Natural language understanding approaches based on joint task of intent detection and slot filling for IoT voice interaction

Pin Ni[1] · Yuming Li[1] · Gangmin Li[2] · Victor Chang[3]

Received: 8 January 2020 / Accepted: 19 February 2020 / Published online: 13 March 2020

Abstract
Internet of Things (IoT) based voice interaction system, as a new artificial intelligence application, provides a new human–computer interaction mode. The more intelligent and efficient communication approach poses greater challenges to the semantic understanding module in the system. Facing with the complex and diverse interactive scenarios in practical applications, the academia and the industry urgently need more powerful Natural Language Understanding (NLU) methods as support. Intent Detection and Slot Filling joint task, as one of the core sub-tasks in NLU, has been widely used in different human–computer interaction scenarios. In the current era of deep learning, the joint task of Intent Detection and Slot Filling has also changed from previous rule-based methods to deep learning-based methods. It is an important problem to explore how to realize the models of these tasks to be refined and targeted designed, and to make the Intent Detection task better serve the improvement of precision of Slot Filling task by connecting the before and after tasks. It has great significance for building a more humanized IoT voice interaction system. In this study, we designed two joint models to realize Intent Detection and Slot Filling joint task. For the Intent Detection type task, one is based on BiGRU-Att-CapsuleNet (hybrid-based model) and the other is based on the RCNN model. Both methods use the BiGRU-CRF model for the Slot Filling type task. The hybrid-based model can enhance the semantic capture capability of a single model. And by combining specialized models built independently for each task to achieve a complete joint task, it can be better to achieve optimal performance on each task. This study also carried out detailed comparative experiments of tasks and joint tasks on multiple datasets. Experiments show that the joint models have achieved competitive results in 7 typical datasets included in multiple scenarios in English and Chinese compared with other models.

Keywords Internet of Things · Artificial intelligence · Natural language understanding · Voice interaction · Intent detection and slot filling · Capsule network

1 Introduction

The way of voice interaction has become a new idea for the communication method of IoT devices. This kind of interaction will greatly reduce the current heavy reliance on Graphical User Interface (GUI) or control methods based on physical buttons. At the same time, it also provides an important direction for the Internet of Things (IoT) to create interconnections: interaction between human and the Internet of Things through a more natural way [21], which includes the type of centralized interface to realize the interaction between multiple objects in the Internet of Things, etc. These interaction methods allow users to communicate directly to individual or multiple connected objects in the Internet of Things in a natural

✉ Victor Chang
V.Chang@tees.ac.uk

Pin Ni
P.Ni2@liverpool.ac.uk

Yuming Li
Y.Li278@liverpool.ac.uk

Gangmin Li
Gangmin.Li@xjtlu.edu.cn

[1] Department of Computer Science, University of Liverpool, Liverpool, UK

[2] Department of Computer Science and Software Engineering, Xi'an Jiaotong-Liverpool University, Suzhou, China

[3] School of Computing, Engineering and Digital Technologies, Teesside University, Middlesbrough, UK

way, while the current IoT interaction methods are limited to programmatic information transfer between devices [31]. Human can give instructions to smart devices around them in a simpler and more natural voice form, and the devices can feedback more simply and understandably. Just as human can communicate with virtual assistants by using natural language, virtual assistants can understand users' intentions and respond accordingly through the semantic understanding module (Fig. 1). How to accurately identify important mentions and corresponding relationships in speech through semantic understanding model, and connect to the accurate entity to call and perform correct operation puts forward higher requirements for the voice control interface. Therefore, the primary goal of build up a smarter voice control interface is to solve the core challenges in natural language understanding and the way of human-computer interaction communication [51].

As the core module of Natural Language Understanding, Intent Detection (ID) aims to classify queries in sentences or question-and-answer tasks into corresponding intent categories using classification. It plays a vital role not only in the voice interaction of the IoT but also in search engines and smart question and answering systems, etc. In simple terms, when the user speaks or inputs a sentence or a text, the Intent Detection can accurately identify which problem it is and then assign it to the corresponding domain system for subsequent activities. This approach can improve the accuracy of a problem matching in the case of excessive problem categories. At the same time, Intent Detection as a type of pre-classification can significantly reduce retrieval time of target slot template matching. Especially in the context of vertical search, it is necessary to divide the query into a variety of specific domains to narrow down the target range for refined retrieval to improve the accuracy and efficiency of the retrieval. And Slot Filling (SF) task is a process that allows the user's intention to be converted into an explicit instruction to complete the information collection. It is equivalent to the precondition of identifying the specific action of the intent. After the conditions are met, new scripts can be triggered, and the voice assistant can continue to perform subsequent actions. With the increasing number of human-computer interaction scenarios in IoT, the scope of intent and entity type coverage has also increased dramatically. How to gradually increase the number of intent and entity types while ensuring the accuracy of them also poses greater challenges to the model aspect. Therefore, on this joint task, the Intent Detection (specific domain) task can improve the performance by having a classification model that performs better in more categories. And a more accurate named entity recognizer in the performance of Slot Filling task and Named Entity Recognition task in a specific domain to help improve the overall effect. Therefore, it is necessary to refine the design and integration of the models of these two tasks to realize the recognition of key information in the semantics (Slot Filling/Entity Recognition) after Intent Understanding (Intent Detection/Clinical Domain Detection), to achieve a more accurate Natural Language Understanding.

In a large number of previous studies [6, 11–13, 17–19, 60, 65, 67], it is more common to use RNN (Recurrent Neural Network) or CNN (Convolutional Neural Network) and corresponding variants to capture context information. However, these two types of methods have their characteristics and therefore have their shortcomings. RCNN (Recurrent Convolutional Neural Network) [22] as a combination and improvement of RNN and CNN. Unlike RNN-type methods, RCNN has no bias on word position in context, it can use the feature of each word in context more uniformly, and it is not necessary to determine the dependent length of the context by setting the window size like CNN. Therefore, the RCNN used in the Intent Detection task can effectively improve the common defects of the existing types of RNN and CNN methods. Additionally, the BiGRU (Bi-directional GRU) improved based on the RNN and the Capsule Network [48] created to improve CNN are two of the most advanced neural network models. Their effective feature modeling ability [36] also lacks attempts on the joint task of ID and SF.

Therefore, in this study, we explore the joint task of Clinical Domain Detection and Entity Recognition (CDD and ER), as well as the joint task of intent detection and slot filling. They are mainly based on related key issues such as how to achieve precise classified the multi-class intents or clinical domains to better service for the subsequent slot filling task or Entity Recognition task, and how the model performs on data with different sample sizes on Intent Detection task, etc. In this work, 1. RCNN model is the first time used on the Intent Detection task and also applied in the application of clinical scenarios. 2. A hybrid model with strong contextual modeling capability, BiGRU-Att-CapsuleNetwork is also specially designed for the Intent Detection task. 3. And combine the BiGRU-CRF model for slot filling task and Entity Recognition task, respectively. Among them, RCNN abandoned some original defects of RNN and CNN, while BiGRU-Att-CapsuleNet combines more advanced RNN variants (BiGRU) and replacement method (Capsule Network) of CNN to achieve better feature modeling and capture ability. Hence, the study employed the RCNN model to Clinical Domain Detection task to enhance the identification accuracy of IoT devices for clinical speech diagnosis in the initial condition of the patient and also used these two models respectively to improve the performance of Intent Detection for voice assistants.

Finally, combined with BiGRU-CRF, a state-of-the-art method for Named Entity Recognition task, the entities are more accurately extracted and slot filled. The experimental results prove that RCNN-BiGRU-CRF and BiGRU-Att-CapsuleNet-BiGRU-CRF are achieved ideal results on the joint task of clinical domain detection and entity recognition in the medical scenario. And on SNIPS, MIT Restaurant corpus and Movie corpus, and three other English and Chinese Intent Detection datasets, which are collected by voice assistant and other devices, achieve the competitive effects in single ID task and ID and SF joint task. These are provided valuable support for the construction of voice clinical diagnosis assistant based on medical scenarios and the move toward a more intelligent voice control system for IoT.

Our contributions are as follows:

- We propose two structures: RCNN-BiGRU-CRF and BiGRU-Att-CapsuleNet-BiGRU-CRF, used for Intent Detection and Slot Filling joint task for IoT Speech Understanding system, and the massive experiments prove our hybrid model structures have high competitiveness, the details can be found in Sects. 3–6.
- Based on the idea of the joint task of intent detection and slot filling, we transferred to the medical field to solve the Natural Language Understanding issues in the construction of clinical voice assistants and proposed the joint task of clinical domain detection and entity recognition. and applied the proposed hybrid model structures and carried out a large number of the corresponding comparative experiments to achieve the joint task, which can be found in Sect. 5.
- The hybrid model structures we propose can well help the semantic understanding module in IoT to realize key functions of Natural Language Understanding in multiple scenarios. As far as we know, there is quite little and limited research work in this field. Specific evaluations can be found in Sects. 5 and 6.

Additionally, as a study involving hybrid-based task models and focusing on joint task methods, the hybrid neural network-based task model can improve the shortcomings of a single model on semantic capture. And the joint task methods also have advantages that other conventional jointly modeling models [25, 32, 33] do not have while avoiding some of the shortcomings of previous approaches. Compared with the conventional jointly modeling methods: it is more difficult to achieve optimal performance on both tasks at the same time. By constructing a dedicated deep learning model for each task independently and using a specific domain knowledge pretrained language model as a word embedding according to each task, the actual effect of each task can be greatly improved. Furthermore, since the performance of task like slot filling is dependent on the effect of predecessor task (the accuracy of Intent Detection will affect the recognition of slot filling), independent professional models for a specific task will more effectively reduce the impact of misjudgment by pre-order task (e.g. Intent Detection). If perform the joint task of Intent Detection and slot filling in a large-scale simultaneous scenario, which will reduce the negative perception of users due to system detection errors.

2 Related work

The current IoT technology [1, 37, 50, 54] has tended to be applied on a large scale. With the great improvement of communication capabilities brought by the development of 5G, the interaction of the Internet of Things has become more frequent and intensive. Such a huge interactive scale cannot be separated from the support of artificial intelligence. As one of the cores of artificial intelligence, speech recognition-based semantic understanding technology has emerged on IoT devices including smart voice assistants. At present, there are many explorations in the field of construction of intelligent voice assistants. Some studies focus on expanding the application of intelligent voice assistants in different scenarios, such as the field of medical and health care [35, 44], intelligent home applications [28], autonomous driving [24], or personal and collective knowledge management [45, 47], etc. Another part of the study focuses on the design and optimization of intelligent voice assistants, such as analyzing from the language expression level of speech recognition [29], or improving the existing development methods and logical framework [57], etc.

As an earlier attempt, Matsuda et al. [30] proposed a method of generating the user interface of an AV controller by using NLU. It can analyze the instructions issued by the user using natural language, and ask the user the questions about corresponding necessary conditions for specific instructions to complete the execution of specific actions. But the study as a rule-based NLU method requires users to know correct answers i.e., the details about the AV system.

Park et al. [38] developed a natural language-based mobile device user interface for mobile devices with limited resources. Its purpose is to translate voice requests into commands that can be understood by mobile devices.

Santos et al. [49] surveyed the latest Intelligent Personal Assistants (IPAs) in general, related IoT protocols, and IPAs based on IoTs. While reviewing recent related technologies, it also described in detail how IoT networks can be improved IPAs functionalities.

The vast majority of IoT-related speech recognition researches mainly focuses on the design of hardware,

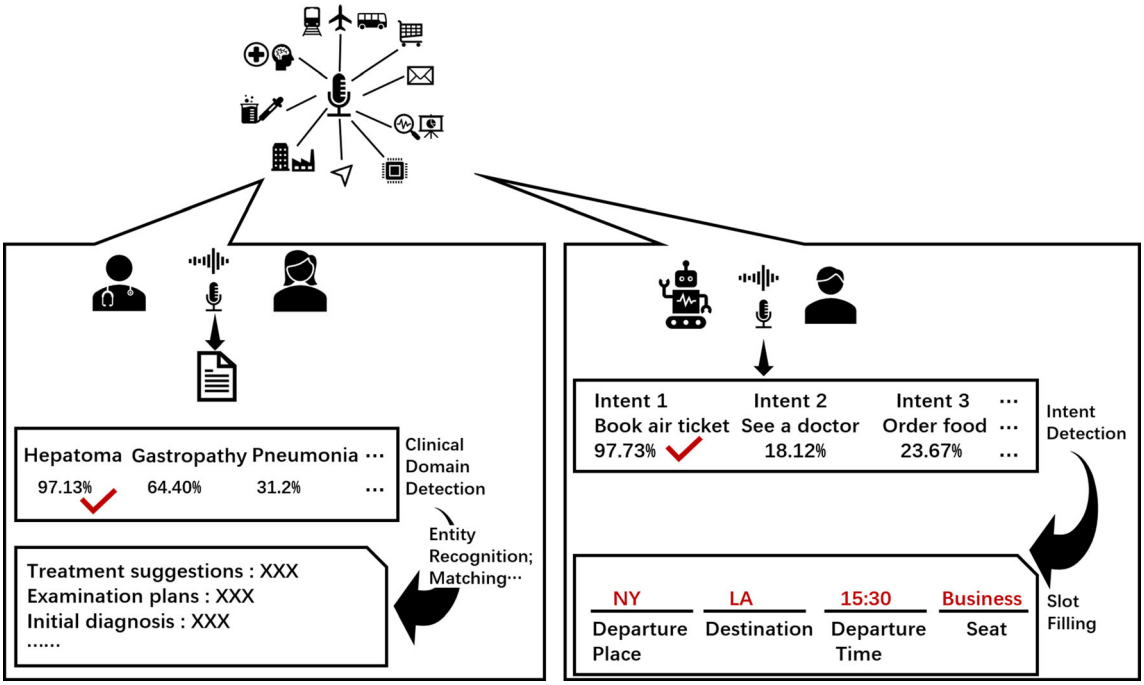

Fig. 1 Speech Understanding for IoT

architecture and security [8, 40, 43, 46, 49], and generally ignores the exploration of natural language understanding based on artificial intelligence. Hence, the latest relevant works for semantic understanding of IoT device design are also extremely rare. However, this is the key part to making IoT speech recognition systems truly work. Therefore, it is urgent to design reasonable and effective methods for the IoT semantic understanding module for different scenarios. In the era of artificial intelligence, deep learning, as the best performing solution at present, provides new ideas for this field.

Several previous studies have explored both Intent Detection and slot filling tasks. Among them, most of these studies [23, 32, 39, 62, 63] have proved that deep learning methods including RNN, CNN, and hybrid methods are suitable for the joint task scenario, and replace conventional methods based on rule-based have become the current state-of-the-art paradigm. Among them, these studies can be divided into the following types of methods:

2.1 RNN-based

Liu and Land [25] provided more contextual information for Intent Detection and slot filling by introducing the Attention mechanism to alignment-based RNN models. Compared with the independent task model, the joint model achieved a 0.56% absolute error reduction on the Intent Detection subtask and 0.23% absolute gain on the slot filling subtask of the benchmark Airline Travel Information System (ATIS) task.

Yu et al. [65] used a Bi-directional model-based RNN semantic frame parsing network structures to consider the interaction effect between the two tasks of Intent Detection and slot filling, and jointly execute these two joint tasks. Experiments show that their method is better than other previous methods on the ATIS dataset [16].

Chen et al. [5] proposed a Bi-directional LSTM (BiLSTM) model based on the Attention mechanism to jointly identify the Intent and Semantic slot filling of Hohhot's public transit queries. The experimental results show that the methods based on character tags are better than those based on word tags, and the proposed model is also better than other original LSTM methods in F1-Score performance.

2.2 CNN-based

Kim [20] used CNN to explore sentence classification tasks, and his research reported several improvements on the CNN-based models on four tasks including sentiment analysis and problem classification.

Xu and Sarikaya [61] described a CNN-based joint Intent Detection and slot filling model. This model can be considered as a neural network version of triangular CRF (TriCRF). Both tasks extract features through the CNN layer. This method is the first attempt of joint training

neural networks on the Intent Classification and slot filling joint task.

Vu [58] provided a CNN architecture for sequence labeling tasks, this structure retains the context words of the sequential information and pays special attention to the context of the current position, introducing them to the model for better perform the classification task. The method does not require prior language knowledge, and it performs better than previous integrated RNN-based models.

2.3 Hybrid-based

Wang et al. [59] proposed a neural network based on Attention-CNN-BiLSTM. The network is used to encode sentences and decoded by the LSTM network with the Attention mechanism, so it comes with contextual semantic information. The method achieved ideal overall performance on the ATIS dataset for Intent Detection and slot filling.

Niu et al. [10] proposed a novel bidirectional association model, which introduced the SF-ID network to establish direct connections for the joint task of Intent Detection and Slot Filling so that they can promote each other. At the same time, a new iterative mechanism is also designed inside the SF-ID network to enhance bidirectional related connections. The method is based on the ATIS [16] and Snips [7] datasets. The accuracy of the model at the sentence level semantic frame is improved by 3.79% and 5.42% respectively compared with other models.

Liu et al. [27] introduced a zero-shot adaptation method called attention-informed mixed-language Training (MLT) to realize cross-language semantic learning on a small number of corpus resources, which is designed for cross-language task-oriented dialogue systems. Compared with other existing state-of-the-art approaches, its method is used in Natural Language Understanding tasks (including Intent Detection and slot filling), and it does not need to use a large amount of bilingual corpus data. It also does not need to use a large amount of bilingual corpus data. A significant zero-shot adaptation performance improvement can be achieved with a small number of word pairs. Since most of the previous work focused on using semantic level information to calculate attention weights, Chen and Yu [4] introduced word-granularity information and created a fusion gate to integrate with semantic level information to jointly train Intent Detection and slot filling tasks. Gupta et al. [15] designed a framework for modularizing Intent Classification and slot filling joint tasks to enhance the transparency of the overall structure. The research also explored multiple self-attention mechanisms and RNN, CNN-type models, and contributed to the modeling paradigm of Intent Detection and Slot Filling joint task across datasets.

In summary, the hybrid-based method, as a type that improves the defects of the original single neural network model, has better semantic modeling performance for the context and the ability to handle more complex contexts. In contrast, the hybrid-based method is more worth considering.

3 Methodology

3.1 Method structure

On the joint task of the Intent Detection and Slot Filling model, the Intent Detection model can be used as a multi-class or multi-label classification task. By predefining the category to which the content belongs, it will be placed in the correct slot recognizer to improve the accuracy of slot filling. Compared with other integrated models, this type of specialized processing approach has a greater advantage in the processing of various tasks. Therefore, in the method section, we will explain in detail the modules of our joint models with Intent Detection and Slot Filling through two sub-sections, which includes the two models we designed based on BiGRU-Att-CapsuleNet and RCNN respectively for Intent Detection. And a BiGRU-CRF model for Slot Filling task (Fig. 2).

3.2 Intent Detection models

3.2.1 BiGRU-Att-CapsuleNetwork

The model is mainly divided into three parts including the BiGRU layer, the Attention layer, and the Capsule Network layer. Its overall structure is shown in Fig. 2.

In the first layer (Bidirectional GRU layer), as a variant of LSTM, GRU (Gated Recurrent Unit) has the characteristics of a simpler structure, fewer parameters, and better convergence (The neuron structure of GRU can be found in Fig. 3). It mainly consists of two parts: update gate and reset gate. The update gate z is used to control the degree of influence of the output at the previous time $t-1$ on the current hidden layer. The reset gate r is used to control the degree to which the information from the hidden layer is ignored at the previous moment. Here, a larger value of the update gate indicates that the current hidden layer is more affected by the output of the hidden layer at the previous moment, and a smaller reset gate indicates that more information from hidden layer at the previous moment is ignored. With the help of the hidden layer state h, the update gate z can be determined by the new information received by the current state and the historical information

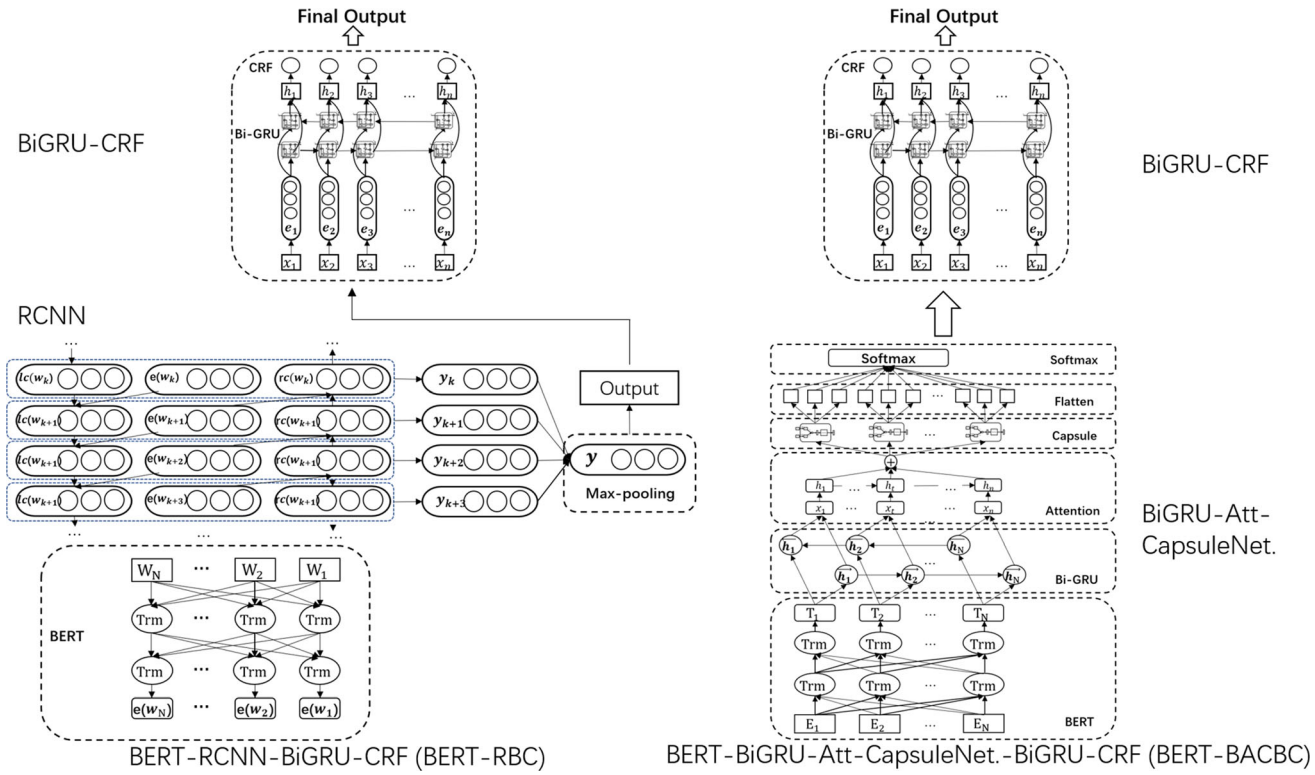

Fig. 2 Overall of the proposed model structures

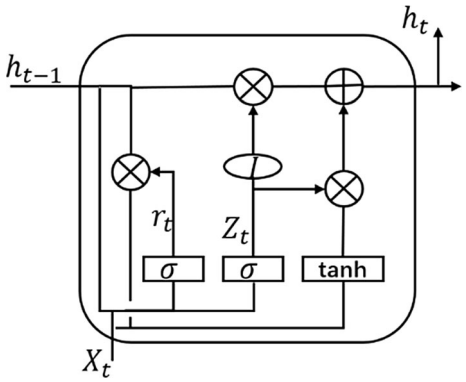

Fig. 3 Neuron Structure of Gated Recurrent Unit (GRU)

that needs to be forgotten; the reset gate r is determined by the information from the candidate state obtained from the historical information. Therefore, the update method of the GRU model can be expressed by the following formula:

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{1}$$

$$z_t = \sigma(W_i \cdot [h_{t-1}, x_t]) \tag{2}$$

$$\tilde{h_t} = \tanh(W_c \cdot [r_t \cdot h_{t-1}, x_t]) \tag{3}$$

$$h_t = (1 - z_t) \cdot c_{t-1} + z_t \cdot \tilde{h_t} \tag{4}$$

By establishing a forward and reverse GRU network for the context, respectively, the modeling from the two directions of the text (from the beginning to the end and from the end to the beginning) can be realized. The two unidirectional GRUs with opposite directions jointly determine the output. At each moment, input information is provided by two GRUs in opposite directions, and the two unidirectional GRUs will jointly determine the output. The current hidden state is jointly determined by the current input x_t, the forward hidden layer state at time $t-1$, i.e. $\overrightarrow{h_{t-1}}$, and the output of the hidden layer state in the reverse direction $h_{t-1}^{\leftarrow}$. Because the bidirectional GRU can be regarded as consisting of two unidirectional GRUs, the hidden state of the BiGRU at time t is obtained by weighted summing the output of the forward hidden state $\overrightarrow{h_{t-1}}$ and the reverse hidden layer state $h_{t-1}^{\leftarrow}$:

$$\overrightarrow{h_t} = GRU(x_i, \overrightarrow{h_{t-1}}) \tag{5}$$

$$h_t = GRU(x_i, h_{t-1}^{\leftarrow}) \tag{6}$$

$$h_i = w_t \overrightarrow{h_t} + v_t h_t + b_t \tag{7}$$

The $GRU()$ function represents a non-linear transformation of the input word vector, encoding the word vector into the corresponding GRU hidden layer state. w_t and v_t

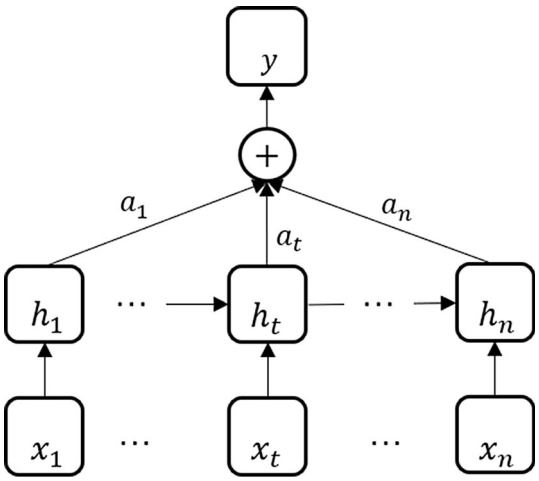

Fig. 4 Structure of Attention

respectively represent the weights of the forward hidden layer state h_t and the reverse hidden state h_t corresponding to the bidirectional GRU at time t, and t in b_t represents the bias corresponding to the hidden layer state at time t. Finally, the information of the forward and backward GRUs is merged to output a vector h_i for each word i. Among them, each Recurrent Unit can capture the dependencies on different time scales.

Therefore, through this layer, the modeling of information of sequential sentence can be realized.

As for Attention (Fig. 4), as a mechanism to capture important features in the context, it calculates the probability weights of word vectors at different moments using probability weight allocation, so that important features in the sentence can get more attention to improve the effect of feature extraction by hidden layer. The vector s from the initial hidden layer state to the updated hidden layer state represents the weighted coefficient a_1 of each hidden state in the new hidden layer state, and the sum of the product of each hidden layer state h_i at the initial input. v_i and w_i represent the weight coefficient matrix at the ith time, b_i represents the corresponding offset at the i-th time, and e_i represents the value determined by the hidden layer state vector h_i at the i-th time. The formula that converts the input initial state to the new attention state is as follows:

$$s = \sum_{i=1}^{l} \alpha_i h_i \tag{8}$$

$$\alpha_i = \frac{\exp(e_i)}{\sum_{j=1}^{n}(e_j)} \tag{9}$$

$$e_i = v_i \tanh(w_i h_i + b_i) \tag{10}$$

The Attention mechanism is used to enhance the semantic vector representation of the target word in the context and input to the next layer to improve the overall semantic representation effect of the model.

The Capsule Network is one of the currently best alternatives to CNN models, where the prediction vector $u_{i|j}$ represents the output of each capsule neuron in the last layer to the neurons in the next layer with different intensity connections. Through the matrix multiplication calculation of the output vector v_i of the BiGRU layer and the transformation matrix w_{ij}, the important spatial feature relationships between high-dimension and low-dimension features in the text can be encoded. A dynamic routing algorithm is used to calculate the coupling coefficient and weight the sum of the input vectors. With the help of a nonlinear "compression" activation function, the vector is compressed to a value between $0-1$ as a probability, and its original direction is maintained.

The most fundamental difference between a capsule network and a conventional artificial neural network is the unit structure of the network. For conventional neural network, the calculation of neurons can be divided into the following three steps:

Step 1. Scalar-weighted calculation to the input. Step 2. Sum the weighted input scalars. Step 3. Non-linear transformation of the scalar.

For capsules, it is calculated in four steps (Fig. 5):

Step 1. Multiply the input vectors, where v_1 and v_2 are from the output of the previous capsule. Within a single capsule, multiply v_1 and v_2 by w_1 and w_2, respectively. And new u_1 and u_2 can be obtained.

Step 2. Scalar weight the input vectors, multiply u_1 and c_1, multiply u_2 and c_2, where c_1 and c_2 are both scalars, and $c_1 + c_2 = 1$.

Step 3. Sum the obtained vectors $S = c_1 u_1 + c_2 u_2$.

Step 4. Non-linear transformation of vectors, convert the resulting vectors s, i.e., through the function $Squash(s) = \frac{\|s\|^2}{1+\|s\|^2}\frac{s}{\|s\|}$ to get the result s as the output of this capsule, and the result v can be used as the input of the next capsule.

$$u^1 = W^1 v^1, u^2 = W^2 v^2 \tag{11}$$

$$s = c_1 u^1 + c_2 u^2 \tag{12}$$

$$v = Squash(s) = \frac{\|s\|^2}{1+\|s\|^2}\frac{s}{\|s\|} \tag{13}$$

Through the flattening operation of the capsule network, the relative spatial relationships and directions of high and low dimension features are modeled and finally converted into the vector of the highest weighted feature in the text. The fully connected dense layer is used to judge the final output category using Softmax.

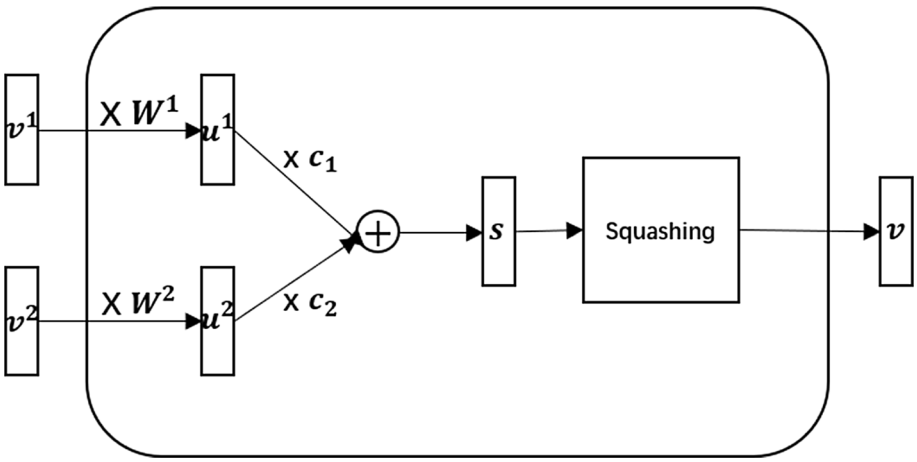

Fig. 5 The Operation of a Single Capsule

3.2.2 RCNN

Since conventional feature expression methods ignore context, semantics and word order information, higher-order N-gram, Tree Kernels, etc. are applied to feature expression. But they also have the disadvantage of sparseness, which affects accuracy in subsequent tasks. Therefore, the methods based on deep learning and word embedding have advantages over conventional feature representation methods in terms of feature extraction. From the point of view of improving convenience and performance, they do not need to manually formulate feature rules and solving data sparseness, but they still have some disadvantages. On the tasks about text classification, recurrent neural network (RNN) and convolutional neural network (CNN), as two commonly used deep learning representatives, have been widely used in various classification tasks. Among them, RNN is good at capturing context of sequence. But as a biased model, it has a greater preference for words in the back of the text sequence and usually has a higher weight for the words at such positions, but this does not take into account the fact that important words (key components) may appear anywhere in the text. As an unbiased model, CNN can obtain more important features through max-pooling than captured by Recursive or Recurrent Neural Network. But at the same time, due to it is difficult to determine the window size, the CNN model is prone to problems such as information loss or huge parameter space. Therefore, to solve the above problems, Lai et al. [22] proposed the Recurrent Convolutional Neural Network (RCNN) for text classification. The model uses a bidirectional loop structure, so it has less noise than the conventional window-based neural networks and can extract context to the maximum. And it through the max-pooling layer to automatically determine which feature has a more important role.

The embedding method of each word mainly consists of three parts (using concat): Left Context (*lc*), Embedding of the word itself (*e*), and Right Context (*rc*) (Fig. 6). Among them, w represents a word, and $e(w_i)$ is the embedding of the word, which is obtained through pre-training by the Skip-gram method.

$$x_i = [c_l(w_i); e(w_i); c_r(w_i)] \tag{14}$$

f represents a non-linear activation function, c_l fuses the information of the word in front of the current word, c_r fuses the information behind the word. Since each x_i is derived from this encoding method, it can be fused the information of context altogether to make long-range dependent predictions.

$$c_l(w_i) = f\left(W^{(l)}c_l(w_{i-1}) + W^{(sl)}e(w_{i-1})\right) \tag{15}$$

$$c_l(w_i) = f\left(W^{(r)}c_l(w_{i+1}) + W^{(sl)}e(w_{i+1})\right) \tag{16}$$

Through multi-layer perceptron processing and adding a *tanh* activation function, $y^{(2)}$ can be obtained, i.e., the score vector of the word for each category:

$$y_i^{(2)} = \tanh\left(W^{(2)}x_i + b^{(2)}\right) \tag{17}$$

Use max-pooling to get $y^{(3)}$:

$$y^{(3)} = \max_{i=1}^{n} y_i^{(2)} \tag{18}$$

After the processed by $y^{(3)}$, the vector of final score can be obtained by connecting a multilayer perceptron layer and a softmax layer:

$$y^{(4)} = W^{(4)}y_i^{(3)} + b^{(4)} \tag{19}$$

Apply the Softmax function to $y^{(4)}$ to convert the output into a probability:

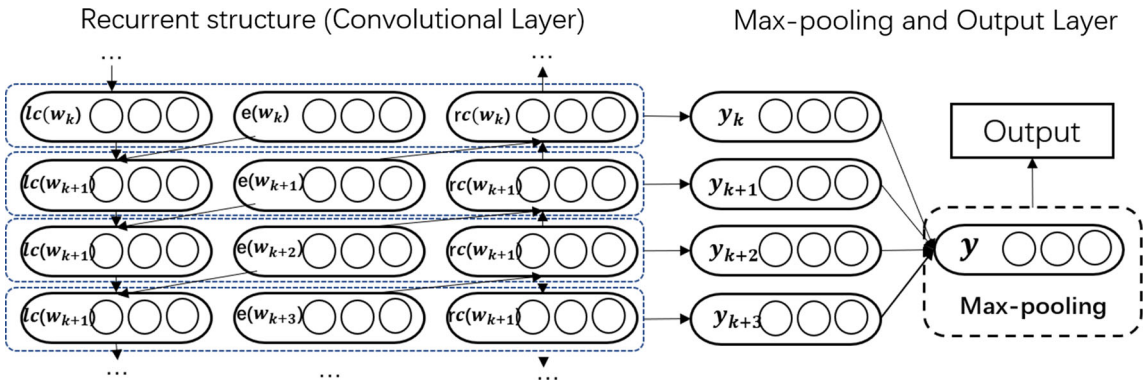

Fig. 6 Structure of Recurrent Convolutional Neural Network (RCNN)

$$p_i = \frac{\exp\left(y_i^{(4)}\right)}{\sum_{k=1}^{n} \exp\left(y_k^{(4)}\right)}. \tag{20}$$

3.3 Slot Filling model

3.3.1 BiGRU-CRF

BiGRU-CRF, as an Entity Recognition model, is used for the Slot Filling task after Intent Detection. Here, the Slot Filling task can be transformed into the Named Entity Recognition task. The specific method of BiGRU-CRF is to obtain a vector representation of each word by pre-trained language model, perform further semantic encoding by BiGRU, and finally, output to CRF (Conditional Random Field) layer to predict the maximum probability sequence label. Among them, the BiGRU in BiGRU-CRF has the same structure as the BiGRU used in the previous BiGRU-Att-CapsuleNet and is also used for sequence modeling tasks (e.g. Slot Filling task), which can capture the long-term context information. However, the original BiGRU model does not consider the dependencies between labels. For example, on some sequence labeling tasks, some labels cannot appear consecutively. Therefore, the model cannot use h_t to make label decisions independently. The CRF can obtain the global optimal label sequence by considering the adjacent relationship between the labels, therefore, using CRF to model the label sequence. Through the sequence $x = (x_1 + x_2 + \cdots + x_n)$ output by the BiGRU layer and its corresponding label $y = (y_1 + y_2 + \cdots + y_n)$, where the matching scores for a given input and output can be calculated:

$$s(x, y) = \sum_{i=1}^{n} \left(W_{y_{i-1}, y_i} + \mathrm{P}_{i, y_i}\right) \tag{21}$$

$p_{(i,y_i)}$ represents the score of the $y_i - th$ label of the character, and $W_{i,j}$ represents the transition score (W is transition matrix) of the label.

$$P_i = W_s h^{(t)} + b_s \tag{22}$$

$h^{(t)}$ is the hidden state of the input data $x^{(t)}$ at time t of the previous layer, the parameters are weight matrix and maximum conditional likelihood estimation for CRF, respectively, which training set is x_i, y_i, and the likelihood function formula is as follows:

$$L = \sum_{i=1}^{n} \log(P(y_i|x_i)) + \frac{\lambda}{2}\|\theta\|^2 \tag{23}$$

where P represents the probability corresponding to the original sequence to the predicted sequence:

$$P(y|x) = \frac{e^{s(x,y)}}{\sum_{y \in Y_x} e^{s(x,y)}} \tag{24}$$

Therefore, after identifying the speech intent, entity recognition through BiGRU-CRF can be used to fill the predefined slots (the output of CRF to determine the identified entity category).

4 Experiments and tasks

4.1 Experiments

Most of the experiments were tested under the environment of Intel Xeon E5-2678 v3 CPU, Dual 2.50GHz RAM, Dual Nvidia GeForce GTX 1080 Ti GPU. Among them, the number of batch size is 32, Sequence length is 256, epoch setting is 100, which also sets the 10 epochs' early stopping. At the same time, we also employed BERT [9], CMed-BERT [36] ELMo [42], GloVe and KazumaChar [41] respectively as word embedding in several tasks to achieve more accurate language representation.

4.2 Description of tasks

In the semantic understanding module, the purpose is to switch user questions and answers from the open domain (recognize the user's intention) into a closed domain dialog (a dialog that requires clear task details after identifying the user's intention, i.e., Slot Filling task). And this logical judgment is the entry condition, which is composed of condition groups and conditions, which form a set of "AND", "OR", and "NOT", i.e.:

- Two condition groups are "OR" relationship.
- The condition within the condition group is an "AND" relationship.
- The condition itself can be a "NOT" relationship.

Therefore, the semantic understanding module identifies the condition group in the user's narrative content (e.g., the patient's chief complaint is "Palpitations with fever"), or the user instruction contains the condition group "booking an air ticket" through Clinical Domain Detection (Intent Detection) and switch into the closed conversation on the corresponding topic. The input and output of the closed domain are enumerable. For example, the instruction "book an air ticket" requires three inputs: "Departure Time", "Departure Place", and "Destination". Therefore, the closed-domain dialogue between machines and humans needs to have a clear purpose and process.

Therefore, when the user's instruction lacks some required conditions, the dialogue system needs to ask questions actively and collect all the required conditions before execution. The process is also called "Clarification". The required condition here is a "Slot", and the conditions for its extension are also a "Clarification" (e.g., Business-class and Economy-class sections). Therefore, if the instruction process of "booking an air ticket" is simplified into three "Slots", then the slot filling process can be transformed into: "Departure Place", "Destination", "Departure Time" and other slots, which may also include "Seat Selection" and additional clarifications.

Furthermore, the attributes or capabilities of the slot also include: 1. whether the slot needs to be non-null (e.g., the required condition); 2. The order of Clarifications (Priority issues in the process of "Clarification" for multiple "Slots"); 3. Same-level Slot or Dependent Slot (whether the slots are independent of each other, whether the subsequent slots depend on the previous results, e.g., the three slots (same-level) of the "booking an air ticket" instruction listed above; country number slot of mobile number (Dependent), etc.) 4. Interface Slot (slot content obtained from other sources, e.g., via GPS to access the geographic location, etc.) and Keyword Slot (obtain slot content through keywords of user dialogue); 5. Slot Priority (Sort the selection order for multiple slots, e.g., the user specifies the contents of a slot in the dialog, which should take precedence over the slot content obtained by the interface slot); 6. Multiple rounds of memory status (users turn to other tasks when clarifying the slot, and return to the closed domain dialog to continue clarifying the slot status after finishing the task, etc.).

Therefore, the dialogue system process of the entire IoT semantic understanding module can be divided into:

- Open-domain multi-round dialogues (recognize user's intents).
- Entry conditions (switch into the closed domain dialog according to the set entry conditions).
- Closed domain dialogue (i.e., the process of Slot Filling task, allowing users to fill in the necessary slots through clarification so that tasks can be performed concretely).

In summary, its core task can be summarized as a joint task of the Intent Detection and Slot Filling type. Based on this idea, we try to translate them into Clinical Domain Detection and Entity Recognition joint task applied to clinical voice assistants, and Intent Detection and Slot Filling joint task based on universal IoT voice control scenarios.

5 Clinical domain detection and entity recognition joint task

In the medical scenario, the clinical voice assistant first needs to make a preliminary diagnosis of the patient's chief complaint, and match the candidate disease with the highest probability as the standard. Based on the preliminary diagnosis results, a pre-trained entity recognition model based on the domain knowledge of the corresponding disease type is called, to realize the identification of key mention or entity in the main complaint of the patient, and classify the corresponding entity types to achieve the slot template of specific illness can be filled. This joint task is based on the idea of Intent Detection and Slot Filling joint task, to matching the symptoms in the chief complaint and the corresponding candidate disease. And according to the judged results, it provides the clarification of the non-null slots for diagnosing related diseases and make the patient return the specific slot content to improve the entire chief complaint, to finish the process of collecting information related to the disease. This will provide a basic technical solution for the construction of an IoT clinical voice assistant that can provide valuable clinical decision-making information such as treatment recommendations, examination plans, and an initial diagnosis for patients, etc.

Table 1 Description of CMedBERT pre-training corpora [23]

Corpus source	Description	Size (Character)
Medical books	13 Chinese mainstream medical books for training corpus, including *Clinical Drug Therapy*, *Psychiatry*, etc.	4,384,503
SAHSU	The electronic medical records (EMRs) collected from the Second Affiliated Hospital of Soochow University, including 5090 electronic medical records.	2,002,202
Online resources	All the data were collected from 4 professional Chinese health websites: "39 Health (39.net)", "XunYiWenYao (XYWY.com)", "Feihua Health (fh21.com.cn)", "NetEase Health (jiankang.163.com)", which contents including medical encyclopedia, Q&A, blog and forum, etc.	29,092,216

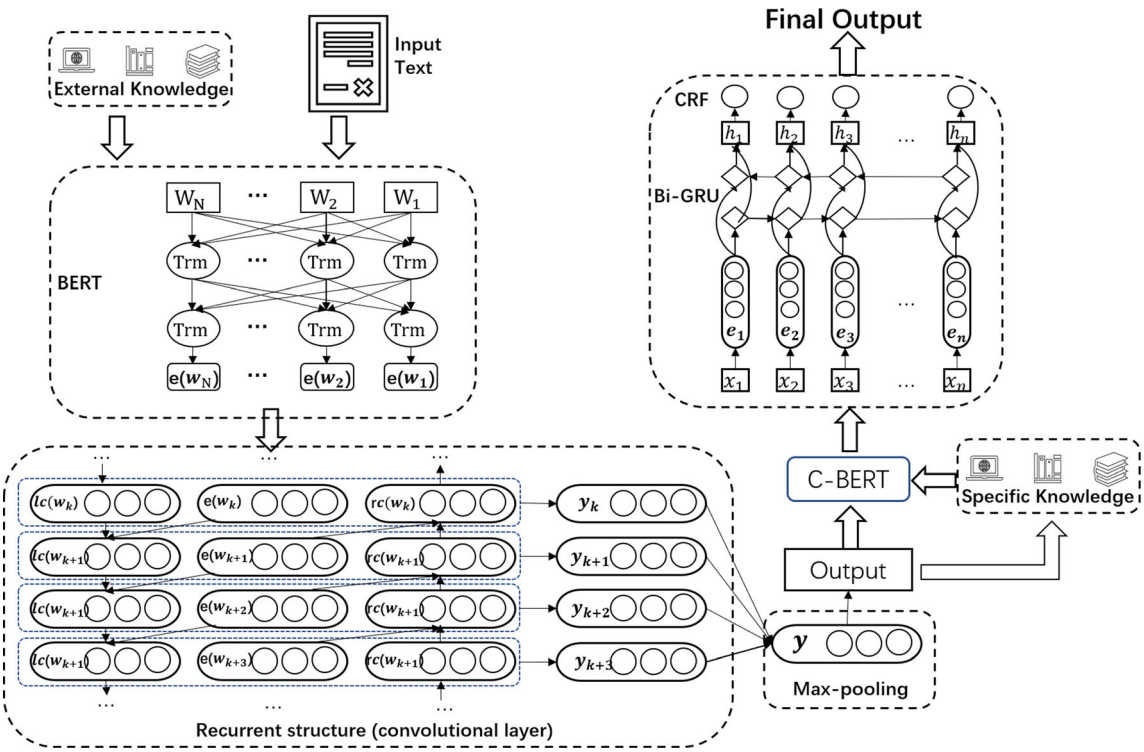

Fig. 7 CMedBERT-RCNN and C-BERT-BiGRU-CRF model structures for the joint task of Clinical Domain Detection and Entity Recognition

5.1 Pre-trained language models

We develop on our previously proposed CMedBERT as a pre-trained language model layer, which is used to build the model structures for the joint task of Clinical Domain Detection and Entity Recognition to evaluate the performance of the proposed structures on the actual datasets. Among them, CMedBERT (Table 1) is used as a pre-trained language model for the Clinical Domain Detection task, and C-BERT is used for the Entity Recognition task in the same way. Different from CMedBERT, Category-based BERT (C-BERT) is another pre-trained language model which is determined by the results of the Clinical Domain Detection, training corpus is specific domain knowledge related to this category (disease). We have prepared some Category-based pre-trained language models with specific domain knowledge, and take one type of the C-BERT (a C-BERT pre-trained based on EMR data of the disease "hepatoma" (extracted from SAHSU)) as an example to evaluate in detail (Fig. 7).

5.2 Evaluations

Our experimental results are evaluated on an integrated dataset (CCKS-SAHSU), which includes the real-world electronic medical record dataset (SAHSU) [36] and the largest academic evaluation dataset in the Chinese Clinical NLP field (CCKS 2019) [2]. Through the testing of 16

Table 2 The comparison results on the Clinical Domain Detection task

Task	Original model				Pre-trained model			
	Model	P	R	F1	Model	P	R	F1
Clinical Domain Detection	BERT-CNN-LSTM	0.4287	0.3762	0.3823	CMedBERT-CNN-LSTM	0.5731	0.5273	0.5285
	BERT-CNN	0.5544	0.5300	0.5103	CMedBERT-CNN	0.5921	0.5562	0.5448
	BERT-BiGRU	0.6091	0.5918	0.5741	CMedBERT-BiGRU	0.6044	0.5870	0.5678
	BERT-CNN-GRU	0.6296	0.5835	0.5791	CMedBERT-CNN-GRU	0.6465	0.5785	0.5735
	BERT-BiLSTM	0.6006	0.5696	0.5657	CMedBERT-BiLSTM	0.6359	0.6259	0.6136
	BERT-Dropout-BiGRU	0.6561	0.6506	0.6310	CMedBERT-Dropout-BiGRU	0.6522	0.6195	0.6213
	BERT-Dropout-AVRNN	0.6862	0.6585	0.6417	CMedBERT-Dropout-AVRNN	0.6787	0.6569	0.6556
	BERT-AVRNN	0.6610	0.6701	0.6487	CMedBERT-AVRNN	0.7040	0.6716	0.6596
	BERT-AVCNN	0.6706	0.6605	0.6487	CMedBERT-AVCNN	**0.7858**	0.7268	0.7170
	BERT-RCNN	0.7389	0.7114	0.7028	**CMedBERT-RCNN**	0.7456	***0.7531***	***0.7372***
	BERT-BiGRU-Att-CapsNet.	0.7463	0.7410	0.7295	**CMedBERT-BiGRU-Att-CapsNet.**	***0.7654***	**0.7590**	**0.7389**

The numbers in bold represent the largest value in each column; the numbers in bold italics represent the second highest value in each column

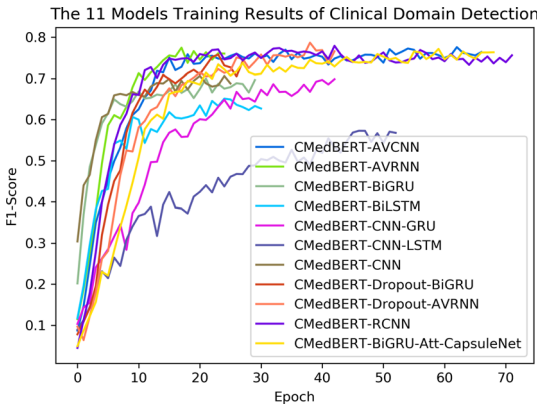

Fig. 8 The 11 types model structure of Clinical Domain Detection task

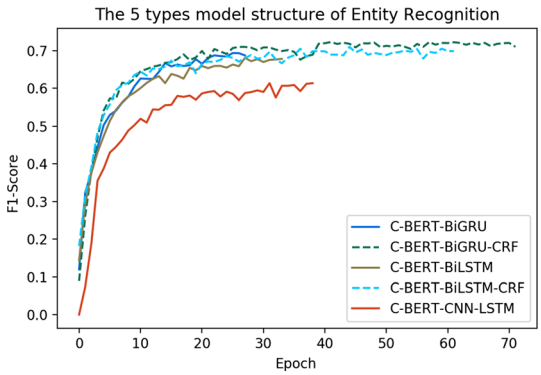

Fig. 9 The 5 types model structure of Entity Recognition task

model structures on the Clinical Domain Detection task and 10 model structures on the Entity Recognition task, respectively, the effectiveness of our proposed model structures on medical tasks can be confirmed.

On the Clinical Domain Detection task, CMedBERT-based RCNN, and BiGRU-Att-CapsuleNet. obtained F1-Scores of 73.89% and 73.72%, respectively. These are also the two models that get the top 2 F1-Score on this task. It is also worth noting that CMedBERT-AVCNN also achieved ideal performance in the accuracy, and the results are closer to our methods than other models. This may also be one of the ideal candidate models for this task (Table 2, Fig. 8). Additionally, compared with the performance of downstream models based on the BERT language model,

Table 3 The comparison results on the Entity Recognition task (Based on "hepatoma" disease data in the CCKS-SAHSU integrated dataset)

Task	Original model				Pre-trained model			
	Model	P	R	F1	Model	P	R	F1
Entity Recognition	BERT-CNN-LSTM	0.6330	0.6910	0.6421	C-BERT-CNN-LSTM	0.6481	0.7248	0.6839
	BERT-BiLSTM	0.6852	0.7640	0.7214	C-BERT-BiLSTM	0.7095	0.7536	0.7303
	BERT-BiGRU	0.7047	0.7437	0.7221	C-BERT-BiGRU	0.7125	0.7635	0.7367
	BERT-BiLSTM-CRF	0.6897	0.7716	0.7282	C-BERT-BiLSTM-CRF	0.7201	0.7874	0.7516
	BERT-BiGRU-CRF	0.7214	0.7820	0.7499	**C-BERT-BiGRU-CRF**	**0.7240**	**0.8072**	**0.7632**

The numbers in bold represent the largest value in each column; the numbers in bold italics represent the second highest value in each column

Table 4 Comparative experimental results on a large number sample of Intent Detection datasets

Model	SNIPS			SMP2018		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score
BiLSTM	0.9247	0.9211	0.9219	0.9235	0.9294	0.9181
BiGRU	0.9335	0.9351	0.9339	0.9273	0.9068	0.9131
AVRNN	0.9440	0.9442	0.9440	0.9003	0.8744	0.8755
CNN-LSTM	0.9326	0.9289	0.9301	0.8633	0.8400	0.8333
DPCNN	0.9247	0.8487	0.8543	0.9002	0.8233	0.8488
CNN-GRU	0.9491	0.9450	0.9461	0.8623	0.8603	0.8553
Dropout-AVRNN	0.9440	0.9440	0.9436	0.8987	0.8807	0.8840
Dropout-BiGRU	0.9450	0.9406	0.9419	0.8925	0.8698	0.8767
RCNN	0.9506	0.9484	***0.9489***	0.9521	0.9426	**0.9439**
BiGRU-Att-CapsuleNet. (BAC)	0.9624	0.9612	**0.9615**	0.9334	0.9402	***0.9289***

The numbers in bold represent the largest value in each column; the numbers in bold italics represent the second highest value in each column

Table 5 Comparative experimental results on a small number sample of Intent Detection datasets

Model	AskUbuntu			WebApplication		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score
BiLSTM	0.6706	0.6250	0.6197	0.4896	0.5625	0.4958
BiGRU	0.7285	0.6000	0.6114	0.5469	0.5625	0.5179
AVRNN	0.6698	0.6417	0.6446	0.6750	0.6875	***0.6406***
CNN-LSTM	0.5763	0.5667	0.5627	0.5500	0.5625	0.5198
DPCNN	0.5422	0.5763	0.5502	0.6000	0.6875	0.6079
CNN-GRU	0.6524	0.6417	0.6374	0.5000	0.6250	0.5312
Dropout-AVRNN	0.6733	0.5917	0.5912	0.6094	0.6250	0.5845
Dropout-BiGRU	0.6367	0.4083	0.3601	0.5437	0.6250	0.5521
RCNN	0.7975	0.8000	**0.7859**	0.4979	0.6250	0.5406
BiGRU-Att-CapsuleNet. (BAC)	0.7820	0.6750	***0.6686***	0.7115	0.7500	**0.6930**

The numbers in bold represent the largest value in each column; the numbers in bold italics represent the second highest value in each column

CMedBERT has improved the performance of most downstream models to varying degrees. And on the Entity Recognition task, the performance of multiple models based on the C-BERT language model proves that domain classification will greatly improve the performance of the Entity Recognition task. Compared with the original BERT language model, the pre-trained C-BERT based on "hepatoma" EMR data has particular help in the accuracy, recall rate and F1-Score of the corresponding domain entity recognition (Table 3, Fig. 9).

6 Intent Detection and Slot Filling joint task

6.1 Intent Detection task

Here, we discuss the performance of our methods in detail in four Intent Detection datasets. These include two large-sample datasets: SNIPS (English) [7, 55], SMP2018 (Chinese) [66], and two small-sample English datasets: AskUbuntu [52], WebApplication [52].

Based on the above results (Tables 4, 5), it is shown that RCNN and BiGRU-Att-CapsuleNet are greatly improved in the tests on different datasets, which compared to RNN and CNN or hybrid type methods in the comparison experiments. By comparing the test results on the small datasets, it can be found that when the number of samples participating in training in each category is greater, the effect of RCNN is better (the "support" value in AskUbuntu test set is 120 but the number in WebApplication is 16). At the same time, the results on the large sample datasets can be found that under the same training conditions (parameters and pre-trained models are equivalents) (Fig. 10), the overall performance of English Intent Detection task (SNIPS) is better than that of Chinese task (SMP2018), this may be related to many factors: semantic ambiguity caused by Chinese word segmentation; pre-

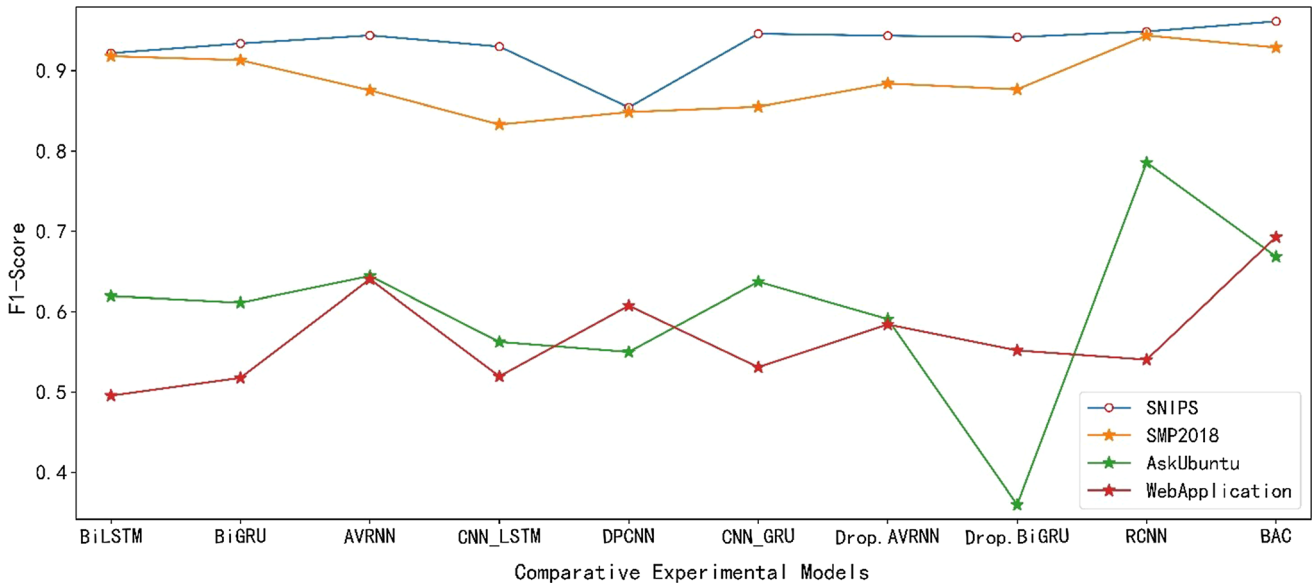

Fig. 10 Visualization of test results on the above four Intent Detection datasets

trained corpus of language models in Chinese are generally less than in English; the test datasets in SMP2018 have fewer samples than SNIPS datasets, etc. Additionally, compared with the results on large sample datasets, deep learning models generally have poor performance on small sample datasets. This again confirms that deep learning-based methods often require larger sample sizes to realize their true potential. At the same time, the deep learning methods based on RNN and CNN types have a large gap compared with the effect achieved by the conventional machine learning methods in [52] in the same datasets. Therefore, the use of conventional machine learning methods in the classification of small sample intents may be more suitable.

6.2 Slot Filling task

In this section, 1. The Slot Filling task is evaluated as an independent experimental task on two datasets: MIT Restaurant corpus and Movie corpus [56]. 2. The Slot Filling task is also evaluated as part of the joint task with the Intent Detection task on the SNIPS dataset (The dataset used here is the version provided by [56]). The specific content is as follows:

1. The independent evaluation of the Slot Filling task is mainly based on the two datasets, MIT Restaurant corpus, and Movie corpus [56]. Our proposed methods are also compared with various other cutting-edge Slot Filling methods with the most advanced pre-trained language models. The BiGRU-CRF model with the

Table 6 Slot F1-score of MIT Restaurant corpus and Movie corpus [34]

Model Movie_trivia10k13	Restaurant	Movie_eng	
Dom-Gen-Adv [26]	0.7425	0.8303	0.6351
Joint Dom Spec and Gen-Adv [26]	0.7447	0.8533	0.6533
Data Augmentation via Joint Variational Generation [64]	0.7300	0.8290	0.6570
ELMo-BiLSTM	0.7754	0.8537	0.6797
ELMo-BiLSTM-CRF	0.7977	***0.8736***	0.7183
ELMo -Enc-dec focus [68]	0.7877	0.8668	0.7085
GloVe and KazumaChar-BiLSTM	0.7802	0.8633	0.6855
GloVe and KazumaChar-BiLSTM-CRF	0.7984	**0.8761**	***0.7190***
GloVe and KazumaChar-Enc-dec focus	***0.7998***	0.8682	0.7110
BERT-BiGRU-CRF	**0.8147**	0.8698	**0.7239**

The numbers in bold represent the largest value in each column; the numbers in bold italics represent the second highest value in each column

Table 7 Results of Intent Dectection task and Slot Filling task on the SNIPS [7, 56]

Model	Intent Detection (Acc.)	Slot Filling (F1)
Slot-Gated [14]	0.9700	0.8880
Joint BERT [3]	0.9860	0.9700
ELMo-BiLSTM [53]	0.9900	0.9692
ELMo-BiLSTM-CRF	0.9914	0.9575
ELMo-Enc-dec focus [68]	0.9871	0.9622
GloVe and KazumaChar-BiLSTM	***0.9914***	0.9624
GloVe and KazumaChar-BiLSTM-CRF	0.9886	0.9631
GloVe and KazumaChar-Enc-dec focus [68]	0.9843	0.9606
Joint BERT-CRF [3]	0.9840	0.9670
BERT-BiLSTM	0.9886	0.9692
BERT-BiLSTM-CRF	0.9886	0.9700
BERT-Enc-dec focus [68]	0.9871	**0.9717**
BERT-RCNN-BiGRU-CRF (BERT-RBC)	0.9900	***0.9703***
BERT-BiGRU-Att-CapsuleNet.-BiGRU-CRF (BERT-BACBC)	**0.9921**	0.9699

The numbers in bold represent the largest value in each column; the numbers in bold italics represent the second highest value in each column

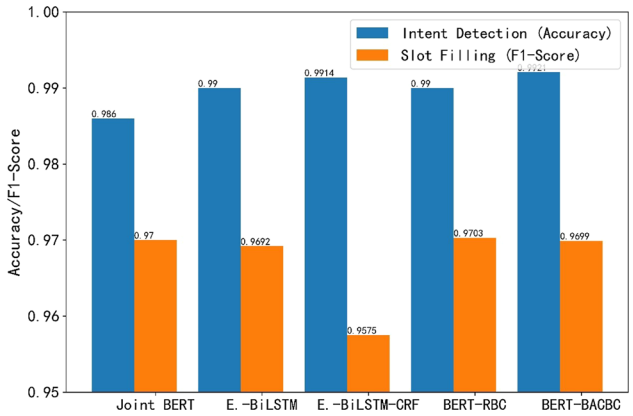

Fig. 11 The performance of the top 4 ranked methods in Intent Detection task on full ID and SF joint task (SNIPS)

BERT language model as word embedding obtained the highest F1-Score in the Restaurant and Movie_trivia10k13 datasets, which were 81.47% and 72.39%, respectively (Table 6). Therefore, it can be proved that RCNN and BiGRU-Att-CapsuleNet have a strong and competitive ability in context capture on classification tasks.

2. The two proposed joint models are also fully evaluated in a full ID and SF joint task on the SNIPS dataset (Table 7, Fig. 11). Among them, BERT-RBC (BERT-RCNN-BiGRU-CRF) also achieved an F1-Score of 97.03% on the Slot Filling task, ranking second in all tested models. At the same time, BERT-BACBC (BERT-BiGRU-Att-CapsuleNet.-BiGRU-CRF) achieved the highest accuracy (99.21%) compared to other methods on the Intent Detection task. On the Slot Filling task, its F1-Score reached 96.99%, which is only 0.04% lower than the second place (BERT-RBC). This proves that based on RCNN and BiGRU-Att-CapsuleNet, these two outstanding intent detection models, combined with BiGRU-CRF, a current state-of-the-art entity recognition method, can be well adapted for ID and SF joint task and help for improve the performance of the joint task. At the same time, the ideal performance of RCNN and BiGRU-Att-CapsuleNet on the pre-task (Intent Detection) has also laid a good foundation for more accurate slot identification in the further.

7 Conclusion and future work

The vision of direct communication between humans and devices in the Internet of Things through voice interaction approaches poses a greater challenge to Natural Language Understanding. This requires continuous exploration of the key tasks in the NLU. As one of the core joint tasks in Natural Language Understanding, Intent Detection and Slot Filling joint task have been widely used in human-computer interaction scenarios such as Question and Answering robots, Dialog Management, Search Engines, etc. Its specific methods can be transformed into two types of subtasks: Classification and Named Entity Recognition. Firstly, classify the language or text content according to the intent category, and then identify the entity from the text and filled into the preset slot through Named Entity Recognition to complete the capture and understanding of the content details described by the user. If the models of the two tasks can be designed more refined and targeted, to enhance the performance of the Intent Detection task to better serve the precision of Slot Filling task through forward and backward connection, it will effectively improve the overall effect of semantic understanding for IoT.

Therefore, this work designed two structures for the joint task, including two methods respectively based on RCNN and BiGRU-Att-CapsuleNetwork for Intent Detection task and Clinical Domain Detection task, and the use of BiGRU-CRF model to realize the Slot Filling task and Entity Recognition task. Hence, two structures, RCNN-BiGRU-CRF (RBC) and BiGRU-Att-CapsuleNet.-BiGRU-CRF (BACBC), are used for the full Intent Detection and Slot Filling joint task and Clinical Domain Detection and Entity Recognition joint task, respectively. Among them,

on the CDD and ER joint task, the RCNN and BiGRU-Att-CapsuleNet based on the CMedBERT language model respectively achieved 73.72% and 73.89% F1-Scores on the CDD task. And the BiGRU-CRF model based on C-BERT also obtained 76.32% F1-Score on ER task, these performances are among the top in the evaluations. This proves that the preprocessing of Clinical Domain Detection will help improve the effectiveness of the downstream Entity Recognition task.

On the ID and SF joint task, the BERT-BACBC model achieved 99.21% accuracy on the SNIPS dataset, and the BERT-RBC reached 97.03% F1-Score on the Slot Filling task, ranking first and second respectively in the compared models. Additionally, the F1-Score of the BERT-BACBC model on the Slot Filling task is only 0.04% lower than the second one, and the BERT-RBC reaches 99.00% accuracy on the Intent Detection task, both ranking third on the corresponding task. At the same time, the two models also achieved ideal performance compared with the comparative models on Chinese and English Intent Detection tasks with different sample sizes. Experimental results show the competitiveness of the proposed model on Intent Detection tasks. It proves that by classifying more accurate intents, will lay a good foundation for achieving the ideal performance of the subsequent slot filling models. As shown in the results of multiple comparative experiments in the study, ID and SF based on a large number of samples have achieved performance that can be applied in practice, but the performance on small sample data is not ideal, which is a common problem in reality (low resource problem). Therefore, subsequent research will focus more on improving the performance of such limited conditions. This research provides some valuable method references for building an ideal IoT-based semantic understanding module, which is a useful exploration for the move towards a more humanized IoT voice interaction system.

Funding Funding was provided by VC Research (Grant No. VCR 0000021).

Compliance with ethical standards

Conflict of interest The authors declared that they have no conflicts of interest to this work.

References

1. Behera TM, Mohapatra SK, Samal UC, Khan MS, Daneshmand M, Gandomi AH (2019) Residual energy based cluster-head selection in wsns for iot application. IEEE Internet Things J 6:5132
2. CCKS2019: Shared tasks—2019 china conference on knowledge graph and semantic computing. CCKS (2019). http://www.ccks2019.cn/?page_id=62. Accessed 3 Aug 2019
3. Chen Q, Zhuo Z, Wang W (2019) Bert for joint intent classification and slot filling. arXiv preprint arXiv:1902.10909
4. Chen S, Yu S (2019) Wais: Word attention for joint intent detection and slot filling. Proc AAAI Conf Artif Intell 33:9927–9928
5. Chen T, Lin M, Li Y (2019) Joint intention detection and semantic slot filling based on blstm and attention. In: 2019 IEEE 4th international conference on cloud computing and big data analysis (ICCCBDA), pp 690–694. IEEE
6. Chen YN, Hakkani-Tür D, Tür G, Gao J, Deng L (2016) End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding. In: Interspeech, pp 3245–3249
7. Coucke A, Saade A, Ball A, Bluche T, Caulier A, Leroy D, Doumouro C, Gisselbrecht T, Caltagirone F, Lavril T, et al (2018) Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. arXiv preprint arXiv:1805.10190
8. de Barcelos Silva A, Gomes MM, da Costa CA, da Rosa Righi R, Barbosa JLV, Pessin G, De Doncker G, Federizzi G (2020) Intelligent personal assistants: a systematic literature review. Expert Syst Appl 147:113193
9. Devlin J, Chang MW, Lee K, Toutanova K (2019) Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp 4171–4186
10. Niu P, Chen Z, Song M (2019) A novel bi-directional interrelated model for joint intent detection and slot filling. In: Proceedings of the 57th annual meeting of the association for computational linguistics, pp 5467–5471
11. Firdaus M, Bhatnagar S, Ekbal A, Bhattacharyya P (2018) Intent detection for spoken language understanding using a deep ensemble model. In: Pacific Rim international conference on artificial intelligence, pp 629–642. Springer
12. Firdaus M, Kumar A, Ekbal A, Bhattacharyya P (2019) A multi-task hierarchical approach for intent detection and slot filling. Knowl Based Syst 183:104846
13. Gong Y, Luo X, Zhu Y, Ou W, Li Z, Zhu M, Zhu KQ, Duan L, Chen X (2019) Deep cascade multi-task learning for slot filling in online shopping assistant. Proceedings of the AAAI conference on artificial intelligence 33:6465–6472
14. Goo CW, Gao G, Hsu YK, Huo CL, Chen TC, Hsu KW, Chen YN (2018) Slot-gated modeling for joint slot filling and intent prediction. In: Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, Volume 2 (Short Papers), pp 753–757
15. Gupta A, Hewitt J, Kirchhoff K (2019) Simple, fast, accurate intent classification and slot labeling for goal-oriented dialogue systems. In: Proceedings of the 20th annual SIGdial meeting on discourse and dialogue, pp 46–55
16. Hemphill CT, Godfrey JJ, Doddington GR (1990) The atis spoken language systems pilot corpus. In: Speech and natural language: proceedings of a workshop held at Hidden Valley, Pennsylvania, June 24–27, 1990
17. Iosif E, Klasinas I, Athanasopoulou G, Palogiannidi E, Georgiladakis S, Louka K, Potamianos A (2018) Speech understanding for spoken dialogue systems: from corpus harvesting to grammar rule induction. Comput Speech Lang 47:272–297
18. Jiao L, Yanling L, Min L (2019) Review of intent detection methods in the human-machine dialogue system. J Phys Conf Ser 1267:012059

19. Kim J, Jeong Y, Lee JH (2019) Speaker-informed time-and-content-aware attention for spoken language understanding. Comput Speech Lang 60:101022
20. Kim Y (2014) Convolutional neural networks for sentence classification. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1746–1751
21. Kranz M, Holleis P, Schmidt A (2010) Embedded interaction: Interacting with the internet of things. IEEE Internet Comput 14(2):46–53
22. Lai S, Xu L, Liu K, Zhao J (2015) Recurrent convolutional neural networks for text classification. In: Twenty-ninth AAAI conference on artificial intelligence
23. Li Y, Ni P, Peng J, Zhu J, Dai Z, Li G, Bai X (2019) A joint model of clinical domain classification and slot filling based on RCNN and BiGRU-CRF. In: 2019 IEEE international conference on big data (Big Data). IEEE, pp 6133–6135
24. Lin SC, Hsu CH, Talamonti W, Zhang Y, Oney S, Mars J, Tang L (2018) Adasa: A conversational in-vehicle digital assistant for advanced driver assistance features. In: The 31st annual ACM symposium on user interface software and technology. ACM, pp 531–542
25. Liu B, Lane I (2016) Attention-based recurrent neural network models for joint intent detection and slot filling. Interspeech 2016:685–689
26. Liu B, Lane I (2017) Multi-domain adversarial learning for slot filling in spoken language understanding. arXiv preprint arXiv:1711.11310
27. Liu Z, Shin J, Xu Y, Winata GI, Xu P, Madotto A, Fung P (2019) Zero-shot cross-lingual dialogue systems with transferable latent variables. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th International joint conference on natural language processing (EMNLP-IJCNLP), pp 1297–1303
28. Luria M, Hoffman G, Zuckerman O (2017) Comparing social robot, screen and voice interfaces for smart-home control. In: Proceedings of the 2017 CHI conference on human factors in computing systems, pp. 580–628. ACM
29. Matani J, Gervais P, Calvo M, Feuz S, Deselaers, T (2018) Matching language and accent in virtual assistant responses. Technical Disclosure Commons. https://www.tdcommons.org/dpubs_series/1239/. Accessed 19 Dec 2019
30. Matsuda M, Nonaka T, Hase T (2006) An av control method using natural language understanding. IEEE Trans Consumer Electr 52(3):990–997
31. Mehrabani M, Bangalore S, Stern B (2015) Personalized speech recognition for internet of things. In: 2015 IEEE 2nd world forum on internet of things (WF-IoT). IEEE, pp 369–374
32. Mesnil G, Dauphin Y, Yao K, Bengio Y, Deng L, Hakkani-Tur D, He X, Heck L, Tur G, Yu D et al (2014) Using recurrent neural networks for slot filling in spoken language understanding. IEEE/ACM Trans Audio Speech Lang Process 23(3):530–539
33. Mesnil G, He X, Deng L, Bengio Y (2013) Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In: Interspeech, pp 3771–3775
34. MIT-CSAIL: MIT restaurant corpus and mit movie corpus. MIT-CSAIL (2014). https://groups.csail.mit.edu/sls/downloads/. Accessed 15 Oct 2019
35. Morris RR, Kouddous K, Kshirsagar R, Schueller SM (2018) Towards an artificially empathic conversational agent for mental health applications: system design and user perceptions. J Med Internet Res 20(6):e10148
36. Ni P, Li Y, Zhu J, Peng J, Dai Z, Li G, Bai X (2019) Disease diagnosis prediction of emr based on BiGRU-ATT-capsnetwork model. In: 2019 IEEE international conference on big data (Big Data). IEEE, pp 6166–6168
37. Paranjothi A, Khan MS, Zeadally S, Pawar A, Hicks D (2019) GSTR: Secure multi-hop message dissemination in connected vehicles using social trust model. Internet Things 7:100071
38. Park SY, Byun J, Rim HC, Lee DG, Lim H (2010) Natural language-based user interface for mobile devices with limited resources. IEEE Trans Consumer Electr 56(4):2086–2092
39. Peng B, Yao K, Jing L, Wong KF (2015) Recurrent neural networks with external memory for spoken language understanding. In: Natural Language Processing and Chinese Computing. Springer, pp 25–35
40. Peng CY, Chen RC (2018) Voice recognition by google home and raspberry pi for smart socket control. In: 2018 Tenth international conference on advanced computational intelligence (ICACI). IEEE, pp 324–329
41. Pennington J, Socher R, Manning C (2014) Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
42. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. In: Proceedings of NAACL-HLT, pp 2227–2237
43. Petnik J, Vanus J (2018) Design of smart home implementation within iot with natural language interface. IFAC-PapersOnLine 51(6):174–179
44. Pradhan A, Mehta K, Findlater L (2018) Accessibility came by accident: use of voice-controlled intelligent personal assistants by people with disabilities. In: Proceedings of the 2018 CHI conference on human factors in computing systems. ACM, p 459
45. Reis A, Paulino D, Paredes H, Barroso J (2017) Using intelligent personal assistants to strengthen the elderlies' social bonds. In: International conference on universal access in human–computer interaction. Springer, pp 593–602
46. Rubio-Drosdov E, Díaz-Sánchez D, Almenárez F, Arias-Cabarcos P, Marín A (2017) Seamless human-device interaction in the internet of things. IEEE Trans Consumer Electr 63(4):490–498
47. Saad U, Afzal U, El-Issawi A, Eid M (2017) A model to measure qoe for virtual personal assistant. Multimed Tools Appl 76(10):12517–12537
48. Sabour S, Frosst N, Hinton GE (2017) Dynamic routing between capsules. In: Advances in neural information processing systems, pp 3856–3866
49. Santos J, Rodrigues JJ, Casal J, Saleem K, Denisov V (2016) Intelligent personal assistants based on internet of things approaches. IEEE Syst J 12(2):1793–1802
50. Sekaran K, Khan MS, Patan R, Gandomi AH, Krishna PV, Kallam S (2019) Improving the response time of m-learning and cloud computing environments using a dominant firefly approach. IEEE Access 7:30203–30212
51. Shilin I, Kovriguina L, Mouromtsev D, Wohlgenannt G, Ivanitskiy R (2018) A method for dataset creation for dialogue state classification in voice control systems for the internet of things. In: R. Piotrowski's readings in language engineering and applied linguistics, pp 96–106
52. Shridhar K, Dash A, Sahu A, Pihlgren GG, Alonso P, Pondenkandath V, Kovács G, Simistira F, Liwicki M (2019) Subword semantic hashing for intent classification on small datasets. In: 2019 International joint conference on neural networks (IJCNN). IEEE, pp 1–6
53. Siddhant A, Goyal A, Metallinou A (2019) Unsupervised transfer learning for spoken language understanding in intelligent agents. Proceedings of the AAAI conference on artificial intelligence 33:4959–4966
54. Singanamalla V, Patan R, Khan MS, Kallam S (2019) Reliable and energy-efficient emergency transmission in wireless sensor networks. Internet Technol Lett 2(2):e91

55. Snipsco: Nlu-benchmark. Github (2019). https://www.github.com/snipsco/nlu-benchmark. Accessed 07 Oct 2019
56. sz128: Slot filling and intent detection of SLU. Github (2019). https://www.github.com/sz128/slot_filling_and_intent_detection_of_SLU. Accessed 15 Oct 2019
57. Vtyurina A, Fourney A (2018) Exploring the role of conversational cues in guided task support with virtual assistants. In: Proceedings of the 2018 CHI conference on human factors in computing systems. ACM, p 208
58. Vu NT (2016) Sequential convolutional neural networks for slot filling in spoken language understanding. Interspeech 2016:3250–3254
59. Wang Y, Tang L, He T (2018) Attention-based cnn-blstm networks for joint intent detection and slot filling. In: Chinese computational linguistics and natural language processing based on naturally annotated big data. Springer, pp 250–261
60. Xu C, Li Q, Zhang D, Cui J, Sun Z, Zhou H (2020) A model with length-variable attention for spoken language understanding. Neurocomputing 379:197–202
61. Xu P, Sarikaya R (2013) Convolutional neural network based triangular crf for joint intent detection and slot filling. In: 2013 IEEE workshop on automatic speech recognition and understanding. IEEE, pp 78–83
62. Yao K, Peng B, Zhang Y, Yu D, Zweig G, Shi Y (2014) Spoken language understanding using long short-term memory neural networks. In: 2014 IEEE spoken language technology workshop (SLT). IEEE, pp 189–194
63. Yao K, Zweig G, Hwang MY, Shi Y, Yu D (2013) Recurrent neural networks for language understanding. In: Interspeech, pp 2524–2528
64. Yoo KM, Shin Y, Lee Sg (2019) Data augmentation for spoken language understanding via joint variational generation. Proceedings of the AAAI conference on artificial intelligence 33:7402–7409
65. Yu S, Shen L, Zhu P, Chen J (2018) ACJIS: A novel attentive cross approach for joint intent detection and slot filling. In: 2018 International joint conference on neural networks (IJCNN). IEEE, pp 1–7
66. yuanxiaosc: Smp2018. Github (2018). https://github.com/yuanxiaosc/SMP2018. Accessed 14 Oct 2019
67. Zhang X, Wang H (2016) A joint model of intent determination and slot filling for spoken language understanding. IJCAI 16:2993–2999
68. Zhu S, Yu K (2017) Encoder-decoder with focus-mechanism for sequence labelling based spoken language understanding. In: 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 5675–5679