Multi-Domain Graph-Enhanced Joint Intent and Slot Learning

Wu Shi, Weihao Yuan,Yang Wang,Yindong Dong,and Gaojian Xu*

Abstract—**Comprehension of spoken language is essential in dialog systems, as it supports two fundamental tasks: intent classification and slot filling. At present, federated modeling methodologies prevail in the domain of spoken language comprehension. Nevertheless, current models encounter constraints in accurately representing the interrelations among activities and utilizing cross-domain semantic data. This study introduces the Multi-Domain Graph-Enhanced Joint Intent and Slot Learning (MDG-JISL) paradigm to tackle these difficulties. MDG-JISL amalgamates a pre-trained BERT model with a self-trained FastText model to create superior sentence-level and word-level representations. A syntactic dependency tree is employed to create graph structures among words, which are subsequently refined by the implementation of a Graph Convolutional Network (GCN) to more effectively capture relationship properties. A Conditional Random Field (CRF) model is utilized for decoding, enhancing the model's efficacy in natural language processing tasks. Experimental findings indicate that MDG-JISL attains remarkable performance on cross-domain datasets, with Slot(F1), Intent (Acc), and Overall (Acc) metrics of 98.7%, 98.4%, and 92.4%, respectively. The results validate the model's efficacy in intent classification and slot filling.**

Index Terms—**Graph Convolutional Networks, Intent Recognition and Slot Filling, BERT, Feature Fusion, Dependency Syntax Analysis.**

I. Introduction

Rapid advancements and breakthroughs in artificial intelligence have facilitated the pervasive use of Natural Language Processing (NLP) technologies in daily life, particularly in interactive systems such as chatbots and voice assistants. Notable instances include Apple's Siri [1], Microsoft's Cortana [2], and Baidu's Xiaodu. These sophisticated human-computer dialogue systems exemplify fundamental elements of artificial intelligence and human-computer interface technology. In these systems, Spoken Language Understanding (SLU) is essential for transforming users' natural language inputs into structured semantic data that computers can interpret, facilitating precise comprehension and replies to human communication.

SLU involves two core tasks: intent recognition and slot labeling. Intent recognition aims to determine the user's purpose from their input, typically framed as a task involving sentence categorization. On the other hand, slot labeling focuses on identifying critical information from the text and assigning it to predefined categories, where each word is classified according to its corresponding semantic role. Therefore, slot labeling is generally viewed as a sequence annotation task. The effectiveness of these two subtasks directly impacts the accuracy and overall performance of the spoken language understanding system, significantly improving the user experience. Table I delineates the intent and slot details for the phrase “查看大棚的温度” (Verify the temperature inside the shed).

TABLE I
SENTENCE ANALYSIS TABLE

sentence	查	看	大	棚	的	温	度
slot	O	O	B-Place	I-Place	O	B-Item	I-Item
Intent	Check the temperature						
Domain	Agriculture						

Traditional methods for the two sub-tasks of SLU often employ a distinct modeling approach. Intent classification encompasses conventional machine learning algorithms, including Support Vector Machine (SVM) [3] and Random Forest (RF) [4], alongside contemporary deep learning approaches such as Recurrent Neural Network (RNN) [5] and Bidirectional Long Short-Term Memory (BiLSTM) [6]. The primary methods for slot filling include the Hidden Markov Model (HMM) [7], Conditional Random Fields (CRF) [8], and Long Short-Term Memory (LSTM) networks [9]. Nevertheless, these algorithms frequently neglect the semantic relationships between the two tasks, failing to leverage shared information to improve overall performance [10]. Consequently, extensive research has aimed to integrate these activities into a unified framework. The initial efforts employed a three-layer CRF architecture that incorporated token features, slot labels, and intent labels, demonstrating the effectiveness of pipeline execution for these subtasks [11]. The emergence of deep learning has accelerated the development of collaborative models, leading to promising outcomes. Zhou et al. [12] presented a collaborative modeling approach utilizing a two-layer LSTM network. In this setup, the upper hidden layer corresponds to slot labels, while the lower layer represents intent labels. Additionally, they introduced a collaborative loss function exhibiting strong generalization properties. Zheng et al. [13] proposed employing a BiLSTM encoder-decoder for intent detection and semantic parsing in navigation dialogues, but distinct

Manuscript received July 12, 2024; revised December 21, 2024.

The work was supported by the Key Natural Science Research Projects in Universities in Anhui Province (Program No. 2022AH050896)

Wu Shi is an undergraduate at School of Information and Artificial Intelligence, Anhui Agricultural University, Hefei, 230036, China. (e-mail: WShi@stu.ahau.edu.cn).

Weihao Yuan is a postgraduate at School of Computer Science and Tech nology, Xidian University, Xi’an, 710126, China. (e-mail: 704843913@qq. com).

Yang Wang is an undergraduate at School of Engineering, Anhui Agricul tural University, Hefei, 230036, China. (e-mail: 2846078977@qq.com).

Yindong Dong is a lecturer at School of Information and Artificial Intelligence, Anhui Agricultural University, Hefei, 230036, China. (e-mail: dongyindong66@163.com).

Gaojian Xu is an associate professor at School of Information and Artificial Intelligence, Anhui Agricultural University, Hefei, 230036, China. (corresponding author, e-mail: xugj@ahau.edu.cn).

losses are utilized for intent and slots. Liu et al. [14] introduced an RNN-LSTM model that employs a single loss function for both tasks, but its ability to generalize is limited because there is no direct connection between the two processes. Firdaus et al. [15] introduced a transformer-based multilingual multitasking model that does intent detection and slot filling in three languages with a shared phrase encoder.

The recent integration of the pre-trained BERT model has led to significant advancements in SLU. Chen et al. [16] innovated the application of BERT for the simultaneous tasks of intent classification and slot filling, yielding substantial performance improvements. Guo et al. [17] further enhanced the approach by employing an attention mechanism to encode the hidden states of various sub-labels into context vectors.The context vectors are then input into the slot-filling encoder, addressing the problem of label length disparities caused by BERT's Word Piece implementation, which segments each input into many sub-labels.

Owing to significant advancements in deep learning methodologies and their ability to evaluate complex language models and comprehend human language, researchers are increasingly employing these techniques for more sophisticated tasks. Graph neural networks (GNNs) have attracted considerable attention for their capacity to model the structural and relational properties of data, making them especially suitable for tasks such as intent classification and slot filling, which demand a comprehensive understanding of language and contextual connections. He et al. [18] proposed a unified approach that applies a Graph Convolutional Network (GCN) on a dependency tree to integrate syntactic structures, enabling the simultaneous learning of both intent detection and slot filling. For sentence dependency analysis, Tang et al. [19] combined a GCN with grammatical structural information, representing words as nodes in a graph and annotated grammatical relationships as the connecting edges. Wei et al. [20] proposed the Wheel-network Attention Network, which creates a network of intent and slot nodes, leveraging an attention mechanism to enhance the flow of information between these nodes. The integration of intent nodes enables the acquisition of discourse-level semantic data, crucial for slot filling, while the inclusion of slot nodes aids in retrieving keyword information relevant to intent, hence improving the overall effectiveness of SLU.

A cross-domain model for slot filling and intent classification, named MDG-JISL, is proposed, leveraging the integration of associative information through a GCN. This work makes the following key contributions:

(1) Annotated several Chinese datasets for semantic slot filling and intent classification across diverse domains and trained corresponding FastText models for each annotated domain.

(2) In the fine-grained text embedding layer, sentence-level and word-level features are treated independently, facilitating a more nuanced treatment of each feature type.

(3) By constructing a collocation matrix derived from phrase dependencies, the detailed features are fed into a GCN, which adeptly captures the intricate links among words in a sentence. This method significantly improves the effectiveness of intent recognition and slot filling tasks. Sophisticated decoding skills from CRF are utilized, leading to enhanced intent recognition and slot extraction accuracy.

II. MDG-JISL

The MDG-JISL model is structured into three key components: the basic feature extraction module, the module that integrates contextual information for feature extraction, and the label representation module. The total structure is illustrated in Fig. 1. The model initially extracts sentence-level and word-level features independently, subsequently integrating them to establish the model's fundamental features. Subsequently, it analyzes the syntactic dependency structures of the sentences, constructs an adjacency matrix, and feeds this matrix along with the basic features into a graph convolutional network for feature representation. The derived feature vectors are fed into the output layer for decoding, resulting in the intent categorization of the input sentence and the identification of word slots.

A. Basic Feature Extraction Module

This module utilizes the BERT model to derive sentence-level representations, emphasizing the complex meaning-related relationships embedded in intricate sentence structures. Subsequently, the FastText word vector method is employed to encode word-level feature vectors. The word vectors are input into a BiLSTM, which produces an improved representation that augments the comprehension of contextual interdependencies across words. The sentence-level and word-level properties are ultimately merged as the fundamental components of the model.

a) Word-level feature-based embedding

During the feature selection phase, FastText is employed to generate word vector representations. This popular word embedding method is distinguished by its ability to model the internal composition of words using n-gram features [21]. During the slot filling process, FastText employs n-gram features to capture word morphology, hence enabling the efficient handling of atypical words and variations in word forms. In intent classification, FastText's word vectors provide comprehensive global semantic information, aiding in the comprehension of the sentence's overall meaning and its classification into predefined intents.

b) BiLSTM

The BiLSTM network is a modification of the LSTM architecture that acquires information in both forward and backward directions within sequential data, hence improving the comprehension of long-term dependencies [22]. By analyzing data sequences bidirectionally, BiLSTM successfully alleviates the gradient vanishing and exploding issues typically encountered by conventional recurrent neural networks. In intent classification, BiLSTM effectively captures the contextual information inherent in the sequence by examining the complete user utterance. BiLSTM effectively collects essential information from user input for slot filling and assigns it to the appropriate slots via its bidirectional comprehension of sequential data. For a given input vector v_i, the BiLSTM output $h_{bi-lstm}$ is obtained according to the following equation:

$$i_t = \sigma(W_i[v_i; h_{t-1}] + b_i) \quad (1)$$

$$f_t = \sigma\left(W_f[v_i; h_{t-1}] + b_f\right) \quad (2)$$

$$o_t = \sigma(W_o[v_i; h_{t-1}] + b_o) \quad (3)$$

$$\tilde{c} = \tanh(W_c[v_i; h_{t-1}] + b_c) \quad (4)$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{c} \quad (5)$$

$$h_t = o_t * \tanh(c_t) \quad (6)$$

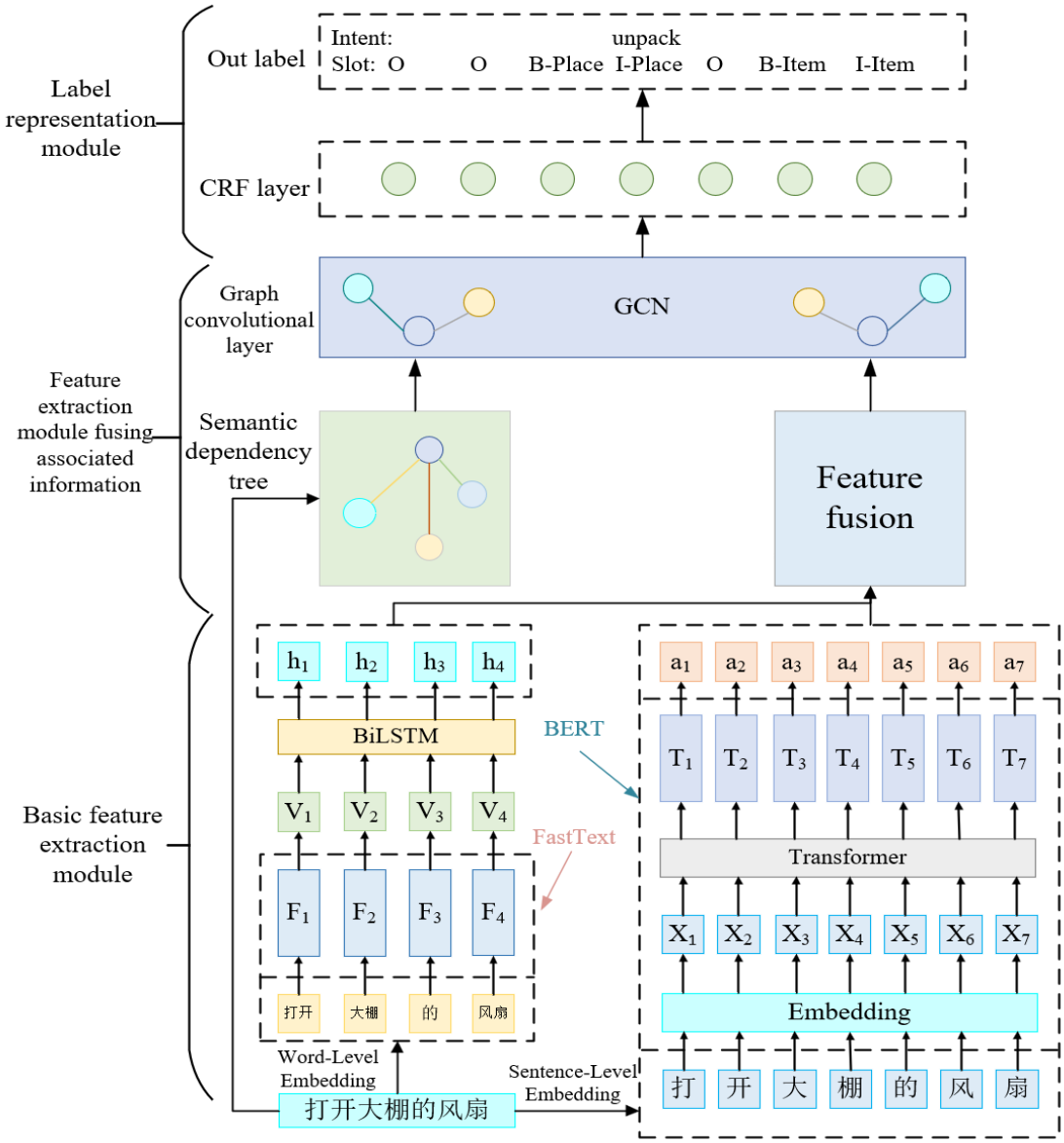

Fig. 1. Model Architecture Diagram

$$h_{bi-lstm} = \overrightarrow{h_t} + \overleftarrow{h_t} \quad (7)$$

In this context, "σ" denotes the sigmoid function, evaluated elementwise to ascertain the extent of information flow. The "*" symbol signifies element-wise multiplication. During each time step t, the input vector is denoted as v_i, while h_t signifies the hidden vector that defines the current state. The weight matrices W_i, W_f, W_o, W_c and the bias terms b_i, b_f, b_o, b_c are parameters acquired through the optimization process. The hidden state $\overrightarrow{h_t}$ is produced by the forward LSTM, whereas $\overleftarrow{h_t}$ is generated by the reverse LSTM.

c) Sentence-level feature-based embedding

BERT is a deep learning model built upon a multilayer bidirectional Transformer encoder [23]. The core operation relies on a multi-head self-attention mechanism combined with linear transformations, enhanced by residual connections. This architecture allows BERT to obtain extensive contextual information from textual data. BERT's inputs comprise three categories: word embeddings generated by the Word-Piece algorithm, segment embeddings indicating separate text sections, and positional embeddings that represent the word sequence. The model undergoes preliminary training on a vast, unlabeled text corpus to capture general linguistic patterns, making it highly versatile for different text processing tasks. In text categorization, BERT employs a [CLS] token at the start of the input, and the corresponding output vector acts as a representative feature for the entire input sequence. BERT utilizes [SEP] tokens to distinguish between two sentences and assigns separate segment embeddings to each. In sequence labeling tasks, BERT utilizes the output vectors of each word position, ensuring an exhaustive representation of each word's semantics within its context. Fig. 2 illustrates a schematic representation of the model's architecture.

d) Feature Fusion

This paper proposes a multi-tiered feature representation that amalgamates sentence-level and word-level data. Fig. 3 illustrates the process. The input sentences are first segmented using Spacy to enable the extraction of word-level features. Subsequently, each word is vectorized using a self-trained FastText model, and the resulting word vectors are input into a BiLSTM network to generate word embeddings. The BERT model simultaneously analyzes the entire text to generate sentence-level embeddings. The word embeddings and sentence embeddings are ultimately combined at the

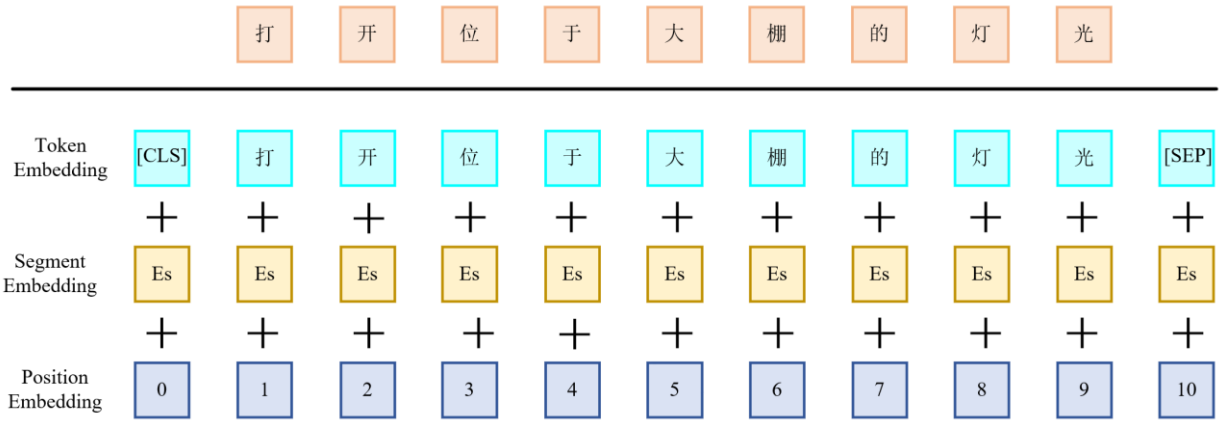

Fig. 2 Input Representations of BERT

feature level, yielding a comprehensive representation that includes both specific word-level details and the broader semantic context of the phrase.

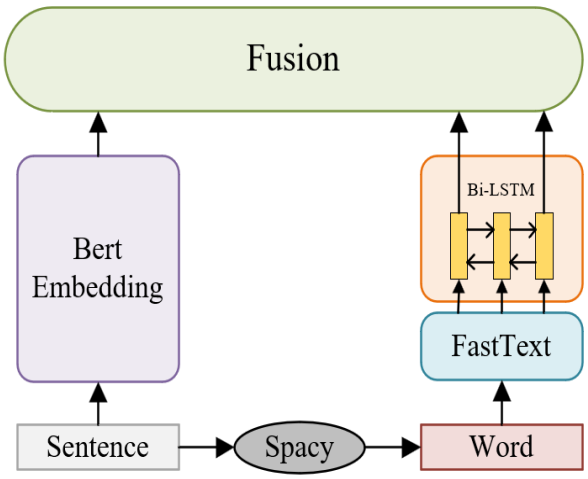

Fig. 3 Feature Fusion Diagram

B. Feature Extraction Module Fusing Associative Information

In the initial phase of text feature extraction, the significance of syntactic information has been inadequately recognized, despite employing the BERT model to capture global semantic attributes at the sentence level and the self-trained FastText model to acquire nuanced word-level information. Syntactic information is essential for language comprehension, as it accurately delineates the text's structure and the relationships among words, which are vital for elucidating sentence structure and constituent relationships. Consequently, it is evident that examining word interactions is crucial for attaining a deep comprehension of sentence objectives and slot relationships. This work presents an innovative approach to enhance feature representation and expand the model's understanding of language structure by using the dependent syntactic structure of sentences as supplementary information. This methodology is expected to significantly improve the model's ability to process complex linguistic structures, hence enhancing overall text comprehension.

a) Associative Information via Dependency Syntax

The LTP tool was utilized in this research to create comprehensive dependency analysis graph. Fig. 4 illustrates the dependency analysis tree produced form the phrase "关闭大棚的传感器和灯光" (turning off the sensors and lighting in the greenhouse).

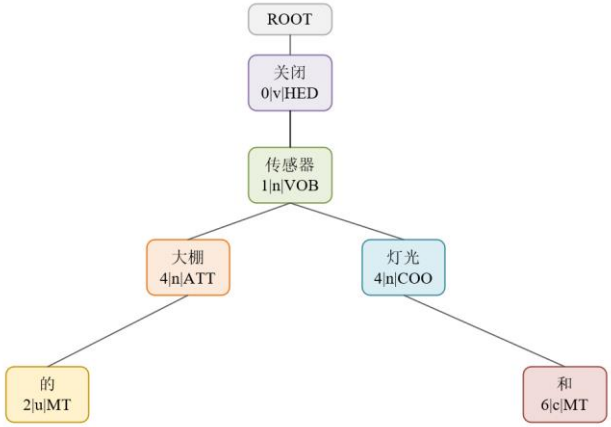

Fig. 4 Dependency Syntax Tree

The graph is represented by an n × n adjacency matrix, A, where A_{ij}=1 signifies the existence of an edge from word X_i to word X_j. Fig. 5 illustrates the adjacency matrix that delineates the relationships among the terms in the phrase.

	关闭	大棚	的	传感器	和	灯光
关闭	1	0	0	1	0	0
大棚	0	1	1	1	0	0
的	0	1	1	0	0	0
传感器	1	1	0	1	0	1
和	0	0	0	0	1	1
灯光	0	0	0	1	1	1

Fig. 5 Neighborhood Matrix Diagram

b) GCN-based feature representation

Graph Convolutional Networks (GCNs) have garnered significant attention for their effectiveness in encoding graphical structural information [24]. GCNs excel in capturing syntactic and semantic relationships among words, hence facilitating the modeling of complex linguistic structures. In intention classification, GCNs excel in

understanding and representing intricate relationships among words. GCNs have a robust capacity to encode structural information, facilitating the precise identification of nonlinear relationships between words for enhanced slot filling accuracy. By constructing an adjacency matrix that represents sentence dependencies and integrating comprehensive fundamental features into the GCN, GCNs enhance the representation of intricate word relationships within sentences, thereby improving the performance of intent recognition and slot extraction tasks.

GCN enhances the representation of each node by integrating neighboring information through operations performed on the graph. This process is shown in Fig. 6. The input channels of the GCN are denoted by *I*, signifying the dimensionality of the feature vector for each node X_i. The number of output channels is represented as O, according to the dimension of the updated node representation H_i. The label Y_i of each node is finally forecasted based on these parameters.

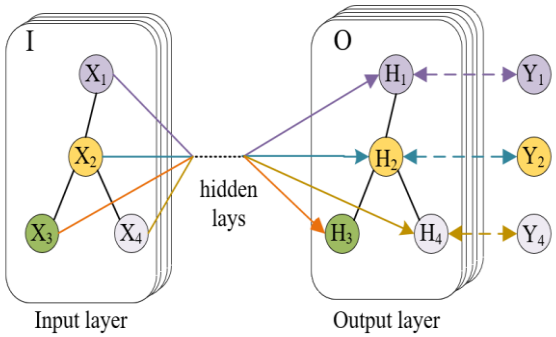

Fig. 6 GCN Model Diagram

In the L-layer GCN, the input vector is represented as $h_i^{(l-1)}$ and the output vector as $h_i^{(l)}$, where i signifies the ith node and l indicates the lth layer, it may be derived.

$$h_i^{(l)} = \sigma\left(\sum_{j=1}^{n} \widetilde{A_{ij}} W^{(l)} h_j^{(l-1)} / d_i + b^{(l)}\right) \tag{8}$$

$$\widetilde{A_{ij}} = A_{ij} + E \tag{9}$$

where d_i denotes the degree of the vertices, A_{ij} represents the adjacency matrix, E signifies the identity matrix, $W^{(l)}$ indicates a linear transformation, $b^{(l)}$ refers to a bias term, and σ is a nonlinear function.

C. Label Representation Module

In the label representation module, the output of the GCN, represented as a vector with dimensions [V, E] (where V signifies the number of nodes and E indicates the hidden dimension), is first converted into a vector with dimensions [batchsize, max_qen, hidden_side] via the fully connected layer, thereby preparing it for the CRF layer [25]. The procedure for conversion is outlined as follows:

$$h_{fc} = W_{fc} \cdot h_{gcn} + b_{fc} \tag{10}$$

where $h_{gcn}$is the output of the GCN, W_{fc} and b_{fc} are the weights and biases of the fully connected layer, and h_{fc} is the output of the fully connected layer.

A CRF layer is subsequently established to anticipate the label of each slot and the aim of the entire sequence depending on the input feature vector.

$$P(y|x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^{n} \theta_i f_i(y_{-1}, y, x_i)\right) \tag{11}$$

In this case, Z(x) represents the normalization factor, while θ_i is the weight parameter associated with the feature function.

For slot filling, the parameters [batchsize, max_qen, hidden_sides] function as inputs to the CRF, which is responsible for predicting a label for each slot. For intent classification, the input to the CRF is represented by the dimension [batchsize, hidden_size], facilitating the CRF component in determining the intent for the full sequence.

III. Experiment and Result Analysis

A. Dataset Labeling

A dataset consisting of five domains was annotated with a total of 1,500 samples, including several everyday tasks such as agriculture, music, health, meal, and travel. The collection has five domains, each including unique intentions and corresponding phrase examples. Additionally, the keyword slots corresponding to each intent have been designated (for further information, please consult Table II). The dataset was split into two subsets: 1,050 samples for model training and 450 samples for evaluating performance.

B. Evaluation Metrics

In this experiment, the semantic Slot F1 value is adopted as the evaluation index for the slot filling task, as in Eq.12

$$Slot(F1) = \frac{2 \times Precison \times Recall}{Precison + Recall} \tag{12}$$

Where *Precision* is the proportion of identified positive instances that are actually positive, while *Recall* is the proportion of true positive instances that are correctly identified by the model.

Intent Accuracy (Intent Acc) as a metric for the intent recognition task is formulated as follows.

$$Intent(Acc) = \frac{N_{Acc-intent}}{N_{all-intent}} \tag{13}$$

Where $N_{Acc-intent}$ d represents the quantity of correctly predicted intents, and $N_{all-intent}$ denotes the quantity of all predicted intents.

Overall (Acc), as an evaluation metric for sentence-level semantic frames, is formulated as follows.

$$Overall(Acc) = \frac{N_{Acc-sentence}}{N_{all-sentence}} \tag{14}$$

Where $N_{Acc-sentence}$ indicates the amount of correctly identified sentences, and $N_{all-sentence}$ denotes the amount of all identified sentence.

C. Parameter Setting

A range of feature extraction and fusion methodologies is utilized to enhance performance. Word-level attributes are derived using 300-dimensional word vectors trained using the FastText algorithm and subsequently fed into a BiLSTM to capture the contextual interactions among words. Features at the sentence level are obtained from the vectors generated by BERT, which represent the overall semantics of the sentences. In the initial stage, feature extraction is carried out using an LSTM network, which consists of 128 neurons in each layer to capture sequential dependencies. During the fusion phase, a three-layer graph convolutional network (GCN) is employed to integrate structural information, generating an output representation with a dimension of 64 to enhance feature learning and representation. The Adam optimizer is employed for model training, with a learning rate of 0.001 and a decay factor set at 0.9.

TABLE II
DATA SET LABELING (PARTIAL)

Domain	Intent	Quantity	Sentences	Slots
Agriculture	Check the temperature	100	请告诉我现在大棚的温度。 (Please let me know the temperature of the shed right now.)	{Date: 现在, Place: 大棚}
	Check prices	100	我想了解最新的玉米和大豆的市场价格。 （I would like to know the latest market prices for corn and soybeans.）	{Product 1: 玉米, Product 2: 大豆}
	Unpack	100	启动自动化采摘机器，收割成熟的水果和蔬菜。 （Start the automated picking machine to harvest ripe fruits and vegetables.）	{Operation: 启动, Object: 自动化采摘机器, Target: 成熟的水果和蔬菜}
Music	Play music	100	请播放周杰伦的《七里香》并循环播放 (Please play Jay Chou's "Seven Miles" and loop it).	{Singer: 周杰伦, Song: 七里香, Play mode: 循环播放}
	Search for songs	100	我想找一些适合跑步时听的动感音乐。 (I'd like to find some dynamic music for running).	{Occasion: 跑步, Type of music: 动感}
	Collection of songs	100	将《爱在西元前》添加到我的收藏夹中 (Add "Love Before the Western Era" to my favorites)	{Song: 爱在西元前, Operation: 添加}
Health	Make an appointment to register	100	我想预约下周三下午三点的北京协和医院皮肤科专家号。 (I would like to make an appointment with a dermatologist at Peking Union Medical College Hospital next Wednesday at 3pm.)	{Date: 下周三, Time: 下午三点, Location: 北京协和医院, Department: 皮肤科}
	Inquire about symptoms	100	最近总是头疼，伴随着恶心，可能是什么原因？ (Recently, I always have a headache accompanied by nausea, what could be the cause?)	{Symptom 1: 头疼, Symptom 2: 恶心}
	Purchase medications	100	我需要购买一盒感冒药和两瓶维生素C。 (I need to buy a box of cold medicine and two bottles of vitamin C.)	{Medicine 1: 感冒药, Quantity 1: 一盒, Medicine 2: 维生素C, Quantity 2: 两瓶}
Meal	Ordering food	100	我想要一份麻婆豆腐，两碗米饭，还有一瓶可乐。 (I would like an order of Mapo Tofu, two bowls of rice, and a bottle of Coke.)	{Dish 1: 麻婆豆腐, Quantity 1: 一份, Additional 1: 米饭, Quantity 2: 两碗, Additional 2: 可乐, Quantity 3: 一瓶}
	Query the menu	100	能否提供一下你们餐厅的特色菜单和价格列表？ (Can you provide a list of your restaurant's specialty menus and prices?)	{Object of enquiry: 特色菜单, Type of information: 价格列表}
	Feedback comments	100	我对上次点的宫保鸡丁非常满意，味道很正宗。 (I was very satisfied with the Kung Pao Chicken I ordered last time, it tasted very authentic.)	{Dish: 宫保鸡丁, Rating: 非常满意, Description: 味道很正宗}
Travel	Inquire about attractions	100	北京有哪些著名的旅游景点？ (What are the famous tourist attractions in Beijing?)	{Location: 北京, Type: 旅游景点}
	Book a hotel	100	我想预订一间上海的双人间。 (I would like to book a double room in Shanghai.)	{Location: 上海, Room type: 双人间}
	Itinerary planning	100	我打算下个月去云南旅游，能帮我规划一下五天四夜的行程吗？ (I am planning to travel to Yunnan next month, can you help me plan a 5 days and 4 nights itinerary?)	{Time: 下个月, Location: 云南, Duration: 五天四夜}

D. Training the word vector FastText

Initially, the labeled cross-domain Chinese dataset was processed for word segmentation in this experiment. The processed textual data served as input for training the FastText model to produce word vectors. The skip-gram model was employed for training, with the model's essential parameters configured as follows: The vector dimension was set to 300, with a context window size of 5, the amount of training iterations was 10 to adequately facilitate the model's learning of contextual information and vector representations of the vocabulary, and the minimum word frequency was established at 1 to guarantee the inclusion and effective training of all vocabulary words in the dataset. After the training is completed, the model's efficacy is assessed through nearest-neighbor lexical analysis and inter-lexical similarity computation for designated terms, evaluating how well the model captures semantic relationships. Table III presents the specific evaluation results, alongside the corresponding lexical similarity analysis, offering detailed insights into model performance.

TABLE III
FEXTTEXT EFFECTIVENESS EVALUATION FORM

	proximity and similarity		
玉米	(0.9997599720，小麦)	(0.9988487708，大豆)	(0.9941645447，高粱)
七里香	(0.9997599895，青花瓷)	(0.9997599720，夜曲)	(0.9997599795，晴天)
头疼	(0.9963153206，眼睛疲劳)	(0.9962143207，颈椎痛)	(0.9963143213，胃疼)
臭豆腐	(0.9973453204，麻辣烫)	(0.9964553212，烤串）	(0.9963153138，粽子)
双人间	(0.9993153207，单人间)	(0.9992153206，标准间)	(0.9973153204，豪华套房)

Analysis of Experimental Results: As indicated in Table III, we selected one term from each of the five distinct domains: "玉米" (corn), "七里香" (a song title), "头疼" (headache), "臭豆腐" (stinky tofu), and "双人间" (double room) to examine the similarity of their associated terms in detail. The findings reveal that the similarity between these words and

their associated terms exceeds 0.99, strongly indicating that our trained word vector model exhibits exceptional performance.

E. Ablation Experiment

A set of comparative experiments was conducted to assess the effectiveness of the MDG-JISL model and to examine the importance of its individual components through ablation studies. The specifics of the experimental design are outlined in the subsequent section.

FastText Baseline Experiment: This experiment uses exclusively self-trained FastText to generate word-level representations, which are subsequently fed into a conditional random field for classification. This baseline model illustrates the efficacy of a straightforward text representation and classification method that eschews the use of complex network topologies.

Integration of FastText and BiLSTM: A word-level vector representation is obtained from the FastText Baseline. Subsequently, the aforementioned representation is fed into a BiLSTM, yielding a more extensive input for the conditional random field classification.

BERT Baseline Experiment: The BERT model is utilized for the direct extraction of textual features, which are then input into a conditional random field for classification.

Combining FastText and BERT: Integrates textual attributes obtained from BERT with those acquired from FastText and BiLSTM, which function as the primary features employed in the CRF for categorization.

Our Model: The vectors produced by FastText-trained word embeddings and those obtained from BERT are combined to form the essential features. Thereafter, the aforementioned features are transmitted to the GCN, which generates supplementary features for classification utilizing the adjacency matrices of the designated vectors and words.

TABLE IV
TABLE OF ABLATION EXPERIMENT RESULTS

No.	method	Slot (F1)/%	Intent (Acc)/%	Overall (Acc)/%
1	FastText Baseline Experiment	73.2	75.8	69.2
2	Integration of FastText and Bi-LSTM	78.3	77.6	71.1
3	BERT Baseline Experiment	95.1	96.4	87.2
4	Combining FastText and BERT	96.7	97.8	88.9
5	**our model**	**98.7**	**98.4**	**92.4**

The effectiveness of five methods was assessed utilizing the dataset, with the results presented in Table IV. A comparative analysis of the experimental findings reveals that the distinction between Experiment 1 and Experiment 2 pertains to the inclusion or exclusion of a BiLSTM layer. For the test set, Slot (F1) improved by 4.9 percentage points, Intent (Acc) by 1.8, and Overall (Acc) by 1.9, indicating that the BiLSTM layer effectively captures bidirectional information across words and considerably enhances the SLU task. Experiment 4 contrasts with Experiment 3 in that its input CRF features are derived from the features obtained in Experiments 2 and 3. The improvements are clearly evident in the measures for slot (F1), intent (Acc), and overall accuracy (Acc). The aforementioned improvements were 1.6, 1.4, and 1.7 percentage points, respectively. This illustrates the benefits of the BERT pre-trained model in encoding efficiency and highlights the significance of word-level characteristics in semantic understanding tests. In Experiment 5, the GCN layer was executed in accordance with the findings of Experiment 4. The important elements derived from the examination of dependent syntactic relations were used to generate the collocation matrix. The results of Experiment 4 were later included into the GCN layer, which was applied to optimize the performance in the relevant categories. The accuracy rates increased by 2.0, 0.6, and 3.5 percentage points, respectively. This demonstrates the GCN's capability in extracting and employing features inside the graph, hence affirming the efficacy of the MDG-JISL model.

F. Comparison experiments with different word vectors

This study conducts a comparative examination of three word embedding models—FastText, Word2Vec, and GloVe—to demonstrate the advantages of FastText in word vector extraction, with results presented in Table V. The experimental results show that FastText achieves improvements of 6.6, 5.4, and 3.1 in the Slot (F1), Intent (Acc), and Overall (Acc) metrics compared to GloVe, and demonstrates gains of 6.2, 6.5, and 3.5 in these metrics when compared to Word2Vec.FastText employs subword n-gram modeling to effectively manage atypical and unregistered words, resulting in improved performance in intent classification and slot filling tasks. In contrast, Word2Vec depends on the context window to produce word vectors, which struggles with intricate variations in word forms, hence impacting the precision of intent categorization. Although GloVe effectively collects global word co-occurrence data, its performance in slot filling tasks is inferior to that of FastText since it cannot evaluate subwords.

TABLE V
COMPARISON OF WORD-LEVEL EMBEDDING MODELS

method	Slot(F1)/%	Intent (Acc)/%	Overall (Acc)/%
Word2Vec	92.5	91.9	88.9
GloVe	93.1	93.0	89.3
FastText	98.7	98.4	92.4

G. Comparison Experiments Different Graph Convolutional Layers

The optimal number of GCN layers for achieving the most efficient model is determined by assigning a value between 1 and 5 to each layer. The corresponding Slot (F1), Intent (Acc), and Overall (Acc) for different GCN layer numbers are shown in Fig. 7.

The results are meticulously examined, and it is widely acknowledged that the addition of network layers reduces error and enhances accuracy. Consequently, this results in a more complex network, thereby extending the training period and increasing the likelihood of overfitting. This study examines dependent syntactic parsing, which uses a binary system (0 or 1) to define the relationship between two words. This is insufficient for the purpose of intent categorization with slot filling. Analysis of Fig.7, alongside the previously presented data reveals that the model demonstrates enhanced performance in the Slot (F1), Intent (Acc), and Overall (Acc) metrics when the convolution layers are set to 3. This indicates that this configuration is optimal for the present task, yielding the highest performance.

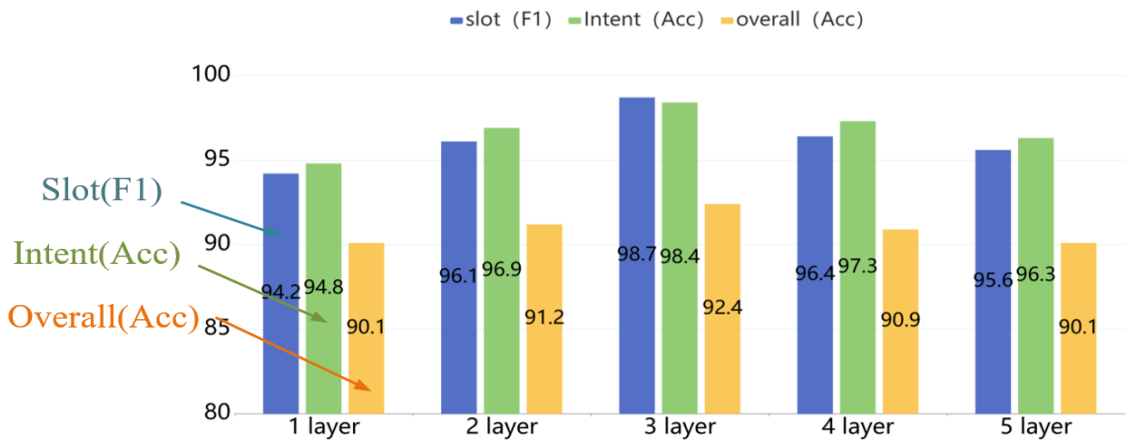

Fig. 7 Comparison of Different GCN Layers

TABLE VI
COMPARISON OF DIFFERENT BASELINE

Models	Slot(F1)/%			Intent (Acc)/%			Overall (Acc)/%		
	Ours	ATIS	SNIPS	Ours	ATIS	SNIPS	Ours	ATIS	SNIPS
Attention BIRNN	94.1	94.1	87.6	91.2	91.3	96.3	78.8	78.3	74.4
Slot-Gated	94.5	95.2	89.0	93.2	95.3	97.4	82.1	84.9	75.1
Joint BERT	96.1	96.0	96.8	92.6	96.8	97.8	88.1	88.3	92.8
SASGBC	96.5	96.8	96.3	97.4	98.1	97.8	90.4	91.6	91.1
Task Conditioned BERT	95.9	96.3	94.7	97.2	97.7	98.0	90.3	87.1	87.3
Co-transformers	96.1	96.5	94.8	97.3	97.8	98.2	90.7	90.2	91.5
our model	**98.7**	**98.4**	**96.9**	**98.4**	**98.7**	**98.8**	**92.4**	**92.5**	**92.6**

H. Comparison Experiments of Different Models

This part provides a comparison of the MDG-JISL model against several baselines, as elaborated in the subsequent paragraphs.

Attention BIRNN [26]: The architecture combines a RNN encoder-decoder with an attention mechanism based on the encoder's hidden state.

Slot-gated [27]: Integrated within the architectural design is a dedicated gate control unit in the LSTM framework, utilizing contextual intent vectors to convey the interactions between slots and intents.

SASGBC [28]: Incorporated as an encoder is BERT, which leverages the semantic associations between slots and intents through a pick-and-pass mechanism, a method for selecting and transmitting information.

Co-interactive transformer [29]: The architectural design employs a synergistic interaction transformer, which proposes a synergistic interaction module that establishes a bidirectional connection between two tasks in order to account for cross-influences.

Task Conditioned BERT [30]: A unified model based on BERT, trained on multiple tasks with augmented inputs, is put forth as a means of tuning the model for target inference.

Table VI illustrates that the MDG-JISL model presented in this paper surpasses all baseline models on every criterion. The MDG-JISL model demonstrates enhancements of 1.6, 1.1, and 1.7 percentage points in the Slot (F1), Intent (Acc), and Overall (Acc) metrics, respectively, compared to the existing ideal baseline model in the labeled dataset. The experimental results further corroborate the model's efficacy in utilizing graph convolutional networks to extract graph characteristics, hence substantially enhancing the overall performance of the SLU task.

Experimental findings using publicly available datasets, such as ATIS and SNIPS, also demonstrate the model's superiority. In the SNIPS dataset, the Slot (F1) score attains 96.9%, the Intent (Acc) is 98.8%, and the Overall (Acc) is 92.6%. The enhancement of MDG-JISL on the SNIPS dataset surpasses that of current models, illustrating its robust applicability to multi-domain applications. Furthermore, the model achieves Slot (F1) of 98.4%, Intent (Acc) of 98.7%, and Overall (Acc) of 92.5% on the ATIS dataset, thereby demonstrating its exceptional generalization capabilities across standard datasets.

I. Visualization analysis

The efficacy of this paper's model is unequivocally illustrated by the intention-word correlation scores of the ideal comparative models Co-Transformer and MDG-JISL, as shown in Table VI. In Fig. 8, the input text is shown on the horizontal scale, while the expected intent categories are represented on the vertical scale, with darker hues reflecting a stronger association. In the sentence “请问大棚的温暖程度如何” (How warm is the greenhouse), the suggested model has a superior correlation score for predicting the intent category "Check the temperature". This indicates that GCNs that integrate window sequence attributes can enhance the significance of key words, hence providing more relevant information to aid the slot-filling process. Conversely, the Co-Transformer model identified the intent of the phrase as "Unpack". This inaccuracy may arise due to the predominance of the "Unpack" function in specific terms, along with the Co-Transformer layer's inadequate ability to capture certain data features while evaluating the interaction effects of the two objectives, leading to erroneous prediction

J. Summary

This study introduces the MDG-JISL model, which efficiently extracts key features by combining BERT and FastText. Additionally, the model employs the adjacency matrix derived from dependent syntactic analysis to augment features through graph convolutional networks. The model's output layer employs a CRF to enhance prediction accuracy. The model exhibited exceptional performance, achieving 98.7% in Slot (F1), 98.4% in Intent (Acc), and 92.4% in

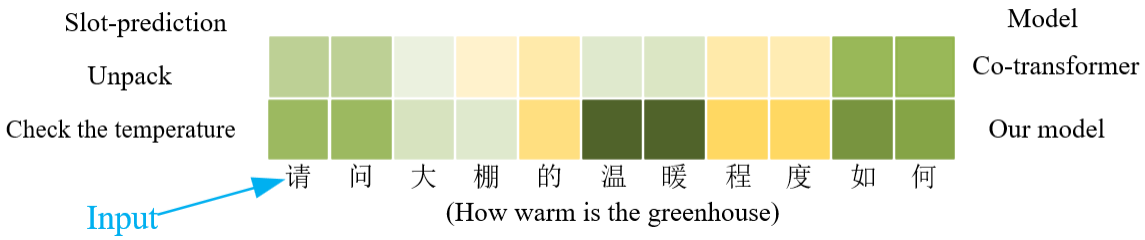

Fig. 8 Visualization of Intention Word Relevance Score

Overall (Acc), across datasets from several domains. Notwithstanding the favorable outcomes, there exists an opportunity for the model's improvement. The deployment of many complex models may result in increased computational costs and prolonged processing times, thus hindering the model's applicability in real-time or resource-limited settings Subsequently, additional work will concentrate on optimizing the model architecture to reduce computational requirements and assessing its effectiveness across a wider array of diverse tasks and circumstances. This will further validate the model's effectiveness in practical applications, particularly in areas such as intelligent assistants and question-and-answer systems.

REFERENCES

[1] R. Sarikaya, P.A. Crook, A. Marin, M. Jeong, J.P. Robichaud, A. Celikyilmaz, Y.B. Kim, A. Rochette, O.Z. Khan, X. Liu and D. Boies, "An overview of end-to-end language understanding and dialog management for personal digital assistants," in 2016 IEEE Spoken Language Technology Workshop (slt), pp391-397,2016.

[2] J.R. Bellegarda, "Spoken language understanding for natural interaction: The siri experience," Natural Interaction with Robots, Knowbots and Smartphones: Putting Spoken Dialog Systems into Practice, pp3-14,2013.

[3] M. Mendoza and J. Zamora, "Identifying the intent of a user query using support vector machines," in International Symposium on String Processing and Information, pp. 131-142,2009.

[4] K. Baati and M. Mohsil, "Real-time prediction of online shoppers' purchasing intention using random forest," in Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, pp43-51,2020.

[5] L. Li, W. Zhao, C. Xu, C. Wang, Q. Chen and S. J. I. T. o. V. T. Dai, "Lane-change intention inference based on RNN for autonomous driving on highways," IEEE Transactions on Vehicular Technology, vol. 70, no. 6, pp5499-5510, 2021.

[6] K. Sreelakshmi, P. Rafeeque, S. Sreetha and E. Gayathri, "Deep bi-directional LSTM network for query intent detection," Procedia Computer Science, vol. 143, pp939-946, 2018.

[7] Q. Deng, J. Wang, and D. Soffker, "Prediction of human driver behaviors based on an improved HMM approach," in 2018 IEEE Intelligent Vehicles Symposium (IV), pp2066-2071,2018.

[8] S. Liu, T. He and J. Dai, "A survey of CRF algorithm-based knowledge extraction of elementary mathematics in Chinese," Mobile Networks Applications, vol. 26, pp1891-1903,2021.

[9] M. Jbene, S. Tigani, R. Saadane, and A. Chehri, "A robust slot filling model based on lstm and crf for iot voice interaction." in 2022 IEEE Globecom Workshops (GC Wkshps), pp922-926, 2022.

[10] L. Hou, Y. Li, C. Li, and M. Lin, "Review of research on task-oriented spoken language understanding," Journal of Physics: Conference Series, vol.1267, no.1,2019.

[11] M. Jeong and G. G. Lee, "Triangular-chain conditional random fields, IEEE Transactions on Audio," Speech, Language Processing, vol.16, no.7, pp1287-1302 ,2008.

[12] Q. Zhou, L. Wen, X. Wang, L. Ma, and Y. Wang, "A hierarchical lstm model for joint tasks," in Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data: 15th China National Conference, pp234-335 ,2016.

[13] Y. Zheng, Y. Liu, and J. H. Hansen, "Intent detection and semantic parsing for navigation dialogue language processing," in 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp1-6,2017.

[14] B. Liu and I. Lane, "Recurrent neural network structured output prediction for spoken language understanding," in Proc. NIPS Workshop on Machine Learning for Spoken Language Understanding and Interactions, 2015.

[15] M. Firdaus, A. Ekbal, and E. Cambria, "Multitask learning for multil ingual intent detection and slot filling in dialogue systems." Informat ion Fusion, pp299-315,2023.

[16] Y. Chen, and Z. Luo, "Pre-trained joint model for intent classificatio n and slot filling with semantic feature fusion." Sensors, vol.23, no.5, 2023.

[17] Y. Guo, Z. Xie, X. Chen, L. Wang, Y. Zhao, and G. Wu, "AWTE-B ERT: Attending to Wordpiece Tokenization Explicitly on BERT for Joint Intent Classification and SlotFilling" CoRR,2022.

[18] K. He, S. Lei, Y. Yang, H. Jiang, and Z. Wang, "Syntactic graph con volutional network for spoken language understanding," in Proceedi ngs of the 28th International Conference on Computational Linguisti cs, pp2728-2738, 2020.

[19] H. Tang, D. Ji and Q. J. N. Zhou, "End-to-end masked graph-based CRF for joint slot filling and intent detection," Neurocomputing vol.413, pp348-359,2020.

[20] P. Wei, B. Zeng and W. Liao, "Joint intent detection and slot filling with wheel-graph attention networks," Journal of Intelligent Fuzzy Systems, vol.42, no.3, pp2409-2420, 2022.

[21] I. N. Khasanah, "Sentiment classification using fasttext embedding and deep learning model." Procedia Computer Science, pp343-350.2021.

[22] Y. Fang, H. Fu, H. Tao, X. Wang, and L. Zhao, "Bidirectional LSTM with multiple input multiple fusion strategy for speech emotion recognition," IAENG International Journal of Computer Science, vol.48, no.3, pp613-618,2021.

[23] Y. Wang, X. Cheng, and X. Meng, "Sentiment analysis with an integ rated model of BERT and bi-LSTM based on multi-head attention m echanism," IAENG International Journal of Computer Science, vol.5 0, no.1, pp255-262,2023.

[24] S. Li, Z. Li, J. Wu, J. Miao, Y. Bai, X. Yu, and K. Li, "Attention Feature Fusion Graph Convolutional Network for Target-Oriented Opinion Words Extraction.," Engineering Letters, vol.31, no.3, pp1273-1280,2023.

[25] H. Yang, L. Wang, and Y. Yang, "Named Entity Recognition in Electronic Medical Records Incorporating Pre-trained and Multi-Head Attention," IAENG International Journal of Computer Science, vol.51, no.4, pp401-408,2024.

[26] B. Liu and I. Lane, "Joint online spoken language understanding and language modeling with recurrent neural networks," arXiv preprint arXiv:1609.01462. 2016.

[27] C.W. Goo, G. Gao, Y.K. Hsu, C.L. Huo, T.C. Chen, K.W. Hsu, and Y.N. Chen, "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, vol.2, pp753-757,2018.

[28] C. Wang, Z. Huang, and M. Hu, "SASGBC: Improving sequence labeling performance for joint learning of slot filling and intent detection," in Proceedings of 2020 the 6th International Conference on Computing and Data Engineering, pp29-33,2020.

[29] L. Qin, T. Liu, W. Che, B. Kang, S. Zhao, and T. Liu, "A co-interactive transformer for joint slot filling and intent detection," in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp8193-8197 ,2021.

[30] D. Tavares, P. Azevedo, D. Semedo, R. Sousa, and J. Magalhaes, "Task Conditioned BERT for Joint Intent Detection and Slot-filling," EPIA Conference on Artificial Intelligence, pp467-480, 2023.