments, the processor is further configured to build a BTR prediction model based on historical data.

[0010] In some embodiments, the processor is further configured to predict the first BTR for the first delivery option and the second BTR for the second delivery option for the order for each of the plurality of discount options, using the BTR prediction model.

[0011] In some embodiments, the processor is further configured to: obtain information about a budget for a discount of the fare for the second delivery option; select the discount option from the plurality of discount options, further based on the budget, using an optimisation module; and output, by the optimisation module, the selected discount option.

[0012] In some embodiments, the processor is further configured to train a post mapping model based on functions of the BTR prediction model and the selected discount option output by the optimisation module.

[0013] In some embodiments, the processor is further configured to use the post mapping model, instead of the optimisation module, to select the discount option from the plurality of discount options, in an offline simulation.

[0014] In some embodiments, the processor is further configured to: upon receipt of the request, query an inter-arrival rate (IAR) signal; predict overall time efficiency (OTE) based on the IAR signal; and select the discount option from the plurality of discount options, further based on the predicted OTE.

[0015] In some embodiments, the processor is further configured to predict the OTE, further based on a batching configuration.

[0016] In some embodiments, the processor is further configured to select the discount option from the plurality of discount options, further based on a fare for the first delivery option.

[0017] According to various embodiments, there is a method for facilitating determining a fare for an on-demand service, the method comprising: receiving a request for a fare-check for an order for the on-demand service from a computing device; predicting a first book-through rate (BTR) for a first delivery option and a second BTR for a second delivery option for the order, for each of a plurality of discount options for the fare for the second delivery option, wherein the second delivery option offers a slower delivery at a lower fare than the first delivery option; computing at least one of an expected revenue and an expected profit for

the order for each of the plurality of discount options, based on the first BTR and the second BTR; selecting a discount option from the plurality of discount options, based on the at least one of the expected revenue and the expected profit; and determining the fare for the second delivery option based on the selected discount option.

[0018] In some embodiments, the method further comprises: in response to the request, providing the computing device with information about the fare for the second option based on the selected discount option.

[0019] In some embodiments, the method further comprises: building a BTR prediction model based on historical data.

[0020] In some embodiments, the method further comprises: predicting the first BTR for the first delivery option and the second BTR for the second delivery option for the order for each of the plurality of discount options, using the BTR prediction model.

[0021] In some embodiments, the method further comprises: obtaining information about a budget for a discount of the fare for the second delivery option; selecting the discount option from the plurality of discount options, further based on the budget, using an optimisation module; and outputting, by the optimisation module, the selected discount option.

[0022] In some embodiments, the method further comprises: training a post mapping model based on functions of the BTR prediction model and the selected discount option output by the optimisation module.

[0023] In some embodiments, the method further comprises: using the post mapping model, instead of the optimisation module, to select the discount option from the plurality of discount options, in an offline simulation.

[0024] In some embodiments, the method further comprises: upon receipt of the request, querying an inter-arrival rate (IAR) signal; predicting overall time efficiency (OTE) based on

the IAR signal; and selecting the discount option from the plurality of discount options, further based on the predicted OTE.

[0025] In some embodiments, the method further comprises: predicting the OTE, further based on a batching configuration.

[0026] In some embodiments, the method further comprises: selecting the discount option from the plurality of discount options, further based on a fare for the first delivery option.

[0027] According to various embodiments, a data processing apparatus configured to perform the method of any one of the above embodiments is provided.

[0028] According to various embodiments, a computer program element comprising program instructions, which, when executed by one or more processors, cause the one or more processors to perform the method of any one of the above embodiments is provided.

[0029] According to various embodiments, a computer-readable medium comprising program instructions, which, when executed by one or more processors, cause the one or more processors to perform the method of any one of the above embodiments is provided. The computer-readable medium may include a non-transitory computer-readable medium.

BRIEF DESCRIPTION OF THE DRAWINGS

[0030] The invention will be better understood with reference to the detailed description when considered in conjunction with the non-limiting examples and the accompanying drawings, in which:

- FIG. 1 is an exemplary diagram showing a conventional user interface of a fare-check for an order for an on-demand service.
- FIGS. 2 and 3 illustrate infrastructures of a system including a server for determining a fare for an on-demand service according to various embodiments.

- FIG. 4 illustrates a block diagram of a server for determining a fare for an on-demand service according to various embodiments.
- FIG. 5 illustrates a flowchart for a method for determining a fare for an on-demand service according to various embodiments.
- FIG. 6 illustrates a data flow diagram of a server for determining a fare for an on-demand service according to various embodiments.
- FIG. 7 illustrates a data flow diagram of a server for determining a fare for an on-demand service according to various embodiments.
- FIGS. 8 and 9 are exemplary diagrams showing a user interface of a fare-check for an order for an on-demand service according to various embodiments.

DETAILED DESCRIPTION

[0031] The following detailed description refers to the accompanying drawings that show, by way of illustration, specific details and embodiments in which the disclosure may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the disclosure. Other embodiments may be utilized and structural, and logical changes may be made without departing from the scope of the disclosure. The various embodiments are not necessarily mutually exclusive, as some embodiments can be combined with one or more other embodiments to form new embodiments.

[0032] Embodiments described in the context of one of a server and a method are analogously valid for the other server and method. Similarly, embodiments described in the context of a server are analogously valid for a method, and vice-versa.

[0033] 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.

[0034] 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.

[0035] As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0036] Throughout the description, the term “module” may be understood as an application specific integrated circuit (ASIC), an electronic circuit, a combinational logic circuit, a field programmable gate array (FPGA), a processor which executes code, other suitable hardware components which provide the described functionality, or any combination thereof. The term of “module” may include a memory which stores code executed by the processor.

[0037] In the following, embodiments will be described in detail.

[0038] FIGS. 2 and 3 illustrate infrastructures of a system 200 including a server 100 for providing information for an on-demand service according to various embodiments.

[0039] As shown in FIG. 2, the system 200 may include, but is not limited to, the server 100, a database system 140, a network 150, a plurality of first computing devices 160 each associated with a plurality of users 161 (who in some contexts herein may also be referred to as a “requester”, a “consumer”, a “passenger”, a “Pax” or a “customer”), a plurality of second computing devices 170 (not shown) each associated with a plurality of delivery service providers 171 (who in some contexts herein may also be referred to as a “driver”, a “driver partner”, a “Dax”, a “delivery partner” or a “delivery agent”), and a plurality of third computing devices 180 (not shown) each associated with a plurality of item providers 181

(who in some contexts herein may also be referred to as a “merchant”, a “Mex”, a “food provider” or a “restaurant”). In some embodiments, the users 161 may include a consumer (who in some contexts herein may also be referred to as an “eater” or a “passenger”) for the on-demand service. For example, a user 161a may be the same as the consumer. As another example, the user 161a may be different from the consumer and use the on-demand service for and/or on behalf of the consumer.

[0040] In some embodiments, the on-demand service may be a service allowing the user 161a to fulfil the user’s demand via an immediate access to items and/or services. The user 161a may request the on-demand service, such as a transport service (also referred to as an “e-hailing service”) or an item delivery service, using a user interface presented on a computing device 160a. The user 161a may make an order for the on-demand service.

[0041] In some embodiments, the user 161a may use an application, for example, a mobile application, provided by a server 100. For example, the server 100 may be controlled and/or managed by an on-demand service platform provider. The application may be installed in the first computing device 160a associated with the user 161a, to interact with the on-demand service.

[0042] In some embodiments, the network 150 may include, but is not limited to, a Local Area Network (LAN), a Wide Area Network (WAN), a Global Area Network (GAN), or any combination thereof. The network 150 may provide a wireline communication, a wireless communication, or a combination of the wireline and wireless communication between the server 100 and the plurality of first computing devices 160, between the server 100 and the plurality of second computing devices 170, and between the server 100 and the plurality of third computing devices 180. As shown in FIG. 3, the network 150 may provide the wireline communication, the wireless communication, or the combination of the wireline and wireless communication between the first computing device 160a of the plurality of first computing

devices 160 and a second computing device 170a of the plurality of second computing devices 170.

[0043] In some embodiments, the plurality of first computing devices 160 may be connectable to the server 100 via the network 150. In some embodiments, the plurality of first computing devices 160 may be arranged in data or signal communication with the server 100 via the network 150. In some embodiments, the plurality of first computing devices 160 may include, but is not limited to, at least one of the following: a mobile phone, a tablet computer, a laptop computer, a desktop computer, a head-mounted display and a smart watch. In some embodiments, the plurality of first computing devices 160 may be associated with the plurality of users 161 respectively. For example, the plurality of first computing devices 160 may belong to the plurality of users 161 respectively. For example, the first computing device 160a may belong to the user 161a who is the consumer. As another example, the first computing device 160a may belong to the user 161a requesting the delivery of the item to the consumer who is a recipient of the on-demand service.

[0044] In some embodiments, the first computing device 160a may include a location sensor. In some embodiments, the location sensor may communicate with at least one of a global positioning satellite (GPS) server, a network server, and a Wi-Fi server, to detect a location of the first computing device 160a. In some embodiments, the first computing device 160a may generate information about the location of the first computing device 160a.

[0045] In some embodiments, the server 100, for example, implemented by a server computer, may include a communication interface 110, a processor 120, and a memory 130 (as will be described with reference to FIG. 4).

[0046] In some embodiments, the server 100 may communicate with the plurality of first computing devices 160 via the network 150. In some embodiments, the user 161a may request the on-demand service, for example, a delivery service or a transport service, using a

user interface presented on the first computing device 160a. In some embodiments, to request the on-demand service, the user 161a may make an order (also referred to as a “booking”) for the on-demand service. In some embodiments, the first computing device 160a may receive the order from the user 161a for the on-demand service. The first computing device 160a may send the order to the server 100 via the network 150. In some embodiments, the first computing device 160a may send the information about the location of the first computing device 160a to the server 100 via the network 150. The location of the first computing device 160a may be considered as a location of the user 161a. In some embodiments, the location of the user 161a may be considered as a destination of the on-demand service. In some other embodiments, the first computing device 160a may send information about an address of the user 161a, and the address of the user 161a may be considered as the destination of the on-demand service. In some other embodiments, the first computing device 160a may send information about an address of the recipient of the delivery, and the address of the recipient may be considered as the destination of the on-demand service.

[0047] In some embodiments, on or before making the order, the user 161a may send a request for a fare-check for the order, to check an estimated fare (for example, a delivery fare) for the on-demand service and determine whether to proceed to make the order considering the estimated fare. In some embodiments, the first computing device 160a may receive the request for the fare-check from the user 161a. The first computing device 160a may send the request for the fare-check to the server 100 via the network 150. In some embodiments, the first computing device 160a may send the information about the location of the first computing device 160a to the server 100 via the network 150. The location of the first computing device 160a may be considered as the location of the user 161a. In some embodiments, the location of the user 161a may be considered as the destination of the on-demand service. In some other embodiments, the first computing device 160a may send the

information about the address of the user 161a, and the address of the user 161a may be considered as the destination of the on-demand service. In some other embodiments, the first computing device 160a may send information about the address of the recipient of the delivery, and the address of the recipient may be considered as the destination of the on-demand service. In some embodiments, in response to the request for the fare-check, the server 100 may send information about the fare (i.e. estimated fare) for the order for the on-demand service to the first computing device 160a via the network 150. In some embodiments, the first computing device 160a may receive the information about the fare for the order from the server 100 via the network 150, and display the information about the fare for the order on the user interface presented on the first computing device 160a.

[0048] In some embodiments, the system 200 may further include a database 141. In some embodiments, the database 141 may be a part of the database system 140 which may be external to the server 100. The server 100 may communicate with the database 141. In some other embodiments, although not shown, the database 141 may be implemented locally in the memory 130 of the server 100.

[0049] In some embodiments, the server 100 may communicate with the plurality of second computing devices 170 via the network 150. In some embodiments, the plurality of second computing devices 170 may be arranged in data or signal communication with the server 100 via the network 150. In some embodiments, the plurality of second computing devices 170 may include, but is not limited to, at least one of the following: a mobile phone, a tablet computer, a laptop computer, a desktop computer, a head-mounted display and a smart watch. In some embodiments, the plurality of second computing devices 170 may be associated with the plurality of delivery service providers 171 respectively. For example, the plurality of second computing devices 170 may belong to the plurality of delivery service providers 171 respectively.

[0050] In some embodiments, the server 100 may receive the order from the first computing device 160a. After the server 100 receives the order from the first computing device 160a, the server 100 may allocate (assign) the order to a suitable delivery service provider 171a. In some embodiments, the second computing device 170a associated with the delivery service provider 171a may send information about a location of the second computing device 170a to the server 100 via the network 150. The location of the second computing device 170a may be considered as a location of the delivery service provider 171a. In some embodiments, the location of the delivery service provider 171a may be considered as a current location of the delivery service provider 171a, and may change while the delivery service provider 171a moves to the destination (en route to the destination). In some embodiments, the server 100 may provide the second computing device 170a with a map relating to a route from the current location of the second computing device 170a (which may be considered as the location of the delivery service provider 171a) to the destination for providing the on-demand service, en route to the destination.

[0051] In some embodiments, the server 100 may communicate with the plurality of third computing devices 180 via the network 150. In some embodiments, the plurality of third computing devices 180 may be arranged in data or signal communication with the server 100 via the network 150. In some embodiments, the plurality of third computing devices 180 may include, but is not limited to, at least one of the following: a mobile phone, a tablet computer, a laptop computer, a desktop computer, a head-mounted display and a smart watch. In some embodiments, the plurality of third computing devices 180 may be associated with the plurality of item providers 181 respectively. For example, the plurality of third computing devices 180 may belong to the plurality of item providers 181 respectively.

[0052] FIG. 4 illustrates a block diagram of a server 100 for providing information for an on-demand service according to various embodiments.

[0053] As shown in FIG. 4, the server 100, for example, implemented by a server computer, may include a communication interface 110, a processor 120, and a memory 130.

[0054] In some embodiments, the memory 130 (also referred to as a “database”) may store input data and/or output data temporarily or permanently. In some embodiments, the memory 130 may be configured to store instructions. In some embodiments, the memory 130 may store program code which allows the server 100 to perform a method 300 (as will be described with reference to FIG. 5). In some embodiments, the program code may be embedded in a Software Development Kit (SDK). The memory 130 may include an internal memory of the server 100 and/or an external memory. The external memory may include, but is not limited to, an external storage medium, for example, a memory card, a flash drive, and a web storage.

[0055] In some embodiments, the communication interface 110 may allow a plurality of first computing devices 160 to communicate with the processor 120 of the server 100 via the network 150, as shown in FIGS. 2 and 3. As shown in FIGS. 2 and 3, each of the plurality of first computing devices 160 may belong to each of users 161 who want to make an order for an on-demand service. In some embodiments, the communication interface 110 may transmit signals to the plurality of first computing devices 160, and/or receive signals from the plurality of first computing devices 160, via the network 150. For example, as shown in FIGS. 2 and 3, a first computing device 160a may belong to a user 161a who wants to make an order for an on-demand service, and the communication interface 110 may transmit signals to the first computing device 160a, and/or receive signals from the first computing device 160a via the network 150.

[0056] In some embodiments, the communication interface 110 may allow a plurality of second computing devices 170 to communicate with the processor 120 of the server 100 via the network 150, as shown in FIGS. 2 and 3. As shown in FIGS. 2 and 3, each of the plurality

of second computing devices 170 may belong to each of a plurality of delivery service providers 171 who may pick up an item from an item provider 181a and deliver the item to the consumer (i.e. a destination) and/or who may transport the consumer to the destination. In some embodiments, the communication interface 110 may transmit signals to the plurality of second computing devices 170, and/or receive signals from the plurality of second computing devices 170, via the network 150.

[0057] In some embodiments, the communication interface 110 may allow a plurality of third computing devices 180 to communicate with the processor 120 of the server 100 via the network 150, as shown in FIG. 2. As shown in FIG. 2, each of the plurality of third computing devices 180 may belong to each of a plurality of item providers 181 who may prepare an item, for example, food, for the order. In some embodiments, the communication interface 110 may transmit signals to the plurality of third computing devices 180, and/or receive signals from the plurality of third computing devices 180, via the network 150.

[0058] The processor 120 may include, but is not limited to, a microprocessor, an analogue circuit, a digital circuit, a mixed-signal circuit, a logic circuit, an integrated circuit, a Central Processing Unit (CPU), a Graphics Processing Unit (GPU), a Digital Signal Processor (DSP), a Field Programmable Gate Array (FPGA), an Application Specific Integrated Circuit (ASIC), 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 the processor 120.

[0059] In some embodiments, the processor 120 may be connectable to the communication interface 110. In some embodiments, the processor 120 may be arranged in data or signal communication with the communication interface 110 to transmit/receive the signals.

[0060] In some embodiments, the communication interface 110 may receive a plurality of orders (also referred to as “a plurality of bookings”) for the on-demand service from the plurality of first computing devices 160 each associated with the plurality of users 161 (who

in some contexts herein may also be referred to as “consumers 161” or “requesters 161”). In some embodiments, the processor 120 may receive the plurality of orders for the on-demand service from the communication interface 110. For example, the communication interface 110 may receive an order (also referred to as a “booking”) for the on-demand service from the first computing device 160a associated with the user 161a (who in some contexts herein may also be referred to as a “consumer 161a” or a “requester 161a”), and the processor 120 may receive the order for the on-demand service from the communication interface 110.

[0061] In some embodiments, on or before making the order, the first computing device 160a may send a request for a fare-check for the order to the communication interface 110, so that the user 161a may check an estimated fare (for example, a delivery fare) for the on-demand service and determine whether to proceed to make the order considering the estimated fare. In some embodiments, the first computing device 160a may send information about a location of the first computing device 160a to the communication interface 110. The location of the first computing device 160a may be considered as the location of the user 161a. In some embodiments, the location of the user 161a may be considered as a destination of the on-demand service. In some other embodiments, the first computing device 160a may send information about an address of the user 161a, and the address of the user 161a may be considered as the destination of the on-demand service. In some other embodiments, the first computing device 160a may send information about an address of a recipient of the delivery, and the address of the recipient may be considered as the destination of the on-demand service. In some embodiments, the communication interface 110 may send the request for the fare-check along with the location of the first computing device 160a and/or the destination of the on-demand service to the processor 120. In some embodiments, the processor 120 may receive the request for the fare-check along with the location of the first computing device 160a and/or the destination of the on-demand service from the communication interface 110.

[0062] In some embodiments, in response to the request for the fare-check, the processor 120 may calculate the fare for the order for the on-demand service. For example, the processor 120 may calculate the fare for the order based on at least one of the location of the first computing device 160a and the destination of the on-demand service. In some embodiments, the processor 120 may send information about the fare (i.e. estimated fare) for the order for the on-demand service to the communication interface 110. In some embodiments, the communication interface 110 may send the information about the fare for the order to the first computing device 160a. In some embodiments, the first computing device 160a may receive the information about the fare for the order from the communication interface 110, and display the information about the fare for the order on the user interface presented on the first computing device 160a.

[0063] In some embodiments, after the user 161a checks the fare (i.e. estimated fare) presented on the first computing device 160a, the user 161a may make the order for the on-demand service based on the fare. In some embodiments, the first computing device 160a may send the order for the on-demand service to the communication interface 110, and the communication interface 110 may send the order for the on-demand service to the processor 120. In some embodiments, the processor 120 may allocate the order to the suitable delivery service provider 171a among the plurality of the delivery service providers 171. In some embodiments, the processor 120 may control a second computing device 170a associated with the delivery service provider 171a to display a map relating to a route to from a location of the second computing device 170a (which may be considered as the location of the delivery service provider 171a) to a destination for providing the on-demand service, while the delivery service provider 171a moves to the destination (en route to the destination). In some embodiments, the destination may be a location of the first computing device 160a (which may be considered as the location of the consumer 161a (i.e. the user 161a)). In some

other embodiments, the destination may be a location of a recipient (i.e. the consumer) of the on-demand service that the requester 161a (i.e. the user 161a) has designated.

[0064] In some embodiments, the processor 120 may provide one or more delivery options, for example, a plurality of delivery options based on a speed of the delivery. For example, the processor 120 may provide two delivery options as follows:

- a) An option of a standard delivery at a standard delivery fare (which may be a generic product of an on-demand service platform provider) (also referred to as a “standard option” or a “first delivery option”); and
- b) An option of a saver delivery at a saver delivery fare (which may be a lower fare for a slower delivery on the user’s side and a higher batch rate on the on-demand service platform provider’s side) (also referred to as a “saver option” or a “second delivery option”).

In some embodiments, the second delivery option may offer the slower delivery, for example, by batching orders, at the lower fare than the first delivery option.

[0065] As another example, the processor 120 may provide three delivery options as follows:

- a) An option of a priority delivery at a priority delivery fare (also referred to as a “priority option” or a “third delivery option”);
- b) An option of a standard delivery at a standard delivery fare (which may be the generic product of the on-demand service platform provider) (also referred to as a “standard option” or a “first delivery option”); and
- c) An option of a saver delivery at a saver delivery fare (which may be the lower fare for the slower delivery on the user’s side and the higher batch rate on the on-demand service platform provider’s side) (also referred to as a “saver option” or a “second delivery option”).

In some embodiments, the second delivery option may offer the slower delivery, for example, by batching orders, at the lower fare than the first delivery option. In some embodiments, the third delivery option may offer a faster delivery, for example, by prioritising in allocation of the order to the delivery service provider 171a, at the higher fare than the first delivery option.

[0066] In some embodiments, the processor 120 may predict a book-through rate (BTR) (hereinafter, referred to as a “first BTR”) for the first delivery option and a book-through rate (BTR) (hereinafter, referred to as a “second BTR”) for the second delivery option for the order for each of a plurality of discount options for the fare for the second delivery option. In some embodiments, the book-through rate (BTR) may refer to a ratio of confirmed bookings (orders) for the on-demand service after the fare-check (i.e. checking the estimated fare) for the order for the on-demand service. In some embodiments, the processor 120 may obtain the plurality of (candidate) discount options for the fare for the second delivery option. In some embodiments, the plurality of discount options may be pre-determined, for example, by the processor 120 and/or an on-demand service platform provider. In some embodiments, the plurality of discount options may be in the form of a percentage discount (for example, 5%) or an amount of discount (for example, \$1). In some embodiments, the plurality of discount options may be customised based on factors, including, but not limited to, a city, a time (for example, hour), and the item providers 181 (merchants). In some embodiments, for each of the plurality of discount options for the fare for the second delivery option, the processor 120 may predict the first BTR and the second BTR. It may be appreciated that a discount for the fare for the second delivery option may affect the first BTR as well as the second BTR. In some embodiments, the processor 120 may output a plurality of sets of the first BTR and the second BTR each corresponding to the plurality of discount options.

[0067] In some embodiments, the processor 120 may compute at least one of an expected revenue and an expected profit for the order for each of the plurality of discount options,

based on the first BTR and the second BTR. In some embodiments, the processor 120 may compute a fare for the first delivery option (also referred to as a “standard delivery fare”), for example, using a conventional pricing model. In some embodiments, the processor 120 may compute a fare for the second delivery option (also referred to as a “saver delivery fare”) based on each of the plurality of discount options for the fare for the second delivery option. For example, the processor 120 may output a plurality of fares for the secondary delivery options, each corresponding to the plurality of discount options. In some embodiments, the processor 120 may compute the at least one of the expected revenue and the expected profit, further based on the computed fare for the first delivery option and the computed fare for the second delivery option (which may relate to the each of the plurality of discount options).

[0068] In some embodiments, the expected revenue for the order may refer to a total amount of income generated by a sale of the on-demand service relating to the order, and the expected profit for the order may refer to an amount of income that remains after accounting for all expenses and costs. In some embodiments, for each of the plurality of sets of the first BTR and the second BTR each corresponding to the plurality of discount options, the processor 120 may compute the at least one of the expected revenue and the expected profit. For example, the processor 120 may output a plurality of expected revenues each corresponding to the plurality of sets of the first BTR and the second BTR. As another example, the processor 120 may output a plurality of expected profits each corresponding to the plurality of sets of the first BTR and the second BTR.

[0069] In some embodiments, the processor 120 may select a discount option, for example, one discount option, from the plurality of discount options, based on the at least one of the expected revenue and the expected profit which is computed based on the first BTR and the second BTR for each of the plurality of discount options. In some embodiments, the processor 120 may select the discount option which may relate to the highest expected

revenue. In some other embodiments, the processor 120 may select the discount option which may relate to the highest expected profit.

[0070] In some embodiments, the processor 120 may determine the fare for the second delivery option based on the selected discount option. In some embodiments, the processor 120 may apply the selected discount option to the fare for the second delivery option, and determine the fare for the second delivery option accordingly.

[0071] In some embodiments, the processor 120 may, in response to the request for the fare-check, provide the first computing device 160a with information about the fare for the second delivery option based on the selected discount option. In some embodiments, the processor 120 may provide the first computing device 160a with the information about the fare for the second delivery option, along with information about the fare for the first delivery option. In some embodiments, the first computing device 160a may present the fare for the first delivery option and the fare for the second delivery option. In some embodiments, the user 161a may select one of the first and the second delivery options for the order for the on-demand service after considering the fares (i.e. estimated fares) and/or expected time of arrival (ETA).

[0072] In some embodiments, the processor 120 may build a BTR prediction model based on historical data. In some embodiments, the processor 120 may include the BTR prediction model. In some embodiments, the processor 120 may obtain the historical data from at least one of the memory 130 and a database 141 of a database system 140 (as shown in FIGS. 2 and 3) to build the BTR prediction model. In some embodiments, the processor 120 may build the BTR prediction model to predict the first BTR for the first delivery option and the second BTR for the second delivery option. In some embodiments, the processor 120 may predict the first BTR for the first delivery option and the second BTR for the second delivery option for the order for each of the plurality of discount options for the fare for the second delivery option, using the BTR prediction model. In some embodiments, the processor 120

may train the BTR prediction model based on predicted data and actual data, for example, the predicted first BTR and the predicted second BTR for each of the plurality of discount options and an actual first BTR and an actual second BTR for the selected discount option.

[0073] In some embodiments, the processor 120 of the server 100 may use the BTR prediction model to predict the first BTR for the first delivery option and the second BTR for the second delivery option for each of the plurality of discount options, for the fare for the second delivery option, for each fare-check of a plurality of fare-checks (i.e. a plurality of requests for the fare-checks received from the user 161a or the plurality of users 161). In some embodiments, given the predicted first BTR and the predicted second BTR, the processor 120 may compute fare-check level expected revenue and/or expected profit for each of the plurality of discount options for the fare for the second delivery option.

[0074] In some embodiments, the processor 120 may build an optimisation module. In some embodiments, the processor 120 may include the optimisation module. In some other embodiments, the processor 120 may access the optimisation module established at an external server. In some embodiments, the processor 120 may obtain information about a budget for a discount of the fare for the second delivery option, for example, from at least one of the memory 130 and the database 141. In some embodiments, the processor 120 may select the discount option from the plurality of discount options, further based on the budget, using the optimisation module. In some embodiments, the processor 120 may output the selected discount option, for example, by the optimisation module.

[0075] In some embodiments, the processor 120 may formulate and solve an optimisation problem to redistribute the budget towards the fare-checks with higher revenue uplift potentials. In some embodiments, the optimisation module may redistribute the budget towards the fare-checks with the higher revenue uplift potentials. In some embodiments, the

optimisation module may output the selected discount option for each fare-check, for example, just as a phase 1 (stage 1) method operated using the BTR prediction model.

[0076] In some embodiments, the processor 120 may train a post mapping model based on functions of the BTR prediction model and the selected discount option output by the optimisation module. In some embodiments, the processor 120 may use the post mapping model, instead of the optimisation module, to select the discount option from the plurality of discount options, in an offline simulation/environment.

[0077] It may be appreciated that, one of concerns of the above two-stage (two-phase) approach (i.e. a stage 1 (phase 1) of operations using the BTR prediction model; and a stage 2 (phase 2) of operations using the optimisation module) may be that in the online environment, the processor 120 may need to query the BTR prediction model $|D|$ times for each fare-check, where D is the set of discount options (i.e. the plurality of discount options), which may lead to high queries per second (QPS). Also, the processor 120 may need to implement the optimisation module in a backend, which may be costly. Therefore, the processor 120 may consider training an additional simple post mapping from functions (features) (similar to the ones used in the BTR prediction model) to the selected discount option output by the optimisation module, so that in the online environment, the server 100 may only need to serve the post mapping model. The validity of using this mapping to replace the optimisation module may be provided by the offline simulation.

[0078] As described above, to build a final discount model for the fare for the second delivery option, the server 100 may use the post mapping model, by considering similar features (functions) as the BTR prediction model, but with some simplification. The server 100 may try to remove too-granular features such as IDs (identifiers) of the item providers 180 (also referred to as “Mex IDs”) to improve generalisability of the final discount model.

Using fewer features and a simpler model structure (for example, trees instead of neural nets) may lead to better efficiency and lower costs of the online service for determining the fare.

[0079] In some embodiments, the processor 120 may, upon receipt of the request for the fare-check, query an inter-arrival rate (IAR) signal. For example, the processor 120 may query the IAR signal to the memory 130 and/or the database 141. In some embodiments, the processor 120 may predict overall time efficiency (OTE) based on the IAR signal. In some embodiments, the processor 120 may predict the OTE, further based on a batching configuration (also referred to as a “batching configuration parameter”). In some embodiments, the processor 120 may select the discount option from the plurality of discount options, further based on the predicted OTE.

[0080] In some embodiments, the processor 120 may predict the OTE for a predetermined time slot at a predetermined area, based on a market condition signal and the batching configuration. In some embodiments, the OTE may refer to a signal relating to a system overall time efficiency. In some embodiments, the processor 120 may use an OTE prediction model (also referred to as an “OTE uplift prediction module (model)”), to predict the OTE. In some embodiments, the processor 120 may include the OTE prediction model. In some other embodiments, the processor 120 may access the OTE prediction model established at an external server. In some embodiments, the batching configuration may include at least one of a batching interval, an order delivery delay, and a maximum amount of time the order can be kept (also referred to as a “max pooling duration”). In some embodiments, the processor 120 may predict the IAR for the predetermined time slot at the predetermined area. The processor 120 may use the predicted IAR as the market condition signal. In some embodiments, the OTE prediction model may obtain the batching configuration and the market condition signal, and predict the OTE.

[0081] In some embodiments, the OTE prediction model may rely on two sets of parameters/signals as follows:

a) IAR signal (from an online feature provider/system)

IAR for the 9 neighbouring geohashes (including self-geohash) where an item provider 180a (i.e. a merchant) resides

b) Batching configuration

i) Batching interval

How often the processor 120 will attempt to attempt batching.

This may be a city-level configuration.

ii) Max Pooling Duration

How long an order (i.e. a booking) stays in the batching system.

This may be a merchant/vehicle type ID-level configuration.

iii) Delivery delay

This may be dynamic, conditioned on whether the batching is an upfront batching (i.e. batching of orders before allocation to the delivery service provider 171a) or an in-transit batching (i.e. batching of orders after allocation to the delivery service provider 171a).

[0082] In some embodiments, for the IAR signals, the processor 120 may query from the online feature provider/system. For the batching configurations, for example, the delivery delay, the processor 120 may make assumptions as follows:

a) The orders may be a base JITA (just-in-time orders); and

b) The orders may be upfront batched, ignoring the in-transit batches.

Given these assumptions, the processor 120 may grab the corresponding configuration variables, and re-use the OTE prediction model for the OTE uplift prediction.

[0083] In some embodiments, the processor 120 may select the discount option from the plurality of discount options, further based on a fare for the first delivery option (i.e. the standard delivery fare).

[0084] As described above, the various embodiments may consider user's 161 choice behaviour (for example, the BTR) together with the batching efficiency uplift of each discount option. Specifically, the various embodiments may model user price elasticities and avoid providing overly-high discount values to highly elastic user groups to reduce revenue and/or profit cannibalization. In this manner, the various embodiments may dynamically adjust a discount of certain options (for example, the second delivery option) for each coming individual fare-check in real-time, such that the overall booking-to-convert (BTC), revenue, and profit may be maximised under the constraint of the same overall discount rate. The various embodiments may automatically identify and address revenue/profit cannibalization from user choices and improve business metrics. In addition, by using the post mapping model, the various embodiments may achieve fast and low-costs serving.

[0085] FIG. 5 illustrates a flowchart for a method 300 for providing information for an on-demand service according to various embodiments. According to various embodiments, the method 300 for providing the information for the on-demand service may be provided.

[0086] In some embodiments, the method 300 may include a step 301 of receiving a request for a fare-check for an order for the on-demand service from a computing device.

[0087] In some embodiments, the method 300 may include a step 302 of predicting a first BTR for a first delivery option and a second BTR for a second delivery option for the order, for each of a plurality of discount options, for the fare for the second delivery option. In some embodiments, the second delivery option offers a slower delivery at a lower fare than the first delivery option.

[0088] In some embodiments, the method 300 may include a step 303 of computing at least one of an expected revenue and an expected profit for the order for each of the plurality of discount options, based on the first BTR and the second BTR.

[0089] In some embodiments, the method 300 may include a step 304 of selecting a discount option from the plurality of discount options, based on the at least one of the expected revenue and the expected profit.

[0090] In some embodiments, the method 300 may include a step 305 of determining the fare for the second delivery option based on the selected discount option.

[0091] FIG. 6 illustrates a data flow diagram of a server for determining a fare for an on-demand service according to various embodiments. FIG. 6 shows a logic of training/simulation time (also referred to as a “training logic”).

[0092] In some embodiments, the training logic may include three regular machine learning (ML) pipeline jobs as follows. First, the training logic may prepare data samples and relevant features daily, with necessary additional OTE prediction/imputation works, if such information is not directly available in an offline datastore (for example, a cold start in new cities). Then, the training logic may train the BTR prediction model weekly with the collected fare-check samples. Finally, the training logic may formulate the optimisation problem with inputs estimated by the BTR prediction model and solve for the optimal saver discount, and then train the discount model. A simulation report may also be generated in this pipeline, based on which a decision on whether the new discount model should replace the previous one will be made.

[0093] FIG. 7 illustrates a data flow diagram of a server for determining a fare for an on-demand service according to various embodiments. FIG. 7 shows a logic of testing/inference time (also referred to as a “test time logic”).

[0094] In some embodiments, the test time logic may be as follows. First, for each fare-check request, the test time logic may query an IAR signal from the online feature provider/system and use these features to get predicted OTE via a predetermined model (for example, Catwalk model). Second, the test time logic may use the predetermined model to predict the saver discount given the predicted OTE and the other related features such as the standard delivery fee.

[0095] Hereinafter, the BTR prediction model will be described in detail with reference to FIGS. 6 and 7.

[0096] The BTR prediction model may be responsible for estimating choice probability q_i given price p_i , time SLA (service level agreement) T_i for each option i (in a standard option, a saver option, etc.), and other contextual information X . As the saver option price may influence user behaviour in multiple pages, and one user action sequence may generate multiple fare-check events, selecting right data as training samples may be non-trivial. In the current stage, to simplify the modelling difficulty, the processor 120 may focus on a basket page choice behaviour and ignore user behaviours with regards to the other pages. In particular, each data sample may correspond to a unique set of available options and prices within an UCP (unique checking price) session (for example, uniquely determined by Pax (passenger) ID, Mex ID, drop-off geohash 6, and consecutive time gap < 30 mins + no order made). The corresponding label may be a final choice of the user for all the fare-checks related to this data sample.

[0097] *Feature Selection*

[0098] The saver option choice probability may be affected by one or more aspects of factors as follows:

- Option details (for example, price + time SLA);
- Drop-off location (for example, as embedding);

- General fare information (for example, nsns (no surge no surcharge) fare (i.e. base fare before surge and surcharge), surges, Mex fees, etc.);
- General delivery information (for example, distance and ETA, etc.);
- General time information (for example, hour, day of week, time to peak, etc.); and
- Mex information (for example, Merchant ID).

[0099] *Feature Engineering*

[00100] For the feature engineering, the processor 120 may apply standard pre-processing approaches for NN (neural network) models as follows:

- For continuous features, the processor 120 may do min-max normalisation after potentially necessary log-transform, and also let the model to learn a default Fill NA value (In some cases, some features are unavailable in the request for the fare-check. The corresponding values (empty) are called “not a number (NA)” values. In addition, here, the processor 120 may leverage the model to learn a value to replace these NA values) in an end-to-end manner;
- For categorical features, the processor 120 may do embedding with the embedding size as 8.
- Structural features (option price + time SLA) may only be scaled manually (for example, price / 10, time SLA in hour units) so that parameters may be interpreted.

[00101] *Model Architecture*

[00102] Currently, the processor 120 may use a general NN-based model for BTR modelling.

The structural model may be described as $\vec{q} = \text{Softmax}(NN(\vec{p}, \vec{T}, X))$, where p refers to prices (fare) of the plurality of discount options, T refers to ETAs of the plurality of discount options, X refers to the other features, NN refers to a generalized neural network function, and q refers to the BTR prediction, and where the structural variables may be concatenated with

other general information and fed into the MLP $NN(\cdot)$ together. The Softmax refers to a softmax function, for example, $f(x | x, y, z) = e^x / (e^x + e^y + e^z)$.

[00103] In particular, for this MLP (multilayer perceptron), the processor 120 may put a fill NA layer and an embedding layer on a bottom of the model. The fill NA layer may be a custom layer responsible for pre-processing and handling missing values in continuous features. It may ensure that continuous feature inputs are clean and ready for the model's subsequent layers. The embedding layer may be responsible for converting input categorical features into dense vector representations (embeddings). The processor 120 may use 16 as embedding size. This layer may help the model better understand underlying relationships between categorical features.

[00104] After that, a batch normalisation layer followed by a LeakyReLU (Rectified Linear Unit) activation layer may be added on the concatenation of fill NA layer and embedding layer. This structure may be stacked 3 times with hidden unit size = [64,32,16] separately.

[00105] For an output layer, it may be basically a SoftMax linear transformation on the final output layer.

[00106] As for hyper parameters, the processor 120 may use CrossEntropyLoss and AdamOptimizer. The learning rate may be 1e-4 and batch size may be 2048.

[00107] *Definition of the Revenue & Profit*

[00108] To compute the expected revenue and the expected profit for the fare-check under the specific saver discount, besides the choice probability estimated from the BTR prediction model, the processor 120 may also need to compute the revenue and profit for each option in a fare-check. Following the definition of revenue and profit in the phase 1 (stage 1), the processor 120 may implement the revenue and the profit for the standard and saver options as follows:

- Standard revenue = basket value * Mex commission rate + Standard delivery fare

- Saver revenue = basket value * Mex commission rate + Saver delivery fare + Standard revenue * delta OTE * batch size
- Standard profit = basket value * Mex commission rate + Standard delivery fare - Dax fee
- Saver profit = basket value * Mex commission rate + Saver delivery fare - Dax fee + Standard profit * delta OTE * batch size

where the basket values are collected in fare-check level, and Mex commission rate, Dax fee, and batch size may be collected in Mex level according to historical data.

[00109] Hereinafter, a linear optimisation will be described in detail with reference to FIGS. 6 and 7.

[00110] Given the estimated revenue r_{ij} and profit c_{ij} for each fare-check i and saver discount option j , the processor 120 may consider the following linear optimisation problem:

$$\begin{aligned}
 & \max_{x_{ij} \in \{0,1\}} \sum_{ij} r_{ij} x_{ij} \\
 & s. t. \sum_{ij} c_{ij} x_{ij} \geq C_{base} \\
 & \sum_j x_{ij} = 1, \forall i
 \end{aligned}$$

where r_{ij} and c_{ij} refer to the revenue and profit for the fare-check i (in the historical dataset) and the discount option j ; C_{base} refers to the actual total profit from all the fare-checks in the historical dataset; and x_{ij} refers to the decision variable whether the processor 120 selects the discount option j for the fare-check i , and where a decision variable $x_{ij} \in \{0,1\}$ represents whether the processor 120 may select the discount option j for the fare-check i . In this formulation, the processor 120 may only consider a lower bound constraint of an aggregated

profit level, but additional constraints on aggregated booking number and discount rate may also be added.

[00111] The above problem may be tractably solved with a dual formulation, which may be a m -d non-smooth convex optimisation problem, where m is the number of linear constraints.

[00112] FIGS. 8 and 9 are exemplary diagrams showing a user interface 160a of a fare-check for an order for an on-demand service according to various embodiments.

[00113] In some embodiments, the processor 120 may provide a plurality of delivery options, for example, three delivery options 162, based on a speed of the delivery. As shown in FIGS. 8 and 9, after a request for the fare-check for the order, the three delivery options 162 may be presented on a booking page, as follows:

- a) An option of a priority delivery at a priority delivery fare 163 (also referred to as a “priority option”);
- b) An option of a standard delivery at a standard delivery fare 164 (which may be a generic product of an on-demand service platform provider) (also referred to as a “standard option”); and
- c) An option of a saver delivery at a saver delivery fare 165 (which may be a lower fare for a slower delivery on the user’s side and a higher batch rate on the on-demand service platform provider’s side) (also referred to as a “saver option”).

[00114] In some embodiments, the processor 120 may predict a first BTR for the first delivery option 164 and a second BTR for the second delivery option 165 for each of a plurality of discount options for the fare for the second delivery option 165, and compute at least one of an expected revenue and an expected profit for the order for each of the plurality of discount options based on the first BTR and the second BTR. Thereafter, the processor 120 may select a discount option based on the at least one of the expected revenue and the

expected profit. The processor 120 may determine the fare for the second delivery option 165 based on the selected discount option.

[00115] As shown in FIG. 8, for example, if applying a first discount option, which offers more discount than the second discount option, to the fare for the second delivery option 165 may contribute higher expected revenue and/or expected profit (for example, at lunch peak time), the processor 120 may determine the fare for the second delivery option 165 based on the first discount option. For example, the fare for the second delivery option 165 may be determined as \$4.9. As shown in FIG. 9, for example, if applying the second discount option, which offers less discount than the first discount option, to the fare for the second delivery option 165 may contribute higher expected revenue and/or expected profit (for example, at afternoon tea time), the processor 120 may determine the fare for the second delivery option 165 based on the second discount option. For example, the fare for the second delivery option 165 may be determined as \$4.7.

[00116] Although not shown, in some other embodiments, the processor 120 may predict the first BTR for the first delivery option 164, the second BTR for the second delivery option 165, and a book-through rate (BTR) (hereinafter, referred to as a “third BTR”) for the third delivery option 163, and compute at least one of the expected revenue and the expected profit for the order for each of the plurality of discount options based on the first BTR, the second BTR, and the third BTR. Thereafter, the processor 120 may select the discount option based on the at least one of the expected revenue and the expected profit. The processor 120 may determine the fare for the second delivery option 165 and a fare for the third delivery option 163 based on the selected discount option.

[00117] While the disclosure has been particularly shown and described with reference to specific embodiments, 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 spirit and scope of the

invention as defined by the appended claims. The scope of the invention 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

1. A server for facilitating determining a fare for an on-demand service, the server comprising:

a memory configured to store instructions;

a communication interface configured to receive a request for a fare-check for an order for the on-demand service from a computing device; and

a processor for executing the stored instructions and configured to:

predict a first book-through rate (BTR) for a first delivery option and a second BTR for a second delivery option for the order, for each of a plurality of discount options for the fare for the second delivery option, wherein the second delivery option offers a slower delivery at a lower fare than the first delivery option;

compute at least one of an expected revenue and an expected profit for the order for each of the plurality of discount options, based on the first BTR and the second BTR;

select a discount option from the plurality of discount options, based on the at least one of the expected revenue and the expected profit; and

determine the fare for the second delivery option based on the selected discount option.

2. The server according to claim 1, wherein the processor is further configured to, in response to the request, provide the computing device with information about the fare for the second delivery option based on the selected discount option.

3. The server according to claim 1 or claim 2, wherein the processor is further configured to build a BTR prediction model based on historical data.

4. The server according to claim 3, wherein the processor is further configured to predict the first BTR for the first delivery option and the second BTR for the second delivery option for the order for each of the plurality of discount options, using the BTR prediction model.
5. The server according to claim 3 or claim 4, wherein the processor is further configured to:
 - obtain information about a budget for a discount of the fare for the second delivery option;
 - select the discount option from the plurality of discount options, further based on the budget, using an optimisation module; and
 - output, by the optimisation module, the selected discount option.
6. The server according to claim 5, wherein the processor is further configured to train a post mapping model based on functions of the BTR prediction model and the selected discount option output by the optimisation module.
7. The server according to claim 6, wherein the processor is further configured to use the post mapping model, instead of the optimisation module, to select the discount option from the plurality of discount options, in an offline simulation.
8. The server according to any one of claims 1 to 7, wherein the processor is further configured to:
 - upon receipt of the request, query an inter-arrival rate (IAR) signal;
 - predict overall time efficiency (OTE) based on the IAR signal; and

select the discount option from the plurality of discount options, further based on the predicted OTE.

9. The server according to claim 8, wherein the processor is further configured to predict the OTE, further based on a batching configuration.

10. The server according to any one of claims 1 to 9, wherein the processor is further configured to select the discount option from the plurality of discount options, further based on a fare for the first delivery option.

11. A method for facilitating determining a fare for an on-demand service, the method comprising:

receiving a request for a fare-check for an order for the on-demand service from a computing device;

predicting a first book-through rate (BTR) for a first delivery option and a second BTR for a second delivery option for the order, for each of a plurality of discount options for the fare for the second delivery option, wherein the second delivery option offers a slower delivery at a lower fare than the first delivery option;

computing at least one of an expected revenue and an expected profit for the order for each of the plurality of discount options, based on the first BTR and the second BTR;

selecting a discount option from the plurality of discount options, based on the at least one of the expected revenue and the expected profit; and

determining the fare for the second delivery option based on the selected discount option.

12. The method according to claim 11, further comprising: in response to the request, providing the computing device with information about the fare for the second option based on the selected discount option.
13. The method according to claim 11 or claim 12, further comprising: building a BTR prediction model based on historical data.
14. The method according to claim 13, further comprising: predicting the first BTR for the first delivery option and the second BTR for the second delivery option for the order for each of the plurality of discount options, using the BTR prediction model.
15. The method according to claim 13 or claim 14, further comprising:
obtaining information about a budget for a discount of the fare for the second delivery option;
selecting the discount option from the plurality of discount options, further based on the budget, using an optimisation module; and
outputting, by the optimisation module, the selected discount option.
16. The method according to claim 15, further comprising: training a post mapping model based on functions of the BTR prediction model and the selected discount option output by the optimisation module.
17. The method according to claim 16, further comprising: using the post mapping model, instead of the optimisation module, to select the discount option from the plurality of discount options, in an offline simulation.

18. The method according to any one of claims 11 to 17, further comprising:
upon receipt of the request, querying an inter-arrival rate (IAR) signal;
predicting overall time efficiency (OTE) based on the IAR signal; and
selecting the discount option from the plurality of discount options, further based on the predicted OTE.
19. The method according to claim 18, further comprising: predicting the OTE, further based on a batching configuration.
20. The method according to any one of claims 11 to 19, further comprising: selecting the discount option from the plurality of discount options, further based on a fare for the first delivery option.