

# **METHOD AND APPARATUS FOR GENERATING PROMPT DATA BASED ON KNOWLEDGE GRAPH**

## **TECHNICAL FIELD**

[0001] One or more embodiments of this specification relate to the field of computer technologies, and in particular, to a method and an apparatus for generating a prompt based on a knowledge graph.

## **BACKGROUND**

[0002] Recently, a large language model has become popular worldwide. The large language model is a language model-based language generation technology, and can generate realistic natural language texts, including dialogues, stories, news, etc. People can intuitively perceive the great progress of the large language model in the fields of natural language understanding, natural language generation, etc. The language model is more widely applied, and can be further applied to machine translation, sentiment analysis, speech recognition, and other fields. With the application and development of the language model, people also recognize that there is room for improvement in correctness of facts understood by the language model in terms of natural language understanding, and there are still some limitations on credibility and controllability of generated content. A prompt has been proven to be valid in the language model. After the language model is obtained through training, the prompt can be used to direct the language model to provide a better answer. When the prompt includes privacy data, privacy protection needs to be performed on a process of generating and applying the prompt. How to generate a large quantity of high-quality prompts is a current problem.

[0003] Therefore, it is hoped that there can be an improved solution to efficiently generate a high-quality prompt, so as to improve a language prediction effect of the language model based on the prompt.

## SUMMARY

**[0004]** One or more embodiments of this specification describe a method and an apparatus for generating a prompt based on a knowledge graph, to efficiently generate a high-quality prompt. Specific technical solutions are as follows:

5 **[0005]** According to a first aspect, an embodiment provides a method for generating a prompt based on a knowledge graph, including:

**[0006]** obtaining a first reasoning rule and a matched first instance subgraph, where the first instance subgraph is from the knowledge graph, and the first reasoning rule includes a reasoning condition and a reasoning result;

10 **[0007]** obtaining a first question and answer template constructed based on the first reasoning rule, where the first question and answer template includes a question template and an answer template, the answer template includes a cause template and a result template, the question template and the result template are obtained by performing text conversion on the reasoning result, and the cause template is obtained by performing text conversion on the reasoning condition; and

15 **[0008]** generating a target text based on the first question and answer template and the first instance subgraph, where the target text includes a question text and an answer text, the answer text includes a cause text and a result text, and the target text is used as a prompt to adjust a language model.

**[0009]** In an implementation, the step of obtaining a first reasoning rule and a matched first instance subgraph includes:

**[0010]** obtaining several reasoning rules of the knowledge graph, where the several reasoning rules include the first reasoning rule; and

**[0011]** determining several instance subgraphs that match the first reasoning rule from the knowledge graph, where the several instance subgraphs include the first instance subgraph.

25 **[0012]** In an implementation, the step of obtaining a first reasoning rule and a matched first instance subgraph includes:

**[0013]** reading a first instance subgraph in the knowledge graph;

**[0014]** obtaining several reasoning rules of the knowledge graph; and

**[0015]** matching the first instance subgraph with the several reasoning rules, to obtain a  
30 matched first reasoning rule comprised by the several reasoning rules.

- [0016] In an implementation, the step of reading a first instance subgraph in the knowledge graph includes:
- [0017] receiving a to-be-queried first question text; and
- [0018] determining a first instance subgraph associated with the first question text from the knowledge graph.
- [0019] In an implementation, the question template is determined in the following manner:
- [0020] converting a text corresponding to the reasoning result into a general question, and determining the question template based on a conversion result.
- [0021] In an implementation, a text corresponding to the first reasoning rule includes several rule elements, and the several rule elements correspond to several instance elements in the first instance subgraph; and
- [0022] the step of determining the question template based on a conversion result includes:
- [0023] converting a text that is in the conversion result and that corresponds to the several rule elements into several to-be-filled slots, to obtain the question template.
- [0024] In an implementation, the result template is determined in the following manner:
- [0025] combining a preset word representing a meaning of "therefore" with a text corresponding to the reasoning result, and determining the result template based on a combination result.
- [0026] In an implementation, the cause template is determined in the following manner:
- [0027] combining a preset word representing a meaning of "because" with a text corresponding to the reasoning condition, and determining the cause template based on a combination result.
- [0028] In an implementation, the result template further includes a to-be-filled probability descriptor; and
- [0029] the step of generating a target text includes:
- [0030] obtaining a first evaluation indicator of the first reasoning rule;
- [0031] determining a probability descriptor corresponding to the first evaluation indicator from a preset correspondence between an evaluation indicator and a probability descriptor, filling the probability descriptor into the result template, and using a result template obtained after the filling as a prefilled result template; and
- [0032] generating the target text based on the question template, the cause template, the

prefilled result template, and the first instance subgraph.

**[0033]** In an implementation, the step of generating a target text includes:

**[0034]** obtaining a first evaluation indicator of the first reasoning rule;

**[0035]** determining a probability descriptor corresponding to the first evaluation indicator as  
5 a first probability descriptor from a preset correspondence between an evaluation indicator and a probability descriptor; and

**[0036]** generating the target text, so that the first probability descriptor is included at a predetermined location of the target text.

**[0037]** In an implementation, the first question and answer template includes several to-be-  
10 filled slots, and the several slots correspond to several rule elements in the first reasoning rule; and the step of generating a target text includes:

**[0038]** determining several instance elements that are in the first instance subgraph and that correspondingly match the several rule elements, and filling the several instance elements into the several slots, to obtain the target text.

15 **[0039]** According to a second aspect, an embodiment provides an apparatus for generating a prompt based on a knowledge graph, including:

**[0040]** a data obtaining module, configured to obtain a first reasoning rule and a matched first instance subgraph, where the first instance subgraph is from the knowledge graph, and the first reasoning rule includes a reasoning condition and a reasoning result;

20 **[0041]** a template obtaining module, configured to obtain a first question and answer template constructed based on the first reasoning rule, where the first question and answer template includes a question template and an answer template, the answer template includes a cause template and a result template, the question template and the result template are obtained by performing text conversion on the reasoning result, and the cause template is obtained by performing text  
25 conversion on the reasoning condition; and

**[0042]** a text generation module, configured to generate a target text based on the first question and answer template and the first instance subgraph, where the target text includes a question text and an answer text, the answer text includes a cause text and a result text, and the target text is used as a prompt to adjust a language model.

30 **[0043]** According to a third aspect, an embodiment provides a computer-readable storage medium. The computer-readable storage medium stores a computer program. When the computer

program is executed in a computer, the computer is enabled to perform the method in any implementation of the first aspect.

[0044] According to a fourth aspect, an embodiment provides a computing device, including a memory and a processor. The memory stores executable code. When the processor executes the executable code, the method in any implementation of the first aspect is implemented.

[0045] In the method and apparatus provided in the embodiments of this specification, a question and answer template is constructed by using a reasoning rule obtained from a knowledge graph, and the question and answer template is combined with an instance subgraph that matches the reasoning rule in the knowledge graph to generate a prompt. The question and answer template includes a question template and an answer template that includes a cause template and a result template. Therefore, a prompt generated based on the question and answer template includes a question text and an answer text that includes a cause text and a result text. A text including a reasoning process can be generated as a prompt by using the reasoning rule and high-quality data in the knowledge graph. This type of prompt is logically strong and highly efficient in a generation process.

## BRIEF DESCRIPTION OF DRAWINGS

[0046] To describe the technical solutions in the embodiments of this specification more clearly, the following briefly describes the accompanying drawings needed for describing the embodiments. Clearly, the accompanying drawings in the following descriptions show merely some embodiments of this specification, and a person of ordinary skill in the art can still derive other drawings from these accompanying drawings without creative efforts.

[0047] FIG. 1 is a schematic diagram of an implementation scenario of an embodiment disclosed in this specification;

[0048] FIG. 2 is a schematic flowchart of a method for generating a prompt based on a knowledge graph according to an embodiment;

[0049] FIG. 3 is a schematic diagram of structures of and a relationship between a first reasoning rule R1 and a first question and answer template QA1; and

[0050] FIG. 4 is a schematic block diagram of an apparatus for generating a prompt based on a knowledge graph according to an embodiment.

## DESCRIPTION OF EMBODIMENTS

**[0051]** The solutions provided in this specification are described below with reference to the accompanying drawings.

**[0052]** FIG. 1 is a schematic diagram of an implementation scenario of an embodiment disclosed in this specification. A corresponding question and answer template can be constructed based on a reasoning rule. An instance subgraph can be extracted from a knowledge graph. When a reasoning rule and an instance subgraph that match each other are obtained, the instance subgraph can be combined with a question and answer template corresponding to the reasoning rule to generate a prompt.

**[0053]** The knowledge graph is intended to describe various entities or concepts that exist in the real world and their relationships. The knowledge graph forms a huge semantic network graph, and is a knowledge base for expressing knowledge. The knowledge graph can express a large amount of complex knowledge in a more orderly manner. Data in the knowledge graph is usually characterized by high factual correctness, controllability, interpretability, etc. It should be emphasized that all information or data mentioned in the embodiments of this specification is used when authorization from a corresponding data object is obtained.

**[0054]** The knowledge graph includes a plurality of nodes representing entities and a connecting edge representing a relationship between nodes. The node and the connecting edge can be referred to as elements in the knowledge graph. Examples of some nodes and connecting edges in a knowledge graph are listed in FIG. 1. Circles and gray dots represent nodes, and connecting lines between nodes represent connecting edges. Gray dots and straight lines are schematic diagrams of more nodes and connecting edges. The entity is a thing in the real world, for example, a place name, a drug, an organization, an institution, a device, a number, etc. The entity can be represented by an entity word, and the entity word has a noun nature. For example, "cola", "beverage", etc. are entity names. The relationship is used to express a connection between different entities. For example, in a connection relationship "Cola-is-a beverage", the relationship is "is", reflecting relationship data such as "Cola is a beverage".

**[0055]** In the knowledge graph, the node includes information such as a node name and a node type, and the connecting edge includes information such as a relationship type. For example, in the knowledge graph shown in FIG. 1, a node type of "convenience store xx" is "merchant", node

types of "cola" and "orange juice" are "product", a relationship type between "convenience store xx" and "cola" is "purchases", a relationship type between "convenience store xx" and "orange juice" is "purchases", a relationship type between "cola" and "beverage" is "is", and a relationship type between "cola" and "orange juice" is "is". The relationship further includes a relationship attribute. Attributes of the relationship type "purchases" include quantity of times>k1 and quantity<k2.

**[0056]** The reasoning rule is obtained based on a node type and a relationship type in the knowledge graph, and is used as logic for summarization and reasoning. Rule elements in the reasoning rule include a node type, a relationship type, etc. The relationship type includes a relationship type existing in the knowledge graph and a predefined relationship type. The reasoning rule usually includes a reasoning condition and a reasoning result. With reference to the reasoning rule example shown in FIG. 1, "{A merchant} [purchases] {a product} (for a plurality of times), and {the product} [belongs to] {a category} → {The merchant} [prefers] {the category}" is a reasoning rule. The arrow follows a reasoning condition and is followed by a reasoning result, {} represents a node type, and [] represents a relationship type.

**[0057]** There can be a plurality of sources of the reasoning rule. For example, the reasoning rule can be obtained from the knowledge graph by using a rule extraction algorithm, or can be obtained through summarization by an expert based on experience. When outputting the reasoning rule, the rule extraction algorithm outputs an evaluation indicator such as a confidence and/or coverage of the reasoning rule. For example, in FIG. 1, the confidence of the reasoning rule is 0.85. The evaluation indicator is used to evaluate an effect of the reasoning rule. For example, the confidence is used to reflect the credibility of the reasoning rule, and the coverage is used to reflect a range of an instance subgraph hit by the reasoning rule in the knowledge graph. When a question and answer template is constructed, the evaluation indicator can be converted into a probability descriptor in the question and answer template. That the reasoning rule hits an instance subgraph in the knowledge graph can also be understood as that the reasoning rule matches the instance subgraph, or the instance subgraph meets the reasoning rule.

**[0058]** The instance subgraph is a relationship graph including several hops of neighbor nodes centered around a specific node in the knowledge graph. The instance subgraph can include several triples. The triples include a triple with a central node as a head node or a tail node, and a triple with a neighbor node of a central node as a head node or a tail node. The triple includes a head

node, a connecting edge, and a tail node. For example, FIG. 1 shows an instance subgraph in which the convenience store xx is used as a central node in the knowledge graph.

**[0059]** A prompt is an input form or template designed by researchers for downstream tasks, to help a pre-trained language model "recall" what the language model has "learned" during pre-training. The prompt can further direct the pre-trained language model to perform fine tuning, to direct the language model to answer in a desired manner.

**[0060]** To more efficiently generate a high-quality prompt, the embodiments of this specification provide a method for generating a prompt based on a knowledge graph. The method includes the following steps: Step S210: Obtain a first reasoning rule and a matched first instance subgraph, where the first reasoning rule includes a reasoning condition and a reasoning result; step S220: Obtain a first question and answer template constructed based on the first reasoning rule, where the first question and answer template includes a question template and an answer template, the answer template includes a cause template and a result template, the question template and the result template are obtained by performing text conversion on the reasoning result, and the cause template is obtained by performing text conversion on the reasoning condition; and step S230: Generate a target text based on the first question and answer template and the first instance subgraph, where the target text includes a question text and an answer text, the answer text includes a cause text and a result text, and the target text is used as a prompt to adjust a language model.

**[0061]** The following describes the embodiments in detail with reference to FIG. 2.

**[0062]** FIG. 2 is a schematic flowchart of a method for generating a prompt based on a knowledge graph according to an embodiment. The method is performed by a computing device, and the computing device can be implemented by any apparatus, device, platform, device cluster, etc. having computing and processing capabilities. The knowledge graph can be stored in the computing device, or can be stored in another device. There are one or more reasoning rules of the knowledge graph, and a first reasoning rule R1 is any one of several reasoning rules. A first instance subgraph G1 is any one of a plurality of instance subgraphs hit by the first reasoning rule R1. The instance subgraph is from the knowledge graph.

**[0063]** The following describes in detail the steps of the method for generating a prompt.

**[0064]** In step S210, the first reasoning rule R1 and the matched first instance subgraph G1 are obtained.

**[0065]** The first instance subgraph G1 is from the knowledge graph. The first reasoning rule

R1 includes a reasoning condition R1\_1 and a reasoning result R1\_2.

[0066] This embodiment can be applied to a plurality of implementation scenarios. For example, this embodiment can be executed offline, and several reasoning rules are collected in advance, to generate a prompt. For another example, when online question answering is performed by using a language model, after a to-be-queried question is received, a prompt is generated based on the to-be-queried question, and the prompt is used to assist the language model in determining an answer to the to-be-answered question.

[0067] In an implementation, during execution, step S210 can include: obtaining several reasoning rules of the knowledge graph, where the several reasoning rules include the first reasoning rule R1; and determining several instance subgraphs that match the first reasoning rule R1 from the knowledge graph, where the several instance subgraphs include the first instance subgraph G1. When there are a large quantity of instance subgraphs that match the first reasoning rule R1 in the knowledge graph, the plurality of instance subgraphs can be sampled.

[0068] "Several" means "one or more". An instance subgraph that matches the first reasoning rule R1 means that the instance subgraph meets the reasoning condition R1\_1 in the first reasoning rule R1.

[0069] In an implementation, during execution, step S210 can include: reading a first instance subgraph G1 in the knowledge graph; obtaining several reasoning rules of the knowledge graph; and then matching the first instance subgraph G1 with each of the several reasoning rules, to obtain a matched first reasoning rule R1.

[0070] The reading a first instance subgraph G1 in the knowledge graph can be specifically determining a first instance subgraph G1 associated with a to-be-queried first question text from the knowledge graph when receiving the first question text. The first question text is any question text. When the first question text is received, an instance word in the first question text can be analyzed, and a first instance subgraph G1 corresponding to the entity word is determined from the knowledge graph based on the entity word.

[0071] In this embodiment, only the first reasoning rule and the matched first instance subgraph are used as examples to describe how to generate a prompt. When there are a plurality of reasoning rules and a plurality of instance subgraphs, a prompt can be generated for any reasoning rule and a corresponding instance subgraph by using the method provided in this embodiment.

**[0072]** In step S220, a first question and answer template QA1 constructed based on the first reasoning rule R1 is obtained.

**[0073]** A plurality of question and answer templates can be constructed based on the first reasoning rule R1, and the first question and answer template QA1 can be one of the plurality of question and answer templates. Primary parts of the plurality of question and answer templates can be the same, and words used in secondary parts of the plurality of question and answer templates can be different.

**[0074]** The step of constructing a first question and answer template QA1 can be that after pre-execution, the first question and answer template QA1 is stored in specified space, and the first question and answer template QA1 is obtained from the specified space when being required. Alternatively, the first question and answer template QA1 can be constructed in step S220.

**[0075]** FIG. 3 is a schematic diagram of structures of and a relationship between the first reasoning rule R1 and the first question and answer template QA1. The first question and answer template QA1 includes a question template Q1 and an answer template A1. The answer template A1 includes a cause template A1\_1 and a result template A1\_2. The question template Q1 and the result template A1\_2 are obtained by performing text conversion on the reasoning result R1\_2, and the cause template A1\_1 is obtained by performing text conversion on the reasoning condition R1\_1.

**[0076]** The computing device can construct the first question and answer template QA1 based on the first reasoning rule R1 and template construction logic. For example, the question template Q1 can be determined by using the following template construction logic: converting a text corresponding to the reasoning result R1\_2 into a general question, and determining the question template Q1 based on a conversion result.

**[0077]** For example, for the reasoning result R1\_2 "{The merchant} [prefers] {the category}" in the reasoning rule in FIG. 1, the text is converted into a general question, and a text "Does {the merchant} [prefer] {the category}" of a conversion result can be obtained.

**[0078]** There can be another type of template construction logic, and the text of the reasoning result is not necessarily converted into a general question. The text of the reasoning result can be converted into a special question based on a reasoning focus of the reasoning result. For example, a question about a subject or an object is asked.

**[0079]** The result template A1\_2 can be determined by using the following template

construction logic: combining a preset word representing a meaning of "therefore" with a text corresponding to the reasoning result R1\_2, and determining the result template A1\_2 based on a combination result. The word representing the meaning of "therefore" includes hence, so, therefore, etc. If different words are selected, different question and answer templates corresponding to the first reasoning rule R1 can be obtained. Specifically, the word representing the meaning of "therefore" can be placed in a start part or another part of the text corresponding to the reasoning result R1\_2.

**[0080]** For example, for the reasoning result R1\_2 "{The merchant} [prefers] {the category}" in the reasoning rule in FIG. 1, if "therefore" is placed in a start part of the reasoning result, an obtained combination result can be "therefore {the merchant} [prefers] {the category}".

**[0081]** The cause template A1\_1 can be determined by using the following template construction logic: combining a preset word representing a meaning of "because" with a text corresponding to the reasoning condition R1\_1, and determining the cause template A1\_1 based on a combination result. The word representing the meaning of "because" includes due to, because, etc. Specifically, the word representing the meaning of "because" can be placed in a start part or another part of the text corresponding to the reasoning condition R1\_1.

**[0082]** For example, for the reasoning condition R1\_1 "{A merchant} [purchases] {a product} (for a plurality of times), and {the product} [belongs to] {a category}" in the reasoning rule in FIG. 1, if "because" is placed in a start part of the reasoning condition, an obtained combination result can be "because {a merchant} [purchases] {a product} (for a plurality of times), and {the product} [belongs to] {a category}". The combination result can be further more properly adjusted in terms of word order, to obtain "because {a merchant} [purchases] {a product} for a plurality of times, and {the product} [belongs to] {a category}".

**[0083]** In specific implementation, when a text corresponding to the first reasoning rule R1 includes several rule elements, the several rule elements correspond to several instance elements in the first instance subgraph G1. The rule element can include a node type, a relationship type, and a custom type. The instance element includes a node and a relationship. The node herein can be replaced with a node name, and the relationship can be replaced with a relationship type.

**[0084]** The reasoning rule and the instance subgraph in FIG. 1 are used as examples for description. Rule elements in the reasoning rule include node types {merchant}, {product}, and {category}, relationship types [purchases], [is], and [prefers], etc. The relationship type [purchases]

corresponds to the relationship type between "convenience store xx" and "cola" and the relationship type between "convenience store xx" and "orange juice" in the instance subgraph. The relationship type [is] corresponds to the relationship type between "cola" and "beverage" and the relationship type between "orange juice" and "beverage". The relationship type [prefers] is a custom relationship type, and is also a relationship type derived from the reasoning rule. The node types {merchant}, {product}, and {category} correspond to node types in the instance subgraph.

**[0085]** When the question template Q1 is determined based on the conversion result, a text that is in the conversion result and that corresponds to the several rule elements can be converted into several to-be-filled slots, to obtain the question template Q1. Alternatively, the text may not be converted into a to-be-filled slot, but the text that is in the conversion result and that corresponds to the several rule elements is marked as a to-be-replaced character.

**[0086]** When the result template A1\_2 is determined based on the combination result, a text that is in the combination result and that corresponds to the several rule elements is converted into several to-be-filled slots, to obtain the result template A1\_2.

**[0087]** When the cause template A1\_1 is determined based on the combination result, a text that is in the combination result and that corresponds to the several rule elements is converted into several to-be-filled slots, to obtain the cause template A1\_1.

**[0088]** In step S230, a target text is generated based on the first question and answer template QA1 and the first instance subgraph G1, where the target text is used as a prompt. The target text includes a question text and an answer text, and the answer text includes a cause text and a result text. The target text is used as a prompt, and can be subsequently used to adjust a language model.

**[0089]** The target text can be obtained by combining the first question and answer template QA1 with the first instance subgraph G1. The first question and answer template QA1 includes elements corresponding to rule elements in the first reasoning rule R1, and the instance element in the first instance subgraph G1 corresponds to the rule element in the first reasoning rule R1. Therefore, a correspondence between the element in the first question and answer template QA1 and the instance element in the first instance subgraph G1 can be determined.

**[0090]** In an implementation, when the first question and answer template QA1 includes several to-be-filled slots, and the several slots correspond to several rule elements in the first reasoning rule R1, several instance elements that are in the first instance subgraph G1 and that correspondingly match the several rule elements can be determined, and the several instance

elements can be filled into the several slots, to obtain the target text.

[0091] To enrich a meaning of the generated prompt, a correspondence between an evaluation indicator of a reasoning rule and a probability descriptor can be further preset. A larger evaluation indicator value indicates higher credibility or a higher occurrence possibility of a reasoning result and a higher possibility represented by a corresponding probability descriptor. The evaluation indicator can include a confidence and coverage. In Table 1, the confidence is used as an example, and a correspondence between different confidence values and probability descriptors is listed.

**Table 1**

| Confidence                    | Probability descriptor |
|-------------------------------|------------------------|
| Greater than or equal to 0.99 | Definitely             |
| Greater than or equal to 0.95 | Almost                 |
| Greater than or equal to 0.90 | Highly likely          |
| Greater than or equal to 0.85 | Very likely            |
| ...                           | ...                    |

[0092] Higher closeness of the confidence value to 1 indicates a higher occurrence possibility.

[0093] In an implementation, a to-be-filled probability descriptor can be set in the result template A1\_2. Specifically, a to-be-filled probability descriptor slot can be added. When the target text is generated in step S230, a first evaluation indicator of the first reasoning rule R1 can be obtained, a probability descriptor corresponding to the first evaluation indicator can be determined from the correspondence, the probability descriptor can be filled into the result template A1\_2, and a result template obtained after the filling can be used as a prefilled result template. Then, the target text is generated based on the question template Q1, the cause template A1\_1, the prefilled result template A1\_2, and the first instance subgraph G1.

[0094] In an implementation, a probability descriptor can be added when the target text is generated. When step S230 is performed, a first evaluation indicator of the first reasoning rule R1 can be obtained, and a probability descriptor corresponding to the first evaluation indicator can be determined as a first probability descriptor from the correspondence. Then, the target text is generated, so that the first probability descriptor is included at a predetermined location of the target text. The predetermined location can be a text location set based on experience.

**[0095]** The reasoning rule and the question and answer template in FIG. 1 are used as examples below for description. The confidence of the reasoning rule is 0.85, and a corresponding probability descriptor is "very likely". In the question template, [] and {} represent to-be-filled slots, a text in {} represents a node type, and a text in [] represents a relationship type. A

5 correspondence between a question and answer template and an instance subgraph is listed in Table 2.

**Table 2**

|                              |                                                                                                                                                                                                                                 |                      |           |         |              |            |          |
|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|-----------|---------|--------------|------------|----------|
| Question and answer template | Q: Does {the merchant} [prefers] {the category}?<br>A: {The merchant} [purchases] {the product} for a plurality of times, and {the product} [belongs to] {the category}, and therefore {the merchant} [prefers] {the category}. |                      |           |         |              |            |          |
| Instance subgraph            | Node type or relationship type                                                                                                                                                                                                  | Convenience store xx | Purchases | Cola    | Orange juice | Is         | Beverage |
|                              | Instance element                                                                                                                                                                                                                | Merchant             | Purchases | Product | Product      | Belongs to | Category |

**[0096]** In Table 2, the first row is specific content of the question and answer template, and the

10 second row is an instance element included in the instance subgraph and a corresponding node type or relationship type. The instance element is correspondingly filled into a to-be-filled slot. For example, the merchant "convenience store xx" is filled into the {merchant} slot, and the probability descriptor "very likely" is filled into the {probability descriptor} slot. After the filling, a target text shown in Table 3 is obtained.

**Table 3**

|               |                                                           |                                                                                                                           |
|---------------|-----------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Question text | Does {the convenience store xx} [prefers] {the beverage}? |                                                                                                                           |
| Answer text   | Cause text                                                | {The convenience store xx} [purchases] {cola} and {orange juice} for a plurality of times, and {the cola} and {the orange |

|  |             |                                                                                           |
|--|-------------|-------------------------------------------------------------------------------------------|
|  |             | juice} [are] {beverages}                                                                  |
|  | Result text | Therefore {it is very likely that} {the convenience store xx}<br>[prefers] {the beverage} |

**[0097]** In the question text and the answer text, {} and [] merely represent that there are originally to-be-filled slots herein, and the actual text does not include the symbols.

**[0098]** In this embodiment, the generated prompt includes the question text and the answer text, and the answer text is divided into the cause text and the result text. In this way, a result obtaining process is more clearly displayed by using the prompt. In addition, the evaluation indicator of the reasoning rule is converted into a corresponding probability descriptor and added to the prompt. Therefore, the prompt is semantically richer and the answer is more accurate.

**[0099]** The language model in this embodiment is a natural language processing model trained based on a deep learning technology and a large-scale corpus. By learning a large quantity of language samples, the language model can learn language structures and rules, and can generate proper natural language texts. The language model can be applied to the fields of question and answer, machine translation, text generation, sentiment analysis, speech recognition, etc., and is one of important technologies in natural language processing. The language model in this embodiment can include a large language model and a small/medium language model.

**[0100]** The method for generating a prompt provided in this embodiment can be automatically performed in batches by using the computing device. This reduces manual participation, and can significantly improve efficiency. A large quantity of reasoning rules are accumulated in the knowledge graph. These reasoning rules are descriptions at a schema level, and the schema is defined. Therefore, the question and answer template can directly use the schema, to efficiently generate a prompt.

**[0101]** The prompt generated in this embodiment is logically strong. In a process of constructing the question and answer template, both a type constraint of the schema and a reasoning rule constraint are met. Therefore, a logically strong prompt can be obtained. The logically strong prompt helps improve a logic capability and a reasoning capability of the language model. There is a correspondence between an evaluation indicator and a probability descriptor. Therefore, in this embodiment, an accurate descriptor can be added to the prompt, so that the prompt is more accurately described, to facilitate more refined learning and reasoning of the

language model.

**[0102]** In this embodiment, the prompt is generated by using the reasoning rule, so that the reasoning rule is reused. In addition, data in the knowledge graph is verified and constrained by the schema, and is of higher quality.

5 **[0103]** In this specification, "first" in words such as the first reasoning rule, the first instance subgraph, the first question and answer template, and the first evaluation indicator and corresponding "second" (if present) in this specification are merely used for ease of distinguishing and description, and have no limitation meaning.

**[0104]** Specific embodiments of this specification are described above, and other  
10 embodiments fall within the scope of the appended claims. In some cases, the actions or steps described in the claims can be performed in a sequence different from that in the embodiments and desired results can still be achieved. In addition, the process depicted in the accompanying drawings does not necessarily need a particular sequence or consecutive sequence to achieve the desired results. In some implementations, multitasking and parallel processing are possible or may  
15 be advantageous.

**[0105]** FIG. 4 is a schematic block diagram of an apparatus for generating a prompt based on a knowledge graph according to an embodiment. The apparatus embodiment corresponds to the method embodiment shown in FIG. 2. The apparatus 400 is deployed in a computing device. The computing device can be implemented by any apparatus, device, platform, device cluster, etc.  
20 having computing and processing capabilities. The apparatus 400 includes:

**[0106]** a data obtaining module 410, configured to obtain a first reasoning rule and a matched first instance subgraph, where the first instance subgraph is from the knowledge graph, and the first reasoning rule includes a reasoning condition and a reasoning result;

**[0107]** a template obtaining module 420, configured to obtain a first question and answer  
25 template constructed based on the first reasoning rule, where the first question and answer template includes a question template and an answer template, the answer template includes a cause template and a result template, the question template and the result template are obtained by performing text conversion on the reasoning result, and the cause template is obtained by performing text conversion on the reasoning condition; and

30 **[0108]** a text generation module 430, configured to generate a target text based on the first question and answer template and the first instance subgraph, where the target text includes a

question text and an answer text, the answer text includes a cause text and a result text, and the target text is used as a prompt to adjust a language model.

**[0109]** In an implementation, the data obtaining module 410 includes a first obtaining submodule and a first determining submodule (not shown in the figure);

5 **[0110]** the first obtaining submodule is configured to obtain several reasoning rules of the knowledge graph, where the several reasoning rules include the first reasoning rule; and

**[0111]** the first determining submodule is configured to determine several instance subgraphs that match the first reasoning rule from the knowledge graph, where the several instance subgraphs include the first instance subgraph.

10 **[0112]** In an implementation, the data obtaining module 410 includes a first reading submodule, a second obtaining submodule, and a first matching submodule (not shown in the figure);

**[0113]** the first reading submodule is configured to read a first instance subgraph in the knowledge graph;

15 **[0114]** the second obtaining submodule is configured to obtain several reasoning rules of the knowledge graph; and

**[0115]** the first matching submodule is configured to match the first instance subgraph with each of the several reasoning rules, to obtain a matched first reasoning rule.

**[0116]** In an implementation, the first reading submodule is specifically configured to:

20 **[0117]** receive a to-be-queried first question text; and

**[0118]** determine a first instance subgraph associated with the first question text from the knowledge graph.

**[0119]** In an implementation, the apparatus 400 further includes a first determining module (not shown in the figure), configured to determine the question template in the following manner:

25 **[0120]** converting a text corresponding to the reasoning result into a general question, and determining the question template based on a conversion result.

**[0121]** In an implementation, a text corresponding to the first reasoning rule includes several rule elements, and the several rule elements correspond to several instance elements in the first instance subgraph; and

30 **[0122]** when the first determining module determines the question template based on the conversion result, the following operation is performed:

**[0123]** converting a text that is in the conversion result and that corresponds to the several rule elements into several to-be-filled slots, to obtain the question template.

**[0124]** In an implementation, the apparatus 400 further includes a second determining module (not shown in the figure), configured to determine the result template in the following manner:

5 **[0125]** combining a preset word representing a meaning of "therefore" with a text corresponding to the reasoning result, and determining the result template based on a combination result.

**[0126]** In an implementation, the apparatus 400 further includes a third determining module (not shown in the figure), configured to determine the cause template in the following manner:

10 **[0127]** combining a preset word representing a meaning of "because" with a text corresponding to the reasoning condition, and determining the cause template based on a combination result.

**[0128]** In an implementation, the result template further includes a to-be-filled probability descriptor; and the text generation module 430 includes a third obtaining submodule, a second determining submodule, and a first generation submodule (not shown in the figure);

**[0129]** the third obtaining submodule is configured to obtain a first evaluation indicator of the first reasoning rule;

**[0130]** the second determining submodule is configured to: determine a probability descriptor corresponding to the first evaluation indicator from a preset correspondence between an evaluation indicator and a probability descriptor, fill the probability descriptor into the result template, and use a result template obtained after the filling as a prefilled result template; and

**[0131]** the first generation submodule is configured to generate the target text based on the question template, the cause template, the prefilled result template, and the first instance subgraph.

25 **[0132]** In an implementation, the text generation module 430 includes a fourth obtaining submodule, a third determining submodule, and a second generation submodule (not shown in the figure);

**[0133]** the fourth obtaining submodule is configured to obtain a first evaluation indicator of the first reasoning rule;

**[0134]** the third determining submodule is configured to determine a probability descriptor corresponding to the first evaluation indicator as a first probability descriptor from a preset correspondence between an evaluation indicator and a probability descriptor; and

**[0135]** the second generation submodule is configured to generate the target text, so that the first probability descriptor is included at a predetermined location of the target text.

**[0136]** In an implementation, the first question and answer template includes several to-be-filled slots, and the several slots correspond to several rule elements in the first reasoning rule; and

5 the text generation module 430 is specifically configured to:

**[0137]** determine several instance elements that are in the first instance subgraph and that correspondingly match the several rule elements, and fill the several instance elements into the several slots, to obtain the target text.

**[0138]** The apparatus embodiments correspond to the method embodiments. For specific  
10 descriptions, references can be made to the descriptions in the method embodiments. Details are omitted here for simplicity. The apparatus embodiments are obtained based on the corresponding method embodiments, and have the same technical effects as the corresponding method embodiments. For specific descriptions, references can be made to the corresponding method embodiments.

15 **[0139]** An embodiment of this specification further provides a computer-readable storage medium. The computer-readable storage medium stores a computer program. When the computer program is executed in a computer, the computer is enabled to perform the method in any one of FIG. 1 to FIG. 3.

**[0140]** An embodiment of this specification further provides a computing device, including a  
20 memory and a processor. The memory stores executable code. When the processor executes the executable code, the method in any one of FIG. 1 to FIG. 3 is implemented.

**[0141]** The embodiments of this specification are described in a progressive manner. For the same or similar parts of the embodiments, mutual references can be made between the embodiments. Each embodiment focuses on a difference from other embodiments. In particular,  
25 the embodiments of the storage medium and the computing device are basically similar to the method embodiments, and therefore are described briefly. For related parts, references can be made to some descriptions in the method embodiments.

**[0142]** A person skilled in the art should be aware that in the above-mentioned one or more examples, functions described in the embodiments of this specification can be implemented by  
30 hardware, software, firmware, or any combination thereof. When being implemented by software, these functions can be stored in a computer-readable medium or transmitted as one or more

instructions or code on a computer-readable medium.

**[0143]** The objectives, technical solutions, and beneficial effects of the embodiments of this specification are further described in detail in the specific implementations described above. It should be understood that the above-mentioned descriptions are merely specific implementations  
5 of the embodiments of this specification, and are not intended to limit the protection scope of this specification. Any modification, equivalent replacement, improvement, etc. made based on the technical solutions in this specification shall fall within the protection scope of this specification.

## CLAIMS

1. A method for generating a prompt based on a knowledge graph, comprising:

obtaining a first reasoning rule and a matched first instance subgraph, wherein the first instance subgraph is from the knowledge graph, and the first reasoning rule comprises a reasoning condition and a reasoning result;

obtaining a first question and answer template constructed based on the first reasoning rule, wherein the first question and answer template comprises a question template and an answer template, the answer template comprises a cause template and a result template, the question template and the result template are obtained by performing text conversion on the reasoning result, and the cause template is obtained by performing text conversion on the reasoning condition; and

generating a target text based on the first question and answer template and the first instance subgraph, wherein the target text comprises a question text and an answer text, the answer text comprises a cause text and a result text, and the target text is used as a prompt to adjust a language model.

2. The method according to claim 1, wherein the step of obtaining a first reasoning rule and a matched first instance subgraph comprises:

obtaining several reasoning rules of the knowledge graph, wherein the several reasoning rules comprise the first reasoning rule; and

determining several instance subgraphs that match the first reasoning rule from the knowledge graph, wherein the several instance subgraphs comprise the first instance subgraph.

3. The method according to claim 1, wherein the step of obtaining a first reasoning rule and a matched first instance subgraph comprises:

reading a first instance subgraph in the knowledge graph;

obtaining several reasoning rules of the knowledge graph; and

matching the first instance subgraph with the several reasoning rules, to obtain a matched first reasoning rule comprised by the several reasoning rules.

4. The method according to claim 3, wherein the step of reading a first instance subgraph in the knowledge graph comprises:

receiving a to-be-queried first question text; and

determining a first instance subgraph associated with the first question text from the

knowledge graph.

5. The method according to claim 1, wherein the question template is determined in the following manner:

converting a text corresponding to the reasoning result into a general question, and  
5 determining the question template based on a conversion result.

6. The method according to claim 5, wherein a text corresponding to the first reasoning rule comprises several rule elements, and the several rule elements correspond to several instance elements in the first instance subgraph; and

the step of determining the question template based on a conversion result comprises:

10 converting a text that is in the conversion result and that corresponds to the several rule elements into several to-be-filled slots, to obtain the question template.

7. The method according to claim 1, wherein the result template is determined in the following manner:

combining a preset word representing a meaning of "therefore" with a text corresponding to  
15 the reasoning result, and determining the result template based on a combination result.

8. The method according to claim 1, wherein the cause template is determined in the following manner:

combining a preset word representing a meaning of "because" with a text corresponding to  
the reasoning condition, and determining the cause template based on a combination result.

20 9. The method according to claim 1, wherein the result template further comprises a to-be-filled probability descriptor; and

the step of generating a target text comprises:

obtaining a first evaluation indicator of the first reasoning rule;

determining a probability descriptor corresponding to the first evaluation indicator from a  
25 preset correspondence between an evaluation indicator and a probability descriptor, filling the probability descriptor into the result template, and using a result template obtained after the filling as a prefilled result template; and

generating the target text based on the question template, the cause template, the prefilled result template, and the first instance subgraph.

30 10. The method according to claim 1, wherein the step of generating a target text comprises:  
obtaining a first evaluation indicator of the first reasoning rule;

determining a probability descriptor corresponding to the first evaluation indicator as a first probability descriptor from a preset correspondence between an evaluation indicator and a probability descriptor; and

5 generating the target text, so that the first probability descriptor is comprised at a predetermined location of the target text.

11. The method according to claim 1, wherein the first question and answer template comprises several to-be-filled slots, and the several slots correspond to several rule elements in the first reasoning rule; and the step of generating a target text comprises:

10 determining several instance elements that are in the first instance subgraph and that correspondingly match the several rule elements, and filling the several instance elements into the several slots, to obtain the target text.

12. An apparatus for generating a prompt based on a knowledge graph, comprising:

15 a data obtaining module, configured to obtain a first reasoning rule and a matched first instance subgraph, wherein the first instance subgraph is from the knowledge graph, and the first reasoning rule comprises a reasoning condition and a reasoning result;

20 a template obtaining module, configured to obtain a first question and answer template constructed based on the first reasoning rule, wherein the first question and answer template comprises a question template and an answer template, the answer template comprises a cause template and a result template, the question template and the result template are obtained by performing text conversion on the reasoning result, and the cause template is obtained by performing text conversion on the reasoning condition; and

25 a text generation module, configured to generate a target text based on the first question and answer template and the first instance subgraph, wherein the target text comprises a question text and an answer text, the answer text comprises a cause text and a result text, and the target text is used as a prompt to adjust a language model.

13. A computer-readable storage medium, wherein the computer-readable storage medium stores a computer program, and when the computer program is executed in a computer, the computer is enabled to perform the method according to any one of claims 1 to 11.

30 14. A computing device, comprising a memory and a processor, wherein the memory stores executable code, and when the processor executes the executable code, the method according to any one of claims 1 to 11 is implemented.