

SYSTEM AND METHOD FOR PREDICTING DELIVERY TIME FOR BATCH ORDERS

TECHNICAL FIELD

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

BACKGROUND

[0002] In a delivery market (e.g., food delivery) with limited driver supply, allocation of orders to drivers could be poor, resulting in some users who placed orders not getting their orders fulfilled. As for the drivers, the rate of investment (ROI) on a delivery job may not be as high as passenger delivery jobs because of the extra effort and waiting time involved in waiting at a merchant store.

[0003] With order batching, i.e. combining multiple orders into one driver trip, the probability that user will be allocated a driver will go up. Drivers also make more income per hour. Food orders from the same and/or nearby merchant stores can be batched together and delivered to the user or to several nearby user locations. The batch will be determined by adding to a driver's in-transit route when they are on the way to a pick-up, with some imposed constraints, such as delivery time to eater, route match (less detour), capacity, etc.

[0004] While maximum delivery time delay can be capped by setting a constraint during batch creation, a problem arises there is a need to predict the delivery time for an order as:

prediction happens before driver allocation (and batch creation), and an additional delivery time due to batching can be anywhere between zero and the maximum delay amount. Current existing approach of delivery prediction is only based on a single order assumption and does not consider the order batching aspect of it.

SUMMARY

[0005] Therefore, there is a need to accurately predict a delivery time for batch orders. There is also a need to increase user and driver satisfaction.

[0006] An advantage of the present disclosure may include a dynamically adjusted delivery time in real-time resulting in more accurate delivery time predictions for batch orders.

[0007] An advantage of the present disclosure may include higher user satisfaction due to increased order allocation rate.

[0008] These and other aforementioned advantages and features of the aspects herein disclosed will be apparent through reference to the following description and the accompanying drawings. Furthermore, it is to be understood that the features of the various aspects described herein are not mutually exclusive and can exist in various combinations and permutations.

[0009] The present disclosure generally relates to a system for predicting a delivery time for batch orders. The system may include one or more processor(s); and a memory having instructions stored therein, the instructions, when executed by the one or more processor(s), may cause the one or more processor(s) to: identify a first location, wherein one or more merchants are located in the first location; identify a second location, wherein one or more users may be located in the second location; predict a dynamic buffer time; wherein the delivery time may include the dynamic buffer time, and wherein the one or more processor(s)

is configured to predict the dynamic buffer time based on one or more batch orders between a first bearing of the first location and the second location and a second bearing of the first location and a third location.

[0010] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time based on at least one of: a proportion of batch orders or a number of batch orders between the first bearing of the first location and the second location and the second bearing of the first location and the third location.

[0011] According to an embodiment, the dynamic buffer time may be additional time due to order batching.

[0012] According to an embodiment, the first bearing and the second bearing may have an angle difference of 45 degrees.

[0013] According to an embodiment, the one or more processor(s) may be configured to allow the one or more users in the second location to batch order from the one or more merchants in the first location.

[0014] According to an embodiment, the one or more processor(s) may be configured to assign a same delivery driver for the batch order.

[0015] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time based on contextual information. The contextual information may include one of: an ordering time, a merchant type, a price range and a basket size of an order.

[0016] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time using machine learning of historical data of single orders. The historical data may be at least one of: an allocation time prediction, a pick-up routing time prediction, a waiting time prediction, an order preparation time prediction and a drop-off routing time prediction.

[0017] According to an embodiment, the one or more processor(s) may be configured to perform skewness data transformation on the historical data prior to using the historical data for predicting the dynamic buffer time.

[0018] The present disclosure generally relates to a method for predicting a delivery time for batch orders. The method may include using one or more processor(s) to: identify a first location, wherein one or more merchants are located in the first location; identify a second location, wherein one or more users may be located in the second location; predict a dynamic buffer time; wherein the delivery time may include the dynamic buffer time, and wherein the one or more processor(s) is configured to predict the dynamic buffer time based on one or more batch orders between a first bearing of the first location and the second location and a second bearing of the first location and a third location.

[0019] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time based on at least one of: a proportion of batch orders or a number of batch orders between the first bearing of the first location and the second location and the second bearing of the first location and the third location.

[0020] According to an embodiment, the dynamic buffer time may be additional time due to order batching.

[0021] According to an embodiment, the first bearing and the second bearing may have an angle difference of 45 degrees.

[0022] According to an embodiment, the one or more processor(s) may be configured to allow the one or more users in the second location to batch order from the one or more merchants in the first location.

[0023] According to an embodiment, the one or more processor(s) may be configured to assign a same delivery driver for the batch order.

[0024] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time based on contextual information. The contextual information may include one of: an ordering time, a merchant type, a price range and a basket size of an order.

[0025] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time using machine learning of historical data of single orders. The historical data may be at least one of: an allocation time prediction, a pick-up routing time prediction, a waiting time prediction, an order preparation time prediction, and a drop-off routing time prediction.

[0026] According to an embodiment, the one or more processor(s) may be configured to perform skewness data transformation on the historical data prior to using the historical data for predicting the dynamic buffer time.

[0027] The present disclosure generally relates to a non-transitory computer-readable medium storing computer executable code comprising instructions for predicting a delivery time for batch orders according to the present disclosure.

[0028] The present disclosure generally relates to a computer executable code comprising instructions for predicting a delivery time for batch orders according to the present disclosure.

[0029] To the accomplishment of the foregoing and related ends, the one or more embodiments include the features hereinafter fully described and particularly pointed out in the claims. The following description and the associated drawings set forth in detail certain illustrative features of the one or more aspects. These features are indicative, however, of but a few of the various ways in which the principles of various aspects may be employed, and this description is intended to include all such aspects and their equivalents.

BRIEF DESCRIPTION OF THE DRAWINGS

[0030] In the drawings, like reference characters generally refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead generally being placed upon illustrating the principles of the present disclosure. The dimensions of the various features or elements may be arbitrarily expanded or reduced for clarity. In the following description, various aspects of the present disclosure are described with reference to the following drawings, in which:

[0031] FIG. 1 illustrates a system according to various embodiments.

[0032] FIG. 2 shows a flowchart of a method according to various embodiments.

[0033] FIG. 3 illustrates an exemplary flowchart of various variables of a food delivery time (FDT) according to various embodiments.

[0034] FIG. 4A illustrates an exemplary bearing chart according to various embodiments.

[0035] FIG. 4B illustrates an exemplary bearing classification table for the exemplary bearing chart of FIG. 4A according to various embodiments.

[0036] FIG. 5A illustrates an exemplary first graph showing symmetric distribution data according to various embodiments.

[0037] FIG. 5B illustrates an exemplary second graph showing right-skewed distribution data according to various embodiments.

[0038] FIG. 5C illustrates an exemplary third graph showing left-skewed distribution data according to various embodiments.

[0039] FIG. 6 illustrates an exemplary machine learning flowchart according to various embodiments.

DETAILED DESCRIPTION

[0040] The following detailed description refers to the accompanying drawings that show, by way of illustration, specific details and embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention. Other embodiments may be utilized and structural, and logical changes may be made without departing from the scope of the invention. The various embodiments are not necessarily mutually exclusive, as some embodiments can be combined with one or more other embodiments to form new embodiments.

[0041] Embodiments described in the context of one of the systems or server or methods or computer program are analogously valid for the other systems or server or methods or computer program and vice-versa.

[0042] 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.

[0043] The word “exemplary” is used herein to mean “serving as an example, instance, or illustration.” Any embodiment or design described herein as “exemplary” is not necessarily to be construed as preferred or advantageous over other embodiments or designs

[0044] 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.

[0045] As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0046] The terms “at least one” and “one or more” may be understood to include a numerical quantity greater than or equal to one (e.g., one, two, three, four, [...], etc.). The term “a plurality” may be understood to include a numerical quantity greater than or equal to two (e.g., two, three, four, five, [...], etc.).

[0047] The words “plural” and “multiple” in the description and the claims expressly refer to a quantity greater than one. Accordingly, any phrases explicitly invoking the aforementioned words (e.g. “a plurality of [objects]”, “multiple [objects]”) referring to a quantity of objects expressly refers more than one of the said objects. The terms “group (of)”, “set [of]”, “collection (of)”, “series (of)”, “sequence (of)”, “grouping (of)”, etc., and the like in the description and in the claims, if any, refer to a quantity equal to or greater than one, i.e. one or more. The terms “proper subset”, “reduced subset”, and “lesser subset” refer to a subset of a set that is not equal to the set, i.e. a subset of a set that contains less elements than the set.

[0048] The term “data” as used herein may be understood to include information in any suitable analog or digital form, e.g., provided as a file, a portion of a file, a set of files, a signal or stream, a portion of a signal or stream, a set of signals or streams, and the like. Further, the term “data” may also be used to mean a reference to information, e.g., in form of a pointer. The term data, however, is not limited to the aforementioned examples and may take various forms and represent any information as understood in the art.

[0049] The term “processor” or “controller” as, for example, used herein may be understood as any kind of entity that allows handling data, signals, etc. The data, signals, etc. may be handled according to one or more specific functions executed by the processor or controller.

[0050] A processor or a controller may thus be or include an analog circuit, digital circuit, mixed-signal circuit, logic circuit, processor, microprocessor, Central Processing Unit (CPU), Graphics Processing Unit (GPU), Digital Signal Processor (DSP), Field Programmable Gate Array (FPGA), integrated circuit, Application Specific Integrated Circuit (ASIC), etc., or any combination thereof. Any other kind of implementation of the respective functions, which will be described below in further detail, may also be understood as a processor, controller, or logic circuit. It is understood that any two (or more) of the processors, controllers, or logic circuits detailed herein may be realized as a single entity with equivalent functionality or the like, and conversely that any single processor, controller, or logic circuit detailed herein may be realized as two (or more) separate entities with equivalent functionality or the like.

[0051] The term “system” (e.g., a drive system, a position detection system, etc.) detailed herein may be understood as a set of interacting elements, the elements may be, by way of example and not of limitation, one or more mechanical components, one or more electrical components, one or more instructions (e.g., encoded in storage media), one or more controllers, etc.

[0052] A “circuit” as user herein is understood as any kind of logic-implementing entity, which may include special-purpose hardware or a processor executing software. A circuit may thus be an analog circuit, digital circuit, mixed-signal circuit, logic circuit, processor, microprocessor, Central Processing Unit (“CPU”), Graphics Processing Unit (“GPU”), Digital Signal Processor (“DSP”), Field Programmable Gate Array (“FPGA”), integrated circuit, Application Specific Integrated Circuit (“ASIC”), etc., or any combination thereof. Any other kind of implementation of the respective functions which will be described below in further detail may also be understood as a “circuit.” It is understood that any two (or more) of the circuits detailed herein may be realized as a single circuit with substantially equivalent functionality, and conversely that any single circuit detailed herein may be realized as two

(or more) separate circuits with substantially equivalent functionality. Additionally, references to a “circuit” may refer to two or more circuits that collectively form a single circuit.

[0053] As used herein, “memory” may be understood as a non-transitory computer-readable medium in which data or information can be stored for retrieval. References to “memory” included herein may thus be understood as referring to volatile or non-volatile memory, including random access memory (“RAM”), read-only memory (“ROM”), flash memory, solid-state storage, magnetic tape, hard disk drive, optical drive, etc., or any combination thereof. Furthermore, it is appreciated that registers, shift registers, processor registers, data buffers, etc., are also embraced herein by the term memory. It is appreciated that a single component referred to as “memory” or “a memory” may be composed of more than one different type of memory, and thus may refer to a collective component including one or more types of memory. It is readily understood that any single memory component may be separated into multiple collectively equivalent memory components, and vice versa. Furthermore, while memory may be depicted as separate from one or more other components (such as in the drawings), it is understood that memory may be integrated within another component, such as on a common integrated chip.

[0054] The following detailed description refers to the accompanying drawings that show, by way of illustration, specific details and aspects in which the present disclosure may be practiced. These aspects are described in sufficient detail to enable those skilled in the art to practice the present disclosure. Various aspects are provided for the present system, and various aspects are provided for the methods. It will be understood that the basic properties of the system also hold for the methods and vice versa. Other aspects may be utilized and structural, and logical changes may be made without departing from the scope of the present

disclosure. The various aspects are not necessarily mutually exclusive, as some aspects can be combined with one or more other aspects to form new aspects.

[0055] To more readily understand and put into practical effect, the present system, method, and other particular aspects will now be described by way of examples and not limitations, and with reference to the figures. For the sake of brevity, duplicate descriptions of features and properties may be omitted.

[0056] It will be understood that any property described herein for a specific system or device may also hold for any system or device described herein. It will also be understood that any property described herein for a specific method may hold for any of the methods described herein. Furthermore, it will be understood that for any device, system, or method described herein, not necessarily all the components or operations described will be enclosed in the device, system, or method, but only some (but not all) components or operations may be enclosed.

[0057] The term “comprising” shall be understood to have a broad meaning similar to the term “including” and will be understood to imply the inclusion of a stated integer or operation or group of integers or operations but not the exclusion of any other integer or operation or group of integers or operations. This definition also applies to variations on the term “comprising” such as “comprise” and “comprises”.

[0058] The term “coupled” (or “connected”) herein may be understood as electrically coupled or as mechanically coupled, e.g., attached or fixed or attached, or just in contact without any fixation, and it will be understood that both direct coupling or indirect coupling (in other words: coupling without direct contact) may be provided.

[0059] FIG. 1 illustrates a system 100 according to various embodiments.

[0060] According to various embodiments, the system 100 may include a server 110, and/or a user device 120.

[0061] In various embodiments, the server 110 and the user device 120 may be in communication with each other through communication network 130. In an embodiment, even though FIG. 1 shows a line connecting the server 110 to the communication network 130, a line connecting the user device 120 to the communication network 130, the server 110, and the user device 120 may not be physically connected to each other, for example through a cable. In an embodiment, the server 110, and the user device 120 may be able to communicate wirelessly through communication network 130 by internet communication protocols or through a mobile cellular communication network.

[0062] In various embodiments, the server 110 may be a single server as illustrated schematically in FIG. 1, or have the functionality performed by the server 110 distributed across multiple server components. In an embodiment, the server 110 may include one or more server processor(s) 112. In an embodiment, the various functions performed by the server 110 may be carried out across the one or more server processor(s). In an embodiment, each specific function of the various functions performed by the server 110 may be carried out by specific server processor(s) of the one or more server processor(s).

[0063] In an embodiment, the server 110 may include a memory 114. In an embodiment, the server 110 may also include a database. In an embodiment, the memory 114 and the database may be one component or may be separate components. In an embodiment, the memory 114 of the server may include computer executable code defining the functionality that the server 110 carries out under control of the one or more server processor 112. In an embodiment, the database and/or memory 114 may include historical data of past order services, e.g., a user location and/or a merchant location, and/or ordering time, and/or a merchant type and/or price range and/or a basket size of an order and/or previous single order historical data and/or previous batch order data. In an embodiment, the memory 114 may

include or may be a computer program product such as a non-transitory computer-readable medium.

[0064] According to various embodiments, a computer program product may store the computer executable code including instructions for predicting a delivery time for batch orders according to the various embodiments. In an embodiment, the computer executable code may be a computer program. In an embodiment, the computer program product may be a non-transitory computer-readable medium. In an embodiment, the computer program product may be in the system 100 and/or the server 110.

[0065] In some embodiments, the server 110 may also include an input and/or output module allowing the server 110 to communicate over the communication network 130. In an embodiment, the server 110 may also include a user interface for user control of the server 110. In an embodiment, the user interface may include, for example, computing peripheral devices such as display monitors, user input devices, for example, touchscreen devices and computer keyboards.

[0066] In an embodiment, the user device 120 may include a user device memory 122. In an embodiment, the user device 120 may include a user device processor 124. In an embodiment, the user device memory 122 may include computer executable code defining the functionality the user device 120 carries out under control of the user device processor 124. In an embodiment, the user device memory 122 may include or may be a computer program product such as a non-transitory computer-readable medium.

[0067] In an embodiment, the user device 120 may also include an input and/or output module allowing the user device 120 to communicate over the communication network 130. In an embodiment, the user device 120 may also include a user interface for the user to control the user device 120. In an embodiment, the user interface may be a touch panel display. In an embodiment, the user interface may include a display monitor, a keyboard or buttons.

[0068] In an embodiment, the system 100 may be used for predicting a delivery time for batch orders. In an embodiment, the memory 114 may have instructions stored therein. In an embodiment, the processor 112 may be configured to identify a first location. The first location may be in or may be a first geohash. The term “geohash” may be predefined geocoded cells of partitioned areas of a city or country. In various embodiments, the first location may be a building such as a shopping mall or food centre. In various embodiment, the first location may be defined based on a predetermined radius or distance. In various embodiments, one or more merchants are located in the first location.

[0069] In an embodiment, the processor 112 may be configured to identify a second location. The second location may be in or may be a second geohash. In various embodiments, the second location may be a defined area such as a housing estate or office buildings or a predefined neighbourhood. In various embodiment, the second location may be defined based on a predetermined radius or distance. The one or more users may be located in the second location.

[0070] In an embodiment, the processor 112 may be configured to predict a dynamic buffer time. The delivery time may include the dynamic buffer time. In an embodiment, the dynamic buffer time may be additional time due to order batching.

[0071] In an embodiment, the processor 112 may be configured to predict the dynamic buffer time based on one or more batch orders between a first bearing of the first location and the second location and a second bearing of the first location and a third location.

[0072] In an embodiment, the first bearing and the second bearing may have an angle difference of 45 degrees.

[0073] In an embodiment, the third location may be in or may be a third geohash. In various embodiments, the third location may be defined based on a predetermined radius or

distance. In an embodiment, the third location may be near the second location wherein the third location may be a predetermined distance away from the second location.

[0074] In an embodiment, the processor 112 may be configured to predict the dynamic buffer time based on at least one of: a proportion of batch orders or a number of batch orders between the first bearing of the first location and the second location and the second bearing of the first location and the third location.

[0075] In an embodiment, the processor 112 may be configured to allow the one or more users in the second location to batch order from the one or more merchants in the first location. In an embodiment, a first user (e.g., user A) in the second location may batch order from a first merchant (e.g., merchant A) and a second merchant (e.g., merchant B) in the first location. In another embodiment, the first user (e.g., user A) and a second user (e.g., user B) may each order from the first merchant (e.g., merchant A). In an embodiment, the first user (e.g., user A) and the second user (e.g., user B) in the second location may each respectively order from the first merchant (e.g., merchant A) and the second merchant (e.g., merchant B) in the first location.

[0076] According to an embodiment, the one or more processor(s) may be configured to assign a same delivery driver for the batch order. In an embodiment, the same delivery driver may be assigned to deliver orders to the one or more users in the second location from the one or more merchants from the first location.

[0077] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time based on contextual information. The contextual information may include one of: an ordering time, a merchant type, a price range and a basket size of an order.

[0078] According to an embodiment, the one or more processor(s) may be configured to predict the dynamic buffer time using machine learning of historical data of single orders.

The historical data may be at least one of: an allocation time prediction (AT), a pick-up routing time (PRT) prediction, a waiting time (WT) prediction, an order preparation time prediction (e.g., a food preparation time (FPT) prediction), and a drop-off routing (DRT) time prediction.

[0079] According to an embodiment, the historical data of single orders may be used as inputs of a machine learning system and/or model. The predicted dynamic buffer time may be the output of the machine learning system. In an embodiment, the system 100 is configured to predict the dynamic buffer time based on the probability of an order getting batched and dynamically adjusted at merchant level in real-time. According to an embodiment, the delivery time may include the dynamic buffer time and one or more of the allocation time prediction (AT), the pick-up routing time (PRT) prediction, the waiting time (WT) prediction, the order preparation time prediction (e.g., the food preparation time (FPT) prediction), and the drop-off routing (DRT) time prediction.

[0080] According to an embodiment, the one or more processor(s) 112 may be configured to perform skewness data transformation on the historical data prior to using the historical data for predicting the dynamic buffer time. In an embodiment, the skewness data transformation may be used to correct right-skewness or left-skewness of the historical data. The one or more processor(s) 112 may be configured to determine whether the historical data is skewed based on mean and median values. For example, if the mean and median values are not the same or not substantially similar, the processor 112 may determine that the data is skewed. The processor may also determine whether the mean and median values are substantially similar based on a pre-determined threshold. That is, if the difference between the mean and median values are within the pre-determined threshold, the processor 112 may determine that the mean and median values are substantially similar. On the other hand, if the difference between the mean and median values are not within the pre-determined threshold,

the processor 112 may determine that the mean and median values are not substantially similar, and the historical data is skewed.

[0081] In an embodiment, an advantage of the present disclosure may include a dynamically adjusted delivery time in real-time resulting in more accurate delivery time predictions for batch orders of around 15% increment in prediction accuracy.

[0082] FIG. 2 shows a flowchart of a method 200 according to various embodiments.

[0083] According to various embodiments, the method 200 for predicting a delivery time for batch orders may be provided. In some embodiments, the method 200 may include a step 202 of using one or more processor(s) of a system (e.g., the system 100) to identify a first location. One or more merchants may be located in the first location. In an embodiment, the method 200 may include a step 204 of using the one or more processor(s) to identify a second location. One or more users may be located in the second location. In an embodiment, the method 200 may include a step 206 of using the one or more processor(s) to predict a dynamic buffer time based on one or more batch orders between a first bearing of the first location and the second location and a second bearing of the first location and a third location. The delivery time may include the dynamic buffer time.

[0084] Steps 202 to 206 are shown in a specific order, however other arrangements are possible, for example, in some embodiments, step 202 may be carried out after step 204. Steps may also be combined in some cases. Any suitable order of steps 202 to 206 may be used.

[0085] FIG. 3 illustrates an exemplary flowchart 300 of various variables of a food delivery time (FDT) according to various embodiments.

[0086] In an embodiment, the FDT prediction may be divided into one or more components such as an allocation time prediction (AT), a pickup routing time prediction (PRT), a waiting time prediction (WT), a food preparation time prediction (FPT), a dropoff routing time prediction (DRT), and a dynamic buffer time prediction (DBT).

[0087] In FIG. 3, a food delivery exemplified, however, other types suitable of delivery orders may be applicable, e.g., grocery orders.

[0088] In an embodiment, in the exemplary flowchart 300, an order 302 may be received. The system 100 may predict an allocation time 304 between the order 302 and the order allocated 306. The allocation time prediction may be a prediction of time required to allocate a driver to the order. The allocation time prediction may be predicted based on supply demand condition.

[0089] In an embodiment, the system 100 may predict a pick-up routing time 308 between the order allocated 306 and a time the allocated driver arrives 310 at the merchant (e.g, restaurant). The pick-up routing time prediction 308 may be a prediction of time required for a driver to travel from his/her current location to restaurant location. The pick-up routing time prediction 308 may be predicted based on at least one of a location, vehicle speed, and traffic condition.

[0090] In an embodiment, the system 100 may predict a food preparation time 312 between the order allocated 306 and a food collection time 314 at the merchant. The food preparation time prediction 312 may be a prediction of time required for required for a merchant to prepare the food. The food preparation time prediction 312 may be predicted based on historical data and/or contextual data.

[0091] In an embodiment, the system 100 may predict a waiting time 316 between the time the allocated driver arrives 310 at the merchant and the food collection time 314 at the merchant. The waiting time prediction 316 may be a prediction of time required for a driver to wait for the food. The waiting time prediction 316 may be predicted based on historical data and/or contextual data.

[0092] In an embodiment, the system 100 may predict a drop-off routing time 318 between the the food collection time 314 and a delivery time 320. The drop-off routing time

prediction 318 may be a prediction of time required for a driver to travel from the restaurant's location to the eater. The drop-off routing time prediction 318 may be predicted based on based on at least one of a location, vehicle speed, and traffic condition.

[0093] In an embodiment, dynamic buffer time prediction (DBT) may be used to predict the delivery time for batch orders. The DBT may be an additional time component to improve FDT prediction for batched orders. In an embodiment, the DBT may be calculated based on the following formulae: $FDT = AT + \max(PRT + WT, FPT) + DRT + DBT$.

[0094] FIG. 4A illustrates an exemplary bearing chart 400 according to various embodiments. FIG. 4B illustrates an exemplary bearing classification table 450 for the exemplary bearing chart 400 of FIG. 4A according to various embodiments.

[0095] In an embodiment, the dynamic buffer time may be a component of estimated time of arrival (ETA) that accounts for extra delivery time due to at least one of order batching, and merchant workload level.

[0096] In an embodiment, the dynamic buffer time may be predicted based on various features such as batching features and/or FDT features. Batching features and FDT features may be aggregated based on historical data. The historical data may be aggregated at predetermined time intervals e.g., at (restaurant x weekday/weekend x 10mins) level.

[0097] In an embodiment, the exemplary bearing chart 400 may include a first location 402. One or more merchants may be located in the first location 402. The exemplary bearing chart 400 may include a second location 404. One or more users may be located in the second location.

[0098] In an embodiment, the exemplary bearing chart 400 may include a first angle 406, a second angle 408, a third angle 410 and a fourth angle 412. The first angle 406 may be perpendicular to the second angle 408 and the fourth angle 412. The first angle 406 may be parallel to the third angle 410 may be perpendicular to each other. The first angle 406 may 0

degrees. The second angle 408 may be 90 degrees. The third angle 410 may be 180 degrees. The fourth angle 412 may be 270 degrees.

[0099] In an embodiment, a bearing is a direction or position, or a direction of movement, relative to a fixed point. In an embodiment, the fixed point may be the first location 402 of the one or more merchants. The bearing may be measured in degrees. In an embodiment, the exemplary bearing chart 400 may include a plurality of bearings (e.g., 8 bearings). Each bearing may have a predetermined angle (e.g., 45 degrees). A total angle of all bearings may be 360 degrees. For example, bearing₁ 414A may have an angle of between 0 to 45 degrees, bearing₂ 414B may have an angle of between 45 to 90 degrees, bearing₃ 414C may have an angle of between 90 to 135 degrees, bearing₄ 414D may have an angle of between 135 to 180 degrees, bearing₅ 414E may have an angle of between 180 to 225 degrees, bearing₆ 414F may have an angle of between 225 to 270 degrees, bearing₇ 414G may have an angle of between 270 to 315 degrees, bearing₈ 414H may have an angle of between 315 to 360 degrees.

[00100] In an embodiment, the one or more processor(s) is configured to predict the dynamic buffer time based on one or more batch orders between a first bearing (e.g., bearing₂ 414B) of the first location 402 and the second location 404 and a second bearing (e.g., bearing₃ 414C) of the first location and a third location. The third location is within the second bearing (e.g., bearing₃ 414C).

[00101] In an embodiment, the batching features may include a proportion of the batched order and/or a number completed of batched orders in a current direction (i.e., the first bearing) and a nearby order direction (i.e., the second bearing), which may be calculated based on the first bearing between the first location of the merchant 402 (Mex) and the second location of the user 404 (Pax) location, and the second bearing of the first location and the third location. The proportion of the batched order may be a proxy of likelihood for an order

being batched. The number completed of batched orders may be a proxy of order density, which correlates with likelihood for an order being batched.

[00102] In an embodiment, the FDT features may include median and/or mean values of the following calculated variable: $FDT - \{AT + \max(PRT + WT, FPT) + DRT\}$. This feature may be a proxy of historical batching buffer time.

[00103] In an embodiment, the contextual features (or contextual information) may include at least one of: a location of a user; ordering time (e.g., hour, day of week, weekday or weekend), a merchant type, a price of the order, and a basket size of the order.

[00104] In an embodiment, real-time features in the ETA formulae like AT, FPT, WT, PRT and DRT may be predicted based on a single order assumption. The predicted values may subsequently be applied as input features for DBT prediction.

[00105] In an embodiment, a machine learning model may be applied to predict the dynamic buffer time (DBT). The machine learning model may predict the DBT based on at least one of: the historical order's FDT, aggregated features and the outputs of other time components.

[00106] For each historical order, the FDT may set as the target output of the ML model, and its corresponding features are set as the input. The predicted output of the ML model may be denoted as $f(input)$. The model training process may try to minimize the loss between the target and predicted output. The loss function $l(order)$ may be a standard Mean squared error (MSE): $l(order) = (f(input) - FDT)^2$.

[00107] The ML model may automatically achieve a optimized model $f(input)$ with minimized loss $l(order)$ based on the historical orders by repeating the above training process.

[00108] Variable ML models may be applied for this task. For example: Gradient Boosting Decision Tree (GBDT), Neural Networks (NNs), and Logistic Regression (LR).

[00109] FIG. 5A illustrates an exemplary first graph showing symmetric distribution data according to various embodiments. FIG. 5B illustrates an exemplary second graph showing right-skewed distribution data according to various embodiments. FIG. 5C illustrates an exemplary third graph showing left-skewed distribution data according to various embodiments.

[00110] In FIG. 5A the exemplary first graph 500 shows symmetric distribution data, where a mean and a median value is substantially similar.

[00111] In FIG. 5B, a mean value is larger than a median value which may lead to a long tail on the right side of a data center of graph 510.

[00112] In FIG. 5C, a median value is larger than a mean value which may lead to a long tail on the left side of a data center of graph 520.

[00113] According to data characteristics, transformation techniques may be applied to avoid skewness of the model input and output. The transformation methodology may be used to overcome skewness of the data (such as long tail on the left/right side of the data center).

[00114] In an embodiment, the input features may be transformed. The value for other components of FDT (including AT, FPT, WT, PRT and DRT) may be heavily right-skewed due to a long tail in the right part of the center or left-skewed due to long tail in the left part of the center.

[00115] In an embodiment, if the raw features (e.g., skewed data) as applied as inputs, the prediction accuracy of DBT will be affected. A log-transformation may be conducted for these inputs:

$f(x) = \log(\alpha + x)$, where α instead of a constant value 1 (in most of traditional applications 1 is applied) is a variable used to control the transformation, which could be 5 percentile or 10 percentile of the corresponding input, leading to better performances in reducing the skewness of the data.

[00116] In an embodiment, an output (e.g., target output) may be transformed as the skewness problem may occur in the output: FDT. For example, due to the longer FDT for batched order, which may introduce a long tail in the right part of the data center. The transformation may be applied to the output. The ML model may be used to predict the transformed values. The transformed values may be transformed back as the final prediction of DBT.

[00117] The transformation function is not limited to log-transformation. Any other suitable ways may be used, such as square-root transformation, and Box-Cox transformation.

[00118] FIG. 6 illustrates an exemplary machine learning flowchart 600 according to various embodiments.

[00119] In a first step S1 602, data may be prepared. In this step, data of historical orders (e.g., from the last one month) may be collected. The data of historical orders may include details of food delivery time.

[00120] The collected data may be used as training data 604 and may randomly sampled and/or splitted in training, validation and test sets, with a predetermined ratio (e.g., 60:10:30).

[00121] In a second step S2 606, based on the data collected, main features such as AT 608, FPT 610, WT 612, PRT 614, DRT 616, batching features 618, FDT features 620 and contextual features 622 may be obtained by statistically calculation or aggregation according to various embodiments.

[00122] In a third step S3 624, a suitable model 626 may be chosen. For example, Gradient-Boosted Decision Trees (GBDT) may be used as the model.

[00123] In the training, a validation set may be used to incrementally improve the model's ability to predict the actual FDT.

[00124] Once training is completed, the performance may be evaluated with a test dataset. Evaluation allows testing of the model against data that has never been used for training.

[00125] In a fourth step S4 630, deployment 630 may be conducted. The ML model may be integrated into an existing production environment, where the ML model accepts the features as input and returns DBT prediction. The purpose of this step is to make the predictions from a trained ML model as a service available to others via an application programming interface (API).

[00126] While the present disclosure has been particularly shown and described with reference to specific aspects, it should be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the scope of the present disclosure as defined by the appended claims. The scope of the present disclosure is thus indicated by the appended claims and all changes which come within the meaning and range of equivalency of the claims are therefore intended to be embraced.

CLAIMS

What is claimed is:

1. A system for predicting a delivery time for batch orders, the system comprising:
one or more processor(s); and
a memory having instructions stored therein, the instructions, when executed by the one or more processor(s), cause the one or more processor(s) to:
identify a first location, wherein one or more merchants are located in the first location;
identify a second location, wherein one or more users are located in the second location;
predict a dynamic buffer time;
wherein the delivery time comprises the dynamic buffer time, and
wherein the one or more processor(s) is configured to predict the dynamic buffer time based on one or more batch orders between a first bearing of the first location and the second location and a second bearing of the first location and a third location.
2. The system of claim 1, wherein the one or more processor(s) is configured to predict the dynamic buffer time based on at least one of: a proportion of batch orders or a number of batch orders between the first bearing of the first location and the second location and the second bearing of the first location and the third location.
3. The system of claim 1 or 2, wherein the dynamic buffer time is additional time due to order batching.

4. The system of any one of claims 1-3, wherein the first bearing and the second bearing have an angle difference of 45 degrees.
5. The system of any one of claims 1-4, wherein the one or more processor(s) is configured to allow the one or more users in the second location to batch order from the one or more merchants in the first location.
6. The system of claim 5, wherein the one or more processor(s) is configured to assign a same delivery driver for the batch order.
7. The system of any one of claims 1-6,
wherein the one or more processor(s) is configured to predict the dynamic buffer time based on contextual information, and
wherein the contextual information comprises one of: an ordering time, a merchant type, a price range and a basket size of an order.
8. The system of any one of claims 1-7,
wherein the one or more processor(s) is configured to predict the dynamic buffer time using machine learning of historical data of single orders, and
wherein the historical data comprises at least one of: an allocation time prediction, a pick-up routing time prediction, a waiting time prediction, an order preparation time prediction, and a drop-off routing time prediction.

9. The system of any one of claims 1-8, wherein the one or more processor(s) is configured to perform skewness data transformation on the historical data prior to using the historical data for predicting the dynamic buffer time.

10. A method for predicting a delivery time for batch orders, the method comprising using one or more processor(s) to:

identify a first location, wherein one or more merchants are located in the first location;

identify a second location, wherein one or more users are located in the second location;

predict a dynamic buffer time;

wherein the delivery time comprises the dynamic buffer time, and

wherein the one or more processor(s) is configured to predict the dynamic buffer time based on one or more batch orders between a first bearing of the first location and the second location and a second bearing of the first location and a third location.

11. The method of claim 10, comprising using the one or more processor(s) to:

predict the dynamic buffer time based on at least one of: a proportion of batch orders or a number of batch orders between the first bearing of the first location and the second location and the second bearing of the first location and the third location.

12. The method of claim 10 or 11, wherein the dynamic buffer time is additional time due to order batching.

13. The method of any one of claims 10-12, wherein the first bearing and the second bearing have an angle difference of 45 degrees.

14. The method of any one of claims 10-13, comprising using the one or more processor(s) to:

allow the one or more users in the second location to batch order from the one or more merchants in the first location.

15. The method of claim 14, comprising using the one or more processor(s) to assign a same delivery driver for the batch order.

16. The method of any one of claims 10-15, comprising using the one or more processor(s) to:

predict the dynamic buffer time based on contextual information;

wherein the contextual information comprises one of: an ordering time, a merchant type, a price range and a basket size of an order.

17. The method of any one of claims 10-16, comprising using the one or more processor(s) to:

predict the dynamic buffer time using machine learning of historical data of single orders;

wherein the historical data comprises at least one of: an allocation time prediction, a pick-up routing time prediction, a waiting time prediction, an order preparation time prediction, and a drop-off routing time prediction.

18. The method of any one of claims 10-17, comprising using the one or more processor(s) to perform skewness data transformation on the historical data prior to using the historical data for predicting the dynamic buffer time.

19. A non-transitory computer-readable medium storing computer executable code comprising instructions for predicting a delivery time for batch orders according to any one of claims 1 to 18.

20. A computer executable code comprising instructions for predicting a delivery time for batch orders according to any one of claims 1 to 19.