(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
08 February 2018 (08.02.2018)

WIPO | PCT

(10) International Publication Number
WO 2018/023356 A1

(51) International Patent Classification:
G06F 17/28 (2006.01)

(21) International Application Number:
PCT/CN2016/092762

(22) International Filing Date:
01 August 2016 (01.08.2016)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **MICROSOFT TECHNOLOGY LICENSING, LLC** [US/US]; One Microsoft Way, Redmond, Washington 98052 (US).

(72) Inventors; and

(71) Applicants *(for US only)*: **LI, Mu** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **ZHOU, Ming** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **LIU, Shujie** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US).

(74) Agent: **NTD PATENT & TRADEMARK AGENCY LIMITED**; 10th Floor, Block A, Investment Plaza, 27 Jinrongdajie, Xicheng District, Beijing 100033 (CN).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:
— *of inventorship (Rule 4.17(iv))*

(54) **Title:** MACHINE TRANSLATION METHOD AND APPARATUS

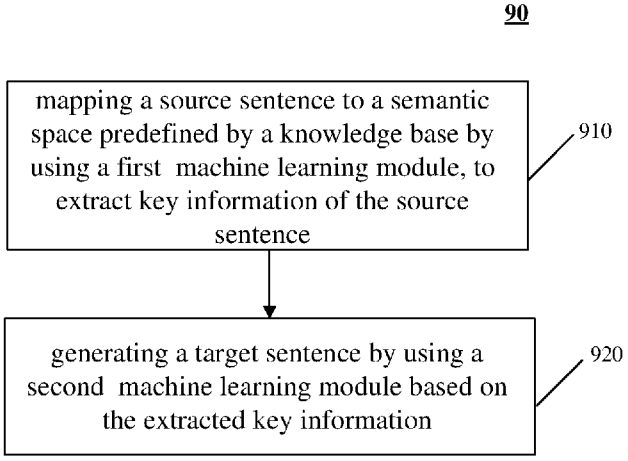

FIG. 9

(57) **Abstract:** A neural network based translation method, wherein the method includes: mapping a source sentence to a semantic space predefined by a knowledge base by using a first machine learning module, to extract key information of the source sentence (910); and generating a target sentence by using a second machine learning module based on the extracted key information (920).

WO 2018/023356 A1

[Continued on next page]

Published:
— *with international search report (Art. 21(3))*

MACHINE TRANSLATION METHOD AND APPARATUS

BACKGROUND

[0001] Machine learning techniques such as Deep Neural Networks (DNN) have achieved excellent performance on difficult learning tasks such as visual object recognition and speech recognition. Some models based on the machine learning techniques such as the deep neural networks have been successfully applied for machine translation. Typically a huge data corpus having tremendous bilingual data pairs is required to train such a DNN-based model in order to achieve satisfied translation performance. Otherwise, if there is not enough bilingual training data, the translation performance would be impaired.

SUMMARY

[0002] The following summary is provided to introduce a selection of concepts in a simplified form that are further described below in the detailed description. This summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject matter.

[0003] According to an embodiment of the subject matter described herein, a machine translation method comprises: mapping a source sentence in a first language to a semantic space predefined by a knowledge base by using a first machine learning module, to extract key information of the source sentence; and generating a target sentence in a second language by using a second machine learning module based on the extracted key information.

[0004] According to an embodiment of the subject matter, a machine translation apparatus comprises: a first machine learning module configured to map a source sentence in a first language to a semantic space predefined by a knowledge base, to extract key information of the source sentence; and a second machine learning module configured to generate a target sentence in a second language based on the extracted key information.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] Various aspects, features and advantages of the subject matter will be more apparent from the detailed description set forth below when taken in conjunction with the drawings, in which use of the same reference number in different figures indicates similar or identical items.

[0006] FIG. 1 illustrates a block diagram of an exemplary environment where embodiments of the subject matter described herein may be implemented;

[0007] FIG. 2 illustrates an existing neural network model for machine translation;

[0008] FIG. 3 illustrates a structure for machine translation utilizing a knowledge-based semantic space according to an embodiment of the subject matter;

[0009] FIG. 4 illustrates a knowledge base organized in a tree structure according to an embodiment of the subject matter;

[0010] FIG. 5 illustrates a machine learning module for extracting key information of a source sentence according to an embodiment of the subject matter;

[0011] FIG. 5A illustrates an exemplary process for obtaining a hidden vector in a recurrent state according to an embodiment of the subject matter;

[0012] FIG. 6 illustrates a machine learning module for generating a target sentence according to an embodiment of the subject matter;

[0013] FIG. 6A illustrates a neural network-based model for generating a target sentence according to an embodiment of the subject matter;

[0014] FIG. 6B illustrates how explicit semantic tuples control the generation of a target sentence according to an embodiment of the subject matter;

[0015] FIG. 7 illustrates a machine learning module for extracting key information of a source sentence according to an embodiment of the subject matter;

[0016] FIG. 7A illustrates a structure for calculating attention weights for respective hidden vectors according to an embodiment of the subject matter;

[0017] FIG. 8 illustrates a machine learning module for generating a target sentence according to an embodiment of the subject matter;

[0018] FIG. 9 illustrates a process for translating a source sentence to a target sentence according to an embodiment of the subject matter;

[0019] FIG. 10 illustrates a block diagram of an apparatus for machine translation according to an embodiment of the subject matter; and

[0020] FIG. 11 illustrates a block diagram of a computer system for machine translation according to an embodiment of the subject matter.

DETAILED DESCRIPTION

[0021] The subject matter described herein will now be discussed with reference to example embodiments. It should be understood these embodiments are discussed only for the purpose of enabling those skilled persons in the art to better understand and thus implement the subject matter described herein, rather than suggesting any limitations on the scope of the subject matter.

[0022] As used herein, the term "includes" and its variants are to be read as open terms that mean "includes, but is not limited to". The term "based on" is to be read as "based at least in part on". The terms "one embodiment" and "an embodiment" are to be read as "at least one implementation". The term "another embodiment" is to be read as "at least one other embodiment". The terms "first", "second", and the like may refer to different or same objects. Other definitions, explicit and implicit, may be included below. A definition of a term is consistent throughout the description unless the context clearly indicates otherwise.

[0023] FIG. 1 illustrates a block diagram of an exemplary environment 10 where embodiments of the subject matter described herein can be implemented. It is to be understood that the structure and functionality of the environment 10 are described only for the purpose of illustration without suggesting any limitations as to the scope of the subject matter described herein. The subject matter described herein can be embodied with a different structure or functionality.

[0024] The environment 10 may include a terminal 100, a server 200 and a terminal 300. The server 200 may support one or more services such as an electronic business (e.g. online shopping), a forum of topics such as sport, movie, politics,

entertainment, or the like. The terminal 100/300 may be a client device such as a mobile phone, a Personal Digital Assistant (PDA), a laptop, a desk computer, a tablet, or the like, which is connected to the server 200 over a network such as Wide Area Network (WAN), Local Area Network (LAN), Wireless Local Area Network (WLAN), cellular network, or the like. The terminal 100/300 may include an input 110/310 and an output 120/320. The input 110/310 may receive text input such as a text sentence, and may also receive speech input. Likely the output 120/320 may provide text output, and may also provide speech output. The subject matter is not limited to the specific form of the input and output.

[0025] In a scenario, the server 200 may support an electronic business website. A translation module 210 in the server 200 may provide translation between a buyer and a seller who speak in different languages, where the buyer and the seller may discuss about the business using their terminals 100 and 300 via the aid of the translation provided by the server.

[0026] In another scenario, the server 200 may support a forum related to a certain topic. For example, the forum may be related to movies. Users from different countries may publish movie related posts using their terminals 100, 300 and so on via the aid of the translation provided by the server.

[0027] It is to be understood that the scenarios are described only for the purpose of illustration without suggesting any limitations as to the scope of the subject matter described herein. The subject matter described herein can be embodied in various scenarios.

[0028] FIG. 2 illustrates an existing neural network model for machine translation. This model may be referred to as sequence to sequence (S2S) model.

[0029] The S2S model tries to learn translation relation in a continuous vector space. As shown in FIG. 2, the S2S framework includes an encoder 210 and a decoder 220. To compress a variable-length source sentence into a fixed-size vector, the encoder 210 may read words one by one of a source sentence and generate a sequence of hidden vectors by using a recurrent neural network (RNN). The words are represented by x_1, x_2 ... x_T, and the hidden vectors h_1, h_2 ... h_T are shown as circles in

FIG. 2. By reading all the source words and generating the hidden vectors recurrently, the final hidden vector h_T should contain all information of the source sentence, and it is used and referred to as a context vector C. Based on the context vector C, another RNN-based neural network is used in the decoder 220 to generate a sequence of hidden vectors h_1, h_2 ... $h_{T'}$ and accordingly a target sentence. The words of the target sentence are represented by y_1, y_2 ... $y_{T'}$ in FIG. 2.

[0030] The context vector C plays a key role in the connection of source and target language spaces, and it should contain all the internal meaning extracted from the source sentence, based on which, the decoder 220 can generate the target sentence keeping the meaning unchanged. To extract the internal meaning which is represented by the context vector C and generate the target sentence, the S2S model usually needs a large number of parameters, and a huge bilingual corpus is required to train them.

[0031] In many cases, the internal meaning of a sentence is not easy to learn, especially when the language is informal. For a same intention, there may be various expressions with very different surface character strings, which aggravates the difficulty of the internal meaning extraction.

[0032] As shown in Table 1, there are three different expressions in Chinese for a same intention, that is, a customer wants a white 4G cellphone with a big screen. The first and second expressions (Source sentence 1 and Source sentence 2) are wordy and contain lots of verbiages. To obtain the internal meaning, the encoder 210 should ignore these verbiages and focus on key information. On the other hand, the encoder 210 should not omit key information while ignoring these verbiages. However it is hard to train the S2S model to accurately focus on the key information and not omit key information, even with a large amount of bilingual training data. For example, as shown in table 1, for the wordy source sentence 1, the S2S model does not generate the translation of "白色的"/"white", and fails to preserve the correct meaning of source sentence 1.

Source1	哎，那个有大屏幕的4G手机吗？要白色的。
Source2	给我推荐个4G手机吧，最好白的，屏幕要大。
Source3	我想买个白色的大屏幕的4G手机。
Intention	I want a white 4G cellphone with a big screen.
Enc-Dec	I need a 4G cellphone with a big screen.

TABLE 1

[0033] FIG. 3 illustrates a knowledge-based model 30 for machine translation using neural networks according to an embodiment of the subject matter.

[0034] In the knowledge-based model 30, a knowledge-based semantic space 330 is utilized to guide the extraction of key information of the source sentence. The semantic space 330 comprises a plurality of semantic tuples. The module 30 includes a key information extraction section 310 and a target generation section 320. The key information extraction section 310 may map a source sentence in a first language (e.g. Chinese) to a semantic space 330 predefined by a knowledge base by using a first neural network to extract key information 340 of the source sentence, wherein the extracted key information is predefined in the knowledge base. The target generation section 320 may generate a target sentence in a second language (e.g. English) by using a second neural network based on the extracted key information 340.

[0035] As shown in FIG. 3, the key information extraction section 310 and the target generation section 320 may be implemented by using neural networks. Source monolingual data and a knowledge base, e.g., the knowledge tree 40 shown in FIG. 4, may be used to train the section 310 to extract key information of a source sentence. For example, the key information may be semantic tuples 340 in the semantic space 330 predefined by the knowledge tree 40. In an embodiment, the extracted semantic tuples 340 may be represented by a vector, which indicates which nodes in the knowledge tree 40 are extracted as the key information of the source sentence. As the knowledge tree with its structure and its nodes representing respective semantic information is predefined, the vector representing the extracted semantic tuples 340 actually indicates explicit semantic information, and thus may be referred to as

explicit semantic vector. For example, if the knowledge base has N nodes, the vector may be a N-bit vector with a bit position in the vector corresponding to a node in the knowledge base, where a bit 1 indicates presence of a corresponding node and a bit 0 indicates non-presence of a corresponding node. It should be understood that the subject matter is not limited to the form of the explicit semantic vector, other methods for representing the extracted key information are also applicable.

[0036] The same knowledge base and target monolingual data may be used to train the target generation section 320 to generate a natural language target sentence based on key information (e.g. the semantic tuples 340) predefined in the knowledge base. For example, the target generation section 320 may be trained to produce a target sentence based on an explicit semantic vector.

[0037] Different from the S2S model 20 using large bilingual corpus for training, the knowledge-based model 30 allows separate training of the key information extraction section 310 (i.e., the source section) and the target generation section 320 (i.e., the target section), and the training on each side only needs monolingual data and corresponding knowledge base. Also the semantic vector obtained in the knowledge-based model 30 is no longer an implicit continuous number vector (e.g., the hidden vector C in FIG. 2), but an explicit semantic vector.

[0038] The semantic space 330 is defined by the knowledge base, and thus the key information can be extracted accurately from source sentence. In such a way, users can easily add external knowledge into the knowledge base to guide the model to generate correct translation results.

[0039] After the training of the key information extraction section 310 and the training of the target generation section 320, the model 30 is ready for translation. For example, as shown in FIG. 3, for a source sentence in Chinese "给我推荐个 4G 手机吧，最好白色，屏幕要大", explicit semantic tuples as shown in 340 are extracted as key information of the source sentence by mapping the source sentence to the knowledge-based semantic space 330, and then a target sentence in English "I want a white 4G cellphone with a big screen" is generated.

[0040] FIG. 4 illustrates a knowledge base organized in a tree structure

according to an embodiment of the subject matter. Given the source sentence as shown in FIG. 3, the semantic tuples 340 are obtained. Since the knowledge base is organized in a tree structure, the semantic tuples 340 may be seen as several paths in the tree. The solid lines show the paths corresponding to the semantic tuples 340, for example, a path Root ->category->cellphone represents the tuple Category.cellphone. It is worth noticing that the knowledge base is language-irrelevant.

[0041] The knowledge base may provide knowledge about semantic information in a specific context. For example, the knowledge base shown in FIG. 4 may provide knowledge about semantic information in electronic business. In a scenario of electronic business, e.g. as shown in FIG. 1, the sentences are typically related to specific contents such as those contained in the knowledge base shown in Fig.4. It should be understood that the knowledge base may be related to any domain such as physics, chemistry, medical, movie, sport, entertainment and so on. Taking advantage of the knowledge base related to a specific context, the data amount required for sufficiently training the source and target neural networks can be significantly reduced compared to the existing S2S model as shown in FIG. 2, and the translation accuracy can be improved.

[0042] FIG. 5 illustrates a source machine learning module 50 for extracting key information of a source sentence according to an embodiment of the subject matter. The machine learning module 50 may be implemented as or may comprise a neural network-based model for extracting key information of a source sentence according to an embodiment of the subject matter. The neural network-based model is shown in FIG. 5 and may be referred to as a source neural network.

[0043] The source machine learning module 50 may include a RNN 510, a knowledge tree 520 and a classifier 530. The classifier 530 may be neural network-based classifier, such as a neural network-based logistic regression (LR) classifier. The RNN 510 may read words represented by x_1, x_2 ... x_T of a source sentence and generate a sequence of hidden vectors h_1, h_2 ... h_T by using equation (1).

$$h_t = f(h_{t-1}, x_t) \quad (1)$$

where f is a non-linear function, which may be as simple as an element-wise

logistic sigmoid function and may be as complex as a long short-term memory (LSTM) unit.

[0044] FIG. 5A illustrates an exemplary process for obtaining a hidden vector h_t based on a previous hidden vector h_{t-1} and a current input x_t at time-stamp t. Given an input sentence $X = (x_1, x_2 \ldots x_T)$, as shown in FIG. 5A, at time-stamp t, an input word x_t is fed into the neural network. With the embedding layer r, the word x_t is mapped into a real vector $r_t = r(x_t)$. Then the real vector r_t is fed into a recurrent function g to get the hidden vector $h_t = g(r_t, h_{t-1})$ at time-stamp t. In an embodiment, a Gated Recurrent Unit (GRU) may be employed as the recurrent function g, in order to model the long dependency and memorize the information of words far from the end.

[0045] Referring back to FIG. 5, after a sequence of hidden vectors $h_1, h_2 \ldots h_T$ are generated by the RNN 510, the final hidden vector h_T may be used as the context vector of the source sentence, which is shown as H in FIG. 5. The process of obtaining the context vector H from a sentence may be referred to as sentence embedding, which may be used to compress the variable-length source sentence into a fixed-size context vector.

[0046] The knowledge base 520 such as the knowledge tree 40 may be used to extract the key information of the source sentence. Given a knowledge base for a specific domain, an intention of this domain may be divided into several classes, while each class has subclasses. All the classes can be organized as a tree structure. The knowledge base in the tree structure may be referred to as a tuple tree. The terms knowledge base, knowledge tree, and tuple tree may be interchangeably used in the description.

[0047] The classifier 530 may be a neural-network-based hierarchical classifier, which may be built as follows: each edge e of the tree has a weight vector W_e, which may be randomly initialized, and learned with training data; the tree may be gone through top-down to find available paths, and for each current node, the classifier 530 may decide which child node may be chosen. As shown in FIG. 4, a node in the tree is corresponding to the edge ending at the node, therefore we may also say that each node of the tree has a weight vector W_e. As shown in FIG. 5, a dot product is

performed on the weight vector W_e of a node and the context vector H of the source sentence, the result of the dot product is input into the classifier 530. The classifier 530 may decide whether the node may be chosen based on the dot product of its weight vector W_e and the context vector H.

Source	啊，那个有大屏幕的4G手机吗？要白色的。
Tuples	*Category.cellphone* *Appearance.color.white* *Appearance.size.big_screen* *Network.4G_network*

TABLE 2

[0048] In an embodiment, a logistic regression (LR) classifier may be used as the classifier 530. As shown in table 2, given the wordy source sentence "啊，那个有大屏幕的 4G 手机吗？要白色的。", the LR classifier 530 may go through the knowledge tree 520, and may choose three child nodes Category, Network and Appearance in the first layer, and in the second layer, may choose a child node Cellphone following its parent node Category, a child node 4G following its parent node network, two child node Size and Color following their parent node Appearance, and in the third layer, may choose a child node Big_screen following its parent node Size, a child node White following its parent node Color. Then the semantic tuples as shown in table 2 are extracted for the source sentence based on the context vector H of the source sentence and the weight vectors of the edges or nodes of the knowledge tree. In an embodiment, the probability to choose an edge e with its child node may be computed as shown in equation (2), and the classifier 530 may decide whether to choose the edge e or the corresponding node based on the probability.

$$p(1|e,H) = \frac{1}{1+e^{-w_e \cdot H}} \tag{2}$$

where $w_e \cdot H$ is the dot product of the weight vector W_e of an edge or node and the context vector H.

[0049] In an embodiment, while the classifier 530 goes through the knowledge tree 520, it may classify each node as bit 1 or 0, for example, the nodes corresponding to the extracted tuples shown in table 2 are classified as 1, and the other nodes are

classified as 0, and accordingly an explicit semantic vector which represents the presence of the explicit semantic tuples in the source sentence is generated. For example, if there are N edges or nodes in the knowledge base, the explicit semantic vector may be an N-bit vector with a bit position in the vector corresponding to a node in the knowledge base. The length N of the obtained semantic vector may be a large number, and usually the extracted tuples may only include a few nodes for a source sentence. That is to say, usually the N-bit vector may be a sparse vector, and it is not necessary to actually generate an N-bit vector, for example, only the position indexes of bit 1s in the semantic vector may be generated. In this case, the generated position indexes also represent the resulting semantic vector.

[0050] FIG. 6 illustrates a target machine learning module 60 for generating a target sentence based on the extracted key information according to an embodiment of the subject matter. The machine learning module may be implemented as or may comprise a neural network-based model for generating a target sentence based on the extracted key information according to an embodiment of the subject matter. The neural network-based model is shown in FIG. 6 and may be referred to as a target neural network.

[0051] In an embodiment, the target neural network may be a RNN. As shown in FIG. 6, based on the explicit semantic vector (shown as C) generated by the source neural network 50, a sequence of hidden vectors h_1, h_2 ... h_T and respectively words y_1, y_2 ... y_T of a target sentence may be generated by the RNN recurrently. For example, at time stamp t, the hidden vector h_t may be obtained by using equation (3), and the probability of the target word y_t may be calculated by using equation (4).

$$h_t = g(h_{t-1}, y_{t-1}, C) \tag{3}$$

$$p(y_t|y_{t-1}, y_{t-2} \cdots y_1, C) = \frac{e^{s(y_t, h_t)}}{\sum_{y'} e^{s(y', h_t)}} \tag{4}$$

[0052] The function g in equation (3) may be a recurrent function in the RNN. The equation (4) may be a soft-max function to produce the probability of the word y_t, and accordingly the word y_t is produced. It should be understood that the subject matter is not limited to a specific recurrent function or a specific soft-max function.

[0053] FIG. 6A illustrates a process of target sentence generation according to an embodiment of the subject matter. In an embodiment, a Gated Recurrent Unit (GRU) may be employed as the recurrent function g in equation (3). For example, a GRU may be implemented as equations (5) to (10).

$$\mathrm{d}_t = \sigma(W^d y_{t-1} + U^d h_{t-1} + V^d c_{t-1}) \tag{5}$$

$$c_t = \mathrm{d}_t \odot c_{t-1} \tag{6}$$

$$\mathrm{r}_t = \sigma(W^r y_{t-1} + U^r h_{t-1} + V^r c_t) \tag{7}$$

$$h'_t = tanh(W y_{t-1} + U(\mathrm{r}_t \odot h_{t-1}) + V c_t) \tag{8}$$

$$\mathrm{z}_t = \sigma(W^z y_{t-1} + U^z h_{t-1} + V^z c_t) \tag{9}$$

$$h_t = (1 - \mathrm{z}_t) \odot h'_t + z_t \odot h_{t-1} + \tanh(V^h c_t) \tag{10}$$

[0054] C_t is initialized as the semantic vector C and updated recurrently based on an extraction gate d_t. σ is a non-linear function. tanh is a tangent function. $\odot$ denotes a bit-wise production. W, U, V, W^d, U^d, V^d, W^r, U^r, V^r, W^z, U^z, V^z are learned parameters by training.

[0055] In the GRU, for each recurrent state t, an extraction gate may be used to retrieve and remove information from the semantic vector C to generate the corresponding target word. For example, as shown in FIG. 6A, for the semantic tuples in table 2, the target RNN may generate the target sentence word by word, until meets the end symbol character: "I want a white 4G cellphone with a big screen.".

[0056] To force the target neural network to generate the target sentence keeping information contained in the semantic vector C unchanged, two additional terms may be introduced into the cost function for training. An example of the cost function is shown as equation (11).

$$\sum_t \log(p(y_t|C)) + \|\sigma r\|_2 + \frac{1}{T}\sum_{j=1}^{T} \|d_t - d_{t-1}\|_2 \tag{11}$$

[0057] The first term in the cost function is log-likelihood cost. The other two terms are introduced penalty terms. $\|\sigma r\|_2$ is for forcing the decoding neural network to extract as much information as possible from the semantic vector C, thus the generated target sentence keeps the same meaning with the source sentence. The third

term is to restrict the extract gate from extracting too much information in the semantic vector C at each time-stamp.

[0058] FIG. 6B illustrates how the extracted semantic tuples control the generation of the target sentence according to an embodiment of the subject matter. Taking the semantic tuple Appearance.color.white as an example, the GRU keeps its feature value almost unchanged until the target word "white" is generated. Almost all the feature values corresponding to the extracted semantic tuples drop from 1 to 0, when the corresponding words are generated, except for the tuple Appearance.size.big_creen. To express the meaning of this tuple, the target neural network may generate two words, "big" and "screen". When the sentence finished, all the feature values may be 0.

[0059] FIG. 7 illustrates a source machine learning module 70 for extracting key information of a source sentence according to an embodiment of the subject matter. The machine learning module 70 may be implemented as or may comprise a neural network-based model for extracting key information of a source sentence according to an embodiment of the subject matter. The neural network-based model is shown in FIG. 7 and may be referred to as a source neural network 70.

[0060] The source machine learning module 70 may include a RNN 710, a knowledge tree 720 and a classifier 730. The RNN 710 may read words represented by x_1, x_2 ... x_T of a source sentence and generate a sequence of hidden vectors h_1, h_2 ... h_T. A weighted sum may be performed to the hidden vectors h_1, h_2 ... h_T with respective weights w_1, w_2, ..., w_T, and the summed hidden vector, shown as *H* in FIG. 7, is used by the classifier to classify the nodes of the knowledge tree. The weights w_1, w_2, ..., w_T may be determined based on the hidden vectors h_1, h_2 ... h_T and the weight vectors of the knowledge tree, and may be referred to as attention weights. By using the attention mechanism, it is helpful to align a target word to a source word, and it would especially benefit the translation of a relative long sentence.

[0061] FIG. 7A illustrates a process for calculating the attention weights for the hidden vectors according to an embodiment of the subject matter. Taking the node "Category" as an example, the hidden vectors h_1, h_2 ... h_T and the weight vector W_e of

the node are input into an attention weight calculating module 740. The attention weight calculating module 740 may generate respective attention weights w_1, w_2, ..., w_T based on the hidden vectors h_1, h_2 ... h_T and the weight vector W_e of the node. In an embodiment, a dot production may be performed to each of the hidden vectors h_1, h_2 ... h_T and the weight vector W_e, and the T dot production results may be input into a softmax function to generate respective probabilities for the hidden vectors h_1, h_2 ... h_T, which then may be normalized as the attention weights w_1, w_2, ..., w_T of the hidden vectors h_1, h_2 ... h_T associated to the node "Category".

[0062] Referring back to FIG. 7, a summed hidden vector H associated to the node "Category" may be obtained by performing a weighted sum to the hidden vectors h_1, h_2 ... h_T with the attention weights w_1, w_2, ..., w_T associated to the node "Category". In an embodiment, the summed hidden vector H associated to the node "Category" may be used by the classifier 730 to classify child nodes of the node "Category", e.g. the child nodes "Computer" and "Cellphone" in the tree. Taking the child node "Computer" as an example, the classifier 730 may classify child node "Computer" as 1 or 0 based on the summed hidden vector H associated to the parent node "Category" and the weight vector of the child node "Computer".

[0063] For each parent node in the tree, the above mentioned process may be performed so as to go through the nodes in the tree. Particularly, for each parent node in the knowledge base, respective attention weights for the plurality of hidden vectors may be generated based on the plurality of hidden vectors and the weight vector of the parent node, a summed hidden vector may be generated by performing weighted sum to the plurality of hidden vectors with the respective attention weights, and the classifier may decide whether to choose a child node of the parent node based on the summed hidden vector and the weight vector of the child node.

[0064] FIG. 8 illustrates a process for generating a target sentence based on the extracted key information according to an embodiment of the subject matter. The extracted key information of a source sentence may be the explicit semantic vector or the semantic tuples as discussed above.

[0065] In an embodiment, the semantic tuples in table 2 have been extracted by

the source neural network 70, the nodes corresponding to the semantic tuples are shown as underlined in the tree as shown in FIG. 8. For a time stamp t, the semantic vector C_t may be generated by performing weighted sum to the weight vectors W_{e1}, W_{e2} ...W_{eN} of the nodes with respective attention weights w_1, w_2, ..., w_N. The weights w_1, w_2, ..., w_N may be determined based on the previous hidden vector h_{t-1} and the weight vectors W_{e1}, W_{e2} ...W_{eN} of the nodes. By using the attention mechanism, it is helpful to align a target word to a source word, and it would especially benefit the translation of a relative long sentence.

[0066] As shown in FIG. 8, the weight vectors W_{e1}, W_{e2} ...W_{eN} of the nodes and the previous hidden vector h_{t-1} may be input into an attention weight calculating module 810. The attention weight calculating module 810 may generate respective attention weights w_1, w_2, ..., w_N based on the weight vectors W_{e1}, W_{e2} ...W_{eN} of the nodes and the previous hidden vector h_{t-1}. In an embodiment, a dot production may be performed to each of the weight vectors W_{e1}, W_{e2} ...W_{eN} of the nodes and the previous hidden vector h_{t-1}, and the N dot production results may be input into a soft-max function to generate respective probabilities for the weight vectors W_{e1}, W_{e2} ...W_{eN} of the nodes, which then may be normalized as the attention weights w_1, w_2, ..., w_N of the weight vectors W_{e1}, W_{e2} ...W_{eN} of the nodes.

[0067] For each recurrent state (e.g. the time stamp t) of the target neural network, respective attention weights for the nodes corresponding to the extracted tuples may be calculated based on the previous hidden vector of the target neural network and the weight vectors of the nodes, wherein the weight vectors of the nodes in the tree are trained in the target neural network, the semantic vector C_t may be obtained by performing weighted sum to weight vectors of the nodes with the respective attention weights, a hidden vector h_t may be generated by using the target neural network based on the semantic vector C_t, and a word y_t of the target sentence may be generated based on the hidden vector h_t.

[0068] It should be noted that although the various models for machine translation are described as being implemented based on RNNs, it should be appreciated that the subject matter is not limited thereto, for example, the models for

machine translation may be implemented based on convolutional neural networks (CNNs). It should be appreciated that the subject matter is not limited to deep neural networks, for example, other kinds of machine learning techniques may be applicable for implementing the subject matter in machine translation.

[0069] FIG. 9 illustrates a process for translating a source sentence to a target sentence according to an embodiment of the subject matter. The method may be started from receiving a source sentence. In order to extract key information of the source sentence, at step 910, the source sentence is mapped to a semantic space predefined by a knowledge base by using a first machine learning module. It should be appreciated that, while the source sentence is mapped to the semantic space, the key information of the source sentence is extracted accordingly. At step 920, a target sentence may be generated by using a second machine learning module based on the extracted key information.

[0070] In an embodiment, the semantic space comprises a plurality of semantic tuples, and the source sentence may be mapped to one or more tuples in the semantic space. In an embodiment, the knowledge base may be organized in a tree structure, a path consisting of at least one node of the tree structure defines a semantic tuple in the semantic space.

[0071] In an embodiment, a plurality of hidden vectors may be obtained based on the source sentence by using a recurrent neural network (RNN) in the first machine learning module, and the one or more semantic tuples may be extracted by using a neural network-based classifier in the first machine learning module based on at least one hidden vector of the plurality of hidden vectors and the knowledge base.

[0072] In an embodiment, the target sentence may be generated by using the second machine learning module based on a semantic vector corresponding to the one or more semantic tuples.

[0073] In an embodiment, one or more nodes representing the one or more semantic tuples may be chosen from the knowledge base by using the neural network-based classifier based on the final hidden vector of the plurality of hidden vectors and respective weight vectors of nodes in the knowledge base.

[0074] In an embodiment, the semantic vector indicates presence of the chosen one or more nodes and non-presence of the other nodes, among the nodes in the knowledge base.

[0075] In an embodiment, for each parent node in the knowledge base, respective weights may be generated for the plurality of hidden vectors based on the plurality of hidden vectors and a weight vector of the parent node, a summed hidden vector may be obtained by performing weighted sum to the plurality of hidden vectors with the respective weights; and a child node of the parent node may be classified as being chosen or not by using the neural network-based classifier based on the weighted hidden vector and the weight vector of the child node.

[0076] In an embodiment, for each recurrent state of a RNN in the second machine learning module, respective weights may be generated for the chosen one or more nodes based on a previous hidden vector of the RNN and one or more weight vectors of the chosen one or more nodes, the semantic vector may be obtained by performing weighted sum to one or more weight vectors of the chosen one or more nodes with the respective weights, a hidden vector may be obtained by using the RNN based on the semantic vector, and a word of the target sentence may be produced based on the hidden vector.

[0077] In an embodiment, the first machine learning module may be trained by using a first data corpus in the first language and the knowledge base, for example, may be trained by using data pairs of the first language data and explicit semantic vectors which represents explicit semantic tuples predefined in the knowledge base, the second machine learning module may be trained by using a second data corpus in the second language and the knowledge base, for example, may be trained by using data pairs of explicit semantic vectors which represents the explicit semantic tuples predefined in the knowledge base and the second language data.

[0078] FIG. 10 illustrates a block diagram of an apparatus 100 for machine translation according to an embodiment of the subject matter. The apparatus 100 may include a source machine learning module 1010 and a target machine learning module 1020. The apparatus 100 may be an embodiment of the translation module 210 of FIG.

1.

[0079] The source machine learning module 1010 may be configured to map a source sentence in a first language to a semantic space predefined by a knowledge base, to extract key information of the source sentence, wherein the extracted key information may be predefined in the knowledge base. The target machine learning module 1020 may be configured to generate a target sentence in a second language based on the extracted key information.

[0080] It should be appreciated that the source machine learning module 1010 and the target machine learning module 1020 may perform the respective operations or functions as described above with reference to FIGs. 1 to 9 in various embodiments of the subject matter.

[0081] The source machine learning module 1010 and the target machine learning module 1020 may be implemented in various forms of hardware, software or combinations thereof. In an embodiment, the modules may be implemented separately or as a whole by one or more hardware logic components. For example, and without limitation, illustrative types of hardware logic components that can be used include Field-programmable Gate Arrays (FPGAs), Application-specific Integrated Circuits (ASICs), Application-specific Standard Products (ASSPs), System-on-a-chip systems (SOCs), Complex Programmable Logic Devices (CPLDs), etc. In another embodiment, the modules may be implemented by one or more software modules, which may be executed by a general central processing unit (CPU), a graphic processing unit (GPU), a Digital Signal Processor (DSP), etc.

[0082] FIG. 11 illustrates a block diagram of a computer system 200 for machine translation according to an embodiment of the subject matter. According to one embodiment, the computer system 200 may include one or more processors 2010 that execute one or more computer readable instructions (i.e. the elements described above which are implemented in the form of the software) stored or encoded in computer readable storage medium (i.e. memory) 2020. The computer system 200 may include an output device 2030 such as display and an input device 440 such as a keyboard, mouse, touch screen, etc. The computer system 200 may include a

communication interface 2050 for communicating with other devices such as the terminals 100 and 300 as shown in FIG. 1.

[0083] In an embodiment, the computer-executable instructions stored in the memory 2020, when executed, may cause the one or more processors 2010 to: map a source sentence in a first language to a semantic space predefined by a knowledge base by using a first machine learning module, to extract key information of the source sentence, wherein the extracted key information is predefined in the knowledge base, and generate a target sentence in a second language by using a second machine learning module based on the extracted key information.

[0084] It should be appreciated that the computer-executable instructions stored in the memory 2020, when executed, may cause the one or more processors 2010 to perform the respective operations or functions as described above with reference to FIGs. 1 to 9 in various embodiments of the subject matter.

[0085] According to an embodiment, a program product such as a machine-readable medium is provided. The machine-readable medium may have instructions (i.e. the elements described above which are implemented in the form of the software) thereon which, when executed by a machine, cause the machine to perform the operations or functions as described above with reference to FIGs. 1 to 9 in various embodiments of the subject matter.

[0086] It should be noted that the above-mentioned solutions illustrate rather than limit the subject matter and that those skilled in the art would be able to design alternative solutions without departing from the scope of the appended claims. In the claims, any reference signs placed between parentheses shall not be construed as limiting the claim. The word "comprising" does not exclude the presence of elements or steps not listed in a claim or in the description. The word "a" or "an" preceding an element does not exclude the presence of a plurality of such elements. In the system claims enumerating several units, several of these units can be embodied by one and the same item of software and/or hardware. The usage of the words first, second and third, et cetera, does not indicate any ordering. These words are to be interpreted as names.

CLAIMS

1. A machine translation method, comprising:

mapping a source sentence in a first language to a semantic space predefined by a knowledge base by using a first machine learning module, to extract key information of the source sentence; and

generating a target sentence in a second language by using a second machine learning module based on the extracted key information.

2. The method of claim 1, wherein the semantic space comprises a plurality of semantic tuples, and wherein the mapping the source sentence in the first language to the semantic space comprises mapping the source sentence to one or more tuples in the semantic space.

3. The method of claim 2, wherein the knowledge base is organized in a tree structure, a path consisting of at least one node of the tree structure defines a semantic tuple in the semantic space.

4. The method of claim 3, wherein the mapping the source sentence in the first language to the semantic space comprises:

generating a plurality of hidden vectors based on the source sentence by using a first recurrent neural network (RNN) in the first machine learning module; and

extracting the one or more semantic tuples by using a neural network-based classifier in the first machine learning module based on at least one hidden vector of the plurality of hidden vectors and the knowledge base.

5. The method of claim 2 or 4, wherein the generating the target sentence comprises:

generating the target sentence by using the second machine learning module

based on a semantic vector corresponding to the one or more semantic tuples.

6. The method of claim 5, wherein the extracting the one or more semantic tuples comprises: choosing one or more nodes representing the one or more semantic tuples from the knowledge base by using the neural network-based classifier based on the final hidden vector of the plurality of hidden vectors and respective weight vectors of nodes in the knowledge base.

7. The method of claim 6, wherein the semantic vector indicates presence of the chosen one or more nodes and non-presence of the other nodes, among the nodes in the knowledge base.

8. The method of claim 5, wherein the mapping the source sentence in the first language to the semantic space comprises:

for each parent node in the knowledge base,

generating respective weights for the plurality of hidden vectors based on the plurality of hidden vectors and a weight vector of the parent node;

generating a summed hidden vector by performing weighted sum to the plurality of hidden vectors with the respective weights; and

deciding whether to choose a child node of the parent node by using the neural network-based classifier based on the summed hidden vector and the weight vector of the child node.

9. The method of claim 8, wherein the generating the target sentence comprises:

for each recurrent state of a second RNN in the second machine learning module,

generating respective weights for the chosen one or more nodes based on a previous hidden vector of the second RNN and one or more weight vectors of the chosen one or more nodes;

generating the semantic vector by performing weighted sum to one or more weight vectors of the chosen one or more nodes with the respective weights;

generating a hidden vector by using the second RNN based on the semantic vector; and

generating a word of the target sentence based on the hidden vector.

10. The method of one of claim 1, wherein the first machine learning module is trained by using a first data corpus in the first language and the knowledge base, the second machine learning module is trained by using a second data corpus in the second language and the knowledge base.

11. A machine translation apparatus, comprising:

a first machine learning module configured to map a source sentence in a first language to a semantic space predefined by a knowledge base, to extract key information of the source sentence; and

a second machine learning module configured to generate a target sentence in a second language based on the extracted key information.

12. The apparatus of claim 11, wherein the semantic space comprises a plurality of semantic tuples, and wherein the first machine learning module is configured to map the source sentence to one or more tuples in the semantic space.

13. The apparatus of claim 12, wherein the knowledge base is organized in a tree structure, a path consisting of at least one node of the tree structure defines a semantic tuple in the semantic space.

14. The apparatus of claim 13, wherein the first machine learning module is further configured to:

generate a plurality of hidden vectors based on the source sentence by using a first recurrent neural network (RNN); and

extract the one or more semantic tuples by using a neural network-based classifier based on at least one hidden vector of the plurality of hidden vectors and the

knowledge base.

15. The apparatus of claim 12 or 14, wherein the second machine learning module is further configured to:

generate the target sentence based on a semantic vector corresponding to the one or more semantic tuples.

16. The apparatus of claim 15, wherein the first machine learning module is further configured to: choose one or more nodes representing the one or more semantic tuples from the knowledge base by using the neural network-based classifier based on the final hidden vector of the plurality of hidden vectors and respective weight vectors of nodes in the knowledge base.

17. The apparatus of claim 16, wherein the semantic vector indicates presence of the chosen one or more nodes and non-presence of the other nodes, among the nodes in the knowledge base.

18. The apparatus of claim 15, wherein the first machine learning module is further configured to:

for each parent node in the knowledge base,

generate respective weights for the plurality of hidden vectors based on the plurality of hidden vectors and a weight vector of the parent node;

generate a summed hidden vector by performing weighted sum to the plurality of hidden vectors with the respective weights; and

deciding whether to choose a child node of the parent node by using the neural network-based classifier based on the summed hidden vector and the weight vector of the child node.

19. The apparatus of claim 18, wherein the second machine learning module is further configured to:

for each recurrent state of a second RNN,

generating respective weights for the chosen one or more nodes based on a previous hidden vector of the second RNN and one or more weight vectors of the chosen one or more nodes;

generating the semantic vector by performing weighted sum to one or more weight vectors of the chosen one or more nodes with the respective weights;

generating a hidden vector by using the second RNN based on the semantic vector; and

generating a word of the target sentence based on the hidden vector.

20. A computer system, comprising:

one or more processors; and

a memory storing computer-executable instructions that, when executed, cause the one or more processors to:

map a source sentence in a first language to a semantic space predefined by a knowledge base by using a first machine learning module, to extract key information of the source sentence; and

generate a target sentence in a second language by using a second machine learning module based on the extracted key information.

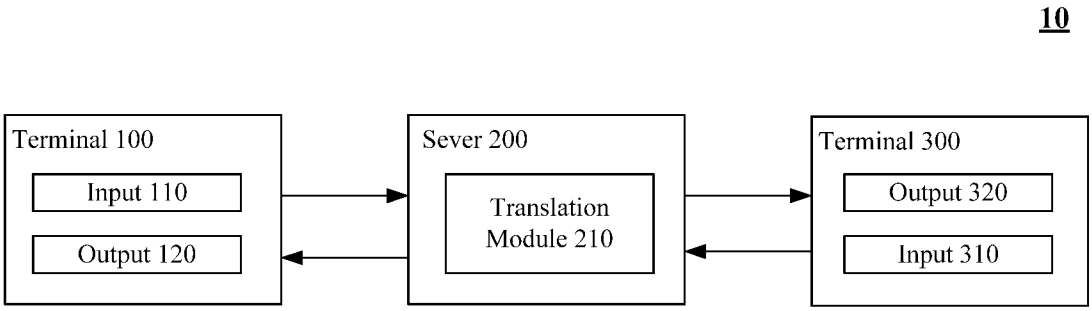
10
Terminal 100
Input 110
Output 120
Sever 200
Translation Module 210
Terminal 300
Output 320
Input 310

FIG.1

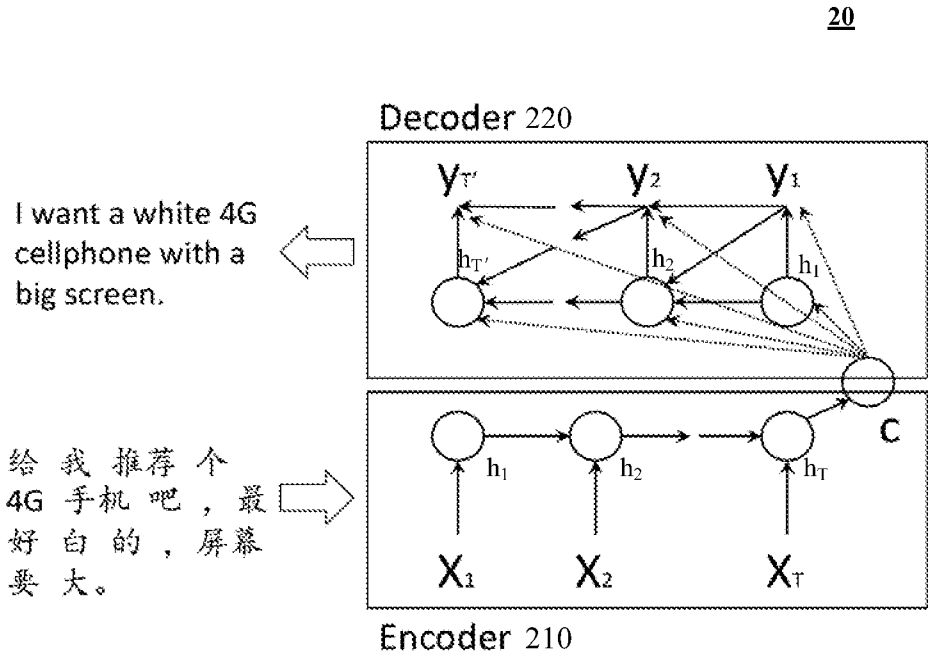
20
Decoder 220
I want a white 4G cellphone with a big screen.
给 我 推荐 个 4G 手机 吧 ， 最 好 白 的 ， 屏幕 要 大。
C
Encoder 210

FIG. 2

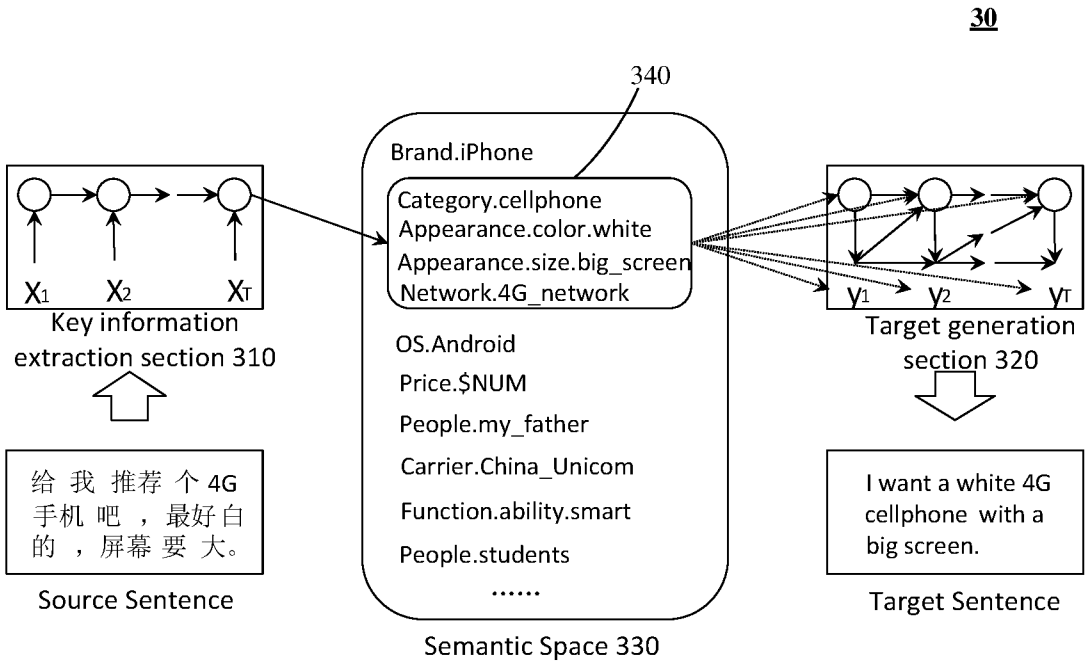
30
340
Brand.iPhone
Category.cellphone
Appearance.color.white
Appearance.size.big_screen
Network.4G_network
OS.Android
Price.$NUM
People.my_father
Carrier.China_Unicom
Function.ability.smart
People.students
.....
X1 X2 XT
Key information extraction section 310
给 我 推荐 个 4G 手机 吧 ， 最好 白 的 ， 屏幕 要 大。
Source Sentence
Semantic Space 330
y1 y2 yT
Target generation section 320
I want a white 4G cellphone with a big screen.
Target Sentence

FIG. 3

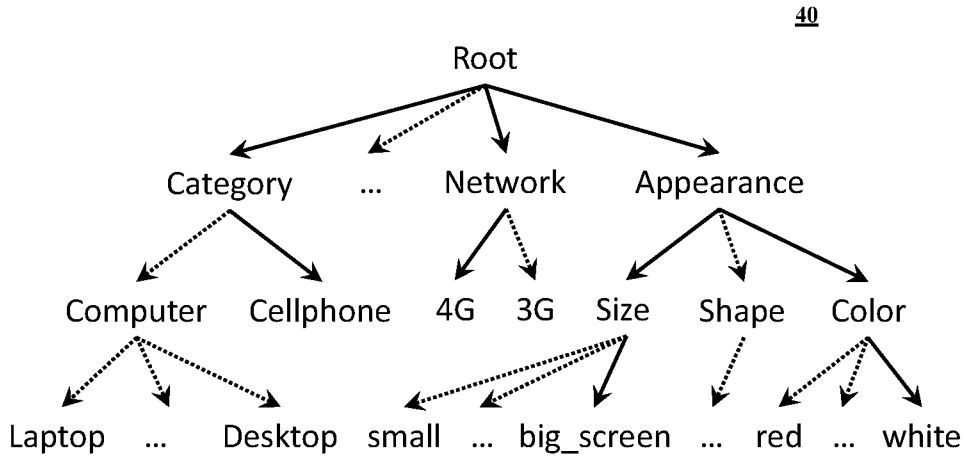
40
Root
Category
...
Network
Appearance
Computer
Cellphone
4G
3G
Size
Shape
Color
Laptop
...
Desktop
small
...
big_screen
...
red
...
white

FIG. 4

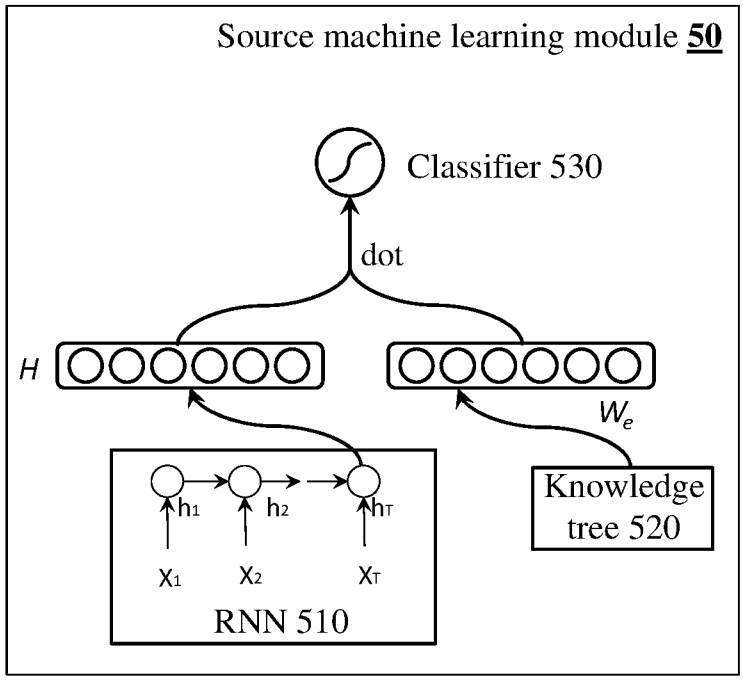
Source machine learning module 50
Classifier 530
dot
H
W_e
Knowledge tree 520
h_1
h_2
h_T
x_1
x_2
x_T
RNN 510

FIG. 5

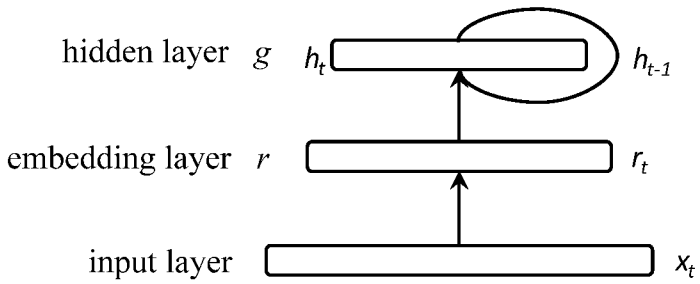
hidden layer g
h_t
h_{t-1}
embedding layer r
r_t
input layer
x_t

FIG. 5A

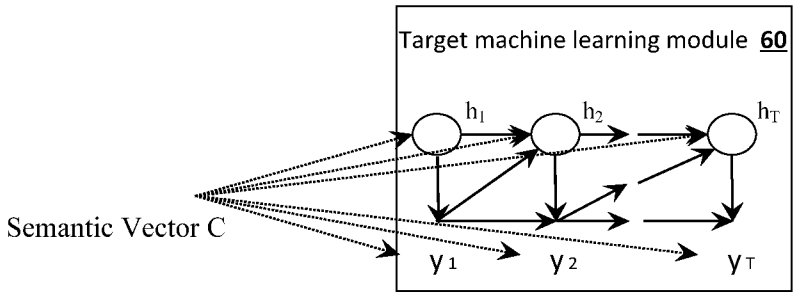

FIG. 6

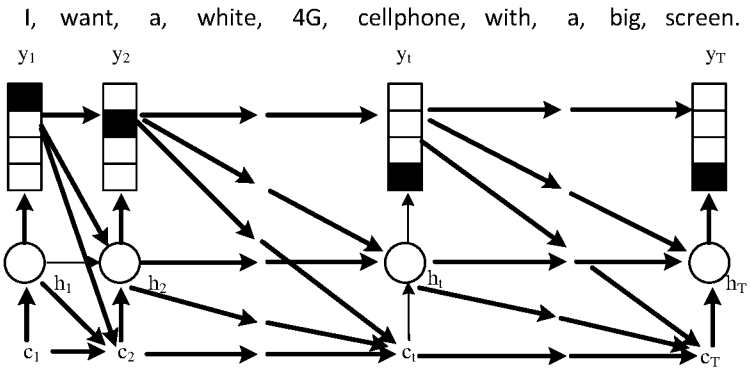

FIG. 6A

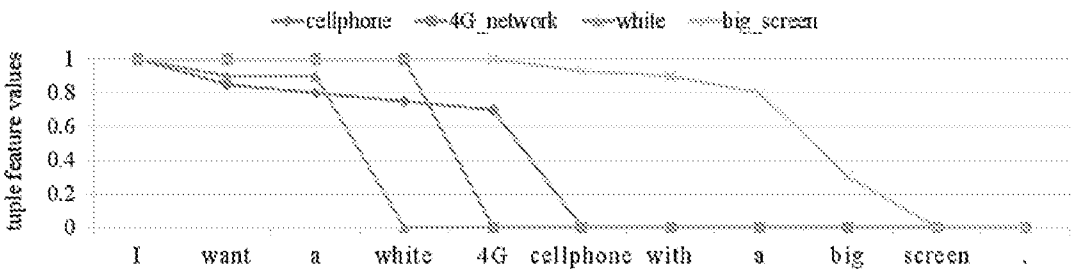

FIG. 6B

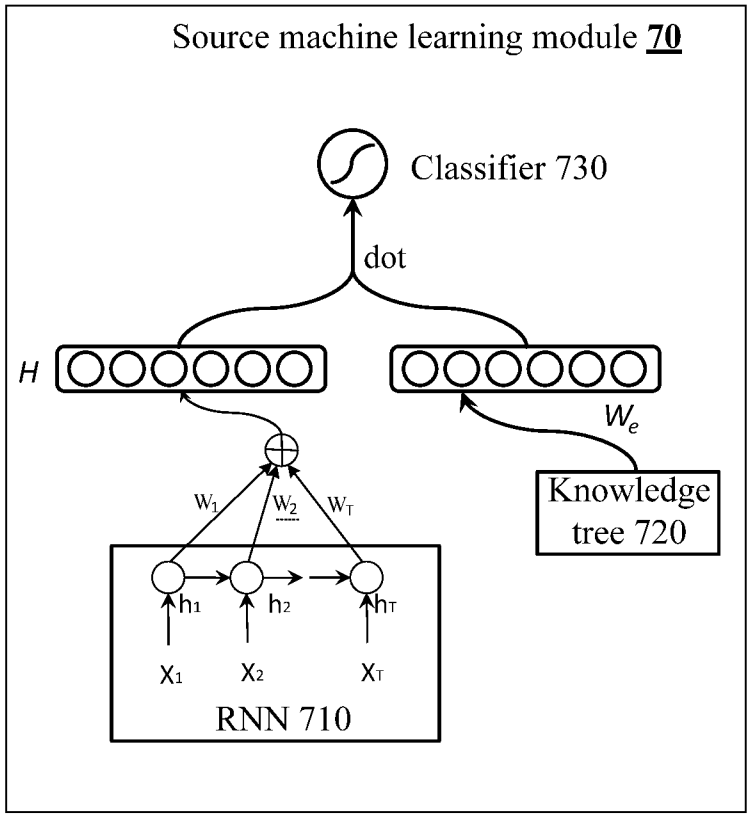
Source machine learning module 70
Classifier 730
dot
H
W_e
Knowledge tree 720
W_1
W_2
W_T
h_1
h_2
h_T
X_1
X_2
X_T
RNN 710

FIG. 7

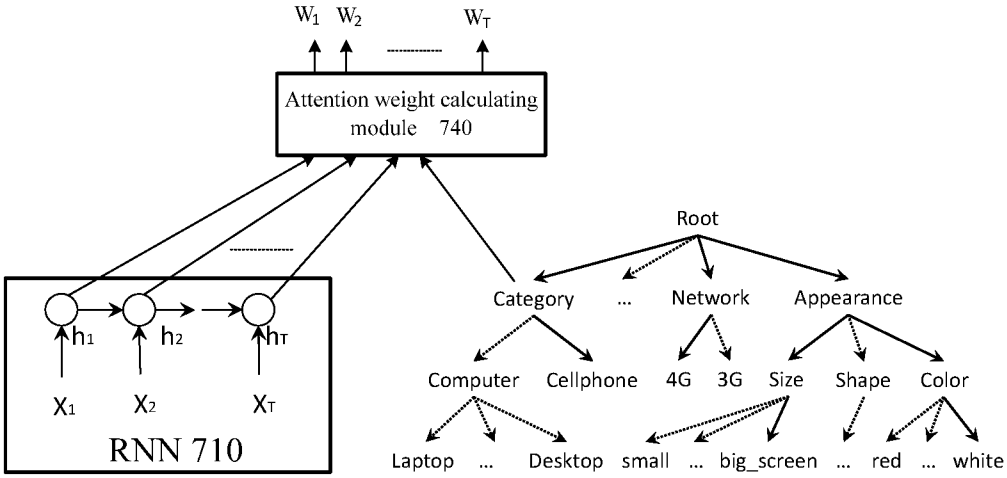
W_1 W_2 W_T
Attention weight calculating module 740
Root
Category
...
Network
Appearance
Computer
Cellphone
4G
3G
Size
Shape
Color
Laptop
...
Desktop
small
...
big_screen
...
red
...
white
h_1
h_2
h_T
X_1
X_2
X_T
RNN 710

FIG. 7A

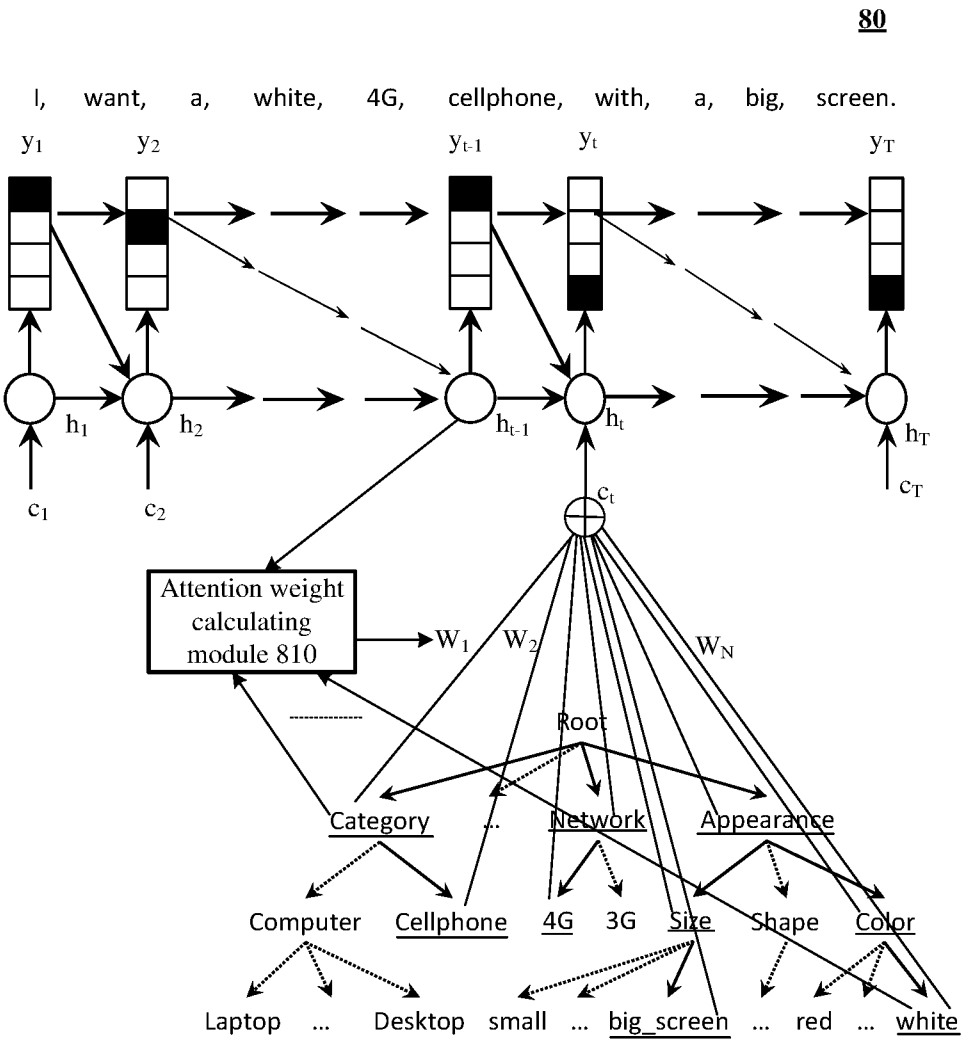
80
I, want, a, white, 4G, cellphone, with, a, big, screen.
y1
y2
yt-1
yt
yT
h1
h2
ht-1
ht
hT
c1
c2
ct
cT
Attention weight calculating module 810
W1
W2
WN
Root
Category
...
Network
Appearance
Computer
Cellphone
4G
3G
Size
Shape
Color
Laptop
...
Desktop
small
...
big_screen
...
red
...
white

FIG. 8

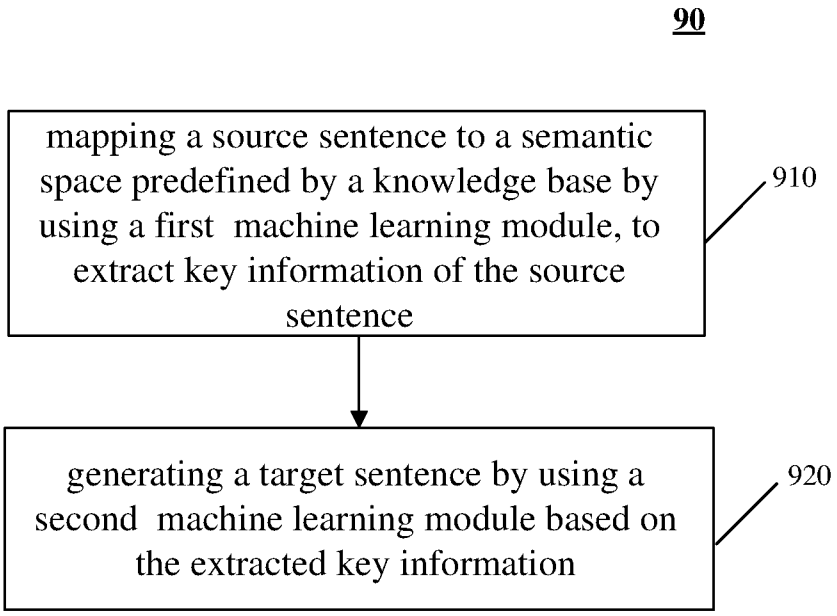
90
mapping a source sentence to a semantic space predefined by a knowledge base by using a first machine learning module, to extract key information of the source sentence
910
generating a target sentence by using a second machine learning module based on the extracted key information
920

FIG. 9

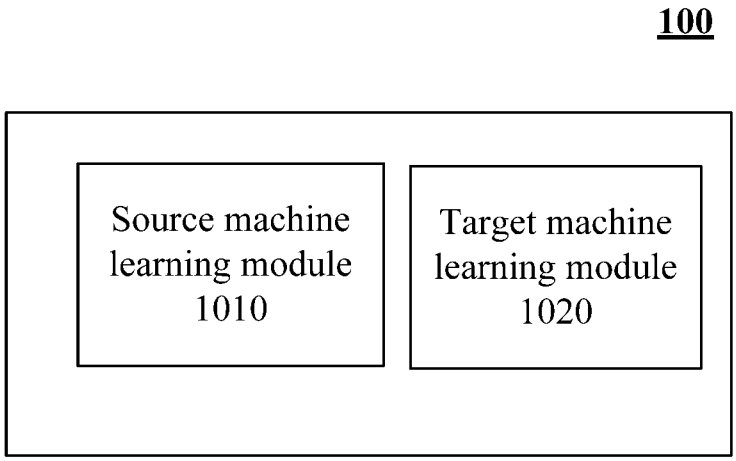
100
Source machine learning module 1010
Target machine learning module 1020

FIG. 10

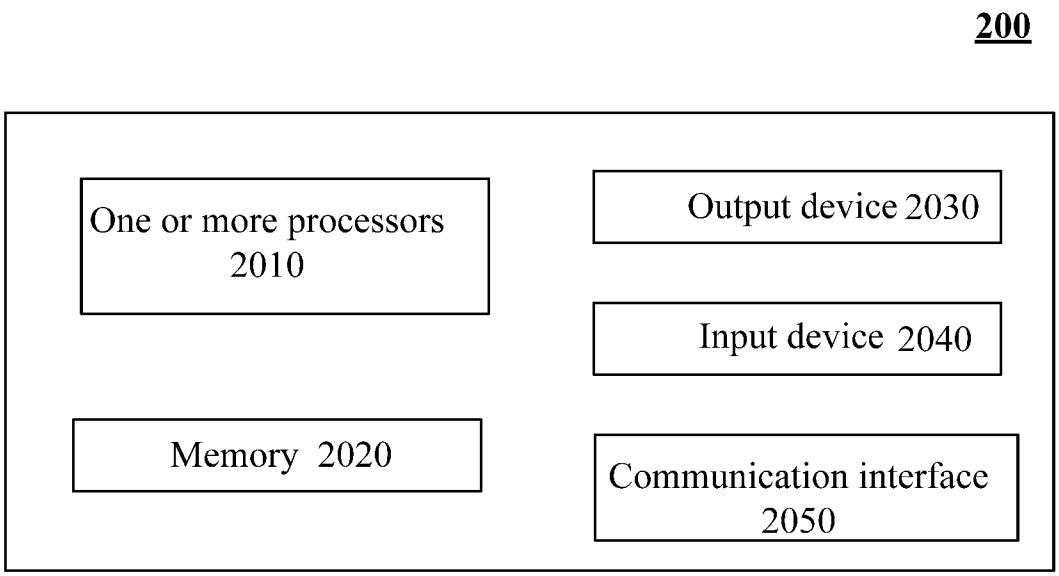
200
One or more processors 2010
Output device 2030
Input device 2040
Memory 2020
Communication interface 2050

FIG. 11

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2016/092762

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/28(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT, CNKI, EPODOC, WPI, IEEE: train+, translat+, semantic, intellig+, learn, knowledge, machin+, neural, network, language, recurrent, tuple, vector, weight, tree

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2008091405 A1 (ANISIMOVICH, KONSTANTIN ET AL.) 17 April 2008 (2008-04-17) description, paragraphs [0010], [0048] to [0050], [0071] to [0078], [0084], and figures 1 to 7	1-20
A	US 2015178271 A1 (ABBYY INFOPOISK LLC) 25 June 2015 (2015-06-25) the whole document	1-20
A	US 2013246322 A1 (CEPT SYSTEMS GMBH) 19 September 2013 (2013-09-19) the whole document	1-20
A	US 2014067728 A1 (ORACLE INTERNATIONAL CORPORATION) 06 March 2014 (2014-03-06) the whole document	1-20

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
10 April 2017	**08 May 2017**

Name and mailing address of the ISA/CN	Authorized officer
STATE INTELLECTUAL PROPERTY OFFICE OF THE P.R.CHINA **6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088** **China**	**XU,Shuxian**
Facsimile No. **(86-10)62019451**	Telephone No. **(86-10)53318920**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2016/092762

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2008091405	A1	17 April 2008	US	2008086299	A1	10 April 2008
				US	2008086300	A1	10 April 2008
				WO	2008045815	A2	17 April 2008
				US	2008086298	A1	10 April 2008
				US	2009070099	A1	12 March 2009
				US	2009182549	A1	16 July 2009
				US	2011257963	A1	20 October 2011
				US	2011270607	A1	03 November 2011
				US	2012010872	A1	12 January 2012
				US	2012109640	A1	03 May 2012
				US	2012173224	A1	05 July 2012
				US	2012232883	A1	13 September 2012
				US	2012239378	A1	20 September 2012
				US	2012259621	A1	11 October 2012
				US	2012271627	A1	25 October 2012
				US	2013024186	A1	24 January 2013
				US	2013024180	A1	24 January 2013
				US	2013041652	A1	14 February 2013
				US	2013054612	A1	28 February 2013
				US	2013191109	A1	25 July 2013
				US	2013211816	A1	15 August 2013
				US	2014114649	A1	24 April 2014
				US	2014129212	A1	08 May 2014
				US	8892423	B1	18 November 2014
				US	2015057992	A1	26 February 2015
				US	2015057991	A1	26 February 2015
				US	2015220515	A1	06 August 2015
				US	2016004766	A1	07 January 2016
US	2015178271	A1	25 June 2015	RU	2013156492	A	27 June 2015
US	2013246322	A1	19 September 2013	HK	1199319	A1	26 June 2015
				JP	2015515674	A	28 May 2015
				AU	2013231564	A1	02 October 2014
				CA	2864946	A1	19 September 2013
				WO	2013135474	A1	19 September 2013
				EP	2639749	A1	18 September 2013
				KR	20140138648	A	04 December 2014
				CN	104169948	A	26 November 2014
US	2014067728	A1	06 March 2014	JP	2015531499	A	02 November 2015
				WO	2014035539	A1	06 March 2014
				CN	104718542	A	17 June 2015
				EP	2891075	A1	08 July 2015