

**SERVER AND METHOD FOR FACILITATING COLLECTING INFORMATION
ABOUT INCIDENT**

TECHNICAL FIELD

[0001] Various embodiments relate to a server and a method for facilitating collecting information about an incident.

BACKGROUND

[0002] Due to development of information technology, a user (also referred to as a “requester”) 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, using a user interface presented on the computing device. After a server for the on-demand service receives an order (also referred to as a “booking”) for the on-demand service from the user, the server may allocate a driver (who in some contexts herein may also be referred to as a “delivery service provider”, a “delivery partner” or a “delivery agent”) to the received order, and the driver may perform the on-demand service for the received order. For example, the driver may move to a destination (that in some contexts herein may also be referred to as a “drop-off point”, a “delivery point”, a “delivery location” or a “meeting point”) for the received order.

[0003] While the driver moves to the destination, an on-demand service platform provider may collect information about an incident from the driver. A real-time incident (RTI) reporting may be a part of a crowdsourcing strategy at the on-demand service platform

provider for collecting ground-truth information by a driver community. The value of collecting such real-time incidents on a ground may be tremendous to the on-demand service platform provider as well as a map company. This crowdsourcing approach may help establish real-time traffic, event information and accurate maps, so that high quality map services, for example, including an ETA (estimated time of arrival), routing, POI (point of interest) search, may be provided.

[0004] Conventionally, the on-demand service platform provider may have built an in-App (also referred to as an “application”) reporting function for the driver to contribute the information about the incident on the way to the destination. For example, in the App provided by the on-demand service platform provider, the driver may report the incident, for example, a road closure, a map issue, a car accident, and a traffic congestion, via a form-based submission flow in the App, for example, by clicking through a few buttons to find the most relevant incident category to report. At a backend, a server of the on-demand service platform provider may collect the reports to update the traffic or send a processed incident report to an internal stakeholder, for example, a map operation team, for a validation.

[0005] FIGS. 1A and 1B illustrate exemplary diagrams showing a user interface of a computing device associated with the driver according to the conventional technology. As shown in FIG. 1A(a), the computing device may display a map relating to a route to the destination for providing the on-demand service, en route to the destination. The map may include an image object (for example, a pin) indicating the destination. The map may further include information about a current road name, an estimated time or arrival, and a distance to the destination. The map may further include an image object 401 to request to send an incident report.

[0006] If the driver touches the image object 401 to request to send the incident report, the computing device may display a pop-up window 402 to select a category of the incident (also

referred to as an “incident category” or an “incident type”), as shown in FIG. 1A(b). For example, the pop-up window 402 may include at least one of a first image object 403 for reporting an existence of a police officer, a second image object 404 for reporting a map issue, a third image object 405 for reporting a congestion, a fourth image object 406 for reporting a road closure, a fifth image object 407 for reporting a car accident, a sixth object 408 for reporting a road condition, and a seventh object 409 for reporting a need of adding a place.

[0007] If the driver, for example, touches the second image object 404 for reporting the map issue, the computing device may display a pop-up window 410 to select a sub-category of the incident (also referred to as an “incident sub-category” or an “incident sub-type”) of the map-issue, as shown in FIG. 1B(a). For example, the pop-up window 410 may include at least one of a first image object 411 for reporting that no turns are allowed, a second image object 412 for reporting a one-way road, and a third image object 413 for reporting an incorrect road name.

[0008] If the driver, for example, touches the third image object 413 for reporting the incorrect road name, the computing device may send the report for the incorrect road name to the server of the on-demand service platform provider, and display a pop-up window 414 for an acknowledgement message (for example, “thank you for helping other drivers”).

[0009] However, the conventional in-App incident reporting may mainly rely on the form-based process through manual clicks and/or touches. There may be a few drawbacks of this approach, as follows:

- a) The driver may need to match the incident they see to the incident type defined in the App. Somehow, it may be stressful for the driver to do a correct matching. In addition, there may be an incorrect incident scenario mapping due to the driver’s misunderstanding.

- b) Conventionally, each submission may mean to report one incident only. However, in the real world, one event may lead to multiple incidents. Oftentimes, the car accident may cause traffic jams and road closure. In this case, the driver may need to submit multiple reports separately, which may not be practical to do voluntarily.
- c) The driver may need to know how to find a correct incident type in a hierarchical reporting structure in the App. For example, in order to report a pothole, the driver may need to click through Hazard, On road, Pothole in a submission flow of the App. It may not be an easy thing for a new driver to know a right way of reporting without familiarity with the App.
- d) Frequent button clicking on the computing device may lead to a distraction of driving of a vehicle, which may be dangerous to the driver.

[0010] Therefore, there is a need to provide a solution for facilitating collecting the information about the incident from the driver, while the driver moves to the destination.

SUMMARY

[0011] Various embodiments may address the above-mentioned problems by providing a hands-free operation through a voice interaction. The various embodiments may allow a driver to report an incident using a plain language, reducing mental efforts required for mapping incident categories and significantly improving the driver's safety. In addition, the various embodiments may be powered by a Generative AI (Artificial Intelligence) to help an efficient and accurate communication, while keeping the driver focused on a road. The various embodiments may improve an RTI quality and a validation speed by making crowdsourcing signals fully automatic and real-time.

[0012] Moreover, according to the various embodiments, during the driver's daily transport and delivery tasks, the driver may need real-time information to boost efficiency, such as traffic conditions, parking locations, and building entrances. A voice UI (User Interface) provided by the various embodiments may provide the real-time information without distracting the driver from driving the vehicle, ensuring the driver's safety while being highly useful. Due to the convenience in the incident reporting, the driver may be encouraged to report the incident that the driver is witnessing, and thus an on-demand service platform provider may easily gather information about the incident and gather a feedback from the driver, and thus a map provided by the on-demand service platform provider may be improved.

[0013] According to various embodiments, there is a server for facilitating collecting information about an incident, the server comprising: a memory configured to store instructions; and a processor for executing the stored instructions and configured to: receive a request to initiate an incident reporting mode from a driver via a computing device associated with the driver; receive a message for the incident from the driver via the computing device; identify an intention from the message; classify the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention; assign the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message to collect the information about the incident; and provide output information based on the collected information about the incident.

[0014] In some embodiments, the processor is further configured to: receive a voice command or a gesture command from the driver via the computing device; determine if the received command relates to a predetermined command; and if it is determined that the

received command relates to the predetermined command, determine the received command as the request to initiate the incident reporting mode.

[0015] In some embodiments, the message includes a voice message, and the processor is further configured to convert the voice message into a text message.

[0016] In some embodiments, the processor is further configured to: merge the text message with a pre-written prompt message to generate a merged message including an instruction to guide a Large Language Model (LLM); input the merged message to the LLM; and control the LLM to identify the intention from the text message and classify the text message into the one or more incident categories among the plurality of predetermined incident categories.

[0017] In some embodiments, the processor is further configured to control the LLM to output at least one of the one or more incident categories that the incident belongs to, an inference logic, a follow-up question to the driver, and an acknowledgement.

[0018] In some embodiments, the processor is further configured to control the computing device to inform the driver of the output of the LLM.

[0019] In some embodiments, the processor is further configured to request the driver to provide additional information about the incident via the computing device, based on the follow-up question output from the LLM.

[0020] In some embodiments, the processor is further configured to activate an image capturing device to obtain an image capturing an external environment, to supplement the text message for the incident, based on the follow-up question output from the LLM.

[0021] In some embodiments, the processor is further configured to send an acknowledgement message to the computing device to end the incident reporting mode, based on the acknowledgement output from the LLM.

[0022] In some embodiments, the processor is further configured to broadcast the incident to one or more computing devices each associated with one or more drivers located in a predetermined range of the driver.

[0023] In some embodiments, the output information includes updating a display of one or more computing devices each associated with one or more subsequent drivers on a same route, based on the collected information about the incident.

[0024] According to various embodiments, there is a method for facilitating collecting information about an incident, the method comprising: receiving a request to initiate an incident reporting mode from a driver via a computing device associated with the driver; receiving a message for the incident from the driver via the computing device; identifying an intention from the message; classifying the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention; assigning the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message to collect the information about the incident; and providing output information based on the collected information about the incident.

[0025] In some embodiments, the method further comprises: receiving a voice command or a gesture command from the driver via the computing device; determining if the received command relates to a predetermined command; and if it is determined that the received command relates to the predetermined command, determining the received command as the request to initiate the incident reporting mode.

[0026] In some embodiments, the message includes a voice message, and the method further comprises: converting the voice message into a text message.

[0027] In some embodiments, the method further comprises: merging the text message with a pre-written prompt message to generate a merged message including an instruction to guide a

Large Language Model (LLM); inputting the merged message to the LLM; and controlling the LLM to identify the intention from the text message and classify the text message into the one or more incident categories among a plurality of predetermined incident categories.

[0028] In some embodiments, the method further comprises: controlling the LLM to output at least one of the one or more incident categories that the incident belongs to, an inference logic, a follow-up question to the driver, and an acknowledgement.

[0029] In some embodiments, the method further comprises: controlling the computing device to inform the driver of the output of the LLM.

[0030] In some embodiments, the method further comprises: requesting the driver to provide additional information about the incident via the computing device, based on the follow-up question output from the LLM.

[0031] In some embodiments, the method further comprises: activating an image capturing device to obtain an image capturing an external environment, to supplement the text message for the incident, based on the follow-up question output from the LLM.

[0032] In some embodiments, the method further comprises: sending an acknowledgement message to the computing device to end the incident reporting mode, based on the acknowledgement output from the LLM.

[0033] In some embodiments, the method further comprises: broadcasting the incident to one or more computing devices each associated with one or more drivers located in a predetermined range of the driver.

[0034] In some embodiments, the output information includes updating a display of one or more computing devices each associated with one or more subsequent drivers on a same route, based on the collected information about the incident.

[0035] According to various embodiments, a data processing apparatus configured to perform the method of any one of the above embodiments is provided.

[0036] 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.

[0037] 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

[0038] 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:

- FIGS. 1A and 1B illustrate exemplary diagrams showing a user interface of a computing device associated with a driver according to a conventional technology.
- FIGS. 2 and 3 illustrate infrastructures of a system including a server for facilitating collecting information about an incident according to various embodiments.
- FIG. 4 illustrates a block diagram of a server for facilitating collecting information about an incident according to various embodiments.
- FIG. 5 illustrates a flowchart for a method for facilitating collecting information about an incident according to various embodiments.
- FIGS. 6A and 6B illustrate exemplary diagrams showing a user interface of a computing device associated with a driver according to various embodiments.
- FIG. 7 illustrates a data flow diagram of a server for facilitating collecting information about an incident according to various embodiments.

- FIG. 8 illustrates a data flow diagram of a Large Language Model (LLM), a plurality of predetermined agents, and a geo data/service layer for facilitating collecting information about an incident according to various embodiments.
- FIG. 9 illustrates a block diagram of a processor of a server for facilitating collecting information about an incident according to various embodiments.

DETAILED DESCRIPTION

[0039] 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.

[0040] 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.

[0041] 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.

[0042] 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.

[0043] As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0044] 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.

[0045] In the following, embodiments will be described in detail.

[0046] FIGS. 2 and 3 illustrate infrastructures of a system 200 including a server 100 for facilitating collecting information about an incident according to various embodiments.

[0047] Throughout the description, the incident (also referred to as an “event”), may refer to an instance of something happening while driving. In some embodiments, the incident may include, but is not limited to, an existence of a police officer, a map issue, a congestion, a road closure, a car accident, a road condition, and a missing place.

[0048] 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 first computing device 160 associated with a user 161 (also referred to as a “requester”), a plurality of second computing devices 170 (not shown) each associated with a plurality of drivers 171 (also referred to as a “delivery service provider”, 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 (for example, “food providers” or “restaurants”). In some embodiments, the user 161 may include a consumer (also referred to as an “eater”) for the on-demand service. For example, the user

161 may be the same as the consumer. As another example, the user 161 may be different from the consumer and use the on-demand service for the consumer.

[0049] In some embodiments, the on-demand service may be a service allowing the user 161 to fulfill the user's demand via an immediate access to items and/or services. The user 161 may request the on-demand service, such as a transport service or an item delivery service, using a user interface presented on the computing device 160. The user 161 may make an order for the on-demand service.

[0050] In some embodiments, the user 161 may use an application (also referred to as an "App"), 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 160 associated with the user 161, to interact with the server 100.

[0051] In some embodiments, the plurality of drivers 171 may use the application provided by the server 100. The application may be installed in the plurality of second computing devices 170 each associated with the plurality of drivers 171, to interact with the server 100.

[0052] 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 first computing device 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 160 and a second computing device 170a of the plurality of second computing devices 170.

[0053] In some embodiments, the first computing device 160 may be connectable to the server 100 via the network 150. In some embodiments, the first computing device 160 may be arranged in data or signal communication with the server 100 via the network 150. In some embodiments, the first computing device 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 first computing device 160 may be associated with the user 161. For example, the first computing device 160 may belong to the user 161 who is the consumer. As another example, the first computing device 160 may belong to the user 161 requesting the delivery of the item to the consumer who is a recipient of the on-demand service. Although not shown, it may be appreciated that the system 200 may further include a plurality of first computing devices each associated with, for example, belonging to, a plurality of users.

[0054] In some embodiments, the first computing device 160 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 160. In some embodiments, the first computing device 160 may generate information about the location of the first computing device 160.

[0055] 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).

[0056] In some embodiments, the server 100 may communicate with the first computing device 160 via the network 150. In some embodiments, the first computing device 160 may receive a request (hereinafter, referred to as an “order”) from the user 161 for the on-demand service. The first computing device 160 may send the order to the server 100 via the network 150. In some embodiments, the first computing device 160 may send the information about

the location of the first computing device 160 to the server 100 via the network 150. The location of the first computing device 160 may be considered as a location of the user 161. In some embodiments, the location of the user 161 may be considered as a destination of the on-demand service. In some other embodiments, the first computing device 160 may send information about an address of the user 161, and the address of the user 161 may be considered as the destination of the on-demand service. In some other embodiments, the first computing device 160 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.

[0057] 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.

[0058] 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. The plurality of second computing devices 170 may be associated with the plurality of drivers 171 respectively. For example, the plurality of second computing devices 170 may belong to the plurality of drivers 171 respectively.

[0059] In some embodiments, the server 100 may receive the order from the first computing device 160. After the server 100 receives the order from the first computing device 160, the

server 100 may allocate (assign) the order to a suitable driver 171a. In some embodiments, the second computing device 170a associated with the driver 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 driver 171a. In some embodiments, the location of the driver 171a may be considered as a current location of the driver 171a, and may change while the driver 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 driver 171a) to the destination for providing the on-demand service, en route to the destination. In some embodiments, the map may include an image object, for example, a pin, indicating the destination.

[0060] 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. 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.

[0061] FIG. 4 illustrates a block diagram of a server 100 for facilitating collecting information about an incident according to various embodiments.

[0062] 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.

[0063] 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.

[0064] In some embodiments, the communication interface 110 may allow one or more computing devices, including a first computing device 160, to communicate with the processor 120 of the server 100 via a network 150, as shown in FIG. 2. In some embodiments, as shown in FIG. 2, the first computing device 160 may belong to a user 161 who wants to make an order for an on-demand service. In some embodiments, the communication interface 110 may transmit signals to the first computing device 160, and/or receive signals from the first computing device 160 via the network 150.

[0065] 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 drivers 171 who may pick up an item from an item service 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.

[0066] 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 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.

[0067] 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.

[0068] 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.

[0069] In some embodiments, the processor 120 may receive a request to initiate an incident reporting mode from the driver 171a via the second computing device 170a associated with the driver 171a. In some embodiments, the communication interface 110 may receive the request to initiate the incident reporting mode from the driver 171a via the second computing device 170a, and the processor 120 may receive the request to initiate the incident reporting mode from the communication interface 110.

[0070] In some embodiments, the processor 120 may receive a contactless command (also referred to as a “touchless command”) from the driver 171a via the second computing device 170a to initiate the incident reporting mode. In some embodiments, the processor 120 may

receive a voice command or a gesture command from the driver 171a via the second computing device 170a. In some embodiments, where the processor 120 receives the voice command, the processor 120 may determine if the received voice command relates to a predetermined voice command. In some embodiments, if it is determined that the received voice command relates to the predetermined voice command, the processor 120 may determine the received voice command as the request to initiate the incident reporting mode. In some other embodiments, where the processor 120 receives the gesture command, the processor 120 may determine if the received gesture command relates to a predetermined gesture command. In some embodiments, if it is determined that the received gesture command relates to the predetermined gesture command, the processor 120 may determine the received gesture command as the request to initiate the incident reporting mode.

[0071] In some embodiments, additionally or alternatively, the processor 120 may receive a contact command from the driver 171a via the second computing device 170a to initiate the incident reporting mode. In some embodiments, the processor 120 may receive a touch input or a click input on a certain icon displayed on the second computing device 170a from the driver 171a. In some other embodiments, the processor 120 may receive a knock input on a screen of the second computing device 170a from the driver 171a. In some embodiments, where the processor 120 receives the contact command, the processor 120 may determine if the received contact command relates to a predetermined contact command. In some embodiments, if it is determined that the received contact command relates to the predetermined contact command, the processor 120 may determine the received contact command as the request to initiate the incident reporting mode.

[0072] In some embodiments, the processor 120 may activate the incident reporting mode, in response to the request to initiate the incident reporting mode. In some embodiments, the processor 120 may receive a message for the incident from the driver 171a, via the second

computing device 170a. In some embodiments, while the incident reporting mode is activated, the processor 120 may record the message for the incident received from the driver 171a via the second computing device 170a.

[0073] In some embodiments, the message may include a voice message. In some embodiments, in the incident reporting mode, the processor 120 may activate detecting an audio signal from an external environment. In some embodiments, in the incident reporting mode, the processor 120 may further activate recording the audio signal from the external environment. For example, in the incident reporting mode, the processor 120 may activate detecting the voice message from the driver 171a. For example, in the incident reporting mode, the processor 120 may activate recording the voice message received from the driver 171a.

[0074] In some embodiments, the processor 120 may convert the voice message into a text message. In some embodiments, the processor 120 may convert the voice message received from the driver 171a into the text message using a speech-to-text recognition technology (for example, using the same language). In some embodiments, the processor 120 may use the text message to identify an intention from the text message.

[0075] In some embodiments, the processor 120 may not convert the voice message into the text message. In some embodiments, the processor 120 may use the voice message to identify the intention from the voice message.

[0076] In some embodiments, the message may include any other type of the message, for example, a text message or a gesture message. In some embodiments, the processor 120 may convert the gesture message into the text message, and use the text message to identify the intention from the text message. In some other embodiments, the processor 120 may not convert the gesture message, and use the gesture message to identify the intention from the gesture message, for example, using a Large Language Model (LLM).

[0077] In some embodiments, the processor 120 may identify the intention from the message. In some embodiments, the processor 120 may identify the driver's 171 intention from the voice message, without converting to the text message. For example, the processor 120 may use at least one of the following technologies/models to identify the driver's 171 intention from the voice message:

- a) Acoustic Feature Extraction: The processor 120 may extract features such as Mel-Frequency Cepstral Coefficients (MFCCs), spectrograms, pitch, energy, and formants from the voice message (in the form of audio signal). These features may capture characteristics of the audio signal and may be used for further processing.
- b) Phoneme Recognition: The processor 120 may break down the audio signal into phonemes, which are the smallest units of sound in a language. The processor 120 may use phoneme recognition models to identify sequences of phonemes without converting the audio signal to text.
- c) Deep Learning Models: The processor 120 may use Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to process raw audio signals and learn representations that correspond to spoken words. Models like WaveNet or wav2vec may be trained directly on audio signal for word recognition.
- d) End-to-End Speech Models: The processor 120 may develop end-to-end models that take raw audio signals as input and output word-level predictions directly. These models may learn to map the audio signals to word labels without intermediate text representation.
- e) Hidden Markov Models (HMMs): The processor 120 may use HMMs to model the temporal sequence of acoustic features. The processor 120 may combine the HMMs with Gaussian Mixture Models (GMMs) or deep neural networks to improve word recognition accuracy.

- f) Connectionist Temporal Classification (CTC): The processor 120 may use CTC loss in neural networks to allow the model to learn alignments between audio frames and word labels. This technique may be useful for handling variable-length input and output sequences.
- g) Attention Mechanisms: The processor 120 may implement attention mechanisms in neural networks to focus on important parts of the audio signal that correspond to specific words. This may help in capturing context and improving word recognition.
- h) Speaker Embeddings: The processor 120 may generate embeddings (fixed-size representations) of the speaker's voice using models like x-vectors or d-vectors. The processor 120 may use these embeddings as input to classifiers to determine words.
- i) Dynamic Time Warping (DTW): The processor 120 may use DTW to measure similarities between different sequences of acoustic features. This technique may align and compare spoken word patterns directly.
- j) Spectrogram Analysis: The processor 120 may convert the audio signal into a spectrogram, which is a visual representation of the spectrum of frequencies. The processor 120 may use image recognition techniques, such as CNNs, to identify patterns in the spectrogram that correspond to specific words.
- k) Large Language Model (LLM): The processor 120 may use the LLM (for example, GPT-4o) configured to understand the audio signal and perform a task (for example, generate human-like text). The LLM may be based on deep learning techniques.

[0078] In some embodiments, the processor 120 may identify the driver's 171a intention from the message which is converted from the voice message. In some embodiments, the processor 120 may merge the text message with a pre-written prompt message to generate a merged message. In some embodiments, the merged message may include an instruction to guide the LLM, so that the LLM may identify the intention from the text message. In some

embodiments, the LLM may refer to a computational model configured to achieve general-purpose language generation and natural language processing tasks such as a classification. It may be appreciated that the LLM may also be referred to as an artificial neural network. In some embodiments, the processor 120 may include the LLM, and thus the LLM may be a part of the processor 120. In some other embodiments, the LLM may be external to the processor 120, and the processor 120 may communicate with the LLM via the communication interface 110.

[0079] In some embodiments, the processor 120 may input the merged message to the LLM. In some embodiments, the processor 120 may control the LLM to identify the intention from the text message. In some embodiments, the LLM may receive the merged message including the instruction from the processor 120, and identify the intention from the text message by analysing and interpreting the text message. As an example, the text message may relate to a single intention. As another example, the text message may relate to multiple intentions. For example, the intentions may include, but are not limited to, a road incident reporting, a traffic reporting, and a nearby POI search.

[0080] In some embodiments, the processor 120 may classify the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention. In some embodiments, the processor 120 may control the LLM to classify the message into the one or more incident categories among a plurality of predetermined incident categories, based on the identified intention. In some embodiments, the LLM may classify the message into the one or more incident categories, based on the identified intention. As an example, the message may be classified to a single incident category corresponding to the single intention. As another example, the message may be classified to multiple incident categories each corresponding to the multiple intentions. For

example, the predetermined incident categories may include, but are not limited to, a road incident reporting category, a traffic reporting category, and a nearby POI search category.

[0081] In some embodiments, the processor 120 may assign the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message and collect the information about the incident. In some embodiments, the plurality of predetermined agents may be a plurality of modules for performing assigned tasks. In some embodiments, the processor 120 may include the plurality of predetermined agents, and thus the plurality of predetermined agents may be a part of the processor 120. In some other embodiments, the plurality of predetermined agents may be external to the processor 120, and the processor 120 may communicate with the plurality of predetermined agents via the communication interface 110. In some embodiments, the LLM may communicate with the plurality of predetermined agents. In some embodiments, the processor 120 may communicate with the plurality of predetermined agents via the LLM.

[0082] For example, the plurality of predetermined agents may include, but are not limited to, an RTI agent corresponding to the road incident reporting category, a traffic agent corresponding to the traffic reporting category, and a place agent corresponding to the nearby POI search category. In some embodiments, each of the predetermined agents may be specialised in the incident category corresponding to that predetermined agent. In some embodiments, each of the predetermined agents may be specialised in a conversational follow-up with the driver 171a to collect the additional information about the incident from the driver 171a.

[0083] In some embodiments, the processor 120 may assign the message to one or more agents corresponding to the classified incident categories, according to the LLM's classification. As an example, the message may be assigned to a single agent corresponding to the single incident category, according to the LLM's classification. As another example,

the message may be assigned to multiple agents each corresponding to the multiple incident categories, according to the LLM's classification.

[0084] In some embodiments, the agent that the message has been assigned may analyse the message and collect the information about the incident. In some embodiments, the processor 120 may provide output information (for example, updating the map, sending an automated message to drivers in the vicinity, requesting drivers in the vicinity to provide an update on the incident, updating a map for subsequent drivers, etc.) based on the collected information about the incident (for example, depending on the nature and/or characteristics of the incident). In some embodiments, the output information updating a display of one or more computing devices each associated with one or more subsequent drivers on a same route as that of the driver 171a, based on the collected information about the incident. For example, the map displayed on the one or more computing devices each associated with the one or more subsequent drivers on the same route may be updated with the information about the incident reported by the driver 171a.

[0085] In some embodiments, the agent that the message has been assigned may confirm the information about the incident and use the collected information about the incident to update the map. In some embodiments, the agent that the message has been assigned may inform a geo data/service layer to update the map, and the geo data/service layer may update the map accordingly (as will be described with reference to FIG. 8). In some other embodiments, the agent that the message has been assigned may collect additional information about the incident. In some embodiments, the agent that the message has been assigned may determine if the additional information about the incident is required to update the map based on the information about the incident, and determine whether to collect the additional information about the incident. In some embodiments, the agent that the message has been assigned may inform the geo data/service layer to update the map according to the confirmed information

about the incident, and the geo data/service layer may update the map accordingly, so that the second computing devices 170 may display the updated map.

[0086] As an example, if the message, for example, the text message is “gate closed 10:00 p.m. to 4 a.m.”, the LLM may classify the text message into “ROAD_CLOSURE – GATE_CLOSED”, and assign the text message to a relevant agent. As another example, if the text message is “the road is slippery”, the LLM may classify the text message into “ROAD_CONDITION – HAZARD”, and assign the text message to a relevant agent. As another example, if the text message is “Road close school days”, the LLM may classify the text message into “ROAD_CLOSURE”, and assign the text message to a relevant agent. As another example, if the text message is “The gate on the other side is closed”, the LLM may classify the text message into “ROAD_CLOSURE – GATE_CLOSED”, and assign the text message to a relevant agent. As another example, if the text message is “There is a fee for the subdivision gate”, the LLM may classify the text message into “TOLL FEE”, and assign the text message to a relevant agent. As another example, if the text message is “Leaves an ID here at BF Resort”, the LLM may classify the text message into “DELIVERY DIFFICULTY”, and assign the text message to a relevant agent. As another example, if the text message is “The merchant is still far from the pin”, the LLM may classify the text message into “LOCATION ACCURACY”, and assign the text message to a relevant agent. As another example, if the text message is “Add place ABC Cafe”, the LLM may classify the text message into “MISSING POI”, and assign the text message to a relevant agent. As described above, in some embodiments, the text message may be assigned to the relevant agent specialised in the incident category that the text message belongs. Advantageously, efficiency and accuracy in collecting the information about the incident may be improved.

[0087] In some embodiments, if the message is in foreign language, the LLM may translate the message, and perform the above-mentioned operations.

[0088] In some embodiments, the processor 120 may control the LLM to output at least one of the one or more incident categories that the incident belongs to, an inference logic, a follow-up question to the driver 171a, and an acknowledgement. In some embodiments, the inference logic may refer to a summary of a ground that the message is classified as the incident category. In some embodiments, the follow-up question to the driver may refer to one or more questions to obtain the additional information about the incident. For example, the one or more questions may relate to more detailed situations about the car accident (for example, an accurate location of the car accident, how many lanes are blocked due to the car accident, etc.). In some embodiments, the acknowledgement may refer to an acknowledgment message to end the incident reporting mode. For example, if the information about the incident that is obtained is enough to update the map, the acknowledgement may be output from the LLM.

[0089] In some embodiments, the processor 120 may control the second computing device 170a to inform the driver 171a of the output of the LLM. In some embodiments, the second computing device 170a may inform the driver 171a of the at least one of the one or more incident categories that the incident belongs to, the inference logic, the follow-up question to the driver 171a, and the acknowledgement. In some embodiments, the processor 120 may control the second computing device 170a to inform the output of the LLM audibly and/or visually. As an example, the second computing device 170a may output an audio signal to inform the driver 171a of the output of the LLM. As another example, the second computing device 170a may display the output of the LLM on the screen of the second computing device 170a. As another example, the second computing device 170a may output the audio signal and display the output of the LLM on the screen of the second computing device 170a at the same time.

[0090] In some embodiments, the processor 120 may request the driver 171a to provide additional information about the incident via the second computing device 170a, based on the follow-up question output from the LLM. In some embodiments, the follow-up question may be determined by the agent that the message has been assigned. In some embodiments, the processor 120 may ask the question to the driver 171a via the second computing device 170a. In some embodiments, the processor 120 may receive an additional voice message for the incident from the driver 171a, in response to the question, via the second computing device 170a. In some embodiments, the processor 120 may record the additional voice message for the incident received from the driver 171a via the second computing device 170a. In some embodiments, the processor 120 may control the agent that the message has been assigned to determine whether the additional voice message for the incident is enough to update the map. If it is determined that the additional voice message for the incident is enough to update the map, the agent that the message has been assigned may inform the LLM to output the acknowledgement, and the second computing device 170a may output the acknowledgment message to end the additional incident reporting mode.

[0091] In some embodiments, the processor 120 may obtain the additional information about the incident from the external environment. In some embodiments, the processor 120 may obtain the additional information about the incident from the external environment, based on the follow-up question output from the LLM. In some embodiments, the follow-up question may be determined by the agent that the message has been assigned. As an example, the processor 120 may obtain the additional information about the incident with requesting the driver 171a to provide the additional information about the incident via the second computing device 170a. As another example, the processor 120 may obtain the additional information about the incident without requesting the driver 171a to provide additional information about the incident via the second computing device 170a. In some embodiments, the processor 120

may supplement the message by obtaining the additional information about the incident from the external environment. In some embodiments, the processor 120 may activate an image capturing device, for example, which is external to the second computing device 170a, to obtain an image capturing the external environment, to supplement the message for the incident, based on the follow-up question output from the LLM. For example, in this manner, the processor 120 may obtain a photo of the car accident from the image capturing device (for example, a Dash Cam), and send the photo to the agent that the message has been assigned for analysing to collect the information about the car accident. In some other embodiments, the processor 120 may control the second computing device 170a to activate a camera of the second computing device 170a, to obtain the image capturing the external environment, to supplement the message for the incident, based on the follow-up question output from the LLM. For example, in this manner, the processor 120 may obtain the photo of the car accident from the second computing device 170a, and send the photo to the agent that the message has been assigned for analysing to collect the information about the car accident.

[0092] In some embodiments, the processor 120 may receive the acknowledgement from the LLM. In some embodiments, whether to issue the acknowledgement may be determined by the agent that the message has been assigned. In some embodiments, the processor 120 may send the acknowledgement message to the second computing device 170a to end the incident reporting mode, based on the acknowledgement output from the LLM.

[0093] In some embodiments, the processor 120 may broadcast the incident to one or more second computing devices 170 each associated with one or more drivers 171 located in a predetermined range of the driver 171a, based on a confirmation of the incident from the LLM. In some embodiments, whether to confirm the incident from the message may be determined by the agent that the message has been assigned. In some embodiments, the processor 120 (for example, including a geo data/service layer) and/or the geo data/service

layer may update the map displayed on the one or more second computing devices 170 each associated with one or more drivers 171 located in the predetermined range of the driver 171a. In some other embodiments, the processor 120 may control the one or more second computing devices 170 each associated with one or more drivers 171 located in the predetermined range of the driver 171a, to output a voice message and/or display a message (for example, in the form of a pop-up window) informing the incident.

[0094] As described above, the various embodiments may relate to a new automated incident reporting process. The various embodiments may use a combination of a voice recognition and a text understanding technology, to enable an automated real-time incident reporting flow.

[0095] As described above, the various embodiments may replace an existing form-based submission flow by using a voice processing technology. With the new submission flow, the driver 171a may only need to talk to the second computing device 170a about what the driver 171a is witnessing on a ground without constraints of an input. The voice processing technique may help to classify the driver's 171a oral description into predefined scenarios automatically. A short oral conversation between the driver 171a and the second computing device 170a may be required to help complete the collection of the information about the incident. This seamless and effortless reporting flow may improve efficiency of the incident reporting, meanwhile, allowing the driver 171a to drive safely without too many interactions with the screen of the second computing device 170a.

[0096] It may be appreciated that the various embodiments may also be applied to an online customer support, which may help to classify users' reported issues. It may be appreciated that the various embodiments may also be applied to other similar reporting systems in existing services/products of the on-demand service platform provider (for example, an existing service which may allow users to contribute POI information onsite).

[0097] FIG. 5 illustrates a flowchart for a method 300 for facilitating collecting information about an incident according to various embodiments. According to various embodiments, the method 300 for facilitating collecting the information about the incident may be provided.

[0098] In some embodiments, the method 300 may include a step 301 of receiving a request to initiate an incident reporting mode from a driver via a computing device associated with the driver.

[0099] In some embodiments, the method 300 may include a step 302 of receiving a message for the incident from the driver via the computing device.

[00100] In some embodiments, the method 300 may include a step 303 of identifying an intention from the message.

[00101] In some embodiments, the method 300 may include a step 304 of classifying the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention.

[00102] In some embodiments, the method 300 may include a step 305 of assigning the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message to collect the information about the incident.

[00103] In some embodiments, the method 300 may include a step 306 of providing output information based on the collected information about the incident.

[00104] According to the various embodiments, as the driver may only need to talk on the computing device about the incident that the driver is witnessing, screen touching may not be required. In addition, the driver may receive an adaptive acknowledgement according to the reported incident type immediately. Moreover, the driver may be safe to initiate reporting anytime and anywhere, without stopping a vehicle.

[00105] FIGS. 6A and 6B illustrate exemplary diagrams showing a user interface of a computing device 170a (also referred to as a “second computing device 170a”) associated with a driver 171a according to various embodiments.

[00106] In some embodiments, as shown in FIG. 6A(a), the computing device 170a associated with the driver 171a may display a map relating to a route to a destination for providing an on-demand service, en route to the destination. The map may include an image object (for example, a pin) indicating the destination. The map may further include information about a current road name, an estimated time or arrival, and a distance to the destination. The map may further include an image object 415 to request to send an incident report. In some embodiments, as shown in FIG. 6A(a), the image object 415 may include a plurality of icons, for example, including a first icon 415a and a second icon 415b. In some embodiments, the first icon 415a may relate to the incident report by a contactless (also referred to as “touchless”) message, for example, a voice message, and the second icon 415b may relate to the incident report by the driver’s 171a contact to a screen of the computing device 170a, for example, a touch input or a click input.

[00107] In some embodiments, if the driver 171a selects the first icon 415a to request to send the incident report, the computing device 170a may initiate an incident reporting mode, and display a pop-up window 416 to receive the voice message for the incident from the driver 171a, as shown in FIG. 6A(b). In some embodiments, the driver 171a may speak to report the incident, and the computing device 170a may receive the voice message for the incident from the driver 171a. In some embodiments, the computing device 170a may record the voice message received from the driver 171a.

[00108] Although not shown, in some embodiments, if the driver 171a who is a first-time user selects the first icon 415a to request to send the incident report, the computing device 170a may play an on-board message, for example, “What traffic incident or map issue did

you encounter? Simply talk to us, like I'm stuck in traffic." Although not shown, in some embodiments, after the driver 171a who is the first-time user finishes speaking to report the incident, the computing device 170a may play a message, for example, "Next time, just tap the icon and record immediately."

[00109] In some embodiments, the computing device 170a may send the voice message to a processor 120 of a server 100. In some embodiments, the processor 120 may convert the voice message to a text message. In some embodiments, the processor 120 may send the text message to a Large Language Model (LLM) to analyse and interpret the text message. In some embodiments, as shown in FIG. 6B(a), the computing device 170a may display a pop-up window 417, while the processor 120 converts the voice message to the text message and the LLM analyses and interprets the text message. In some embodiments, the LLM may identify an intention of the text message, classify the text message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention, and assign the text message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the text message and collect information, including additional information, about the incident.

[00110] In some other embodiments, the processor 120 may not convert the voice message to the text message, and send the voice message to at least one model (as described with reference to FIG. 4) to analyse and interpret the voice message. In some embodiments, as shown in FIG. 6B(a), the computing device 170a may display the pop-up window 417, while the at least one model analyses and interprets the voice message. In some embodiments, the LLM may identify an intention of the voice message, classify the voice message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention, and assign the voice message to one or more agents corresponding to the

classified incident categories among a plurality of predetermined agents, to analyse the voice message and collect information, including additional information, about the incident.

[00111] In some embodiments, the processor 120 may receive an output from the LLM. If the LLM outputs an acknowledgement to end the incident reporting mode, the computing device 170a may display a pop-up window 418 to display an acknowledgement message (for example, “Thank you for reporting incorrect road name. We will work on updating information in our app.”), to inform the driver 171a of the end of the incident reporting mode, as shown in FIG. 6B(b).

[00112] Although not shown, in some embodiments, the user interface may provide the following functions:

- a) Multi-round conversation: Enabling the server 100 to follow up with the driver 171a on report details
- b) Undo operation: Providing an option to undo reports made due to testing or unintentional triggers by the driver 171a
- c) Animated incident icons: Enhancing visual acknowledgements with animated incident icons
- d) Data mining on reports: Clustering data and generating valuable insights using a generative pre-trained transformer
- e) Confidence scoring on reports: Evaluating a quality of the reports
- f) Incident broadcasting: Broadcasting critical incidents to nearby drivers 171
- g) Easy to talk: Implementing an easier entry point to start the incident reporting mode, such as a wake-up word or screen knock at any place in the App
- h) Smart RTI trigger: Triggering a validation/conversation to confirm whether there is an incident based on the driver's 171a odd behaviour (for example, an unusual stop, driving slowly, unexpected U-turn, driving away from the destination, etc.)

[00113] Although not shown, in some embodiments, the processor 120 may process the following scenarios of voice queries of the driver 171a:

- a) Traffic on road: Querying traffic conditions on a specific road
- b) Alternative route: Querying for alternative routes if a road is congested or blocked
- c) Diversion: Querying a new destination for diversion due to an ad-hoc need
- d) Nearby facilities: Querying a car park, a gas station, a cash card top-up location, a toilet, a prayer room, etc.
- e) Entrance: Querying an entrance of a place
- f) Weather: Querying on rain, flood, and weather forecasting
- g) Walking direction: Querying on a walking direction

[00114] FIG. 7 illustrates a data flow diagram of a server 100 for facilitating collecting information about an incident according to various embodiments.

[00115] In some embodiments, in a first step 501, an in-App reporting may be triggered. In some embodiments, there may be two types of triggers for an initiation of an incident reporting process, which may include a contactless (also referred to as “touchless”) trigger and a contact trigger. In some embodiments, the contactless trigger may include, but is not limited to, a voice-command wake-up and a gesture wake-up. In some embodiments, the contact trigger may include, but is not limited to, a button press and knocking a screen. This first step 501 may enable recording of a voice message by a driver 171a.

[00116] In some embodiments, in a second step 502, the voice message may be recorded. In some embodiments, in the second step 502, the voice message, in the form of an audio signal, from the driver’s 171a speaking about the incident that the driver 171a is witnessing may be captured.

[00117] In some embodiments, in a third step 503, the voice message may be converted to a text message. In some embodiments, in the third step 503, the voice message may be converted to the text message using the same language.

[00118] In some embodiments, in a fourth step 504, a scenario of the text message may be classified by a Large Language Model (LLM). In some embodiments, the text message may be assigned to at least one agent among a plurality of predetermined agents, according to the classification. In some embodiments, the text message converted from the voice message in the third step 503 may be merged with a pre-written prompt message for an instruction to the LLM. In some embodiments, the instruction may be to guide the LLM to do the scenario classification and set an expected response. In some embodiments, the merged text message may be an input to the LLM, and an output from the LLM may be one or more predetermined incident categories that the incident belongs to. In some embodiments, besides the incident categories, additional responses may also be provided by the LLM, such as an inference logic, a follow-up question or a general acknowledgement.

[00119] In some embodiments, in a fifth step 505, the output from the LLM (also referred to as an “LLM response” or a “response message”) may be provided to the driver 171a. In some embodiments, the output from LLM may be shown to the driver 171a immediately. In some embodiments, there may be two ways to give a feedback to the driver 171a. For example, a first way may be to show the response message from the LLM in text on the App, and a second way may be to use a text-to-speech (TTS) technology to play back the response message.

[00120] In some embodiments, in a sixth step 506, a follow-up conversation may be proceeded. In some embodiments, the follow-up conversation may be led by the agent that the text message has been assigned, and the agent that the text message has been assigned may communicate with the LLM accordingly. In some embodiments, whether to proceed

with the follow-up conversation may be determined by the agent that the text message has been assigned. In some embodiments, the agent that the text message has been assigned may inform the decision on whether to proceed with the follow-up conversation to the LLM, and the LLM may generate an initial LLM response accordingly. In some embodiments, from the initial LLM response, it may require the driver 171a to provide additional information (also referred to as “more details”) about the incident, for example, a severity of a traffic jam, a correct road name, and a location of a car accident. In some embodiments, a few rounds of follow-up conversation may happen between the driver’s 171a message and the LLM response.

[00121] In some embodiments, in a seventh step 507, an acknowledgement to a completion of the incident reporting may be provided. In some embodiments, whether to issue the acknowledgement to complete the incident reporting may be determined by the agent that the text message has been assigned. In some embodiments, to end the incident reporting process, an acknowledgement message may be sent to a second computing device 170a associated with the driver 171a, so that the driver 171a may know that their reports are well received and processed correctly by the server 100.

[00122] As described above, the various embodiments may allow an interaction between the driver 171a and the server 100 providing the new reporting system. The new technology of the LLM may allow a more flexibility on the driver’s 171a reporting structure, contents and response. The new automated real-time incident reporting flow may achieve a smoothness and effortless reporting experience to the driver 171a with maintaining a driving safety.

[00123] FIG. 8 illustrates a data flow diagram of a Large Language Model (LLM) 601, a plurality of predetermined agents 602, and a geo data/service layer 603 for facilitating collecting information about an incident according to various embodiments.

[00124] In some embodiments, a processor 120 of a server 100 may include the LLM 601, and thus the LLM 601 may be a part of the processor 120. In some other embodiments, the LLM 601 may be external to the processor 120, and the processor 120 may communicate with the LLM 601 via a communication interface 110.

[00125] In some embodiments, the processor 120 may include the plurality of predetermined agents 602, and thus the plurality of predetermined agents 602 may be a part of the processor 120. In some other embodiments, the plurality of predetermined agents 602 may be external to the processor 120, and the processor 120 may communicate with the plurality of predetermined agents 602 via the communication interface 110. In some embodiments, the LLM 601 may communicate with the plurality of predetermined agents 602. In some embodiments, the processor 120 may communicate with the plurality of predetermined agents 602 via the LLM 601.

[00126] In some embodiments, the processor 120 may include the geo data/service layer 603, and thus the geo data/service layer 603 may be a part of the processor 120. In some other embodiments, the geo data/service layer 603 may be external to the processor 120, and the processor 120 may communicate with the geo data/service layer 603 via the communication interface 110. In some embodiments, the plurality of predetermined agents 602 may communicate with the geo data/service layer 603. In some embodiments, the processor 120 may communicate with the geo data/service layer 603 via the LLM 601 and the plurality of predetermined agents 602.

[00127] In some embodiments, the LLM 601 may identify an intention of a text message converted from a voice message of a driver 171a, classify the text message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention, and assign the text message to one or more agents corresponding to the classified incident categories among the plurality of predetermined agents 602, to collect

information, for example, including additional information, about the incident. In some embodiments, the plurality of predetermined agents 602 may include, but are not limited to, an RTI agent 602a corresponding to a road incident reporting category, a traffic agent 602b corresponding to a traffic reporting category, and a place agent 602n corresponding to a nearby POI search category. In some embodiments, each of the predetermined agents 602 may be specialised in the incident category corresponding to that predetermined agent 602. In some embodiments, each of the predetermined agents 602 may be specialised in a conversational follow-up with the driver 171a to collect the additional information about the incident from the driver 171a. For example, the traffic agent may be specialised in at least one of analysing the text message in the field of the traffic, collecting necessary information from the text message in the field of the traffic, generating a question to collect the additional information about the incident, interpreting the driver's 171a additional voice message including the additional information about the incident, confirming the information about the traffic to the geo data/service layer 603 (so that the geo data/service layer 603 may update the map). In this manner, the plurality of predetermined agents 602 may act as a conversational assistant based on the intention of the driver 171a.

[00128] In some embodiments, the processor 120 may answer a question (also referred to as a “query”) of the driver 171a, offer an improved experience for reporting the incident, and/or provide more natural voice interactions. In some embodiments, the driver 171a may ask the question to the processor 120. In some embodiments, the processor 120 may send the driver's 171a question to the LLM 601. In some embodiments, the processor 120 may simultaneously work on the LLM 601 to answer the question. Examples of the question may include, but are not limited to, traffic, a route, and a point of interest (POI). In some embodiments, the LLM may support booking, in response to the driver 171a's request. For example, the plurality of predetermined agents 602 may include a booking agent (not shown). In some embodiments,

the number and types of the plurality of predetermined agents 602 may change, according to a trend of drivers' 171 reporting, questions and requests.

[00129] In some embodiments, the plurality of predetermined agents 602 may perform the conversational follow-up on incident report details with the driver 171a, by obtaining more detailed information from the driver 171a. In this regard, confidence in the incident report of an on-demand service platform provider may be increased. In addition, in some embodiments, the plurality of predetermined agents 602 may request the computing device 170a and/or an image capturing device associated with the driver 171a to obtain an image capturing an external environment, to supplement the driver's 171a incident reporting. For example, the processor 120 may activate the image capturing device, for example, a Dash Cam, to obtain a photo capturing a car accident spot, to supplement the driver's 171a incident reporting, based on the follow-up question output from the plurality of predetermined agents 602. Capturing the photo may be useful for a validation without a need of an operational review. In some embodiments, the plurality of predetermined agents 602 may perform a cross-validation with other drivers, for example, nearby drivers, to verify the driver's 171a incident reporting. In some embodiments, the plurality of predetermined agents 602 may evaluate the driver 171a, for example, assign a score, based on a quality of the driver's 171a historical incident reports.

[00130] In some embodiments, the plurality of predetermined agents 602 may address a core issue of crowdsourcing: a validation. For example, the plurality of predetermined agents 602 may immediately validate one or more predetermined types of incident reports, such as a traffic congestion, turn restrictions, and lane closures, with photos captured from the second computing device 170a and/or the image capturing device. In some embodiments, the plurality of predetermined agents 602 may determine which device (of the second computing device 170a and the image capturing device) will capture the photo, based on a position and capability of the devices, and control the determined device to capture the photo.

[00131] FIG. 9 illustrates a block diagram of a processor 120 of a server 100 for facilitating collecting information about an incident according to various embodiments.

[00132] In some embodiments, the processor 120 may include a plurality of modules. The plurality of modules may include, but are not limited to, an agent core 701, an intention module 702, a memory module 703, a conversation module 704, and an executor module 705.

[00133] In some embodiments, the agent core 701 may communicate with the intention module 702, the memory module 703, the conversation module 704 and the executor module 705.

[00134] In some embodiments, the intention module 702 may perform an intention mapping for a text message converted from a voice message received from a driver 171a. In some embodiments, the intention module 702 may understand the driver's 171a request and subsequently map the request to a plurality of predetermined agents 602. In some embodiments, the intention module 702 may onboard an agent with a new intention, find a similar agent, and map a history (for example, a chat history) to the intention. In some embodiments, the intention module 702 may play a crucial role in interpreting the intention of the driver 171a, and facilitate corresponding interactions.

[00135] In some embodiments, the memory module 703 may perform a state maintenance. In some embodiments, the memory module 703 may summarise the driver's 171a previous interactions within a session to provide consolidated information. In some embodiments, the memory module 703 may maintain a conversation continuity. In some embodiments, the memory module 703 may update the state, update the chat history, and retrieve a past chat session history/state.

[00136] In some embodiments, the conversation module 704 may perform a chat orchestration. In some embodiments, the conversation module 704 may obtain required information from the driver 171a or confirm the intention and actions of the driver 171a. In

some embodiments, the conversation module 704 may orchestrate an actual interaction with the driver 171a. In some embodiments, the conversation module 704 may reply to the driver 171a, ask questions upon requirements (for example, according to instructions of the plurality of predetermined agents 602), and initiate a confirmation question.

[00137] In some embodiments, the executor module 705 may perform an action delegation. In some embodiments, the executor module 705 may instruct one or more agents of the plurality of predetermined agents 602 or interact with external systems. In some embodiments, the executor module 705 may act as a gateway between local operations and external interactions. In some embodiments, the executor module 705 may delegate different operators with the external systems (for example, submission, query, proxy, action, etc.)

[00138] In some embodiments, the intention module 702 may include a retriever to retrieve contents of the text message of the driver 171a, and embedding modules to embed use case descriptions into embedding and to use RAG (Retrieval-Augmented Generation) to map the text message to the use case. In some embodiments, the memory module 703 may include messages in chat models, to enable all drivers/system/server chat message saved and to save the system/server actions as messages. In some embodiments, the memory module 703 may save the chat history in a memory 103 of the server 100 and the state in an orchestration framework which may control the workflow. In some embodiments, the executor module 705 may include an operator. An input definition of the executor module 705 may be a prompt template, and an output definition of the executor module 705 may be an output parser. In some embodiments, the executor module 705 may use a DAG (Direct Acyclic Graph) and the orchestration framework. In some embodiments, the conversation module 704 may include chat models.

[00139] An exemplary conversation flow for a hotspot search is provided as follows.

- a) Confirming Intention by the Intention Module 702

- DAX: I have no jobs here. Waiting for too long already.
- Server: It seems like you are looking for a hotspot for more jobs. Is that correct?
- DAX: Yes.
- Server: I understand that you are looking for a hotspot for more jobs. Please wait while I work on this.

(Operation of the Intention Module 702: [BE: mapped to use case 000300: hotspot navigation, trigger 000300 workflow] [BE: start workflow from “query_operator: hotspot_query_operator”] [BE: collected required information from system: current_location, wheel_type, current_utc_hour] [BE: missing information found: next_destination, initiate conversation to collect])

b) Collecting Required Information by the Conversation Module 704

- Server: Could you please provide me with the location of your next destination? This information will help us accurately determine the most suitable hotspot location for your itinerary.
- DAX: Clementi Rd.

(Operation of the Conversation Module 704: [BE: all required information checked])

c) API Call for Hotspot Search by the Executor Module 705

- Server: Thank you for the information. We are now processing it to determine the most suitable hotspot location based on your itinerary.

(Operation of the Executor Module 705: [BE: call API, get back nearest hotspot places: Jurong East (lat0, lng0)])

d) API Call + App Interaction to Start Navigation by the Conversation Module 704 and subsequently by the Executor Module 705

- Server: We have found a suitable hotspot for you: Jurong East.
- Server: Do you want me to start navigation to the place?

- DAX: Why not?

(Operation of the Conversation Module 704: [BE: trigger branch operator, get next action: "NAVIGATION_START"])

- Server: Great! Let's start heading towards Jurong East for more job opportunities. Remember to follow safe driving guidelines.

(Operation of the Executor Module 705: [BE: end chat and trigger navigation to Jurong East (lat0, lng0)])

[00140] In some embodiments, the processor 120 may define the intention of the driver 171a, define a workflow with operators, and define an operator input/output (I/O). In a step of defining the intention, the processor 120 may describe an agent purpose with detailed descriptions. In the step of defining the intention, the intention module 701 may map the driver's 171a chat history during serving, and the conversation module 704 may confirm the driver's 171a intention. In the step of defining the workflow with the operators, the processor 120 may define the workflow with the operators and dependencies. In the step of defining the workflow with the operators, the agent core 701 may follow the workflow definition to run different operators, and the memory module 703 may track the chat history and state variables across steps. In the step of defining the operator I/O, the processor 120 may define a task, input and output for each operators (support reuse). In the step of defining the operator I/O, the conversation module 704 may run chats to collect information, and the executor module 705 may take actions based on the operator definition.

[00141] As an example of a road incident reporting, in the step of defining the intention, the intention may be defined, for example, as follows: "This use case is to report a road incident, such as accidents, police check, bad traffic, road condition, which can be broadcasted to other drivers to help their navigation." In the step of defining the workflow with the operators, the processor 120 may perform according to following instructions: "op_classification" >>

“op_info_collection” >> “op_api_submission”. In the step of defining the operator I/O, the processor 120 may perform as follows:

a) Operator Classification

- Classifying a free text message into pre-defined class label (also referred to as a “predetermined incident category”) based on description
- Input: text
- Output: pre-defined class label

b) Operator Info Collection

- Conducting Q&A and extracting information based on scheme
- Input: Chat history
- Output: key-value mapping

c) Operator Submission

- Submitting a request to pre-defined API
- Input: request body (from last operator)
- Output: status code

[00142] As an example of a nearby POI search, in the step of defining the intention, the intention may be defined, for example, as follows: “This use case is to enable DAX to search for any POI by its name or any related information.” In the step of defining the workflow with the operators, the processor 120 may perform according to following instructions: “op_info_collection” >> “op_api_retrieval”. In the step of defining the operator I/O, the processor 120 may perform as follows:

a) Operator Info Collection

- Conducting Q&A and extracting information based on scheme
- Input: Chat history
- Output: key-value mapping [keyword]

b) Operator Query

- Submitting a request to pre-defined API to get back response
- Input: request body (keyword)
- Output: response (POI list)

[00143] As described above, the processor 120 may enable a multi-round conversation and a non-templating response, and enable a detailed information collection for map-related issues. In addition, in some embodiments, the processor 120 may provide smart follow-up actions for certain incident types, suggest a new route for reporting drivers after an accident, traffic, road closure reports, and trigger a map edit task in a “task centre” for map-related issues.

[00144] In the following, various examples of this disclosure are illustrated:

[00145] Example 1 is a server for facilitating collecting information about an incident, the server comprising: a memory configured to store instructions; and a processor for executing the stored instructions and configured to: receive a request to initiate an incident reporting mode from a driver via a computing device associated with the driver; receive a message for the incident from the driver via the computing device; identify an intention from the message; classify the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention; assign the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message to collect the information about the incident; and provide output information based on the collected information about the incident.

[00146] Example 2 is the server of Example 1, wherein the processor is further configured to: receive a voice command or a gesture command from the driver via the computing device; determine if the received command relates to a predetermined command; and if it is determined that the received command relates to the predetermined command, determine the received command as the request to initiate the incident reporting mode.

[00147] Example 3 is the server of Example 1 or Example 2, wherein the message includes a voice message, and the processor is further configured to convert the voice message into a text message.

[00148] Example 4 is the server of Example 3, wherein the processor is further configured to: merge the text message with a pre-written prompt message to generate a merged message including an instruction to guide a Large Language Model (LLM); input the merged message to the LLM; and control the LLM to identify the intention from the text message and classify the text message into the one or more incident categories among the plurality of predetermined incident categories.

[00149] Example 5 is the server of Example 4, wherein the processor is further configured to control the LLM to output at least one of the one or more incident categories that the incident belongs to, an inference logic, a follow-up question to the driver, and an acknowledgement.

[00150] Example 6 is the server of Example 5, wherein the processor is further configured to control the computing device to inform the driver of the output of the LLM.

[00151] Example 7 is the server of Example 5 or Example 6, wherein the processor is further configured to request the driver to provide additional information about the incident via the computing device, based on the follow-up question output from the LLM.

[00152] Example 8 is the server of any one of Examples 5 to 7, wherein the processor is further configured to activate an image capturing device to obtain an image capturing an external environment, to supplement the text message for the incident, based on the follow-up question output from the LLM.

[00153] Example 9 is the server of any one of Examples 5 to 8, wherein the processor is further configured to send an acknowledgement message to the computing device to end the incident reporting mode, based on the acknowledgement output from the LLM.

[00154] Example 10 is the server of any one of Examples 1 to 9, wherein the processor is further configured to broadcast the incident to one or more computing devices each associated with one or more drivers located in a predetermined range of the driver.

[00155] Example 11 is the server according to Examples 1 to 10, wherein the output information includes updating a display of one or more computing devices each associated with one or more subsequent drivers on a same route, based on the collected information about the incident.

[00156] Example 12 is a method for facilitating collecting information about an incident, the method comprising: receiving a request to initiate an incident reporting mode from a driver via a computing device associated with the driver; receiving a message for the incident from the driver via the computing device; identifying an intention from the message; classifying the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention; assigning the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message to collect the information about the incident; and providing output information based on the collected information about the incident.

[00157] Example 13 is the method of Example 12, further comprising: receiving a voice command or a gesture command from the driver via the computing device; determining if the received command relates to a predetermined command; and if it is determined that the received command relates to the predetermined command, determining the received command as the request to initiate the incident reporting mode.

[00158] Example 14 is the method of Example 12 or Example 13, wherein the message includes a voice message, and the method further comprises: converting the voice message into a text message.

[00159] Example 15 is the method of Example 14, further comprising: merging the text message with a pre-written prompt message to generate a merged message including an instruction to guide a Large Language Model (LLM); inputting the merged message to the LLM; and controlling the LLM to identify the intention from the text message and classify the text message into the one or more incident categories among the plurality of predetermined incident categories.

[00160] Example 16 is the method of Example 15, further comprising: controlling the LLM to output at least one of the one or more incident categories that the incident belongs to, an inference logic, a follow-up question to the driver, and an acknowledgement.

[00161] Example 17 is the method of Example 16, further comprising: controlling the computing device to inform the driver of the output of the LLM.

[00162] Example 18 is the method of Example 16 or Example 17, further comprising: requesting the driver to provide additional information about the incident via the computing device, based on the follow-up question output from the LLM.

[00163] Example 19 is the method of any one of Examples 16 to 18, further comprising: activating an image capturing device to obtain an image capturing an external environment, to supplement the text message for the incident, based on the follow-up question output from the LLM.

[00164] Example 20 is the method of any one of Examples 16 to 19, further comprising: sending an acknowledgement message to the computing device to end the incident reporting mode, based on the acknowledgement output from the LLM.

[00165] Example 21 is the method of any one of Examples 12 to 20, further comprising: broadcasting the incident to one or more computing devices each associated with one or more drivers located in a predetermined range of the driver.

[00166] Example 22 is the method according to Examples 12 to 21, wherein the output information includes updating a display of one or more computing devices each associated with one or more subsequent drivers on a same route, based on the collected information about the incident.

[00167] 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 collecting information about an incident, the server comprising:

a memory configured to store instructions; and

a processor for executing the stored instructions and configured to:

receive a request to initiate an incident reporting mode from a driver via a computing device associated with the driver;

receive a message for the incident from the driver via the computing device;

identify an intention from the message;

classify the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention;

assign the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message to collect the information about the incident; and

provide output information based on the collected information about the incident.

2. The server according to claim 1, wherein the processor is further configured to:

receive a voice command or a gesture command from the driver via the computing device;

determine if the received command relates to a predetermined command; and

if it is determined that the received command relates to the predetermined command, determine the received command as the request to initiate the incident reporting mode.

3. The server according to claim 1, wherein the message includes a voice message, and the processor is further configured to convert the voice message into a text message.
4. The server according to claim 3, wherein the processor is further configured to:
merge the text message with a pre-written prompt message to generate a merged message including an instruction to guide a Large Language Model (LLM);
input the merged message to the LLM; and
control the LLM to identify the intention from the text message and classify the text message into the one or more incident categories among the plurality of predetermined incident categories.
5. The server according to claim 4, wherein the processor is further configured to control the LLM to output at least one of the one or more incident categories that the incident belongs to, an inference logic, a follow-up question to the driver, and an acknowledgement.
6. The server according to claim 5, wherein the processor is further configured to control the computing device to inform the driver of the output of the LLM.
7. The server according to claim 5, wherein the processor is further configured to request the driver to provide additional information about the incident via the computing device, based on the follow-up question output from the LLM.
8. The server according to claim 5, wherein the processor is further configured to activate an image capturing device to obtain an image capturing an external environment, to

supplement the text message for the incident, based on the follow-up question output from the LLM.

9. The server according to claim 5, wherein the processor is further configured to send an acknowledgement message to the computing device to end the incident reporting mode, based on the acknowledgement output from the LLM.

10. The server according to claim 1, wherein the processor is further configured to broadcast the incident to one or more computing devices each associated with one or more drivers located in a predetermined range of the driver.

11. The server according to claim 1, wherein the output information includes updating a display of one or more computing devices each associated with one or more subsequent drivers on a same route, based on the collected information about the incident.

12. A method for facilitating collecting information about an incident, the method comprising:

receiving a request to initiate an incident reporting mode from a driver via a computing device associated with the driver;

receiving a message for the incident from the driver via the computing device;

identifying an intention from the message;

classifying the message into one or more incident categories among a plurality of predetermined incident categories, based on the identified intention;

assigning the message to one or more agents corresponding to the classified incident categories among a plurality of predetermined agents, to analyse the message to collect the information about the incident; and

providing output information based on the collected information about the incident.

13. The method according to claim 12, further comprising:

receiving a voice command or a gesture command from the driver via the computing device;

determining if the received command relates to a predetermined command; and

if it is determined that the received command relates to the predetermined command, determining the received command as the request to initiate the incident reporting mode.

14. The method according to claim 12, wherein the message includes a voice message, and the method further comprises: converting the voice message into a text message.

15. The method according to claim 14, further comprising:

merging the text message with a pre-written prompt message to generate a merged message including an instruction to guide a Large Language Model (LLM);

inputting the merged message to the LLM; and

controlling the LLM to identify the intention from the text message and classify the text message into the one or more incident categories among a plurality of predetermined incident categories.

16. The method according to claim 15, further comprising: controlling the LLM to output at least one of the one or more incident categories that the incident belongs to, an inference logic, a follow-up question to the driver, and an acknowledgement.

17. The method according to claim 16, further comprising: controlling the computing device to inform the driver of the output of the LLM.

18. The method according to claim 16, further comprising: requesting the driver to provide additional information about the incident via the computing device, based on the follow-up question output from the LLM.

19. The method according to claim 16, further comprising: activating an image capturing device to obtain an image capturing an external environment, to supplement the text message for the incident, based on the follow-up question output from the LLM.

20. The method according to claim 16, further comprising: sending an acknowledgement message to the computing device to end the incident reporting mode, based on the acknowledgement output from the LLM.

21. The method according to claim 12, further comprising: broadcasting the incident to one or more computing devices each associated with one or more drivers located in a predetermined range of the driver.

22. The method according to claim 12, wherein the output information includes updating a display of one or more computing devices each associated with one or more subsequent drivers on a same route, based on the collected information about the incident.