

DEVICE, METHOD, ELECTRONIC TERMINAL AND MEDIUM FOR ONLINE TRAINING OF NATURAL LANGUAGE UNDERSTANDING MODELS

FIELD OF THE INVENTION

[0001] The present disclosure generally relates to natural language understanding (NLU) and, in particular, to a device, method, electronic terminal and medium for training NLU models online.

BACKGROUND OF THE INVENTION

[0002] Natural language understanding, or NLU, is the process of enabling machines to understand the content of text. It plays a crucial role in text processing systems and is an essential component of recommendation, question answering, and search systems. The goal of natural language understanding is to take a user's input and produce a structured interpretation of the meaning the user's input.

[0003] Currently, NLU models are often trained off-line using corpora (collections of linguistic data in a computer database) provided in Excel. However, this approach can be slow to update and makes it difficult to trace data. As a result, there is a pressing need for a more efficient and traceable NLU model training solution.

SUMMARY

[0004] In view of the drawbacks of the above-described related technology, the present application provides a device, method, electronic terminal and medium for training NLU models online, to solve the technical problems that existing NLU models' corpus provision relies on file formats such as MS Excel as a platform, which has low update efficiency and is not convenient for data traceability.

[0005] A first aspect of the present application provides a device for training NLU models online, comprising: a communication interface; a processor, operably connected to the communication interface; and a memory, operably connected to the

processor, wherein the memory stores NLU models and instructions, wherein the instructions, when executed by the processor, cause the processor to perform following operations: calling one of the NLU models from the memory, creating an intent within the NLU model, and defining one or more intent attributes of the intent, and forming a model training corpus data structure based on the one or more intent attributes; obtaining dialog content through the communication interface, performing data maintenance based on the model training corpus data structure and linguistic context of the dialog content, thereby generating a corpus dataset; performing online training of one of the NLU models based on the corpus dataset, to obtain a trained NLU model, and testing the trained NLU model to obtain a tested NLU model; and deploying the tested NLU model to an external electronic device through the communication interface, for understanding, analyzing, and processing natural language acquired in real time.

[0006] In some examples of the first aspect of the present application, the one or more intent attributes defined within the trained NLU model comprise one or more of: an intent name, an intent priority, an intent example, and a downgrading strategy.

[0007] In some examples of the first aspect of the present application, the downgrading strategy comprises: when one corpus datum corresponds to two or more intents, determining a weight of each of the two or more intents according to linguistic environment in which the corpus datum is embedded, and selecting one of the two or more intents with the highest weight and downgrading the other ones of the two or more intents.

[0008] In some examples of the first aspect of the present application, testing the trained NLU model comprises: inputting a test dataset to the trained NLU model, calculating multi-categorization model evaluation indexes based on predicted values output by the trained NLU model, and determining whether the trained NLU model is qualified based on the multi-categorization model evaluation indexes; wherein the

multi-categorization model evaluation indexes comprise one or more of an accuracy index, a precision index, a recall index, and an F1 score index.

[0009] In some examples of the first aspect of the present application, online training of the NLU models comprises: obtaining abnormal corpus data and unconnected data generated during operation of the NLU models; and iteratively training the NLU models using the abnormal corpus data and unconnected data as input data to the models.

[0010] In some examples of the first aspect of the present application, the NLU models comprise one of a BERT (Bidirectional Encoder Representations from Transformers) model, a Transformer (deep learning architecture) model, and a GPT (generative pre-trained) model.

[0011] In some examples of the first aspect of the present application, the processor is further caused to input one or more intents into a ChatGPT model and pre-describe definitions of the intents to generate the corpus dataset by applying the ChatGPT model.

[0012] In some examples of the first aspect of the present application, before online training of the NLU models based on the corpus dataset, the processor is further caused to perform error correction on the corpus dataset by applying the ChatGPT model to correct incorrect labels.

[0013] In some examples of the first aspect of the present application, the processor is further caused to generate the dialog content by engaging the ChatGPT model in a conversation with a speech robot, wherein the processor then generates the corpus dataset based on the dialog content.

[0014] A second aspect of the present application provides a method for training an NLU model online, comprising: calling the NLU model, creating an intent within the NLU model, and defining one or more intent attributes of the intent, and forming a

NLU model training corpus data structure based on the one or more intent attributes; obtaining dialog content, performing data maintenance based on the NLU model training corpus data structure and linguistic context of the dialog content, thereby generating a corpus dataset; performing online training of the NLU model based on the corpus dataset, to obtain a trained NLU model, and testing the trained NLU model to obtain a tested NLU model; and deploying the tested NLU model to an external electronic device, for understanding, analyzing, and processing natural language acquired in real time.

[0015] A third aspect of the present application provides a non-transitory computer-readable storage medium having a computer program stored thereon, wherein the computer program, when executed by a processor, implements the method for training the NLU model online.

[0016] A fourth aspect of the present application provides an electronic terminal comprising: a processor and a memory; wherein the memory comprises a storage for a computer program; wherein the processor executes the computer program to cause the electronic terminal to perform the method for training the NLU model online.

[0017] The method, device, electronic terminal and medium for training NLU models online according to the present application have the following beneficial effects. They support data management for NLU models, which includes management of basic model information, intentions, and corpus data; they also allow for NLU model training, and version rollback and version-specific data viewing post training can be achieved by retraining with bad cases; additionally, the present application can optimize model testing, support closed-loop optimization of NLU models, and improve the efficiency of NLU iteration.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] FIG. 1 is a schematic block diagram of a device for training NLU models online according to an embodiment of the present application.

[0019] FIG. 2 is a flowchart of operations performed by a processor according to an embodiment of the present application.

[0020] FIG. 3 is a schematic table diagram of a corpus dataset according to an embodiment of the present application.

[0021] FIG. 4 is a flowchart of a method for training NLU models online according to an embodiment of the present application.

[0022] FIG. 5 is a schematic block diagram of an electronic electronic terminal according to an embodiment of the present application.

DETAILED DESCRIPTION

[0023] The embodiments of the present application will be described below. Those skilled can easily understand disclosure advantages and effects of the present application according to contents disclosed by the specification. The present application can also be implemented or applied through other different specific embodiments. Various details in this specification can also be modified or changed based on different viewpoints and disclosures without departing from the spirit of the electronic present application. It should be noted that the following embodiments and the features of the embodiments can be combined with each other if no conflict will result.

[0024] It is noted that in the following description, reference is made to the accompanying drawings, which depict several embodiments of the present application. As used herein, the singular forms “a”, “an” and “said/the” are intended to comprise the plural forms, unless the context clearly points out differently. It should be further understood that the terms "include" and "comprise" indicate the

existence of the described features, steps, elements, components, items, categories, and/or groups, but do not exclude the existence, presence, or addition of one or more other features, steps, elements, components, items, categories, and/or groups. As used herein, the terms “or” and “and/or” are inclusive, and are used to include any of the associated listed items and all combinations thereof. Thus, “A, B or C” or “A, B and/or C” means “any of the following: A; B; C; A and B; A and C; B and C; A, B and C”. Exceptions to this definition apply only when combinations of elements, functions, steps are inherently paradoxical in some way.

[0025] In order to make the purpose, technical solutions and advantages of the present application more clearly understood, further details of the technical solutions of the present application are described by means of the following embodiments in conjunction with the accompanying drawings. It should be understood that the specific embodiments described herein are merely illustrative of the invention and are not intended to restrict the scope of the invention.

[0026] Before further explaining the present application, terms and terminology involved in the embodiments of the present application are explained as follows:

[0027] <1> Natural language understanding, or NLU: the process of enabling machines to understand the content of text. The goal of natural language understanding is to take a user query and produce a structured representation of its meaning.

[0028] <2> Intent: generally, a user intent of the user query.

[0029] <3> Domain: generally, a domain to which the user query belongs.

[0030] <4> Chat Generative Pretrained Transformer, or ChatGPT: a chatbot program developed by OpenAI, an AI-driven natural language processing tool that is capable of generating conversations based on patterns and statistical regularities seen in its

pre-training phase, and capable of interactions based on context during a conversation.

[0031] The present application provide a method and device for training NLU models online, and storage media storing executable programs for implementing the method for training NLU models online. Embodiments of the present application will illustrate exemplary implementation scenarios of training NLU models online.

[0032] FIG. 1 is a schematic structural diagram of a device for training NLU models online according to an embodiment of the present application. The device for training NLU models online 100 comprises: a communication interface 110, a processor 120, and a memory 130. The communication interface 110 is configured to send data to or receive data from an external device. The processor 120 is operably coupled to the communication interface 110; the memory 130 is operably coupled to the processor 120. A plurality of NLU models and instructions are stored in the memory 130, wherein the instructions, when executed by the processor 120, cause the processor 120 to perform corresponding tasks.

[0033] As an example, the device for training NLU models online 100 further comprises a display 140, a speaker 150, and a microphone 160. The microphone 160 receives sound signals (e.g., user instructions) and converts the sound signals into electrical signals; the speaker 150 outputs the electrical signals into sound signals (e.g., speech); and the display 140 displays an image or video, for displaying graphical user interfaces of running applications.

[0034] As an example, the memory 130 stores a software development kit (SDK) 132 and a plurality of applications 131 (e.g., a first application 131-1, a second application 131-2). As an example, each of the plurality of applications 131 may comprise a plurality of actions for performing specified functions.

[0035] As an example, a plurality of NLU models 133 and instructions are stored in the memory 130, wherein the instructions, when executed by the processor, cause the processor 120 to perform following steps as shown in FIG. 2:

[0036] Step S2A: by the processor 120, calling a NLU model from the memory 130, creating an intent within the NLU model, and defining one or more intent attributes of the intent, and forming a model training corpus data structure based on the intent attributes.

[0037] In simple terms, a NLU model takes a user's question and outputs a structured interpretation of its meaning. This structured representation comprises at least three parts: domain, intent, and slot. Domain generally refers to the general topic of the question; for example, weather, online shopping, music, train tickets, medical consultation, etc. Intent generally refers to the user intent for asking the question, such as checking the weather, looking for online shopping advice, selecting a song to play, booking/refunding train tickets, looking for medical advice and so on. Slot generally refers to specific information needed to fulfill the intent; for example, to book train tickets, slots needed to fulfill the intent comprise at least departure time, departure location, destination, seat level, and quantity, etc.; for example, to play a song, slots required by the intent comprises at least playback mode, volume, sound mode, timed shutdown, etc. NLU models can be applied to many different domains, and each domain can have many different intents with varying slots, which are not exhaustively numerated herein.

[0038] As an example, one or more attributes of an intent defined within the NLU model comprise one or more of: an intent name, an intent priority, an intent example, and a downgrading strategy.

[0039] Further, the downgrading strategy comprises: when one corpus datum corresponds to two or more intents, determining a weight of each of the two or more intents according to linguistic environment in which the corpus datum is embedded,

and selecting one of the two or more intents with the highest weight and downgrading the other ones of the two or more intents. For example, in a conversation, if the respondent answers “Okay, I will repay today” when asked about his willingness to repay, the answer actually has two intents. “Okay” indicates the intent to answer the question, while “I will repay today” indicates the intent to repay. In the current conversation context, the repayment intent is obviously more important, so the intent to answer the question should be downgraded, i.e., its weight should be reduced to highlighting the repayment intent.

[0040] It should be understood that since the meaning expressed by a corpus is not unique and the intent of the corpus expressed in different contexts is different, it is usually not enough to understand the corpus only from its literal meaning. The corpus needs to be placed in a specific language environment to reveal its true intent. Therefore, in the present application, after obtaining a preliminary intent, technical means for proofreading with the previous language context are also introduced to identify the true intent. For example, if someone says “Wait and see”, it could mean that he wants the other person to wait, or it could be an expression of frustration. In other words, a single sentence can have multiple possible intents. To determine the true intent, it’s important to consider the context of the conversation.

[0041] Step S2B: by the processor 120, obtaining dialog content through the communication interface 110, performing data maintenance based on the model training corpus data structure and linguistic context of the dialog content, thereby generating a corpus dataset.

[0042] As an example, after determining the model training corpus data structure, entries of corpus data can be extracted from the dialog content and stored in accordance with the data structure, and these corpus data eventually form the corpus dataset. As an example, the above-mentioned intent attributes, that is, intent name, intent priority, intent example, and downgrading strategy form the model

training corpus data structure (e.g., they are column headings), as exemplarily shown in FIG. 3.

[0043] Preferably, the data in the above corpus dataset may be generated based on real voice calls with added manual labels, or by using large models such as ChatGPT to generate model outputs. The present application does not restrict how the data is generated. Step S2C: by the processor 120, performing online training of the NLU model based on the corpus dataset, to obtain a trained NLU model, and testing the trained NLU model to obtain a tested NLU model.

[0044] As an example, after being generated, the corpus dataset may be divided into a training dataset and a test dataset in accordance with a predetermined ratio, and the training dataset may be input into the NLU model for on-line training, and after the training is completed, the trained model may be tested using the test dataset.

[0045] A preferred approach for testing the model is to use ten-fold cross-validation. This involves dividing the data into ten parts and conducting ten rounds of training and testing. In each round, nine parts of the data are used for training and the remaining part is used for testing. The final result is the average of performance indicators from each round. Using cross-validation can help reduce the risk of overfitting and improve the model's ability to generalize data.

[0046] As an example, multi-categorization model evaluation indexes used herein for model testing comprise one or more of an accuracy index, a precision index, a recall index, and an F1 score index. Accuracy index measures the proportion of correctly predicted positive and negative examples out of the total number of examples. Precision index measures the proportion of correctly predicted positive examples out of all predicted positive examples. Recall index measures the proportion of correctly predicted positive examples out of all actual positive examples. F1 score index is a balanced measure that takes into account both precision and recall.

[0047] Specifically, the accuracy index, precision index, and recall rate are calculated using the following Equations (1) to (3):

$$[0048] \text{ ACC} = \frac{TP+TN}{TP+FP+FN+T}; \quad \text{Equation (1)}$$

$$[0049] \text{ Precision} = \frac{TP}{TP+FP}; \quad \text{Equation (2)}$$

$$[0050] \text{ Recall} = \frac{TP}{TP+FN}; \quad \text{Equation (3)}$$

[0051] wherein TP (*Ture Positives*) denotes a true positive case, where the prediction is positive and the actual outcome is also positive; wherein FP (*False Positives*) denotes a false positive case, where the prediction is positive but the actual outcome is negative; wherein FN (*False Negatives*)denotes a false negative case, where the prediction is negative but the actual outcome is positive; and wherein TN (*Ture Negatives*) denotes a true negative case, where the prediction is negative and the actual outcome is also negative.

[0052] Step S2D: by the processor 120, deploying the tested NLU model to an external electronic device through the communication interface 110, for understanding, analyzing, and processing natural language acquired in real time.

[0053] As an example, the tested NLU model may be deployed to an external electronic device in a variety of application scenarios, such as an electronic device in the field of finance, an electronic device in the field of intelligent medical care, an electronic device in the field of intelligent manufacturing, or an electronic device in the field of new energy manufacturing, and so on.

[0054] Preferably, online training of the NLU model comprises: obtaining abnormal corpus data and unconnected data generated during operation of the NLU model; and iteratively training the NLU model using the abnormal corpus data and unconnected data as input data to the model.

[0055] It is necessary stating that as an example of the present application, any corpus data that the natural language understanding model fails to understand or

instead produces an incorrect understanding during the actual operation is collected and re-input into a model training port as a bad case. This allows the model to be retrained and improved, creating a closed-loop optimization process for the model. Corpus data that the model produces an incorrect understanding of usually refers to cases where the output of the model does not match the expected intent of the corpus or is inconsistent with its previous language context. Corpus data that are not understood by the model is often a result of overfitting caused by a small amount of training data, which results in a lack of generalization ability and an inability to recognize certain corpus.

[0056] As an example, the NLU model may be one of a BERT model, a Transformer model, and a GPT model, and may also be one of an XLNet (a large bidirectional transformer that uses improved training methodology) model, an ALBERT (a transformers model pretrained on a large corpus of English data in a self-supervised fashion) model, an ERNIE model (a transformer-based model consisting of two stacked modules: 1) textual encoder and 2) knowledgeable encoder), a T5 (Text-to-Text Transfer Transformer) model, an ELMo (word embedding method for representing a sequence of words as a corresponding sequence of vectors.) model, an ULMFiT model(an architecture and transfer learning method that can be applied to NLP tasks), an MTDNN (multi-task deep neural network) model, an OPENAI GPT model, a DeBERTa (Decoding-enhanced BERT with disentangled attention) model, etc.

[0057] Preferably, the processor 120 is further caused to input one or more intents into a ChatGPT model and pre-describe definitions of the intents to generate the corpus dataset using the ChatGPT model.

[0058] It is necessary noting that while a corpus dataset may be provided for the NLU model by manually creating conversational content, conversations created in this manner are more limited in number and content, which is likely to result in a poorer generalization ability of the model. Therefore, creating a large number of

conversations by means of the ChatGPT model greatly expands the content of created conversations and greatly enlarges the corpus dataset, which effectively improves the model's generalization ability and effectively prevents overfitting.

[0059] Preferably, before online training of the NLU model based on the corpus dataset, the processor 120 is further caused to perform error correction on the corpus dataset using the ChatGPT model in order to correct incorrect labels.

[0060] It is necessary noting that the labeling of the corpus data is usually done manually during the model training process, and manual labeling inevitably faces the problem of low efficiency and is very prone to errors, especially when encountering some semantically similar phrases, which are very likely to be semantically entangled, e.g., "I should pay it today, but I might pay it tomorrow" is a sentence whose intention is not clear and is semantically entangled. To address such issues, we can use the ChatGPT model. By setting a specific intention for the model and providing it with a clear definition, we can leverage its vast data resources to accurately classify and determine the true intention behind an ambiguous statement. For example, we can inform the ChatGPT model of our intention to repay and provide it with a clear definition of what that means. Then, by inputting an ambiguous statement into the model, it can determine whether or not there is an intention to repay.

[0061] Preferably, for dialog scenarios that lack dialog content as corpus data support, dialog content can be generated by having the ChatGPT model engage in dialog with a voice robot, and a corpus dataset for model training and testing can be generated based on the generated dialog content.

[0062] It's important to note that for well-established scenarios, such as debt collection, medical consultation, and product sales, there is already an abundance of call data available. This data can be used to extract information and train models. However, for newer scenarios that lack conversation data, it can be challenging to

create realistic dialog scenarios manually. To address this challenge, we can use the ChatGPT model as one robot and another voice robot as the second. By providing a specific scenario and allowing the two robots to engage in free conversation, we can generate call data. This data can then be used to create a dataset for training NLU models.

[0063] Above, the structure of the device for training NLU models online of the present application has been explained in detail. In the following, a method for training NLU models online according to one embodiment of the present application will be described.

[0064] FIG. 4 is a flowchart of the method for training NLU models online according to an embodiment of the present application. The method comprises steps S41-S44:

[0065] Step S41: calling a NLU model, creating an intent within the NLU model, and defining one or more intent attributes of the intent, and forming a model training corpus data structure based on the intent attributes;

[0066] As an example, one or more attributes of an intent defined within the NLU model comprise one or more of: an intent name, an intent priority, an intent example, and a downgrading strategy.

[0067] Further, the downgrading strategy comprises: when one corpus datum corresponds to two or more intents, selecting the most probable intent as the true intent of the corpus datum based on the linguistic context.

[0068] Step S42: obtaining dialog content, performing data maintenance based on the model training corpus data structure and linguistic context of the dialog content, thereby generating a corpus dataset;

[0069] Step S43: performing online training of the NLU model based on the corpus dataset, to obtain a trained NLU model, and testing the trained NLU model to obtain a tested NLU model; and

[0070] Testing the trained NLU model comprises: inputting a test dataset to the trained NLU model, calculating multi-categorization model evaluation indexes based on predicted values output by the trained NLU model, and determining whether the NLU model is qualified based on the multi-categorization model evaluation indexes.

[0071] As an example, the multi-categorization model evaluation indexes comprise one or more of an accuracy index, a precision index, a recall index, and an F1 score index.

[0072] As an example, online training of the NLU model comprises: obtaining abnormal corpus data and unconnected data generated during operation of the NLU model; and iteratively training the NLU model using the abnormal corpus data and unconnected data as input data to the model.

[0073] In some examples, the NLU model comprises any one of a BERT model, a Transformer model, and a GPT model.

[0074] As an example, online training of the NLU model comprises: obtaining abnormal corpus data and unconnected data generated during operation of the NLU model; and iteratively training the NLU model using the abnormal corpus data and unconnected data as input data to the model.

[0075] Step S44: deploying the tested NLU model to an external electronic device, for understanding, analyzing, and processing natural language acquired in real time.

[0076] It is to be noted that the method for training NLU models online and the device for training NLU models online belongs to the same inventive concept, and the specific implementation process of the latter can be found in the method embodiments, and will not be repeated here.

[0077] The method for training NLU models online may be implemented on a electronic terminal side or a server side. As for the hardware structure of an electronic terminal for training NLU models online, please refer to FIG. 5, which

shows an optional hardware structure of the electronic terminal 500 for training NLU models online. The electronic terminal 500 can take various forms, these forms include mobile phones, computers, tablets, personal digital processing devices, and factory background processing equipment.

[0078] The electronic terminal 500 for training NLU models online comprises: at least one processor 501, a memory 502, at least one network interface 504, and a user interface 506. The components in the device are coupled together via a bus system 505. The bus system 505 is used to enable connection and communication between these components. The bus system 505 comprises a power bus, a control bus, a status signal bus, and a data bus. However, for clarity of illustration, the various buses are labeled as the bus system 505 in FIG. 5.

[0079] The user interface 506 may comprise a display, a keyboard, a mouse, a trackball, a click gun, keys, buttons, a tactile pad, or a touch screen.

[0080] The memory 502 may be volatile memory or non-volatile memory, or may comprise both volatile and non-volatile memory. The non-volatile memory may be read-only memory (ROM), programmable read-only memory (PROM), which is used as an external cache. Many forms of RAM are applicable, such as static random access memory (SRAM), synchronous static random access memory (SSRAM). The memory described herein is intended to comprise, but is not limited to, these and any other suitable types of memory.

[0081] The memory 502 is used to store various types of data to support operation of the electronic terminal 500 for training NLU models online. Examples of such data comprise: any executable programs, such as an operating system 5021 and application programs 5022, for operation on the electronic terminal 500 for training NLU models online; the operating system 5021 may comprise various system programs, such as a framework layer, a core library layer, a driver layer, and the like, used to implement various underlying operations as well as to handle hardware-

based tasks. The application programs 5022 may comprise various applications, such as a media player, a browser, etc., for various business scenarios. The method for training NLU models online described above may be comprised in the application programs 5022.

[0082] The method for training NLU models online may be applied in, or realized by, the processor 501. The processor 501 may be an integrated circuit chip with signal processing capabilities. In the implementation process, each step of the above method may be completed by an integrated logic circuit of hardware in the processor 501 or an instruction in a form of software. The processor 501 may be a general-purpose processor, a digital signal processor (DSP), or other programmable logic devices, discrete gates or transistor logic devices, discrete hardware components, etc. The processor 501 may implement or perform the various methods, steps, and logic block diagrams disclosed in the present application. The general-purpose processor 501 may be a microprocessor or any conventional processor. The method of the present application may be directly performed by a hardware decoding processor, or performed by a combination of hardware and software modules in a decoding processor. The software modules may be located in a storage medium, which is located in a memory. The processor reads information from the memory and implements the steps of the method in combination with its hardware.

[0083] In exemplary embodiments, the electronic terminal 500 for training NLU models online may be utilized by one or more of application-specific integrated circuits (ASIC), DSPs, programmable logic devices (PLD), complex programmable logic devices (CPLD), to perform the foregoing method.

[0084] It may be appreciated by those of ordinary skill in the art that all or some of the steps that implement the method embodiments described above may be accomplished by hardware related to computer programs. The aforementioned computer program may be stored in a computer readable storage medium. Operations of the aforementioned methods are performed when the program is

executed; and the aforementioned storage media comprises an ROM, an RAM, a magnetic disk, an optical disk, or any of other various media that can store software programs.

[0085] In the embodiments provided herein, the computer readable and writeable storage medium may comprise read-only memory, random access memory, EEPROM, CD-ROM or other optical disk storage device, disk storage device or other magnetic storage device, flash memory, USB flash disk, removable hard disk, or any other medium capable of storing desired program code in the form of instructions or data structures and capable of being accessed by a computer. Alternatively, any connection may be appropriately referred to as a computer readable medium. For example, if instructions are sent from a website, server, or other remote source using coaxial cable, fiber optic cable, twisted pair, digital subscriber line (DSL), or wireless technology such as infrared, radio, and microwave, the coaxial cable, fiber optic cable, twisted pair, DSL, or wireless technology such as infrared, radio, and microwave are comprised in the definition of the medium. However, it should be understood that computer readable and writeable storage medium and data storage medium do not comprise connections, carriers, signals, or other transient media, but are intended for non-transient, tangible storage medium. Herein, disks and optical discs comprise compact discs (CDs), laser discs, optical disks, digital versatile discs (DVDs), floppy disks, and Blu-ray discs. Disks usually replicate data magnetically, while optical disks use lasers to replicate data optically.

[0086] In summary, the method, device, electronic terminal and medium for training NLU models online of the present application supports data management for NLU models, comprising management of basic model information, intentions, and corpus data; they also allows for NLU model training, and after training, version rollback and version-specific data viewing can be achieved by retraining with bad cases; additionally, the present application can optimize model testing, support closed-loop optimization of NLU models, and improve the efficiency of NLU iteration. Therefore,

the present disclosure effectively overcomes various shortcomings of the prior art and has a high industrial value.

[0087] The above-mentioned embodiments are merely illustrative of the principle and effects of the present application instead of restricting the scope of the present application. Modifications or variations of the above-described embodiments may be made by those skilled in the art without departing from the spirit and scope of the present application. Therefore, all equivalent modifications or changes made by those who have common knowledge in the art without departing from the spirit and technical concept disclosed by the present application shall be still covered by the claims of the present application.

CLAIMS

What is Claimed is:

1. A device for training natural-language-understanding (NLU) models online, comprising:
 - a communication interface;
 - a processor, operably connected to the communication interface; and
 - a memory, operably connected to the processor, wherein the memory stores NLU models and instructions, wherein the instructions, when executed by the processor, cause the processor to perform following operations:
 - calling one of the NLU models from the memory, creating an intent within the NLU model, and defining one or more intent attributes of the intent, and forming a model training corpus data structure based on the one or more intent attributes;
 - obtaining dialog content through the communication interface, performing data maintenance based on the model training corpus data structure and linguistic context of the dialog content, thereby generating a corpus dataset;
 - performing online training of one of the NLU models based on the corpus dataset, to obtain a trained NLU model, and testing the trained NLU model to obtain a tested NLU model; and
 - deploying the tested NLU model to an external electronic device through the communication interface, for understanding, analyzing, and processing natural language acquired in real time.
2. The device for training the NLU models online according to claim 1, wherein the one or more intent attributes defined within the trained NLU model comprise one or more of: an intent name, an intent priority, an intent example, and a downgrading strategy.
3. The device for training the NLU models online according to claim 2, wherein the downgrading strategy comprises: when one corpus datum corresponds to two or more intents, determining a weight of each of the two or more intents according to linguistic environment in which the corpus datum is embedded, and selecting one of the two or more intents with the highest weight and downgrading the other ones of the two or more intents.

4. The device for training the NLU models online according to claim 1, wherein testing the trained NLU model comprises: inputting a test dataset to the trained NLU model, calculating multi-categorization model evaluation indexes based on predicted values output by the trained NLU model, and determining whether the trained NLU model is qualified based on the multi-categorization model evaluation indexes; wherein the multi-categorization model evaluation indexes comprise one or more of an accuracy index, a precision index, a recall index, and an F1 score index.
5. The device for training the NLU models online according to claim 1, wherein online training of the NLU models comprises: obtaining abnormal corpus data and unconnected data generated during operation of the NLU models; and iteratively training the NLU models using the abnormal corpus data and unconnected data as input data to the NLU models.
6. The device for training the NLU models online according to claim 1, wherein the NLU models comprise one of a BERT model, a Transformer model, and a GPT model.
7. The device for training the NLU models online according to claim 1, wherein the processor is further caused to input one or more intents into a ChatGPT model and to pre-describe definitions of the intents to generate the corpus dataset by applying the ChatGPT model.
8. The device for training the NLU models online according to claim 1, wherein before online training of the NLU models based on the corpus dataset, the processor is further caused to perform error correction on the corpus dataset by applying the ChatGPT model to correct incorrect labels.
9. The device for training the NLU models online according to claim 1, wherein the processor is further caused to generate the dialog content by engaging the ChatGPT model in a conversation with a speech robot, wherein the processor then generates the corpus dataset based on the dialog content.
10. A method for training an NLU model online, comprising:
 - calling the NLU model, creating an intent within the NLU model, and defining one or more intent attributes of the intent, and forming a NLU model training corpus

data structure based on the one or more intent attributes;

obtaining a dialog content, performing data maintenance based on the NLU model training corpus data structure and linguistic context of the dialog content, thereby generating a corpus dataset;

performing online training of the NLU model based on the corpus dataset, to obtain a trained NLU model, and testing the trained NLU model to obtain a tested NLU model; and

deploying the tested NLU model to an external electronic device, for understanding, analyzing, and processing natural language acquired in real time.

11. A non-transitory computer-readable storage medium having a computer program stored thereon, wherein the computer program, when executed by a processor, implements the method for training the NLU model online according to claim 10.

12. An electronic terminal comprising: a processor and a memory;

wherein the memory comprises a storage for a computer program; and

wherein the processor executes the computer program to cause the electronic terminal to perform the method for training the NLU model online according to claim 10.