Li *et al. Urban Informatics* (2024) 3:24
https://doi.org/10.1007/s44212-024-00056-6

Urban Informatics

ORIGINAL ARTICLE

Open Access

A novel model based on a transformer for intent detection and slot filling

Dapeng Li[1], Shuliang Wang[2*], Boxiang Zhao[2], Zhiqiang Ma[1] and Leixiao Li[1]

Abstract

Building task-oriented dialogue systems has become a topic of interest in the research community and industry. The task-oriented dialogue system is a closed-domain dialogue system that can perform specific tasks for users. The natural language understanding module of a task-oriented dialogue system is crucial because it is related to a task-oriented dialogue system that provides correctional services for users. The natural language understanding module of a task-oriented dialogue system performs two tasks: intent detection and slot filling. The intent detection task can be regarded as a text classification task; a classification model is trained to predict the intention of the user from the user's input information. The slot filling task can be regarded as a sequence analysis task; a sequence analysis model is trained to predict the details of the user's intention. In this paper, we proposed a novel model based on a transformer encoder for intent detection and slot filling. It follows the encoder-decoder structure, including a vanilla Transformer encoder, a bidirectional LSTM encoder, a linear classification decoder for intent detection, and a conditional random field decoder for slot filling. The experimental results on two public datasets show that our proposed model outperforms the existing methods based on the Transformer and can be combined with BERT to achieve better intent detection and slot filling results.

Keywords Dialogue system, Intent detection, Slot filling, Transformer, BERT

1 Introduction

Large language models are increasingly being utilized in various fields, including urban informatics, as demonstrated by CityGPT. The task-oriented (also referred to as goal-oriented) dialogue system, as part of the urban large language model, has become a topic of interest in the research community and industry (Zhang et al., 2020a, 2020b). Unlike chatbots (Wang et al., 2019, 2020), a task-oriented dialogue system is a closed-domain dialogue system (Gao et al., 2021; Mi et al., 2021) that can perform specific tasks for users, such as querying information, ordering products online, and playing music. Representative products include Siri and Cortana. The task-oriented dialogue system includes modules such as natural language understanding (Liu et al., 2021), dialogue management (Takanobu et al., 2019, 2020; Zhang et al., 2019a, 2019b), and natural language generation (Mi et al., 2019, 2020). The natural language understanding module is crucial because it is related to a task-oriented dialogue system providing correctional services for users.

The natural language understanding module of a task-oriented dialogue system performs two tasks (Wang et al., 2023): intent detection and slot filling. The intent detection task can be regarded as a text classification task; the classification model is trained to predict the intention of the user from the user's input information. Table 1 shows an example of a user asking about weather conditions (Intent: GetWeather) in the SNIPS (Coucke et al., 2018) corpus. The slot filling task can be regarded as a sequence analysis task; the sequence analysis model is trained to

*Correspondence:
Shuliang Wang
slwang2011@bit.edu.cn
[1] College of Data Science and Application, Inner Mongolia University of Technology, Hohhot 010080, China
[2] School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China

Table 1 An example of the SNIPS corpus

Sentence	What	will	the	weather	be	here	?
Slots	O	O	O	O	O	B-current_location	O
Intent	GetWeather						

predict the details of the user's intention. Table 1 shows an example of a user asking about the weather conditions in their location (Slot: B-current_location) in the SNIPS (Coucke et al., 2018) corpus. The natural language understanding module performs these two tasks to obtain the specific needs of users for a task-oriented dialogue system. Wang et al. (2018) found that for models based on deep learning, if these two tasks cooperate with each other, the accuracy of the task-oriented dialogue system to obtain user requirements can be improved.

Recently, transformer research has emerged in the field of natural language processing, and some transformer-based models for intent detection and slot filling have been proposed (Qin et al., 2021; Wang et al., 2021). Although the vanilla Transformer can handle text classification tasks or sequence analysis tasks, it has difficulty accomplishing these two tasks at the same time. Therefore, there are two solutions to this problem: one is to modify the vanilla Transformer to better handle the text classification task and sequence analysis task simultaneously (Wang et al., 2021), and the other is to combine the vanilla Transformer with other methods to build a model that can better handle the text classification task and sequence analysis task simultaneously (Qin et al., 2021). We choose the second solution to build our model. Inspired by the TRANS-BLSTM method (Huang et al., 2020), we integrate a vanilla Transformer with bidirectional LSTM (BiLSTM) as the encoder and a linear classification decoder for intent detection with a conditional random field (CRF) as the slot filling decoder and propose the TLC method. TLC stands for Transformer, LSTM and CRF, which are indispensable in our method. We will further explain this in the model ablation analysis. In addition, we add a residual learning module to our model. The experimental results show that the residual learning module is effective in improving the slot filling effect of our model. However, compared with ResNET (He et al., 2016) in the computer vision research field, TLC cannot improve the effect of intent detection and slot filling through a large-scale overlay of neural network layers. Qin et al. (2021) also found the same problem in their research. We will discuss this problem in the parameter tuning analysis section.

Our contributions in this paper are (1) the proposal of a new transformer-based model for intent detection and slot filling and (2) empirical verification of the effectiveness of our proposed model on two public datasets.

2 Related work

In the past, when statistical learning methods dominated natural language processing research, intent detection and slot filling were regarded as two independent tasks. The support vector machine (SVM) and AdaBoost algorithms had good results for the intent detection task and the conditional random field (CRF) dominated the slot filling task (Mesnil et al., 2013). With the advent of the deep learning era, methods of intent detection and slot filling based on deep learning have become mainstream, such as the Joint Seq model (Hakkani-Tür et al., 2016) based on BiLSTM. Liu and Lane (2016) added an attention mechanism to BiLSTM and proposed the attention BiRNN model. Zhu and Yu (2017) presented the focus mechanism and applied it to the encoder-decoder structure. Goo et al. (2018) added a slot gating mechanism to the attention BiRNN and proposed the slot-gated attention model. Li et al. (2018) added a gating mechanism to their model and proposed the self-attentive model. Wang et al. (2018) recognized the interaction between the intent detection task and slot filling task and proposed the Bi-Model. Zhang et al., (2019a, 2019b) applied the capsule network for intent detection and slot filling and proposed the CAPSULE-NLU model. The SF-ID Network (E et al., 2019) and the CM-Net (Liu et al., 2019) have contributed to improving the interaction and promotion between the intent detection task and the slot filling task. Qin et al. (2019) proposed the stack propagation model, which can effectively improve the performance of intent detection and further alleviate error propagation by adding word-level intent detection, thereby better combining intent information for slot filling.

With the development of pretraining technology, pretraining language models have begun to be used for intent detection and slot filling. Siddhant et al. (2019) used the ELMo (Peters et al., 2018) model as a representation learning method, combined BiLSTM with CRF, and improved the baseline method of intent detection and slot filling. Chen et al. (2019) applied the BERT (Devlin et al., 2019) model to intent detection and slot filling and proposed the JointBERT model. Furthermore, with the rise of graph neural network research, methods of intent detection and slot filling based on graph neural

networks have been proposed, such as the graph LSTM model (Zhang et al., 2020a, 2020b). Recently, Transformer-based methods for intent detection and slot filling have sparked interest in the research community. The Co-Interactive Transformer (Qin et al., 2021) and the SyntacticTF (Wang et al., 2021) are the latest methods developed for intent detection and slot filling based on the Transformer. In addition, Gunaratna et al. (2022) proposed a joint NLU model based on BERT that can improve the slot explanation ability while improving the effect of intent detection and slot filling.

3 Proposed model

We followed current mainstream approaches and regarded intent detection and slot filling as interrelated tasks.

3.1 Problem Formalization

The tasks of intent detection and slot filling can be formalized as Eqs. (1), (2) and (3):

$$y^{intent} = \sigma\left(W^{intent} h_1 + b^{intent}\right) \tag{1}$$

$$y_n^{slot} = \sigma\left(W^{slot} h_n + b^{slot}\right) \tag{2}$$

$$P\left(y^{intent}, y_n^{slot}|x\right) = P\left(y^{intent}|x\right)\prod_{n=1}^{N} P\left(y_n^{slot}|x\right) \tag{3}$$

where y^{intent} represents the user's intention; y_n^{slot} represents the slot value for the user's input information; h_1 and h_n represent the hidden vectors of the user input information in the neural network; $n \in [1, N]$. W^{intent}, b^{intent}, W^{slot} and b^{slot} are the neural network parameters; and σ represents the activation function. The goal of this task is to train the neural network model to predict the correct user intention y^{intent} and slot value y_n^{slot} according to user input information x.

3.2 Model Overview

Our proposed model follows the encoder-decoder structure. The encoder of the TLC model includes two parts: the first part is a vanilla Transformer encoder, and the second part is a bidirectional LSTM (BiLSTM) encoder. We add a residual connection between the Transformer encoder and the BiLSTM encoder. This residual connection plays a key role in promoting the slot filling effect of our model. The decoder of the TLC model includes two parts: a linear classification decoder for intent detection and a CRF decoder for slot filling. The architecture of the proposed TLC model is shown in Fig. 1.

3.3 Encoder

As shown in Fig. 1, the first step of TLC model training requires representation learning, which includes word embedding and positional embedding. The word embedding of the TLC model uses the GloVe (Pennington et al., 2014) method. The positional embedding of the TLC model is a method proposed by Vaswani et al. (2017). We will analyse two kinds of positional embedding methods (Vaswani et al., 2017) in the parameter tuning analysis section. After word embedding and positional embedding, our proposed model performs Add and Dropout (Hinton et al., 2012) operations on the outputs of word embedding and positional embedding, which can be defined as Eq. (4):

$$S = Dropout(P + E) \tag{4}$$

where P represents the output of positional embedding, and E represents the output of word embedding. Then, S is input into the Transformer encoder of the TLC model.

The Transformer encoder is the first encoder of the TLC model. Although a complete Transformer model includes an encoder and a decoder, we use only a Transformer encoder in our proposed model. The first step is to map S to Query, Key and Value and then process it through the multihead attention mechanism. This step can be defined as Eqs. (5), (6) and (7):

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{6}$$

$$H^O = Concat(head_1, \ldots, head_h)W^O \tag{7}$$

where Q represents Query; K represents Key; V represents Value; $1/\sqrt{(d_k)}$ is the scaling factor; and$W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$,$W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$ and $W^O \in \mathbb{R}^{hd_v \times d_{model}}$ are the parameter matrices of linear mapping. Then, H^O is subjected to the Add and LayerNorm (LN) (Ba et al., 2016) operations and input into the feed-forward network (FFN), which can be defined as Eqs. (8) and (9):

$$H^{L1} = LN\left(S + H^O\right) \tag{8}$$

$$FFN\left(H^{L1}\right) = max\left(0, H^{L1}W_1 + b_1\right)W_2 + b_2 \tag{9}$$

where $W_1, b_1, W_2, and b_2$ are parameters of the FFN. Before completing the Transformer encoding, another Add and LayerNorm (LN) (Ba et al., 2016) operation is performed, which can be defined as Eq. (10):

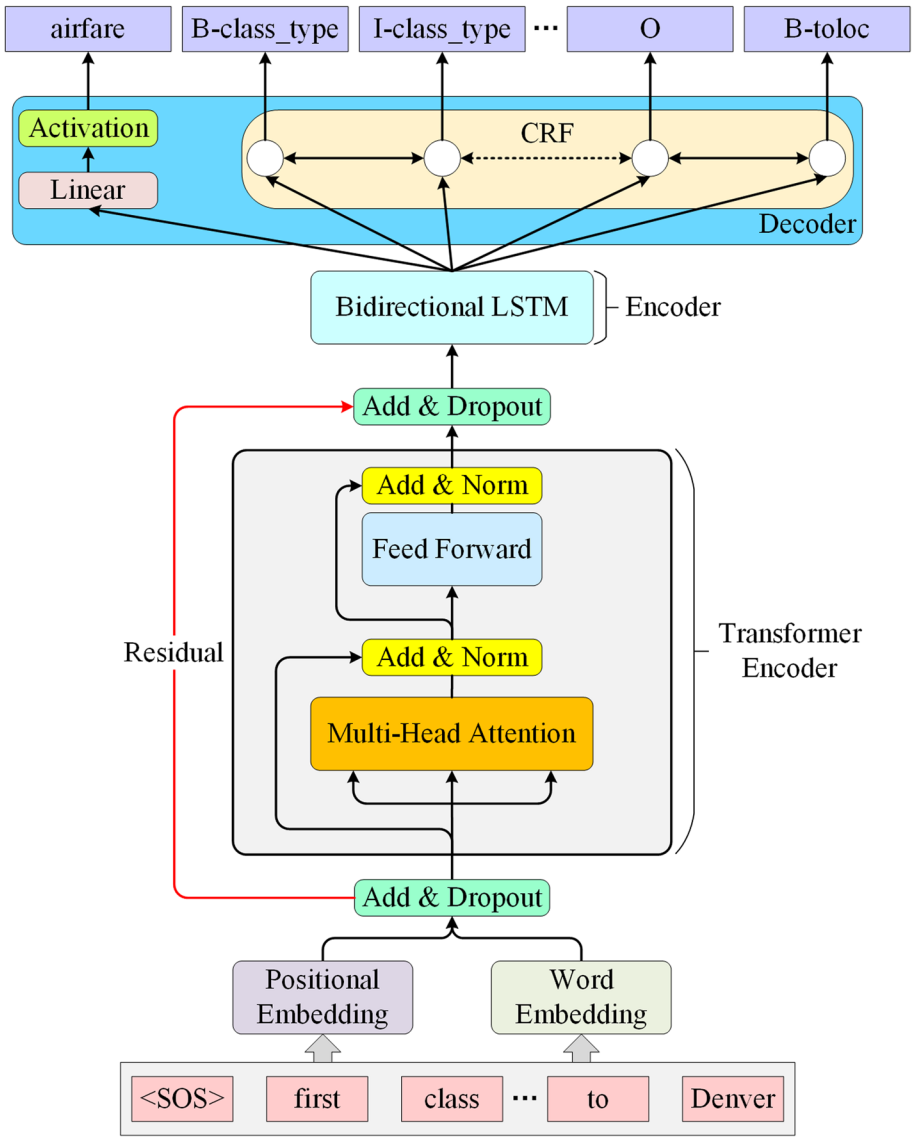

Fig. 1 Architecture of the TLC

$$H^{L2} = LN\left(H^{L1} + FFN\left(H^{L1}\right)\right) \tag{10}$$

Furthermore, before the output of the Transformer is input into the BiLSTM encoder, it is necessary to perform the Add and Dropout (Hinton et al., 2012) operation between S and H^{L2}. Although this process is simple, it is necessary to improve the slot filling effect of the TLC model. This process can be defined as Eq. (11):

$$X = Dropout\left(S + H^{L2}\right) \tag{11}$$

Then, X is expressed as $X = (x_1, x_2, \ldots, x_t)$ and input into the BiLSTM for the second encoding, which can be defined as Eqs. (12), (13) and (14):

$$\overleftarrow{h_t} = LSTM_{bw}\left(\overleftarrow{h_{t+1}}, x_t\right) \tag{12}$$

$$\overrightarrow{h_t} = LSTM_{fw}\left(\overrightarrow{h_{t-1}}, x_t\right) \tag{13}$$

$$H = \left[\overleftarrow{h_t}, \overrightarrow{h_t}\right] \tag{14}$$

where h_t represents the hidden state of the LSTM at time step t, $\overleftarrow{h_t}$ represents the hidden state of the LSTM calculated from back to front at time step t, $LSTM_{bw}$ represents the LSTM function from back to front, $\overrightarrow{h_t}$ represents the hidden state of the LSTM calculated from front to back at time step t, $LSTM_{fw}$ represents the LSTM

function from front to back, and x_t represents the t-th token in X. Finally, after encoding the BiLSTM, the result H is input to the decoder of the TLC model.

3.4 Decoder

The decoder of the TLC model includes two parts: a linear classifier for intent detection and a CRF for slot filling. Since intent detection and slot filling are different types of tasks, the output result H from the BiLSTM encoder needs to be extracted according to the nature of the different tasks. We extract the output of the hidden unit at the last state from H, record it as H^{intent} for intent detection, and input it into the neural network for linear classification, which can be defined as Eq. (15):

$$y^{intent} = \sigma\left(W^{intent}H^{intent} + b^{intent}\right) \tag{15}$$

where σ is the LogSoftmax activation function, and W^{intent} and b^{intent} are neural network parameters.

H is the output of the hidden state of the BiLSTM encoder at all time steps, so it can be directly used for slot filling. We followed the method of Qin et al. (2021) to apply a CRF for the slot filling task in the decoder of our method, which can be defined as Eqs. (16) and (17):

$$C^{slot} = W^{slot}H + b^{slot} \tag{16}$$

$$P\left(y^{slot}|C^{slot}\right) = \frac{\sum_{i=1} expf\left(y_{i-1}, y_i, C^{slot}\right)}{\sum_{y'}\sum_{i=1} expf\left(y_{i-1}', y_i', C^{slot}\right)} \tag{17}$$

where W^{slot} and b^{slot} are training parameters, y_i' is the slot label, and $f\left(y_{i-1}, y_i, C^{slot}\right)$ is responsible for calculating the label score of y_i and the score of transition from y_{i-1} to y_i.

Finally, both the intent detection task and the slot filling task use the negative log likelihood loss (NLLLOSS) function as the loss function for the training of the TLC model. The loss function of the joint training can be defined as Eq. (18):

$$L^{joint} = \alpha L^{intent} + (1-\alpha)L^{slot} \tag{18}$$

where L^{joint} is the loss function of the entire TLC model, L^{intent} is the loss function for the intent detection task, L^{slot} is the loss function for the slot filling task, and the hyperparameter α is used to control the balance between these two tasks during training.

4 Experiments

To test our proposed TLC model for intent detection and slot filling, we choose the SNIPS (Coucke et al., 2018) corpus and the ATIS (Hemphill et al., 1990; Tur et al., 2010) corpus for experiments.

4.1 Datasets and evaluation metrics

The SNIPS corpus is a task-oriented dialogue system corpus collected by the French company SNIPS. It is mainly used to design voice assistants for dialogue systems. The full name of the ATIS corpus is the Air Travel Information System corpus. It is a corpus collected through the Official Airline Guide (OAG, 1990) that contains professional information such as airline bookings, travel, and consultations. It is a commonly used dataset for the evaluation of intention detection and slot filling in task-oriented dialogue systems. Although the SNIPS corpus has fewer types of intent and slots than the ATIS corpus, it has more training data. The statistics of the SNIPS corpus and the ATIS corpus are shown in Table 2.

In the evaluation of the experimental results of intent detection and slot filling, we choose accuracy as the evaluation metric for the intent detection task and the F1 value as the evaluation metric for the slot filling task.

4.2 Experimental Settings

We use the PyTorch (Paszke et al., 2019) deep learning framework to build the TLC model, and all experiments are performed on a single GeForce GTX 1080 Ti GPU. Following the method of Wang et al. (2021) in the word embedding part of our model, the GloVe (Pennington et al., 2014) method is used for word-level embedding, and the Kazuma (Hashimoto et al., 2017) character-level embedding method is used as a supplement. The representation learning dimensions of these methods are 300 and 100, respectively. In the positional embedding part of our model, we choose the learned positional embedding method. The Transformer encoder layer of our model is 2, the Transformer encoder dimension is 400, the number of heads of the multihead attention mechanism is 10, the dimension of the feedforward network is 2048, and the activation function is GELU. The BiLSTM encoder layer of our model is 2, and the hidden size of each LSTM is 200. The batch size for training is 32, and the maximum number of epochs is 200. The learning rate is 0.0001, and the dropout rate is 0.1. The Adam (Kingma and Ba, 2015) method is used as the training optimizer; β_1 and β_2 are set to 0.9 and 0.999, respectively; $\epsilon = 10^{-8}$; and the weight decay is 0. The gradient clipping method is used to prevent overfitting during training, the maximum norm of the gradient is set to 1, and the type of norm is L2. The hyperparameter α is 0.5.

Table 2 Statistics of the SNIPS and ATIS datasets

Dataset	Train	Val	Test	Intents	Slots
SNIPS	13,084	700	700	7	72
ATIS	4478	500	893	18	130

4.3 Experimental Results

We use 12 models as the baseline methods for intent detection and slot filling experiments. These 12 models include Joint Seq (Hakkani-Tür et al., 2016), Slot-Gated Atten (Goo et al., 2018), Self-Attentive Model (Li et al., 2018), Bi-Model (Wang et al., 2018), CAPSULE-NLU (Zhang et al., 2019a, 2019b), SF-ID Network (E et al., 2019), CM-Net (Liu et al., 2019), Stack-Propagation (Qin et al., 2019), JointBERT (Chen et al., 2019), Graph LSTM (Zhang et al., 2020a, 2020b), Co-Interactive Transformer (Qin et al., 2021), SyntacticTF (Wang et al., 2021). The characteristics of these 12 models have been introduced in related studies. It should be noted that the Co-Interactive Transformer (Qin et al., 2021), the SyntacticTF (Wang et al., 2021) and our proposed TLC are models based on the Transformer encoder. In addition, the experimental results of these 12 models are from published papers (Hakkani-Tür et al., 2016; Goo et al., 2018; Li et al., 2018; Wang et al., 2018; Zhang et al., 2019a, 2019b; E et al., 2019; Liu et al., 2019; Qin et al., 2019; Chen et al., 2019; Zhang et al., 2020a, 2020b; Qin et al., 2021; Wang et al., 2021). The experimental results are shown in Table 3. The experimental results of the two datasets show that our proposed TLC model is a better model for intent detection and slot filling. On the SNIPS corpus, the slot filling F1 value of our model is 0.36% higher than that of SyntacticTF (Wang et al., 2021). The intent detection accuracy of our model is 0.15% higher than that of SyntacticTF (Wang et al., 2021). On the ATIS corpus, the slot filling F1 value of our model is 0.09% higher than that of SyntacticTF (Wang et al., 2021), and the accuracy of intent detection of our model is 0.47% higher than that of the Co-Interactive Transformer (Qin et al., 2021). The experimental results show that our proposed TLC model outperforms the previously proposed models based on the Transformer encoder.

Table 3 Results of intent detection and slot filling on the SNIPS and ATIS datasets

Model	SNIPS		ATIS	
	Slot (F1)	Intent (Acc)	Slot (F1)	Intent (Acc)
Joint Seq	87.30	96.90	94.30	92.60
Slot-Gated Atten	88.80	97.00	94.80	93.60
Self-Attentive Model	90.00	97.50	95.10	96.80
Bi-Model	93.50	97.20	95.50	96.40
CAPSULE-NLU	91.80	97.30	95.20	95.00
SF-ID Network	90.50	97.00	95.60	96.60
CM-Net	93.40	98.00	95.60	96.10
Stack-Propagation	94.20	98.00	95.90	96.90
JointBERT	97.00	98.60	96.10	97.50
Graph LSTM	95.30	98.29	95.91	97.20
Co-Interactive Transformer	95.90	98.80	95.90	97.70
SyntacticTF	96.89	99.14	96.01	97.31
TLC (ours)	97.25	99.29	96.10	98.17

5 Discussion

We conduct model ablation analysis and parameter tuning analysis of the TLC model. In addition, we combine the TLC model with BERT.

5.1 Ablation Study

We conduct a model ablation analysis of our proposed model, and the experimental results are shown in Table 4.

Table 4 shows that when we remove the residual learning module, BiLSTM or CRF from the TLC model, the effect of the TLC model will decrease. It is worth noting that the residual learning module removed here is the newly added residual learning of our proposed model, which is the red line in Fig. 1, rather than the residual connection between the internal layers of the vanilla Transformer. When the residual learning module of the TLC model is removed, the slot filling effect of the model is reduced on both datasets. When only BiLSTM is removed from the TLC model, the slot filling effect of the model is reduced on both datasets. When only the CRF is removed from the TLC model, only the slot filling effect on the SNIPS corpus decreases, and the intent detection effect increases. However, considering that it is difficult to achieve a better slot filling effect on the SNIPS corpus, we choose CRF as the slot filling decoder for our proposed model. When Residual Learning, BiLSTM, and CRF are removed at the same time, the effect of our proposed model decreases significantly. This means that it is difficult for the model to complete the intent detection task and slot filling task with vanilla Transformer at the same time. This also further illustrates the vanilla Transformer combined with BiLSTM and CRF in our proposed model is indispensable for intent detection task and slot filling task.

Table 4 Results of the ablation study on the SNIPS and ATIS datasets

Model	SNIPS		ATIS	
	Slot (F1)	Intent (Acc)	Slot (F1)	Intent (Acc)
Without Residual Learning	97.00	99.43	95.94	**98.23**
Without BiLSTM	89.71	99.29	94.34	97.79
Without CRF	96.02	99.57	96.11	98.17
Only vanilla Transformer	87.60	98.86	93.89	97.36
TLC	97.25	99.29	96.10	98.17

5.2 Parameter Tuning Analysis

The Transformer encoder is the core of the TLC model. Therefore, we analyse the parameters of the Transformer encoder in the TLC model. First, because positional embedding plays an important role in the Transformer model, we adjust and analyse the positional embedding method of our proposed model. The experimental results are shown in Table 5. When the TLC model does not use positional embedding, the effect of the TLC model decreases. Therefore, choosing a suitable positional embedding method is vital for our proposed model. Vaswani et al. (2017) proposed sinusoidal positional encoding and learned positional embedding for the Transformer model and found that the effects of these two methods were basically the same in machine translation experiments. However, as shown in Table 5, in the intent detection and slot filling experiments, the sinusoidal positional encoding method has a better intent detection effect, while the learned positional embedding method has a better slot filling effect. Since it is more difficult to achieve a better slot filling effect compared with the baseline method, we choose the learned positional embedding method as the positional embedding method of the Transformer encoder in our proposed model.

Second, it is a common practice to use a multilayer transformer in natural language processing tasks. Therefore, whether the effect of the TLC model can be improved by increasing the number of transformer layers is worthy of further study. We adjust the number of transformer encoder layers in our proposed model, and the experimental results are shown in Table 6. When a 2-layer transformer encoder is used in the model, the experimental effect of our proposed model is the best. In addition, the Co-Interactive Transformer (Qin et al., 2021) and SyntacticTF (Wang et al., 2021) both use a 2-layer Transformer encoder. Therefore, when using the transformer encoder for intent detection and slot filling, we choose 2 as the parameter of the Transformer encoder in our proposed model.

5.3 Combination with BERT

The BERT (Devlin et al., 2019) model, a landmark in the field of natural language processing, excels at handling various natural language processing tasks and can be combined with other methods to achieve better results.

Table 6 Results of the transformer encoder layer tuning experiment

Transformer Encoder Layer	SNIPS		ATIS	
	Slot (F1)	Intent (Acc)	Slot (F1)	Intent (Acc)
1	96.87	99.29	95.86	98.45
2	97.25	99.29	96.10	98.17
3	96.87	99.14	95.96	98.28

The joint NLU (Gunaratna et al., 2022) is a model based on BERT. Qin et al. (2021) used the Co-Interactive Transformer with BERT to achieve better intent detection and slot filling results. Therefore, we combine the TLC model with BERT for intent detection and slot filling. In these experiments, we remove the word embedding and positional embedding in the TLC model and combine the rest of the TLC model with BERT. The number of heads of the multihead attention mechanism in the transformer is adjusted from 10 to 16, and the dimension of the feedforward network is adjusted from 2048 to 1024. The Transformers (Wolf et al., 2020) tool is used to call the BERT for combining with the TLC model. Since the SNIPS corpus is a cased corpus, the base-cased version of BERT is selected. The ATIS corpus is an uncased corpus, so the base-uncased version of BERT is selected. The learning rate is changed from 0.0001 to 0.00005, the max epoch is changed from 200 to 100, and BERTAdam, which is an improved Adam (Kingma and Ba, 2015) optimization method for BERT, is used as the training optimizer. The hyperparameter α of the experiment on the SNIPS corpus remains 0.5, while the hyperparameter α of the experiment on the ATIS corpus is changed to 0.7. Other model parameters and training settings remain unchanged. We follow the method of Qin et al. (2021) and choose stack propagation (Qin et al., 2019) and the Co-Interactive Transformer (Qin et al., 2021) as the research comparison methods. In addition, Wang et al. (2021) believed that BERT and Transformer belong to different technical routes and did not combine SyntacticTF with BERT to study intent detection and slot filling. Therefore, we did not choose SyntacticTF as the comparison method. The experimental results of the combination of the TLC model with BERT are shown in Table 7.

Table 5 Results of the positional embedding adjustment experiment

Method	SNIPS		ATIS	
	Slot (F1)	Intent (Acc)	Slot (F1)	Intent (Acc)
Without Positional Embedding	96.95	99.43	96.07	97.95
Sinusoidal Positional Encoding	97.12	99.43	95.75	98.22
Learned Positional Embedding	97.25	99.29	96.10	98.17

Table 7 Results of TLC combined with BERT on the SNIPS and ATIS datasets

Model	SNIPS		ATIS	
	Slot (F1)	Intent (Acc)	Slot (F1)	Intent (Acc)
Stack-Propagation	94.20	98.00	95.90	96.90
Stack-Propagation + BERT	97.00	99.00	96.10	97.50
Co-Interactive Transformer	95.90	98.80	95.90	97.70
Co-Interactive Transformer + BERT	97.10	98.80	96.10	98.00
joint NLU (based on BERT)	97.24	98.99	96.20	99.10
TLC (ours)	97.25	99.29	96.10	98.17
TLC + BERT (ours)	97.36	99.57	96.32	98.39

As shown in Table 7, BERT can enhance the effect of the TLC model for intent detection and slot filling. When the TLC model is combined with BERT, our model outperforms all the comparison methods on the SNIPS corpus. The slot filling F1 value of our proposed TLC model is 0.12 higher than that of the joint NLU (Gunaratna et al., 2022) on the ATIS corpus. Although the accuracy of intent detection of our proposed TLC model is lower than that of the joint NLU (Gunaratna et al., 2022) model on the ATIS corpus, our model performs better than the joint NLU model on both the SNIPS and the ATIS corpus.

6 Conclusion

In this paper, we propose a novel model based on transformers for intent detection and slot filling. The experimental results show that the proposed method can achieve higher intent detection accuracy and slot filling F1 values than the existing Transformer-based methods. In addition, our proposed model can be combined with BERT to achieve better experimental results of intent detection and slot filling. In the future, we will verify our model on other datasets.

Acknowledgements
We appreciate the advice of Prof. Minlie Huang.

Authors' contributions
Conceptualization: Dapeng Li; Methodology: Dapeng Li; Formal analysis and investigation: Dapeng Li, Shuliang Wang, Boxiang Zhao, Zhiqiang Ma, Leixiao Li; Writing—original draft preparation: Dapeng Li, Shuliang Wang, Boxiang Zhao, Zhiqiang Ma, Leixiao Li; Writing—review and editing: Dapeng Li, Shuliang Wang, Boxiang Zhao, Zhiqiang Ma, Leixiao Li; Funding acquisition: Shuliang Wang, Dapeng Li, Zhiqiang Ma; Supervision: Shuliang Wang.

Funding
This work was supported by the Natural Science Foundation of Inner Mongolia (2023QN06010), the National Natural Science Foundation of China (62076027), and Research Projects of Universities in Inner Mongolia Autonomous Region (JY20220268).

Availability of data and materials
The datasets generated and trained models during the current study are available from the corresponding author on reasonable request.
The datasets generated and models trained during the current study are available from the corresponding author upon reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no conflict of interest.

Received: 11 August 2023 Revised: 1 July 2024 Accepted: 8 August 2024
Published online: 14 August 2024

References

Ba, J., Kiros, J., & Hinton, G. (2016). *Layer Normalization. Arxiv Preprint arXiv, 1607*, 06450.
Chen, Q., Zhuo, Z., Wang, W. (2019). BERT for joint intent classification and slot filling. *arXiv preprint arXiv*: 1902.10909
Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., Primet, M., Dureau, J. (2018). Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv*: 1805.10190.
Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv, 1*, 4171–4186.
E, H., Niu, P., Chen, Z., Song, M.: 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. ACL, Florence, Italy (2019).
Gao, S., Takanobu, R., Peng, W., Liu, Q., Huang, M.: HyKnow: end-to-end task-oriented dialog modeling with hybrid knowledge management. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 1591–1602, ACL, Online (2021).
Goo, C., Gao G., Hsu, Y., Huo, C., Chen, T., Hsu, K., Chen, Y.: 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, pp. 753–757. ACL, New Orleans, Louisiana (2018).
Gunaratna, K., Srinivasan, V., Yerukola, A., Jin, H.: Explainable slot type attentions to improve joint intent detection and slot filling. In: Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 3367–3378, ACL, Abu Dhabi, United Arab Emirates (2022).
Hakkani-Tür, D., Tur, G., Celikyilmaz, A., Chen, Y.V., Gao, J., Deng, L., Wang, Y.: Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM. In: Proc. Interspeech 2016, pp. 715–719. ISCA, San Francisco, USA (2016).
Hashimoto, K., Xiong, C., Tsuruoka, Y., Socher, R.: A joint many-task model: growing a neural network for multiple NLP tasks. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1923–1933. ACL, Copenhagen, Denmark (2017).

He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognitions, pp. 770–778. IEEE, Las Vegas, NV, USA (2016).
Hemphill, T., Godfrey, J., Doddington, G.: The ATIS spoken language systems pilot corpus. In: Proceedings of the DARPA Speech and Natural Language Workshop, pp. 96–101, Morgan Kaufmann, Hidden Valley, PA, USA (1990).
Hinton, G., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv*: 1207.0580
Huang, Z., Xu, P., Liang, D., Mishra, A., Xiang, B. (2020). TRANS-BLSTM: transformer with bidirectional LSTM for language understanding. *arXiv preprint arXiv*: 2003.07000
Kingma, D.P., Ba, J. Adam. (2015). a method for stochastic optimization. In: *Third International Conference on Learning Representations*, San Diego, CA, USA
Li, C., Li, L., Qi, J.: A self-attentive model with gate mechanism for spoken language understanding. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 3824–3833. ACL, Brussels, Belgium (2018).
Liu, B., Lane, I.: Attention-based recurrent neural network models for joint intent detection and slot filling. In: Proc. Interspeech 2016, pp. 685–689, ISCA, San Francisco, CA, USA (2016).
Liu, J., Takanobu, R., Wen, J., Wan, D., Li, H., Nie, W., Li, C., Peng, W., Huang, M.: Robustness testing of language understanding in task-oriented dialog. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 2467–2480, ACL, Online (2021).
Liu, Y., Meng, F., Zhang, J., Zhou, J., Chen, Y., Xu, J.: CM-Net: a novel collaborative memory network for spoken language understanding. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pp. 1051–1060. ACL, Hong Kong, China (2019).
Mesnil, G., He, X., Deng, L., Bengio, Y.: Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In: Proc. Interspeech 2013, pp. 3771–3775, ISCA, Lyon, France (2013).
Mi, F., Chen, L., Zhao, M., Huang, M., Faltings, B.: Continual learning for natural language generation in task-oriented dialog systems. In: Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 3461–3474, ACL, Online (2020).
Mi, F., Huang, M., Zhang, J., Faltings, B.: Meta-learning for low-resource natural language generation in task-oriented dialogue systems. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19), pp. 3151–3157, International Joint Conferences on Artificial Intelligence Organization, Macao, China (2019).
Mi, F., Zhou, W., Cai, F., Kong, L., Huang, M., Faltings, B.: Self-training improves pre-training for few-shot learning in task-oriented dialog systems. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 1887–1898, ACL, Online and Punta Cana, Dominican Republic (2021).
Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: an imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8026–8037. Curran Associates, Vancouver, Canada (2019).
Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. ACL, Doha, Qatar (2014).
Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 2227–2237, ACL, New Orleans, Louisiana, USA (2018).
Qin, L., Che, W., Li, Y., Wen, H., Liu, T.: A stack-propagation framework with token-level intent detection for spoken language understanding. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pp. 2078–2087. ACL, Hong Kong, China (2019).
Qin, L., Liu, T., Che, W., Kang, B., Zhao, S., Liu, T.: A co-interactive transformer for joint slot filling and intent detection. In: 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8193–8197. IEEE, Virtual Conference (2021).
Siddhant, A, Goyal, A., Metallinou, A.: Unsupervised transfer learning for spoken language understanding in intelligent agents. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence, pp. 4959–4966. AAAI Press, Honolulu, Hawaii, USA (2019).
Takanobu, R., Liang, R., Huang, M.: Multi-agent task-oriented dialog policy learning with role-aware reward decomposition. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 625–638, ACL, Online (2020).
Takanobu, R., Zhu, H., Huang, M.: Guided dialog policy learning: reward estimation for multi-domain task-oriented dialog. 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. 100–110, ACL, Hong Kong, China (2019).
Tur, G., Hakkani-Tür, D., Heck, L.: What is left to be understood in ATIS? In: 2010 IEEE Spoken Language Technology Workshop, pp. 19–24. IEEE, Berkeley, CA, USA (2010).
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 6000–6010. Curran Associates, Long Beach, California, USA (2017).
Wang, H., Yang, D., Guo, L., & Zhang, X. (2023). Joint modeling method of question intent detection and slot filling for domain-oriented question answering system. *Data Technologies and Applications*. https://doi.org/10.1108/DTA-07-2022-0281
Wang, J., Wei, K., Radfar, M., Zhang, W., Chung, C.: Encoding syntactic knowledge in transformer encoder for intent detection and slot filling. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence, pp. 13943–13951. AAAI Press, Virtual Conference (2021).
Wang, S., Li, D., Geng, J., Yang, L., & Dai, T. (2019). Learning bi-utterance for multi-turn response selection in retrieval-based chatbots. *International Journal of Advanced Robotic Systems, 16*(2), 1–10.
Wang, S., Li, D., Geng, J., Yang, L., & Leng, H. (2020). Learning to balance the coherence and diversity of response generation in generation-based chatbots. *International Journal of Advanced Robotic Systems, 17*(4), 1–11.
Wang, Y., Shen, Y., Jin, H.: A bi-model based RNN semantic frame parsing model for intent detection and slot filling. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2, pp. 309–314. ACL, New Orleans, Louisiana (2018).
Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P. (2020). HuggingFace's Transformers: state-of-the-art natural language processing. *arXiv preprint arXiv*: 1910.03771
Zhang, C., Li, Y., Du, N., Fan, W., Yu, P.: Joint slot filling and intent detection via capsule neural networks. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 5259–5267. ACL, Florence, Italy (2019).
Zhang, L., Ma, D., Zhang, X., Yan, X., Wang, H.: Graph LSTM with context-gated mechanism for spoken language understanding. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence, pp. 9539–9546. AAAI Press, New York, USA (2020).
Zhang, Z., Huang, M., Zhao, Z., Ji, F., Chen, H., & Zhu, X. (2019b). Memory-augmented dialogue management for task-oriented dialogue systems. *ACM Trans. Inf. Syst., 37*(3), 1–30.
Zhang, Z., Takanobu, R., Zhu, Q., Huang, M., & Zhu, X. (2020b). Recent advances and challenges in task-oriented dialog systems. *Sci China Tech Sci, 63*, 2011–2027.
Zhu, S., Yu, K.: Encoder-decoder with focus-mechanism for sequence labelling based spoken language understanding. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5675–5679, IEEE, New Orleans, LA, USA (2017).

Publisher's Note