Towards Joint Intent Detection and Slot Filling via Higher-order Attention

Dongsheng Chen[1] , **Zhiqi Huang**[1] , **Xian Wu**[2] , **Shen Ge**[2] and **Yuexian Zou**[1*]
[1]School of ECE, Peking University, China
[2]Tencent Medical AI Lab, China
chends@stu.pku.edu.cn, {zhiqihuang, zouyx}@pku.edu.cn, {kevinxwu, shenge}@tencent.com

Abstract

Recently, attention-based models for joint intent detection and slot filling have achieved state-of-the-art performance. However, we think the conventional attention can only capture the first-order feature interaction between two tasks and is insufficient. To address this issue, we propose a unified BiLinear attention block, which leverages bilinear pooling to synchronously explore both the contextual and channel-wise bilinear attention distributions to capture the second-order interactions between the input intent and slot features. Higher-order interactions are constructed by combining many such blocks and exploiting Exponential Linear activations. Furthermore, we present a Higher-order Attention Network (HAN) to jointly model them. The experimental results show that our approach outperforms the state-of-the-art results. We also conduct experiments on the new SLURP dataset, and give a discussion on HAN's properties, i.e., robustness and generalization.

1 Introduction

Spoken language understanding (SLU) that aims to understand user oral instructions typically includes two modules: Intent detection (ID) and Slot filling (SF). Typically, ID is regarded as a semantic utterance classification problem to predict users intent, and SF is usually treated as a sequence labeling problem to extract semantic concepts of the spoken utterance. Take a flight-related utterance as an example, "*Show flights from Seattle to San Diego*", illustrated in Table 1. There are different slot labels for each word in the utterance, and a specific intent for the whole utterance.

Numerous works appealed to improve sentence and slot level semantics via mutual enhancement between ID and SF for their close correlation [Hakkani-Tür *et al.*, 2016; Niu *et al.*, 2019; Zhu *et al.*, 2020; Qin *et al.*, 2021]. In addition, the Multi-Head Attention (MHA) [Vaswani *et al.*, 2017] was introduced and leveraged into the model [Li *et al.*, 2018; Zhang *et al.*, 2019; Qin *et al.*, 2021] to provide the precise focus by linearly fusing the query and key via element-wise sum, and assign a weight to this value to arrive at a weighted sum representing the enhanced representations.

Utter.	Show	flights	from	Seattle	to	San	Diego
Slot	O	O	O	B-fromloc	O	B-toloc	I-toloc
Intent	Flight						

Table 1: An example of utterance with intent and annotated slots using the BIO scheme.

Despite their success, the attention based approaches suffer a limitation, i.e., they mainly exploit the first-order interactions between ID and SF features. We propose to distill second-order feature interactions via bilinear pooling [Kim *et al.*, 2018], which calculates the outer product between two feature vectors, to trigger higher-order feature interactions. This technique triggers second-order feature interactions by establishing pairwise interactions between all queries and keys, and resulting in more expressive representations, which has been successfully applied in the field of Computer Vision [Lin *et al.*, 2015; Fang *et al.*, 2019; Yu *et al.*, 2020; Pan *et al.*, 2020; Kumari and Ekbal, 2021].

This paper introduces a BiLinear attention block illustrated in Figure 2 to trigger the second-order interactions between ID and SF tasks. Note that the significant difference from [Chen *et al.*, 2022] is that we additionally leverage channel-wise bilinear attention simultaneously except for the conventional contextual attention computation. By combining lots of such blocks and equipping them with Exponential Linear Unit (ELU) [Barron, 2017], the model can build even higher-order feature interactions. Furthermore, we proposed the Dynamic Feature Fusion to explicitly fuse intent and slot features, instead of the straightforward concatenation of them. Finally, we present the Higher-order Attention Network (HAN) to leverage higher-order interactions for the SLU task, which is shown in Figure 1(b).

Extensive experiments on two benchmark datasets SNIPS [Coucke *et al.*, 2018] and ATIS [Hemphill *et al.*, 1990] proves the advancement of our approach. To further evaluate the proposed model, we conduct experiments on a recently released SLU dataset, i.e., SLURP [Bastianelli *et al.*, 2020], which is substantially larger and more diverse than existing SLU datasets.

Our contributions in this work are three-fold:

- We propose a novel Higher-order Attention Network

*Contact Author

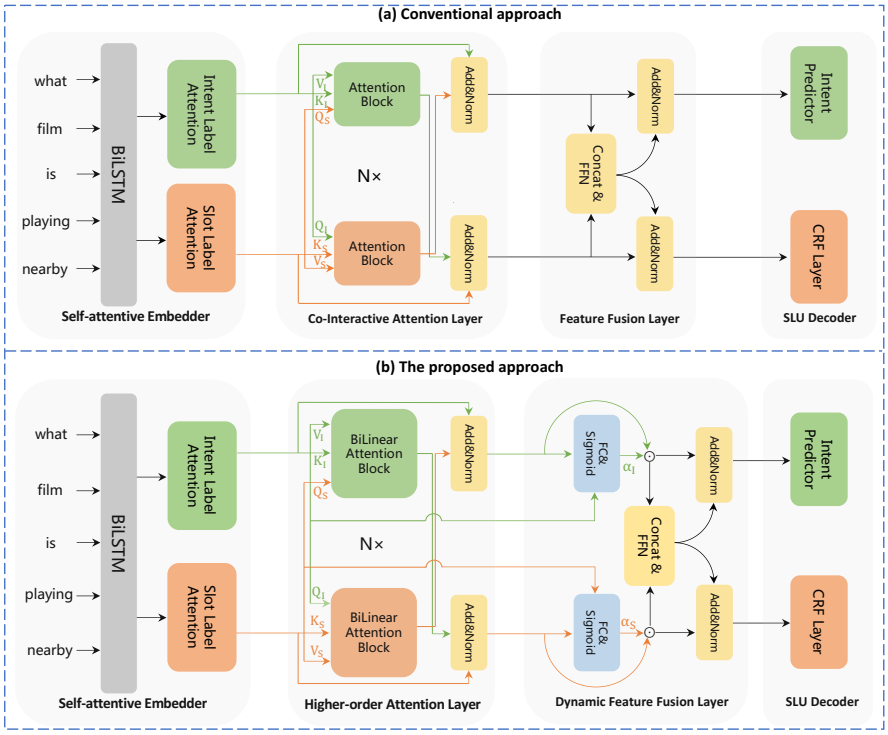

Figure 1: Comparison between (a) conventional approach and (b) the proposed approach (HAN) for SLU task.

(HAN) for SLU tasks, it achieves state-of-the-art results on two benchmark datasets in ID and SF tasks.

- HAN includes two essential modules: Higher-order Attention Layer (HAL) and Dynamic Feature Fusion Layer (DFFL). These two modules can be easily integrated into existing SLU models to boost their performance.
- We further give a deep exploration of the proposed methods from different views, including triggering higher-order interactions, attention visualization, robustness, and generalization analysis.

2 Conventional Approach

We first introduce the conventional approach illustrated in Figure 1(a), which consists of 4 modules: Self-attentive Embedder, Co-Interactive Attention Layer (CAL), Feature Fusion Layer (FFL), and SLU Decoder.

2.1 Self-attentive Embedder

Self-attentive Embedder targets to acquire utterance embeddings [Qin *et al.*, 2019]. Given an input utterance, it uses a shared BiLSTM layer to acquire $\mathbf{H} \in \mathbb{R}^{n \times d}$. Then the label attention [Cui and Zhang, 2019] is introduced to enrich the utterance embeddings with the intent and slot labeling information respectively:

$$\begin{aligned} \mathbf{H_I} &= \mathbf{H} + \text{softmax}(\mathbf{H}(\mathbf{W^I})^\top)\mathbf{W^I}, \\ \mathbf{H_S} &= \mathbf{H} + \text{softmax}(\mathbf{H}(\mathbf{W^S})^\top)\mathbf{W^S}, \end{aligned} \tag{1}$$

where $\mathbf{W^I} \in \mathbb{R}^{N_I \times d}$ and $\mathbf{W^S} \in \mathbb{R}^{N_S \times d}$ denote the label embedding matrix for ID and SF, N_I and N_S denotes the number of labels respectively. With Eq.(1), the acquired $\mathbf{H_I} \in \mathbb{R}^{n \times d}$ and $\mathbf{H_S} \in \mathbb{R}^{n \times d}$ could capture the intent and slot labeling information.

2.2 Co-Interactive Attention Layer (CAL)

The Co-Interactive Attention module is introduced to calculate the cross attention between intent feature vector and slot feature vector. Here we first provide a brief review of the attention module. Given the query $\mathbf{q}$, we can obtain the attention distribution α over a set of keys $\mathbf{K} = \{\mathbf{k}_i\}_{i=1}^n$:

$$\begin{aligned} a_i &= \mathbf{W}_0[\tanh(\mathbf{W}_1\mathbf{q} + \mathbf{W}_2\mathbf{k}_i)], \\ \alpha &= \text{softmax}(\mathbf{a}), \end{aligned} \tag{2}$$

where a_i represents the i^{th} value in $\mathbf{a}$, and $\mathbf{W}_0$, $\mathbf{W}_1$, $\mathbf{W}_2$ are weight matrices. The attention module outputs the enhanced feature $\hat{\mathbf{v}}$ via leveraging all values with contextual attention weights: $\hat{\mathbf{v}} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i$, which can be denoted as:

$$\hat{\mathbf{V}} = \text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}). \tag{3}$$

$\mathbf{H_I}$ and $\mathbf{H_S}$ in Eq.(1) are fed into the CAL to be enriched with both the intent and slot features in a mutual way. As [Vaswani *et al.*, 2017], we first map both $\mathbf{H_I}$ and $\mathbf{H_S}$ to two Query-Key-Value sets $(\mathbf{Q_I}, \mathbf{K_I}, \mathbf{V_I})$ and $(\mathbf{Q_S}, \mathbf{K_S}, \mathbf{V_S})$ through different weight matrices, respectively. Then $\mathbf{Q_I}$, $\mathbf{K_S}$, and $\mathbf{V_S}$ are used as queries, keys, and values, leading to the enhanced values:

$$\begin{aligned} \hat{\mathbf{V}}_\mathbf{I} &= \text{MHA}(\mathbf{Q_I}, \mathbf{K_S}, \mathbf{V_S}), \\ \mathbf{H_I} &= \text{LN}(\mathbf{H_I} + \hat{\mathbf{V}}_\mathbf{I}), \end{aligned} \tag{4}$$

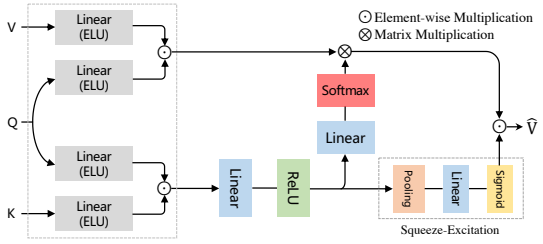

Figure 2: Illustration of the BiLinear Attention Block.

where LN represents the layer normalization [Ba *et al.*, 2016]. Similarly, we take $\mathbf{Q_S}$, $\mathbf{K_I}$, and $\mathbf{V_I}$ to obtain $\mathbf{H_S}$. We finally obtain the enhanced intent features $\mathbf{H_I^N} \in \mathbb{R}^{n\times d}$ and slot features $\mathbf{H_S^N} \in \mathbb{R}^{n\times d}$ after repeating N times.

2.3 Feature Fusion Layer (FFL)

In this subsection, we extend the Feed-Forward Network layer to explicitly fuse intent and slot information. we concatenate $\mathbf{H_I^N}$ and $\mathbf{H_S^N}$ to fuse the intent and slot features:

$$\mathbf{H_{IS}} = [\mathbf{H_I^N}, \mathbf{H_S^N}], \tag{5}$$

where $[\cdot,\cdot]$ indicates concatenation. Next, we adopt the shared FFN to acquire the updated $\hat{\mathbf{H}}_\mathbf{I}^{\mathbf{N}}$, $\hat{\mathbf{H}}_\mathbf{S}^{\mathbf{N}} \in \mathbb{R}^{n\times d}$:

$$\begin{aligned}\hat{\mathbf{H}}_\mathbf{I}^{\mathbf{N}} &= \mathrm{LN}(\mathrm{FFN}(\mathbf{H_{IS}}) + \mathbf{H_I^N}),\\ \hat{\mathbf{H}}_\mathbf{S}^{\mathbf{N}} &= \mathrm{LN}(\mathrm{FFN}(\mathbf{H_{IS}}) + \mathbf{H_S^N}).\end{aligned} \tag{6}$$

2.4 SLU Decoder

SLU Decoder includes an Intent Predictor for ID and a CRF Layer for SF. Specially, we follow [Kim, 2014] to employ the maxpooling on $\hat{\mathbf{H}}_\mathbf{I}^{\mathbf{N}}$ to obtain $\boldsymbol{c}$, which is used to predict the intent label: $\mathbf{o^I} \sim \hat{\mathbf{y}}^\mathbf{I} = \mathrm{softmax}\left(\mathbf{W^I}\boldsymbol{c} + \mathbf{b_I}\right)$.

Besides, we apply a CRF layer [Niu *et al.*, 2019] to model the labels relationships:

$$P(\hat{y}|\mathbf{O_S}) = \frac{\sum_{i=1} \exp f(y_{i-1}, y_i, \mathbf{O_S})}{\sum_{y'}\sum_{i=1} \exp f(y'_{i-1}, y'_i, \mathbf{O_S})}, \tag{7}$$

where $\mathbf{O_S} = \mathbf{W^S}\hat{\mathbf{H}}_\mathbf{S}^{\mathbf{N}} + \mathbf{b_S}$, $\hat{y}$ indicates the predicted label sequence, and $f(y_{i-1}, y_i, \mathbf{O_S})$ computes the transition score from y_{i-1} to y_i.

3 The Proposed Approach

As shown in Figure 1(b), based on the conventional approach, we replace CAL and FFL with Higher-order Attention Layer (HAL) and Dynamic Feature Fusion Layer (DFFL), respectively, and thus obtain the proposed Higher-order Attention Network, i.e., HAN. Before introducing HAL and DFFL, we first describe the proposed BiLinear attention block illustrated in Figure 2.

3.1 BiLinear Attention Block

The BiLinear attention block uses Eq.(8) to replace Eq.(3). Given $\mathbf{Q} = \{\mathbf{q}_i\}_{i=1}^n$, $\mathbf{K} = \{\mathbf{k}_i\}_{i=1}^n$, and $\mathbf{V} = \{\mathbf{v}_i\}_{i=1}^n$ (Their shapes are all $\mathbb{R}^{n\times d}$), the BiLinear attention is formulated in Eq.(8):

$$\hat{\mathbf{V}} = \mathrm{MHA}_{bi}(\mathbf{Q}, \mathbf{K}, \mathbf{V}). \tag{8}$$

Take a query $\mathbf{q}$ from $\mathbf{Q}$, together with the i^{th} $\mathbf{k}_i$ from $\mathbf{K}$ and $\mathbf{v}_i$ from $\mathbf{V}$. Using the bilinear pooling calculation, the bilinear query-key and query-value joint representations $\mathbf{R}_i^k, \mathbf{R}_i^v \in \mathbb{R}^d$ are introduced to trigger the second-order feature interactions:

$$\begin{aligned}\mathbf{R}_i^k &= (\mathbf{W}_q^k\mathbf{q}) \odot (\mathbf{W}_k\mathbf{k}_i),\\ \mathbf{R}_i^v &= (\mathbf{W}_q^v\mathbf{q}) \odot (\mathbf{W}_v\mathbf{v}_i),\end{aligned} \tag{9}$$

where $\mathbf{W}_q^k, \mathbf{W}_k, \mathbf{W}_q^v, \mathbf{W}_v \in \mathbb{R}^{d\times d}$ are all learnable weight matrices.

After that, we first obtain the contextual bilinear attention weights by transforming $\mathbf{R}_i^k$ into the corresponding attention weight through two fully-connected layers and a softmax function:

$$\begin{aligned}&\mathbf{R}_i^{k'} = \mathrm{ReLU}(\mathbf{W}_R^k\mathbf{R}_i^k),\\ &r_i = \mathbf{W}_r\mathbf{R}_i^{k'}, \mathbf{r} = r_1, r_2, \ldots, r_n,\\ &\gamma = \mathrm{softmax}(\mathbf{r}),\end{aligned} \tag{10}$$

where $\mathbf{W}_R^k \in \mathbb{R}^{d\times d}$, $\mathbf{W}_r \in \mathbb{R}^{1\times d}$, and $\mathbf{R}_i^{k'}$ is the converted bilinear query-key representation, r_i is the i^{th} element in $\mathbf{r}$. Element γ_i in γ represents the i^{th} key-value pair's contextual attention weight.

Secondly, we conduct a squeeze excitation calculation using all bilinear query-key representations $\{\mathbf{R}_i^{k'}\}_{i=1}^n$ for channel-wise attention weight calculation, such operation aggregates $\{\mathbf{R}_i^{k'}\}_{i=1}^n$ via average pooling, generating a channel descriptor: $\mathbf{C} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{R}_i^{k'}$. Then we leverage the sigmoid activation on it to obtain channel-wise attention weight:

$$\mathbf{C}' = \sigma(\mathbf{W}_c\mathbf{C}), \tag{11}$$

where $\mathbf{W}_c \in \mathbb{R}^{d\times d}$. σ is the sigmoid activation.

In the final step, both contextual and channel-wise attention are simultaneously exploited to output the value $\hat{\mathbf{v}}_i$:

$$\hat{\mathbf{v}}_i = \sum_{i=1}^{n}\gamma_i\mathbf{R}_i^v \odot \mathbf{C}'. \tag{12}$$

We enumerate $\mathbf{q}_i$ in $\mathbf{Q}$, and get a set of values $\hat{\mathbf{V}} = \{\hat{\mathbf{v}}_i\}_{i=1}^n$ mentioned in Eq.(8).

Expansion via ELU. To further improve the second-order feature interactions in Eq.(9), we introduce higher-order feature interactions by equipping the block with Exponential Linear Unit [Barron, 2017], which is formulated as follows:

$$\mathrm{ELU(x)} = \begin{cases}\mathrm{x}, & \mathrm{x} \geq 0\\ \alpha * (\exp(\mathrm{x}) - 1), & \mathrm{x} < 0\end{cases} \tag{13}$$

where the parameter α is set to 1 by default. We apply ELU to two dot product operations in Eq.(9) and obtain the updated Eq.(14):

$$\begin{aligned}\mathbf{R}_i^k &= \mathrm{ELU}(\mathbf{W}_q^k\mathbf{q}) \odot \mathrm{ELU}(\mathbf{W}_k\mathbf{k}_i),\\ \mathbf{R}_i^v &= \mathrm{ELU}(\mathbf{W}_q^v\mathbf{q}) \odot \mathrm{ELU}(\mathbf{W}_v\mathbf{v}_i).\end{aligned} \tag{14}$$

The higher-order interactions brought by ELU can be proved via Taylor expansion. Technically, given two feature vectors $\mathbf{A}$ and $\mathbf{B}$, their exponential bilinear pooling is formulated as:

$$\begin{aligned}
&\exp\left(\mathrm{W}_A\mathrm{A}\right) \odot \exp\left(\mathrm{W}_B\mathrm{B}\right) \\
&= \left[\exp\left(\mathrm{W}_A^1\mathrm{A}\right) \odot \exp\left(\mathrm{W}_B^1\mathrm{B}\right), \ldots, \exp\left(\mathrm{W}_A^D\mathrm{A}\right) \odot \exp\left(\mathrm{W}_B^D\mathrm{B}\right)\right] \\
&= \left[\exp\left(\mathrm{W}_A^1\mathrm{A} + \mathrm{W}_B^1\mathrm{B}\right), \ldots, \exp\left(\mathrm{W}_A^D\mathrm{A} + \mathrm{W}_B^D\mathrm{B}\right)\right] \\
&= \left[\sum_{i=0}^{\infty} r_i^1 \left(\mathrm{W}_A^1\mathrm{A} + \mathrm{W}_B^1\mathrm{B}\right)^i, \ldots, \sum_{i=0}^{\infty} r_i^D \left(\mathrm{W}_A^D\mathrm{A} + \mathrm{W}_B^D\mathrm{B}\right)^i\right],
\end{aligned}$$

where D is the feature vector's dimension, $\mathrm{W}_A^j/\mathrm{W}_B^j$ is the j^{th} row vector in weight matrices $\mathrm{W}_A/\mathrm{W}_B$.

3.2 Higher-order Attention Layer (HAL)

The HAL consists of N identical sub-layers ($N = 2$), each sub-layer includes a BiLinear attention block and a layer normalization module. Similar to CAL described previously, we first transform $\mathbf{H_I}$ and $\mathbf{H_S}$ acquired from Eq.(1) to two Query-Key-Value sets $(\mathbf{Q_I}, \mathbf{K_I}, \mathbf{V_I})$ and $(\mathbf{Q_S}, \mathbf{K_S}, \mathbf{V_S})$. Then the $\mathbf{Q_I}$, $\mathbf{K_S}$, and $\mathbf{V_S}$ are leveraged as queries, keys, and values, respectively, leading to the enhanced $\mathbf{H_I}$:

$$\begin{aligned}
\hat{\mathbf{V}}_\mathbf{I} &= \mathrm{MHA}_{bi}(\mathbf{Q_I}, \mathbf{K_S}, \mathbf{V_S}), \\
\mathbf{H_I} &= \mathrm{LN}(\mathbf{H_I} + \hat{\mathbf{V}}_\mathbf{I}),
\end{aligned} \tag{15}$$

and we take $\mathbf{Q_S}$, $\mathbf{K_I}$, and $\mathbf{V_I}$ to obtain $\mathbf{H_S}$.

After repeating N times, we can acquire the enhanced intent and slot features, i.e., $\mathbf{H_I^N}$, $\mathbf{H_S^N} \in \mathbb{R}^{n\times d}$, which are endowed with the higher-order feature interactions in between.

3.3 Dynamic Feature Fusion Layer (DFFL)

Inspired by [Cornia *et al.*, 2020] which measures the relevance between the output of the cross attention and the input query in the Meshed Decoder, we propose to fuse intent and slot features $\mathbf{H_I^N}$ and $\mathbf{H_S^N}$ dynamically. First, we compute two weight matrices $\alpha_\mathbf{I}$ and $\alpha_\mathbf{S}$ which refers to the relevance between the output and the input query of the last sub-layer in the HAL, and then obtain the fused features $\mathbf{H_{IS}}$, which can be defined as follows:

$$\begin{aligned}
\alpha_\mathbf{I} &= \sigma(\mathbf{W}_I[\mathbf{Q_I^N}, \mathbf{H_I^N}] + b_I), \\
\alpha_\mathbf{S} &= \sigma(\mathbf{W}_S[\mathbf{Q_S^N}, \mathbf{H_S^N}] + b_S), \\
\mathbf{H_{IS}} &= [\alpha_\mathbf{I} \odot \mathbf{H_I^N}, \alpha_\mathbf{S} \odot \mathbf{H_S^N}],
\end{aligned} \tag{16}$$

where $\mathbf{W}_I, \mathbf{W}_S \in \mathbb{R}^{2d\times d}$, b_I and b_S are learnable bias parameters, $\mathbf{Q_I^N}$ and $\mathbf{Q_S^N}$ are acquired from $\mathbf{H_I^N}$ and $\mathbf{H_S^N}$ using different linear projections. Following Eq.(6), we obtain the updated intent and slot features: $\hat{\mathbf{H}_\mathbf{I}}^\mathbf{N}, \hat{\mathbf{H}_\mathbf{S}}^\mathbf{N} \in \mathbb{R}^{n\times d}$.

3.4 Overall Loss Function

We adopt a joint learning scheme to optimize ID and SF tasks simultaneously as [Goo *et al.*, 2018]. Specially, the total loss function is formulated as:

$$\mathcal{L} = -\sum_{j=1}^{M} \hat{\mathbf{y}}^{\mathbf{I},j} \log(\mathbf{y}^{\mathbf{I},j}) - \sum_{j=1}^{M}\sum_{i=1}^{N_j} \hat{\mathbf{y}}_i^{\mathbf{S},j} \log(\mathbf{y}_i^{\mathbf{S},j}), \tag{17}$$

where $\hat{\mathbf{y}}^{\mathbf{I},j}$ and $\hat{\mathbf{y}}_i^{\mathbf{S},j}$ is the gold intent label and gold slot label separately; M is the number of training data and N_j is the number of tokens in the j^{th} data.

4 Experiments

4.1 Datasets and Setting

We conduct major experiments on SNIPS and ATIS. SNIPS has 13,084 utterances for training, 700 for validation, and 700 for testing. ATIS has 4,478 utterances for training, 500 for validation, and 893 for testing. Both datasets follow the partition as in [Goo *et al.*, 2018]. We also conduct additional experiments on the newly released dataset SLURP[1].

We use Adam as the optimizer. The size of hidden vector d is set to 128, batch size is 32, and learning rate is 1e-3. We evaluate the SLU performance about SF using F1 score, ID using accuracy, and sentence-level semantic frame parsing using overall accuracy. We map each word to a vector using the pretrained 300d GloVe embeddings [Pennington *et al.*, 2014]. All experiments in this paper were conducted on a single NVIDIA GeForce GTX 2080 Ti GPU, and the model was implemented in PyTorch.

4.2 Comparison with Baseline Works

Related Works. We list three mostly related baseline works for comparison, i.e., *Stack-Propagation*, *Graph-LSTM*, and *Co-Interactive*. *Stack-Propagation* first learns the ID task and uses the features acquired from hidden layers of ID to enhance the SF task; *Graph-LSTM* introduces a graph LSTM encoder to derive the token level embedding and the sentence-level embedding which are used in SF and ID respectively; *Co-Interactive* introduces attention modules to mutually enrich the representations of utterances in SF and ID, and achieves the state-of-the-art result. They all have the conventional first-order attention involved.

As shown in Table 2, we can see that HAN outperforms the SOTA result. On both SNIPS and ATIS, HAN achieves an absolute 0.32% and 0.84% on accuracy increase on ID task, an absolute 1.31% and 0.56% on F1 score increase on SF task, an absolute 1.5% and 1.27% increase on the overall accuracy.

Furthermore, we also conduct experiments which uses BERT [Devlin *et al.*, 2019] to update the BiLSTM in Self-attentive Embedder module. We can see that HAN can achieve improved accuracy on both datasets with BERT. Moreover, the performance of conventional network w/ BERT is still lower than HAN w/ BERT and HAN w/o BERT, which demonstrates the advantages of the proposed HAN.

4.3 Ablation Study

We also evaluate the contribution of each proposed modules: 1) BiLinear attention block; 2) Dynamic Feature Fusion module; 3) ELU module. We gradually integrate each module into the base setting (BiLSTM + SLU Decoder) and calculate the performance. As shown in Table 3, all these modules can boost the performance of HAN, and it reaches the best when equipping with ELU module (4.57% and 3.13% absolute overall accuracy gain on two datasets), which proves that exploiting the higher-order feature interactions between ID and SF can improve the performance of each other.

[1] https://github.com/pswietojanski/slurp/tree/master/dataset/slurp

Model	SNIPS			ATIS		
	Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)	Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)
Stack-Propagation [Qin *et al.*, 2019]	94.20	98.00	86.90	95.90	96.90	86.50
Graph-LSTM [Zhang *et al.*, 2020]	95.30	98.29	89.71	95.91	97.20	87.57
Co-Interactive [Qin *et al.*, 2021]	95.90	98.80	90.30	95.90	97.70	87.40
HAN (Ours)	**97.21**†	**99.12**†	**91.80**†	**96.46**†	**98.54**†	**88.67**†
Conventional approach w/ BERT	97.02	98.63	91.28	95.91	97.98	88.24
HAN w/ BERT	**98.26**	**99.33**	**93.54**	**97.23**	**98.74**	**89.31**

Table 2: SF and ID results on two benchmark datasets. † indicates the significant improvement over baselines (p <0.05).

Model	SNIPS			ATIS		
	Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)	Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)
BiLSTM + SLU Decoder	94.19	97.79	85.86	95.32	95.63	84.99
+ (Label attention + shallow concat)	94.39	98.03	87.89	95.55	97.52	85.89
+ Conventional attention (First-order)	95.37	98.34	88.12	95.64	97.43	87.01
+ **BiLinear attention block** (Second-order)	95.35	98.43	88.57	95.83	97.43	87.32
+ **Dynamic Feature Fusion** (Second-order)	95.57	98.57	89.43	95.88	97.56	87.57
+ **ELU** (Higher-order)	96.01	98.69	90.43	95.95	97.89	88.12

Table 3: Ablation study of different component of HAN, the sub-layer number in HAL is set to 1. The "shallow concat" means using FFL.

N	ELU	SNIPS			ATIS		
		Slot	Intent	Overall	Slot	Intent	Overall
1	×	95.57	98.57	89.43	95.88	97.56	87.57
	✓	96.01	98.69	90.43	95.95	97.89	88.12
2	×	**95.86**	**98.57**	**89.71**	**95.92**	**97.76**	**87.79**
	✓	**97.21**	**99.12**	**91.80**	**96.46**	**98.54**	**88.67**
3	×	95.65	98.29	89.57	95.75	97.54	87.12
	✓	95.70	98.57	89.86	95.75	97.42	87.23
4	×	95.51	98.00	89.00	95.75	97.31	87.01
	✓	95.65	98.29	89.11	95.77	97.42	87.01
5	×	95.12	97.86	88.86	95.58	97.54	86.67
	✓	95.32	98.14	88.86	95.83	97.31	86.90

Table 4: The optimal number of sub-layers (N) exploration in HAL.

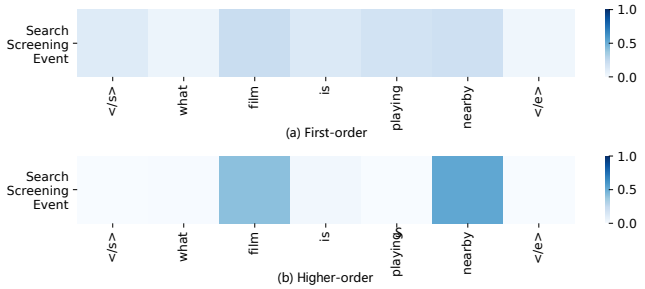

Figure 3: Visualization of attention distribution.

4.4 Optimal Number of Sub-layers in HAL/CAL

We evaluate the number of stacking BiLinear attention blocks in the HAL module, specific, the number N in $\mathbf{H}_\mathbf{I}^\mathbf{N}$ and $\mathbf{H}_\mathbf{S}^\mathbf{N}$. We set the N between 1 to 5 while keeping other components fixed. As shown in Table 4, we can see that the performance of the HAN w/ ELU outperforms the one w/o ELU under all settings of N. In case N equals 2, the performance of HAN on both datasets reaches best.

Similarly, we evaluate the performance of conventional approach under different sub-layer number of CAL on ATIS dataset to find the optimal sub-layer number (the number N in Eq.(5)). As shown in Table 6, it reaches its best performance when $N = 4$, the overall accuracy is 87.23% which is still lower than 88.67% of HAN with $N = 2$. This is probably due to the ELU-equipped BiLinear attention blocks already mining the higher-order interactions between ID and SF tasks, HAN requires less stacking than that of the conventional approach.

4.5 Visualization of Attention

To show the learned higher-order attention, we visualize the first-order attention distribution in Eq.(2) and the higher-order attention distribution of the first sub-equation in Eq.(14) for comparison. In particular, we visualize the attention distribution of the intent feature to each token of slot features. As can be seen in Figure 3, where we use the utterance *"What film is playing nearby"* from the SNIPS dataset. The proposed higher-order attention focuses more on the keyword *"film"* and *"nearby"* that are most relevant to the intent *"Search Screening Event"*, showing a better attention capture ability compared to the conventional first-order attention.

4.6 Robustness Analysis

To verify the robustness of higher-order attention towards hyper-parameters, we evaluate the performance of HAN and conventional approach under different learning rates. From Figure 4, we can see that higher-order model performs better than the first-order model under different learning rate. Besides, we find that under reasonable and task-specific range of the learning rate, i.e., 1e-4 to 1e-2, the higher-order based model performs consistently better than the first-order based

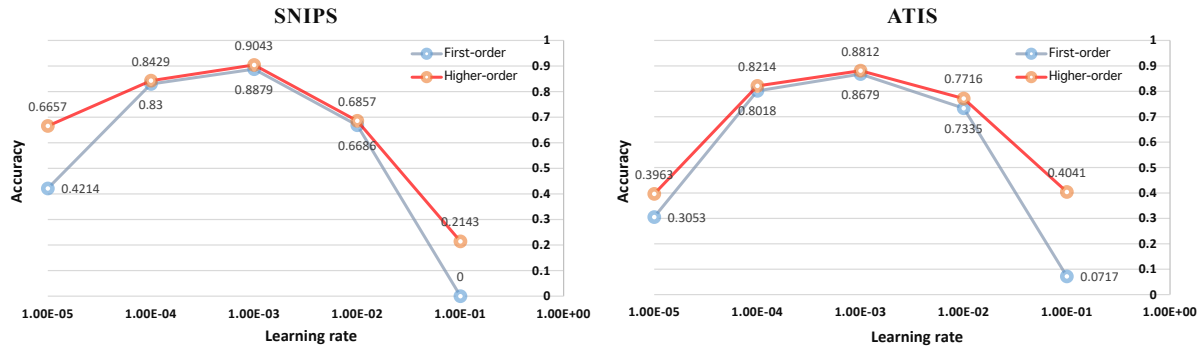

Figure 4: Performance of HAN (the sub-layer number in HAL is set to 1) and conventional model under different learning rates.

Model	HAL	SNIPS			ATIS		
		Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)	Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)
Stack-Propagation	×	94.20	98.00	86.90	95.90	96.90	86.50
	✓	**94.70**	**98.20**	**87.20**	**96.20**	**97.50**	**87.40**
Co-Interactive	×	95.90	**98.80**	90.30	95.90	97.70	87.40
	✓	**96.49**	98.57	**90.78**	**95.98**	**97.87**	**87.68**
HAN (Ours)	×	95.43	98.57	89.29	95.87	97.42	87.12
	✓	**97.21**	**99.12**	**91.80**	**96.46**	**98.54**	**88.67**

Table 5: Impact of higher-order attention on different baselines, i.e., *Stack Propagation*, *Co-Interactive*, and our HAN.

N	ATIS		
	Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)
1	95.64	97.43	87.01
2	95.75	97.54	87.07
3	95.52	97.69	86.98
4	**95.91**	**97.76**	**87.23**
5	95.61	96.98	86.23
HAN (N=2)	**96.46**	**98.54**	**88.67**

Table 6: The optimal number of sub-layers (N) exploration in CAL.

model. When the learning rate is out of this range, i.e., bigger than 1e-2 or smaller than 1e-4, the performance of the first-order based model reduces quickly or even drops to 0 when the learning rate is 0.1. While the performance of higher-order based model, also drops for out of range learning rate, can still maintain compared to the first-order based model, and thus shows its robustness to different learning rates.

4.7 Generalization Analysis

To explore the generalization of the proposed higher-order attention, we further incorporate HAL into two existing baselines (i.e., *Stack-Propagation* and *Co-Interactive*). For *Stack-Propagation*, HAL is inserted after its Self-Attentive Encoder. For *Co-Interactive*, we just replace its Co-Interactive Attention Layer with the HAL. Table 5 shows that baselines with HAL perform better than that with first-order attention in most cases. This verifies the generalization of the higher-order attention on other SLU models to some extent.

Model	SLURP		
	Slot (*F1*)	Intent (*Acc*)	Overall (*Acc*)
Conventional approach	53.95	52.19	30.14
HAN	**55.50**	**53.78**	**33.51**

Table 7: Experimental results on SLURP.

4.8 Evaluation on SLURP

We choose to use SNIPS and ATIS as the evaluation datasets, as they are widely used in previous works, therefore it is easy to compare with previous works. We also conduct an experiment on SLURP, which is a publicly available multi-domain dataset for SLU. As shown in Table 7, under the same setting, our HAN outperforms the conventional approach on three evaluation metrics.

5 Conclusion

This work proposes to trigger the second-order interactions between intent and slot features for the SLU task by introducing a novel BiLinear attention block, which simultaneously exploits both the contextual and channel-wise bilinear attention distributions. Likewise, higher-order feature interactions are further proposed via combining multiple BiLinear attention blocks and applying ELU on them. Furthermore, we introduce the HAN, which achieves new state-of-the-art results on both two benchmark SLU datasets. We also conduct a set of ablation studies to justify the importance of key components in our designed architecture. Lastly, the detailed discussion proves the effectiveness of the higher-order attention in terms of robustness and generalization.

Acknowledgements

Special acknowledgements are given to AOTO-PKUSZ Joint Research Center for its support. This paper is partially supported by Shenzhen Science & Technology Fundamental Research Programs (No: GXWD20201231165807007-20200814115301001).

References

[Ba *et al.*, 2016] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[Barron, 2017] Jonathan T. Barron. Continuously differentiable exponential linear units. *CoRR*, abs/1704.07483, 2017.

[Bastianelli *et al.*, 2020] Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. Slurp: A spoken language understanding resource package. *arXiv preprint arXiv:2011.13205*, 2020.

[Chen *et al.*, 2022] Dongsheng Chen, Zhiqi Huang, and Yuexian Zou. Leveraging bilinear attention to improve spoken language understanding. In *ICASSP*, 2022.

[Cornia *et al.*, 2020] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *CVPR*, 2020.

[Coucke *et al.*, 2018] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *CoRR*, abs/1805.10190, 2018.

[Cui and Zhang, 2019] Leyang Cui and Yue Zhang. Hierarchically-refined label attention network for sequence labeling. *arXiv preprint arXiv:1908.08676*, 2019.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[Fang *et al.*, 2019] Pengfei Fang, Jieming Zhou, Soumava Kumar Roy, Lars Petersson, and Mehrtash Harandi. Bilinear attention networks for person retrieval. In *CVPR*, 2019.

[Goo *et al.*, 2018] Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. Slot-gated modeling for joint slot filling and intent prediction. In *ACL*, 2018.

[Hakkani-Tür *et al.*, 2016] Dilek Hakkani-Tür, Gökhan Tür, Asli Çelikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM. In *INTERSPEECH*, 2016.

[Hemphill *et al.*, 1990] Charles T. Hemphill, John J. Godfrey, and George R. Doddington. The ATIS spoken language systems pilot corpus. In *HLT*, 1990.

[Kim *et al.*, 2018] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. *arXiv preprint arXiv:1805.07932*, 2018.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, 2014.

[Kumari and Ekbal, 2021] Rina Kumari and Asif Ekbal. Amfb: Attention based multimodal factorized bilinear pooling for multimodal fake news detection. *Expert Systems with Applications*, 184:115412, 2021.

[Li *et al.*, 2018] Changliang Li, Liang Li, and Ji Qi. A self-attentive model with gate mechanism for spoken language understanding. In *EMNLP*, 2018.

[Lin *et al.*, 2015] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear CNN models for fine-grained visual recognition. In *ICCV*, 2015.

[Niu *et al.*, 2019] Peiqing Niu, Zhongfu Chen, Meina Song, et al. A novel bi-directional interrelated model for joint intent detection and slot filling. *arXiv preprint arXiv:1907.00390*, 2019.

[Pan *et al.*, 2020] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *CVPR*, 2020.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[Qin *et al.*, 2019] Libo Qin, Wanxiang Che, Yangming Li, Haoyang Wen, and Ting Liu. A stack-propagation framework with token-level intent detection for spoken language understanding. In *EMNLP/IJCNLP*, 2019.

[Qin *et al.*, 2021] Libo Qin, Tailu Liu, Wanxiang Che, Bingbing Kang, Sendong Zhao, and Ting Liu. A co-interactive transformer for joint slot filling and intent detection. In *ICASSP*, 2021.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[Yu *et al.*, 2020] Donghang Yu, Haitao Guo, Qing Xu, Jun Lu, Chuan Zhao, and Yuzhun Lin. Hierarchical attention and bilinear fusion for remote sensing image scene classification. *Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:6372–6383, 2020.

[Zhang *et al.*, 2019] Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and Philip S Yu. Joint slot filling and intent detection via capsule neural networks. In *ACL*, 2019.

[Zhang *et al.*, 2020] Linhao Zhang, Dehong Ma, Xiaodong Zhang, Xiaohui Yan, and Houfeng Wang. Graph lstm with context-gated mechanism for spoken language understanding. In *AAAI*, 2020.

[Zhu *et al.*, 2020] Su Zhu, Ruisheng Cao, and Kai Yu. Dual learning for semi-supervised natural language understanding. *IEEE/ACM TASLP*, 2020.