Towards Multi-Intent Spoken Language Understanding via Hierarchical Attention and Optimal Transport

Xuxin Cheng, Zhihong Zhu, Hongxiang Li, Yaowei Li, Xianwei Zhuang, Yuexian Zou*

School of ECE, Peking University, China
{chengxx, zhihongzhu, lihongxiang, ywl, xwzhuang}@stu.pku.edu.cn, zouyx@pku.edu.cn

Abstract

Multi-Intent spoken language understanding (SLU) can handle complicated utterances expressing multiple intents, which has attracted increasing attention from researchers. Although existing models have achieved promising performance, most of them still suffer from two leading problems: (1) each intent has its specific scope and the semantic information outside the scope might potentially hinder accurate predictions, i.e. *scope barrier*; (2) only the guidance from intent to slot is modeled but the guidance from slot to intent is often neglected, i.e. *unidirectional guidance*. In this paper, we propose a novel Multi-Intent SLU framework termed **HAOT**, which utilizes hierarchical attention to divide the scopes of each intent and applies optimal transport to achieve the mutual guidance between slot and intent. Experiments demonstrate that our model achieves state-of-the-art performance on two public Multi-Intent SLU datasets, obtaining the 3.4 improvement on MixATIS dataset compared to the previous best models in overall accuracy.

1 Introduction

As a core component of task-oriented dialogue systems, spoken language understanding (SLU) aims at accurately comprehending the user's intent by constructing semantic frames (Tur and De Mori 2011; Cheng et al. 2023a; Zhuang, Cheng, and Zou 2023). In general, SLU consists of two subtasks, including slot filling and intent detection (Qin et al. 2019; Zhu et al. 2023a,b; Cheng et al. 2023c). Slot filling could be regarded as a sequence labeling task to predict the slot for each token and intent detection can be treated as a sentence-level semantic classification task to predict the intent of the user.

In real-world scenarios, the given utterance universally includes multiple intents. As a result, researchers begin to explore Multi-Intent SLU (Xu and Sarikaya 2013; Kim, Ryu, and Lee 2017). Gangadharaiah and Narayanaswamy (2019) makes the first attempt to develop a multi-task framework to jointly achieve multi-intent detection and slot filling, aiming to improve the overall performance through more accurately capturing the intents present in the utterances. Recently, Qin et al. (2020) proposes the AGIF framework to introduce the graph attention network (GAT) (Velickovic et al. 2018) and take various intent knowledge to the decoding process of slot filling. Qin et al. (2021) further proposes GL-GIN to develop both a local slot-aware graph and a global intent-slot graph. Song et al. (2022) generates the additional features to leverage the correlation between the slot and intent, which is summarized from the training data. Xing and Tsang (2022b) establishes the heterogeneous label graph and defines relations between slot and intent to leverage the correlations between the different labels, which also improves the performance.

Despite the promising progress that existing Multi-Intent SLU models have made, we discover that most of these models still suffer from two main issues:

Scope Barrier. As shown in Figure 1, different from conventional SLU, the utterance in Multi-Intent SLU has multiple intents and the relationship between these intents is relatively weak (Cheng, Yang, and Jia 2023). As a result, each intent in the utterance has its specific scope and the semantic information outside the scope may potentially impede the accurate predictions. We refer this issue to *scope barrier*. To prevent the negative impact from *scope barrier*, it is vital to divide the scopes for different intent in the original utterance as accurately as possible.

Unidirectional Guidance. Several previous studies (Li, Li, and Qi 2018; Wang, Shen, and Jin 2018; Qin et al. 2019) have verified that slot filling and intent detection are closely tied and it is beneficial to jointly model them. However, most of previous works (E et al. 2019; Qin et al. 2020, 2021) only focus on the unidirectional guidance from intent to slot while the guidance from slot to intent is neglected. In fact, the predicted slot could also help to generate more accurate predictions of intent (Xing and Tsang 2022a). Via achieving bidirectional guidance between slot and intent, the performance of the SLU model can be further enhanced.

In this paper, to tackle the above two issues, we propose a novel SLU framework termed **HAOT**. For *scope barrier*, hierarchical attention (Wang, Chen, and Hu 2019; Geng et al. 2022) is designed to progressively discover the semantic hierarchies layer-by-layer from the utterances in the unsupervised manner. Specifically speaking, we calculate the neighbouring affinity scores among the adjacent tokens, which indicate the tendency to merge the tokens into groups. To keep the consistency of merged groups across different layers, we maintain affinity scores to increase as layer gets deeper. During the training process, the semantically and spatially similar tokens are recursively merged according to affinity scores

*Corresponding author.

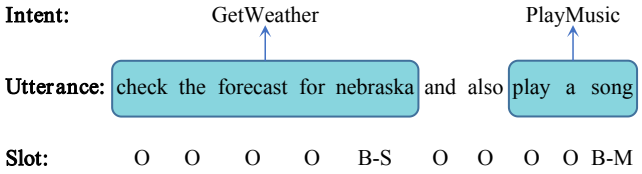

Figure 1: An illustration of *scope barrier* from MixSNIPS dataset, where "B-S" denotes "B-state" and "B-M" denotes "B-music_item". For "play a song", the corresponding intent is "GetWeather" and it is weakly related with "PlayMusic".

and semantic-concentrated clusters are gradually generated. Intuitively, tokens in the same cluster usually have the same intent, so we consider the clusters as the scopes of the different intents. For *unidirectional guidance*, we introduce optimal transport (Kantorovich 2006) to model the mutual guidance between slots and intents. The alignment between slots and intents is regarded as a transportation plan. The distance between the slots and intents is measured by a transportation cost. Through minimizing the transportation cost, the model could achieve the mutual guidance between slots and intents. Experimental results on two public Multi-Intent SLU benchmark datasets MixATIS and MixSNIPS (Hemphill, Godfrey, and Doddington 1990; Coucke et al. 2018) show that HAOT achieves the new state-of-the-art performance. Further analyses also verify the advantages of our method.

In a nutshell, our main contributions are four-fold:

- We propose a novel framework HAOT, which applies the hierarchical attention to tackle *scope barrier* and applies optimal transport to tackle *undirectional guidance*.
- We utilize hierarchical attention mechanism to gradually divide the scopes of each intent in the utterance.
- We utilize optimal transport to creatively treat the mutual guidance between slot and intent as a transportation plan. To our best knowledge, we make the first attempt to apply optimal transport in Multi-Intent SLU.
- Experiment results on two public SLU datasets show that the proposed model outperforms previous best model.

2 Related Work

2.1 Spoken Language Understanding

It is a mainstream to develop a joint SLU model for intent detection and slot filling due to the high correlation between these two subtasks. Liu and Lane (2016) explores how to utilize the alignment information in an encoder-decoder framework to further improve these alignment-based SLU models. Qin et al. (2019) proposes a stack-propagation framework for using the intent information to guide the slot filling. Recently, with the increasing attention to the multi-intent problems, several Multi-Intent SLU models based on graph attention mechanisms have been proposed. GL-GIN (Qin et al. 2021) introduces a non-autoregressive global-local graph interaction framework for parallel decoding in slot filling. Inspired by these success of recent pre-trained models (Li et al. 2021, 2022, 2023b; Zhang et al. 2023a,b; Mao et al. 2023), Zhu et al. (2023c) proposes DGIF to construct a multi-grain intent-slot interactive graph instead of statically incorporating multiple intent information as in previous studies. However, most of existing models still suffer from *scope barrier* and *unidirectional guidance*, which limits the performance. To solve the first problem, Cheng, Yang, and Jia (2023) designs an additional scope recognizer to divide the scopes of different intents. In our approach, we directly add an attention mask to the conventional attention mechanism as an inductive bias, which could help to find the hierarchical structures and form more semantic-concentrated clusters. These generated clusters are regarded as the scopes of the different intents. For the second issue, Xing and Tsang (2022a) proposes the novel Co-guiding Net to achieve the mutual guidance between these two subtasks via a two-stage framework. However, error propagation is an inevitable issue in the two training stages. In our method, we apply optimal transport to model mutual guidance in a single-stage framework, thereby avoiding error propagation, which is more beneficial.

2.2 Hierarchical Attention

Our method is motivated by recent success in hierarchical attention mechanism. Compared with the conventional attention mechanism (Bahdanau, Cho, and Bengio 2015; Vaswani et al. 2017; Zhu, Xu, and Yang 2017), hierarchical attention could leverage more than one level of attention and its superiority has been demonstrated in a range of tasks (Wang et al. 2016). DB-AIAT (Yu et al. 2022) utilizes hierarchical attention to capture the long-term temporal-frequency dependencies and aggregate the global hierarchical contextual information. In this paper, we introduce hierarchical attention to progressively divide the scopes of different intents.

2.3 Optimal Transport

Optimal transport (Kantorovich 2006) is a classic mathematical problem and is initially introduced to solve the problem of minimizing the cost when moving multiple items simultaneously. Peyré, Cuturi et al. (2019) summarizes the theories and the effectiveness of optimal transport. In order to reduce the computational complexity, Kusner et al. (2015) proposes the relaxed form of optimal transport. With the development of machine learning, optimal transport is widely leveraged to compare the different distributions, such as structural matching (Chen et al. 2019), generative models (Arjovsky, Chintala, and Bottou 2017; Balaji, Chellappa, and Feizi 2020), image matching (Zhao et al. 2021), and cross-modal alignment (Chen et al. 2020, 2023; Zhou, Fang, and Feng 2023). In this paper, we utilize the relaxed form of optimal transport to achieve the mutual guidance between slot and intent.

3 Method

In this section, we will begin with the problem definition and then introduce the model architecture, including the encoder, the optimal transport module, the slot decoder, and the intent decoder. Finally, we introduce the final training objective of HAOT. The overview of our method is shown in Figure 2.

3.1 Problem Definition

Given the input utterance $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, where n is the length of $\boldsymbol{x}$. Multi-Intent detection could be formulated

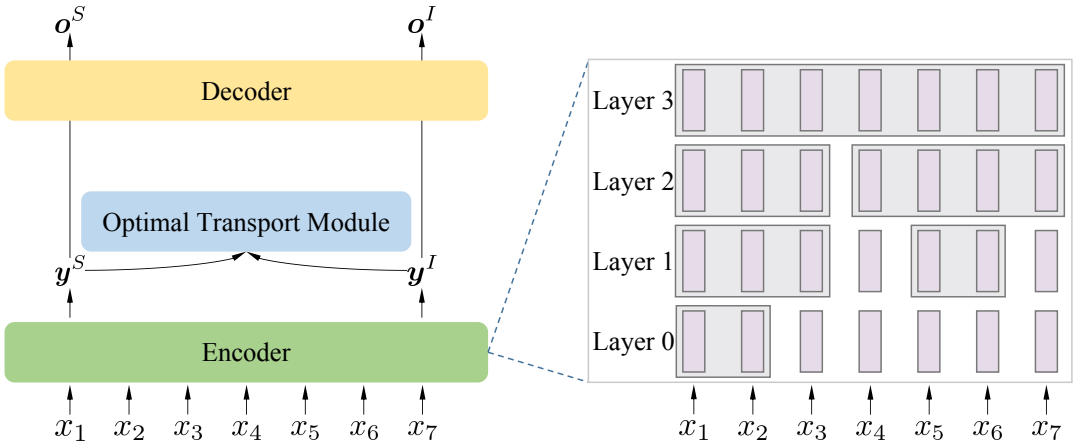

Figure 2: Overview of our proposed framework HAOT, consisting of the Encoder (§3.2), the Optimal Transport Module (§3.3), and the Decoder (§3.4). In the right part, tokens which are shaded indicate that they belong to the same scope.

as a multi-label classification task which predicts the intent labels $\boldsymbol{o}^I = \left(o_1^I, o_2^I, \ldots, o_m^I\right)$, where m denotes the number of intents in $\boldsymbol{x}$. Slot filling is a sequence labeling task which predicts a slot label sequence $\boldsymbol{o}^S = \left(o_1^S, o_2^S, \ldots, o_n^S\right)$.

3.2 Encoder

Our encoder is composed of N_e Transformer (Vaswani et al. 2017) encoder layers, which includes a self-attention layer, a feed-forward layer, and normalization layers. Given an input utterance $\boldsymbol{x}$, the encoder outputs the corresponding hidden states $\boldsymbol{h} = \left(h_1, h_2, \ldots, h_n \in \mathbb{R}^{d_{model}}\right)$, where d_{model} denotes the input and output dimension of Transformer. The preliminary prediction of slots $\boldsymbol{y}^S = \left(y_1^S, y_2^S, \ldots, y_n^S\right)$ and intents $\boldsymbol{y}^I = \left(y_1^I, y_2^I, \ldots, y_n^I\right)$ are calculated as follows:

$$\begin{aligned} y_j^S &= W^S\left(h_j \,||\, \text{Pooling}\left(\boldsymbol{h}\right)\right) + b^S \\ y_j^I &= W^I\left(h_j \,||\, \text{Pooling}\left(\boldsymbol{h}\right)\right) + b^I \end{aligned} \tag{1}$$

where $W^S \in \mathbb{R}^{d_s \times 2d_{model}}$ and $W^I \in \mathbb{R}^{d_i \times 2d_{model}}$ are fully connected matrices, $b^S \in \mathbb{R}^{d_s}$ and $b^I \in \mathbb{R}^{d_i}$ are bias vectors, d_s and d_i denote the categories of the slot labels and intent labels, $||$ denotes the concatenation operation, and Pooling denotes the average polling operation.

To solve *scope barrier*, we replace the conventional attention in the encoder with hierarchical attention. We design an attention mask and apply it to conventional attention, which indicates the trend to merge the tokens that are spatially and semantically similar. As shown in the right part of Figure 2, some clusters are formed in an unsupervised manner as these similar tokens are recursively merged. We regard these clusters as the scopes of different intents. The proposed hierarchical attention is formulated as follows:

$$H = \left(C \odot \text{softmax}\left(\frac{QK^T}{\sqrt{d_h}}\right)\right) V \tag{2}$$

where C denotes the attention mask, $\odot$ denotes Hadamard product, Q denotes query, K denotes key, V denotes value, and d_h denotes the feature dimension.

Inspired by Zhou et al. (2020); Geng et al. (2022); Tseng et al. (2023), to obtain the attention mask C, we first calculate the neighboring attention scores (Wang, Lee, and Chen 2019), which represent the merging trend of adjacent tokens. For any adjacent tokens (x_i, x_{i+1}), the neighboring attention score $s_{i,i+1}$ is calculated as follows:

$$s_{i,i+1} = \frac{\left(x_i W_Q'\right) \cdot \left(x_{i+1} W_K'\right)}{d_s} \tag{3}$$

where W_Q' denotes the query matrix, W_K' denotes the key matrix, and d_s denotes a hyper-parameter as a scaling factor. Note that both W_Q' and W_K' are learnable.

We define the neighboring affinity score $\hat{a}_{i,i+1}$ as the average of the normalized results of $s_{i,i+1}$ and $s_{i+1,i}$:

$$\hat{a}_{i,i+1} = \frac{\text{softmax}\left(s_{i,i+1}\right) + \text{softmax}\left(s_{i+1,i}\right)}{2} \tag{4}$$

To ensure that the merged tokens will not be split again, we add a constraint that the neighboring affinity score should increase as the network goes deeper. The affinity score $a_{i,i+1}^l$ in the l-th layer can be calculated as follows:

$$a_{i,i+1}^l = \begin{cases} a_{i,i+1}^{l-1} + \left(1 - a_{i,i+1}^{l-1}\right) \hat{a}_{i,i+1}, & l \geq 1 \\ \hat{a}_{i,i+1}, & l = 0 \end{cases} \tag{5}$$

Given a token pair (x_i, x_j), the element $C_{i,j}$ of the attention mask matrix C is calculated as follows:

$$C_{i,j} = \begin{cases} \prod_{k=i}^{j-1} a_{k,k+1}, & i < j \\ 1, & i = j \\ C_{j,i}, & i > j \end{cases} \tag{6}$$

The attention mask C is shared by all the attention heads and progressively updated. Many semantically and spatially similar tokens are gradually merged to form several clusters, which are regarded as the scopes of different intents.

3.3 Optimal Transport Module

The optimal transport module is designed to solve *unidirectional guidance* and model the mutual guidance between slot

and intent. We present an innovative perspective to apply optimal transport (Kantorovich 2006) and regard the alignment between slots and intents as the transportation plan. Optimal transport is a classic problem, which is proposed to compare different probability distributions (Santambrogio 2015).

Given a initial state $\alpha = \{\alpha_1, \ldots, \alpha_p\}$ before transportation, a final state $\beta = \{\beta_1, \ldots, \beta_q\}$ after transportation, and the unit cost function $m(\alpha_i, \beta_j)$ represents the unit transport cost from the i-th position in α to the j-th position in β. The objective of optimal transport is to develop a transport plan $\mathbf{T}$ to minimize the total transport cost $\mathcal{D}(\alpha, \beta)$, where each element $\mathbf{T}_{i,j}$ denotes the mass transported from α_i to β_j. The total cost $\mathcal{D}(\alpha, \beta)$ is calculated as follows:

$$\begin{aligned} \mathcal{D}(\alpha, \beta) &= \min_{\mathbf{T} \geq 0} \sum_{i=1}^{p} \sum_{j=1}^{q} \mathbf{T}_{i,j} \cdot m(\alpha_i, \beta_j) \\ s.t. \sum_{j=1}^{q} \mathbf{T}_{i,j} &= \alpha_i, \forall i \in \{1, \ldots, p\}, \\ \sum_{i=1}^{p} \mathbf{T}_{ij} &= \beta_j, \forall j \in \{1, \ldots, q\}. \end{aligned} \tag{7}$$

For the preliminary prediction of slots $\boldsymbol{y}^S$ and intents $\boldsymbol{y}^I$ obtained by the encoder, we utilize optimal transport to measure the distance between them. The corresponding transport cost $\mathcal{D}(\boldsymbol{y}^S, \boldsymbol{y}^I)$ is calculated as follows:

$$\begin{aligned} \mathcal{D}(\boldsymbol{y}^S, \boldsymbol{y}^I) &= \min_{\mathbf{T} \geq 0} \sum_{i=1}^{n} \sum_{j=1}^{n} \mathbf{T}_{i,j} \cdot m(\boldsymbol{y}_i^S, \boldsymbol{y}_j^I) \\ s.t. \sum_{j=1}^{n} \mathbf{T}_{i,j} &= \boldsymbol{y}_i^S, \forall i \in \{1, \ldots, n\}, \\ \sum_{i=1}^{n} \mathbf{T}_{i,j} &= \boldsymbol{y}_j^I, \forall j \in \{1, \ldots, n\}. \end{aligned} \tag{8}$$

We leverage cosine similarity to define the unit cost function $m(\boldsymbol{y}_i^S, \boldsymbol{y}_j^I)$. As the cosine similarity between $\boldsymbol{y}_i^S$ and $\boldsymbol{y}_j^I$ increases, the corresponding unit cost will be lower:

$$m(\boldsymbol{y}_i^S, \boldsymbol{y}_j^I) = 1 - \cos(\boldsymbol{y}_i^S, \boldsymbol{y}_j^I) \tag{9}$$

For the optimal transport problem, some solutions including Sinkhorn (Cuturi 2013) and IPOT (Xie et al. 2019) bring great time complexity, we follow Kusner et al. (2015) to calculate the relaxed transport distance which removes the second constraint to obtain the lower bound of the accurate solution. Then the optimal solution for each slot prediction $\boldsymbol{y}_i^S$ is to move all its mass to the closest intent prediction $\boldsymbol{y}_j^I$ and the transportation matrix becomes:

$$\mathbf{T}_{i,j} = \begin{cases} \frac{1}{n}, & \text{if } j = \arg\min_{j'} m(\boldsymbol{y}_i^S, \boldsymbol{y}_{j'}^I) \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

Then the transport cost $\mathcal{D}(\boldsymbol{y}^S, \boldsymbol{y}^I)$ becomes:

$$\begin{aligned} \mathcal{D}(\boldsymbol{y}^S, \boldsymbol{y}^I) &= \sum_{i=1}^{n} \sum_{j=1}^{n} \mathbf{T}_{i,j} \cdot m(\boldsymbol{y}_i^S, \boldsymbol{y}_j^I) \\ &= \frac{1}{n} \sum_{i=1}^{n} \min_j m(\boldsymbol{y}_i^S, \boldsymbol{y}_j^I) \end{aligned} \tag{11}$$

Similarly, the transport cost $\mathcal{D}(\boldsymbol{y}^I, \boldsymbol{y}^S)$ from $\boldsymbol{y}^I$ to $\boldsymbol{y}^S$ can be derived like Eq. 11, which is formulated as follows:

$$\mathcal{D}(\boldsymbol{y}^I, \boldsymbol{y}^S) = \frac{1}{n} \sum_{j=1}^{n} \min_i m(\boldsymbol{y}_i^S, \boldsymbol{y}_j^I) \tag{12}$$

The transport loss $\mathcal{L}_{OT}$ is defined as follows:

$$\mathcal{L}_{OT} = \frac{\mathcal{D}(\boldsymbol{y}^S, \boldsymbol{y}^I) + \mathcal{D}(\boldsymbol{y}^I, \boldsymbol{y}^S)}{2} \tag{13}$$

3.4 Decoder

The preliminary slot predictions $\boldsymbol{y}^S = \left(y_1^S, y_2^S, \ldots, y_n^S\right)$ are fed into the slot decoder and the final slot predictions $\boldsymbol{o}^S = \left(o_1^S, o_2^S, \ldots, o_n^S\right)$ are as follows:

$$y_j^{S,F} = \text{softmax}(W_F^S y_j^S + b_F^S) \tag{14}$$

$$o_j^S = \arg\max(y_j^{S,F}) \tag{15}$$

where $W_F^S \in \mathbb{R}^{d_s \times d_s}$ is a fully connected matrix and $b_F^S \in \mathbb{R}^{d_s}$ is a bias vector. Similarly, the preliminary intent predictions $\boldsymbol{y}^I = \left(y_1^I, y_2^I, \ldots, y_n^I\right)$ are fed into the intent decoder and token-level voting (Qin et al. 2021) is applied to obtain the final intent predictions $\boldsymbol{o}^I$:

$$y_j^{I,F} = \text{softmax}(W_F^I y_j^I + b_F^I) \tag{16}$$

$$\boldsymbol{o}^I = \{o_k^I | (\sum_{t=1}^{n} \mathbb{1}[\boldsymbol{I}_{(t,k)} > \frac{1}{2}]) > \frac{n}{2}\} \tag{17}$$

where $W_F^I \in \mathbb{R}^{d_i \times d_i}$ is a fully connected matrix, $b_F^I \in \mathbb{R}^{d_i}$ is a bias vector, and $\boldsymbol{I}_{(t,k)}$ denotes the prediction probability of token t for the intent o_k^I. The prediction of each token is considered as a vote and the votes with a probability greater than 0.5 are considered as the positive votes. Only the intents gotten more than half positive votes in all n tokens are added to the final predictions $\boldsymbol{o}^I$.

3.5 Training Objective

Following previous works (Qin et al. 2020; Song et al. 2022; Cheng, Yang, and Jia 2023), the training objective $\mathcal{L}_S$ of slot filling and the training objective $\mathcal{L}_I$ of intent detection are:

$$\mathcal{L}_S \triangleq -\sum_{j=1}^{n} \sum_{i=1}^{n_S} \hat{\boldsymbol{y}}_j^{i,S} \log\left(\boldsymbol{o}_j^{i,S}\right) \tag{18}$$

$$\mathcal{L}_I \triangleq -\sum_{j=1}^{n} \sum_{i=1}^{n_I} \text{CE}(\hat{\boldsymbol{y}}_j^{i,I}, \boldsymbol{o}_j^{i,I}) \tag{19}$$

$$\text{CE}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \hat{\boldsymbol{y}} \log(\boldsymbol{y}) + (1 - \hat{\boldsymbol{y}}) \log(1 - \boldsymbol{y}) \tag{20}$$

where $\hat{\boldsymbol{y}}_j^{i,S}$ is the gold slot label, $\hat{\boldsymbol{y}}_i^I$ is the gold intent label, n_S is the number of the slot labels, and n_I is the number of the intent labels. The final training objective $\mathcal{L}$ is as follows:

$$\mathcal{L} = \lambda \mathcal{L}_S + (1 - \lambda) \mathcal{L}_I + \gamma \mathcal{L}_{OT} \tag{21}$$

where λ and γ are two hyper-parameters.

4 Experiments

4.1 Datasets and Metrics

We conduct all the experiments on two public Multi-Intent SLU datasets[1], including MixATIS dataset and MixSNIPS dataset (Qin et al. 2020). MixATIS dataset is collected from ATIS (Hemphill, Godfrey, and Doddington 1990) and MixSNIPS dataset is collected from SNIPS (Coucke et al. 2018). The detailed statistics of the MixATIS dataset and the MixSNIPS dataset are shown in Table 1.

Dataset	MixATIS	MixSNIPS
Vocabulary Size	722	11241
Intent Categories	17	6
Slot Categories	116	71
Training Set Size	13162	39776
Validation Set Size	756	2198
Test Set Size	828	2199

Table 1: Dataset statistics.

Following the previous works (Qin et al. 2021; Song et al. 2022; Cheng, Yang, and Jia 2023), we evaluate the performance of slot filling with F1 score, intent detection with accuracy, and the utterance-level semantic frame parsing with the overall accuracy which represents that both slots and intents are predicted correctly in the utterance.

4.2 Implementation Details

We leverage an Adam optimizer (Kingma and Ba 2015) with $\beta_1 = 0.9, \beta_2 = 0.98$, and 4k warm-up updates to optimize parameters in our framework, where we linearly increase the learning rate from 5e-4 to 1e-3. The batch size is set to 32. The number of encoder layers N_e is set to 4, the Transformer input and output dimension d_{model} is set to 128, the number of the attention heads is set to 8, and the dropout ratio is set to 0.1. For all hyper-parameters, we perform several experiments and select the values with the best performance. For all the experiments, we select the model that works the best on *dev* set and evaluate it on *test* set. The training process will early-stop if the total loss $\mathcal{L}$ on the *dev* set does not decrease for 3 epochs in order to avoid overfitting. For hyper-parameter λ, we follow Xing and Tsang (2022a) and set it to 0.1 on MixATIS dataset and 0.2 on MixSNIPS dataset. For the hyper-parameter γ, we set it to 0.5 on both datasets. All the experiments are conducted at an Nvidia V100 GPU.

4.3 Main Results

Experimental results on MixATIS and MixSNIPS are shown in Table 2, from which we have the following observations:

(1) HAOT obtains consistent improvements across all the subtasks and datasets. Specifically, it surpasses the previous state-of-the-art models by 3.4% (Overall), 2.7% (Slot), and 3.1% (Intent) on MixATIS and 2.2% (Overall), 0.6% (Slot), and 0.5% (Intent) on MixSNIPS, respectively. This improvement can be attributed to the proposed hierarchical attention mechanism and the proposed optimal transport module. The hierarchical attention mechanism progressively divides the scopes of different intents, which solves *scope barrier*. The optimal transport module achieves the mutual guidance between slot and intent, which solves *unidirectional guidance*.

(2) Compared to slot filling and intent detection, the improvements in overall accuracy are more significant. We believe the reason is that our model achieves the mutual guidance between slot and intent, which allows them to leverage their initial predictions to stimulate each other. As a result, the correct predictions of these two subtasks could be better aligned, leading to the boosted overall accuracy. Co-guiding Net (Xing and Tsang 2022a) also dedicates to achieve the mutual guidance between slot and intent. However, it adopts a two-stage framework, which leads to error propagation despite the introduction of margin penalty. Whereas our framework only contains one stage, which could avoid error propagation and achieve better performance.

(3) Though ChatGPT[2](OpenAI 2023) has verified its superiority in few-shot learning and zero-shot learning tasks, there is still a performance gap between our proposed HAOT and ChatGPT. Specifically, HAOT outperforms ChatGPT by 42.9% (Overall), 48.7% (Slot), and 16.8% (Intent) on MixATIS. A similar situation could also be observed on MixSNIPS, which suggests that ChatGPT may face challenges in understanding spoken utterances, especially for fine-grained information like slots. As a result, developing the framework for SLU remains a crucial task for the NLP community, demanding additional exploration and investigation.

4.4 Ablation Study

In order to verify the advantages from different perspectives, we conduct the ablation studies on MixATIS and MixSNIPS, whose results are shown in the lower part of Table 2.

Effect of Hierarchical Attention: One of the core contributions of our framework is the proposed hierarchical attention, which can progressively divide the scopes of different intents to solve *scope barrier*. To evaluate the effectiveness of hierarchical attention, we conduct an ablation experiment where we replace hierarchical attention in the encoder with conventional attention and refer it to *HAOT w/o Hierarchical Attention* in Table 2. We could observe the dramatic drops in all metrics on both datasets, which confirms that hierarchical attention can make the positive contribution to SLU. We believe it is because hierarchical attention is helpful to prevent the negative impact of *scope barrier* and improve intent detection, thereby enhancing the F1 score of slot filling and overall accuracy through the mutual guidance.

Since hierarchical attention brings more parameters, such as neighboring attention score $s^l_{i,i+1}$, affinity score $a^l_{i,i+1}$, and attention mask C, a natural question is whether the additional parameters involved in HAOT contribute to the final performance. Following Qin et al. (2020, 2021), we replace the hierarchical attention with the conventional attention and expand the number of layers of the Transformer encoder to six layers to validate that the proposed hierarchical attention

[1]https://github.com/LooperXX/AGIF

[2]https://chat.openai.com

Model	MixATIS			MixSNIPS		
	Overall (Acc)↑	Slot (F1)↑	Intent (Acc)↑	Overall (Acc)↑	Slot (F1)↑	Intent (Acc)↑
Bi-Model (Wang, Shen, and Jin 2018)	34.4	83.9	70.3	63.4	90.7	95.6
SF-ID (E et al. 2019)	34.9	87.4	66.2	59.9	90.6	95.0
Stack-Propagation† (Qin et al. 2019)	40.1	87.8	72.1	72.9	94.2	96.0
Joint Multiple ID-SF† (Gangadharaiah and Narayanaswamy 2019)	36.1	84.6	73.4	62.9	90.6	95.1
AGIF♢ (Qin et al. 2020)	40.8	86.7	74.4	74.2	94.2	95.1
GL-GIN♢ (Qin et al. 2021)	43.5	88.3	76.3	75.4	94.9	95.6
LR-Transformer‡ (Cheng, Yang, and Jia 2021)	43.3	88.0	76.1	74.9	94.4	96.6
SDJN♢ (Chen, Zhou, and Zou 2022)	44.6	88.2	77.1	75.7	94.4	96.5
GISCo♢ (Song et al. 2022)	48.2	88.5	75.0	75.9	95.0	95.5
Co-guiding Net♢ (Xing and Tsang 2022a)	51.3	89.8	79.1★	77.5★	95.1	97.7
ReLa-Net♢ (Xing and Tsang 2022b)	52.2★	90.1★	78.5	76.1	94.7	97.6
DARER^2♢ (Xing and Tsang 2023)	49.0	89.2	77.3	76.3	94.9	96.7
SSRAN♢ (Cheng, Yang, and Jia 2023)	48.9	89.4	77.9	77.5★	95.8★	98.4★
ChatGPT♠ (OpenAI 2023)	34.2	43.7	66.1	39.6	59.4	94.9
HAOT w/o Hierarchical Attention	51.8 (3.8↓)	89.2 (3.6↓)	78.1 (4.1↓)	76.9 (2.8↓)	95.2 (1.2↓)	97.5 (1.4↓)
HAOT w/o Hierarchical Attention + More Parameters	52.1 (3.5↓)	89.4 (3.4↓)	78.3 (3.9↓)	77.1 (2.6↓)	95.5 (0.9↓)	97.6 (1.3↓)
HAOT w/o Optimal Transport	51.4 (4.2↓)	88.9 (3.9↓)	77.6 (4.6↓)	76.6 (3.1↓)	94.6 (1.8↓)	97.2 (1.7↓)
HAOT (ours)	**55.6♣**	**92.8♣**	**82.2♣**	**79.7♣**	**96.4♣**	**98.9♣**

Table 2: Results on MixATIS and MixSNIPS datasets. '♣' denotes HAOT outperforms the baselines with $p < 0.01$ under t-test. Results with '♢' indicate that they are from the original papers, results with '†' indicate that they are from Qin et al. (2020), and results with '‡' indicate that they are from Cheng, Yang, and Jia (2023). '★' denotes the previous best results, and the results with '♠' are obtained based on our implementation. Best results are highlighted in bold.

rather than the extra parameters works. We refer it to *HAOT w/o Hierarchical Attention + More Parameters* in Table 2. We observe that there is still a significant performance gap between HAOT and the SLU model with more parameters, which verifies that this improvement indeed comes from the hierarchical attention rather than the involved parameters.

Effect of Optimal Transport: Another core contribution of our framework is the creative application of optimal transport to achieve the mutual guidance between slot and intent. To validate the effectiveness of optimal transport, we remove $\mathcal{L}_{OT}$ in Eq. 21 and refer it to *HAOT w/o Optimal Transport* in Table 2. We can observe that the absence of optimal transport leads to 4.2% and 3.1% overall accuracy drops on these two datasets, respectively. This suggests that optimal transport encourages slot and intent to stimulate each other using their initial predictions, which can improve the performance.

4.5 Comparison of Different Optimal Transport Algorithms

Due to the higher time complexity of Sinkhorn (Cuturi 2013) algorithm and IPOT (Xie et al. 2019) algorithm, we utilize the relaxed moving distance (Kusner et al. 2015) to calculate the lower bound of the original problem. To verify the effectiveness of the relaxed moving distance, we replace it with Sinkhorn and IPOT, respectively. The corresponding results are shown in Table 3. HAOT maintains the nearly comparable performance to the exact solution IPOT and has a significant speed advantage, which demonstrates the superiority of the relaxed moving distance of HAOT.

Model	MixATIS				MixSNIPS			
	Overall (Acc)↑	Slot (F1)↑	Intent (Acc)↑	Speed (Time)↓	Overall (Acc)↑	Slot (F1)↑	Intent (Acc)↑	Speed (Time)↓
Sinkhorn	52.8	90.5	79.4	8.7	78.3	96.1	98.5	9.9
IPOT	55.8	92.9	82.3	7.3	79.8	96.6	99.0	8.6
HAOT	55.6	92.8	82.2	3.7	79.7	96.4	98.9	4.8

Table 3: Results on MixATIS and MixSNIPS datasets. Speed denotes the average training time for each batch.

4.6 Case Study

As shown in Table 4, to further demonstrate the capability of our framework in addressing Multi-Intent SLU, we provide a case study on the MixATIS dataset. Both GL-GIN and Co-guiding Net miss the intent `atis_flight` and also fail to predict the slot of *j31* correctly, while HAOT predicts them correctly. This is because the slot filling of *j31* is negatively affected by `atis_quantity` in GL-GIN and Co-guiding Net due to the lack of scope information, resulting in the incorrect slot prediction. In addition, our HAOT uses optimal transport to model the mutual guidance between slot and intent, which leads to the correct intent prediction. Co-guiding Net also fails to predict the slot of *international* accurately. We believe this is because Co-guiding Net suffers from error propagation, while HAOT avoids this problem.

4.7 Low-Resource Setting

Following previous work (Song et al. 2022), we compare our method with one of previous best SLU baselines Co-guiding Net in the low-resource scenarios, where the ratio of training set is varied from {20%, 40%, 60%, 80%, 100%}. The

	Text:	how	many	canadian	airlines	international	flights	use	j31
Ref.	Intent:	atis_flight	atis_quantity						
	Slot:	O	O	B-airline_name	I-airline_name	I-airline_name	O	O	B-aircraft_code
GL-GIN	Intent:	atis_quantity							
	Slot:	O	O	B-airline_name	I-airline_name	I-airline_name	O	O	*O*
Co-guiding Net	Intent:	atis_quantity							
	Slot:	O	O	B-airline_name	I-airline_name	*O*	O	O	*O*
HAOT	Intent:	atis_flight	atis_quantity						
	Slot:	O	O	B-airline_name	I-airline_name	I-airline_name	O	O	B-aircraft_code

Table 4: Case study of GL-GIN, Co-guiding Net, and HAOT on MixATIS dataset. Text in italic indicates incorrect predictions.

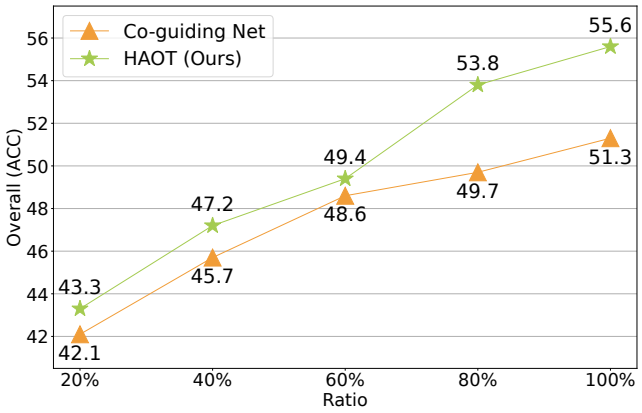

Figure 3: Overall accuracy in the low-resource setting of our method and Co-guiding Net on MixATIS dataset.

comparison results on the MixATIS dataset are illustrated in Figure 3. We could clearly observe that our approach outperforms the baseline in all five proportions of the training set, which further verifies that our hierarchical attention and optimal transport can enhance the performance. Besides, when the ratio of training set exceeds 80%, the superiority of our method becomes more obvious. A possible reason is that as the ratio of training set increases, the *scope barrier* becomes more serious. Via solving *scope barrier* through hierarchical attention, our approach can achieve more remarkable performance improvements as the ratio increases.

4.8 Effect of Pre-trained Model

Pre-trained model has shown its potential in many tasks (Li et al. 2023a,c; Cheng et al. 2023b). To explore the effect of the pre-trained language models, we replace vanilla encoder with RoBETRa (Liu et al. 2019), BERT (Devlin et al. 2019), and XLNet (Yang et al. 2019), respectively. The corresponding results are shown in Table 5, where we find: (1) all three pre-trained language models can further improve the performance of the Multi-Intent SLU models, including GL-GIN, SSRAN, Co-guiding Net, ReLa-Net, and HAOT. We believe the reason is that the pre-trained language models could provide the richer semantic features, which is very beneficial for Multi-Intent SLU. (2) Our HAOT surpasses its counterparts with these pre-trained language models and can achieve the new state-of-the-art performance, which further confirms the superiority of our proposed framework.

Model	**MixATIS**	**MixSNIPS**
RoBERTa	49.7	80.2
GL-GIN + RoBERTa	53.6	82.6
SSRAN + RoBERTa	54.4	83.1
Co-guiding Net + RoBERTa	57.5	85.3
ReLa-Net + RoBERTa	58.4	83.8
HAOT (ours) + RoBERTa	61.8	87.2
BERT	51.6	83.0
GL-GIN + BERT	52.4	83.7
SSRAN + BERT	54.8	84.5
Co-guiding Net + BERT	56.3	85.6
HAOT (ours) + BERT	62.2	87.4
XLNet	52.1	84.8
GL-GIN + XLNet	53.4	85.2
SSRAN + XLNet	55.3	85.6
Co-guiding Net + XLNet	57.6	87.1
HAOT (ours) + XLNet	63.4	89.3

Table 5: Overall performance with three pre-trained models on MixATIS and MixSNIPS datasets.

5 Conclusion

In this paper, we propose the framework HAOT for Multi-Intent SLU, which utilizes hierarchical attention to progressively divide the scopes of different intents and leverages optimal transport to achieve the mutual guidance between slot and intent. Experiments on two public datasets show that our model surpasses previous best models and achieves the new state-of-the-art performance. Model analysis further verifies that our HAOT could also perform well in low-resource scenarios and is compatible with pre-trained models.

Acknowledgements

We thank all the reviewers for the helpful reviews. This paper was partially supported by Shenzhen Science & Technology Research Program (No:GXWD20201231165807007-20200814115301001) and NSFC (No: 62176008).

References

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein Generative Adversarial Networks. In *Proc. of ICML*.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proc. of ICLR*.

Balaji, Y.; Chellappa, R.; and Feizi, S. 2020. Robust Optimal Transport with Applications in Generative Modeling and Domain Adaptation. In *Proc. of NeurIPS*.

Chen, G.; Yao, W.; Song, X.; Li, X.; Rao, Y.; and Zhang, K. 2023. Prompt Learning with Optimal Transport for Vision-Language Models. In *ICLR*.

Chen, L.; Zhang, Y.; Zhang, R.; Tao, C.; Gan, Z.; Zhang, H.; Li, B.; Shen, D.; Chen, C.; and Carin, L. 2019. Improving Sequence-to-Sequence Learning via Optimal Transport. In *Proc. of ICLR*.

Chen, L.; Zhou, P.; and Zou, Y. 2022. Joint Multiple Intent Detection and Slot Filling Via Self-Distillation. In *Proc. of ICASSP*.

Chen, Y.-C.; Li, L.; Yu, L.; El Kholy, A.; Ahmed, F.; Gan, Z.; Cheng, Y.; and Liu, J. 2020. Uniter: Universal image-text representation learning. In *Proc. of ECCV*.

Cheng, L.; Yang, W.; and Jia, W. 2021. A Result based Portable Framework for Spoken Language Understanding. In *Proc. of ICME*.

Cheng, L.; Yang, W.; and Jia, W. 2023. A scope sensitive and result attentive model for multi-intent spoken language understanding. In *Proc. of AAAI*.

Cheng, X.; Cao, B.; Ye, Q.; Zhu, Z.; Li, H.; and Zou, Y. 2023a. Ml-lmcl: Mutual learning and large-margin contrastive learning for improving asr robustness in spoken language understanding. In *Proc. of ACL Findings*.

Cheng, X.; Dong, Q.; Yue, F.; Ko, T.; Wang, M.; and Zou, Y. 2023b. M 3 st: Mix at three levels for speech translation. In *Proc. of ICASSP*.

Cheng, X.; Xu, W.; Zhu, Z.; Li, H.; and Zou, Y. 2023c. Towards spoken language understanding via multi-level multi-grained contrastive learning. In *Proc. of CIKM*.

Coucke, A.; Saade, A.; Ball, A.; Bluche, T.; Caulier, A.; Leroy, D.; Doumouro, C.; Gisselbrecht, T.; Caltagirone, F.; Lavril, T.; et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *ArXiv preprint*.

Cuturi, M. 2013. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In *Proc. of NeurIPS*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of NAACL*.

E, H.; Niu, P.; Chen, Z.; and Song, M. 2019. A Novel Bi-directional Interrelated Model for Joint Intent Detection and Slot Filling. In *Proc. of ACL*.

Gangadharaiah, R.; and Narayanaswamy, B. 2019. Joint Multiple Intent Detection and Slot Labeling for Goal-Oriented Dialog. In *Proc. of NAACL*.

Geng, S.; Yuan, J.; Tian, Y.; Chen, Y.; and Zhang, Y. 2022. HiCLIP: Contrastive Language-Image Pretraining with Hierarchy-aware Attention. In *Proc. of ICLR*.

Hemphill, C. T.; Godfrey, J. J.; and Doddington, G. R. 1990. The ATIS Spoken Language Systems Pilot Corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

Kantorovich, L. V. 2006. On the translocation of masses. *Journal of mathematical sciences*.

Kim, B.; Ryu, S.; and Lee, G. G. 2017. Two-stage multi-intent detection for spoken language understanding. *Multimedia Tools and Applications*.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*.

Kusner, M. J.; Sun, Y.; Kolkin, N. I.; and Weinberger, K. Q. 2015. From Word Embeddings To Document Distances. In *Proc. of ICML*.

Li, C.; Li, L.; and Qi, J. 2018. A Self-Attentive Model with Gate Mechanism for Spoken Language Understanding. In *Proc. of EMNLP*.

Li, J.; Xia, Y.; Yan, R.; Sun, H.; Zhao, D.; and Liu, T. 2021. Stylized Dialogue Generation with Multi-Pass Dual Learning. In *Proc. of NeurIPS*.

Li, Y.; Chen, J.; Li, Y.; Xiang, Y.; Chen, X.; and Zheng, H.-T. 2023a. Vision, Deduction and Alignment: An Empirical Study on Multi-Modal Knowledge Graph Alignment. In *Proc. of ICASSP*.

Li, Y.; Li, Y.; Chen, X.; Zheng, H.-T.; and Shen, Y. 2023b. Active relation discovery: Towards general and label-aware open relation extraction. *Knowledge-Based Systems*.

Li, Y.; Li, Y.; He, Y.; Yu, T.; Shen, Y.; and Zheng, H.-T. 2022. Contrastive learning with hard negative entities for entity set expansion. In *Proc. of SIGIR*.

Li, Y.; Ma, S.; Wang, X.; Huang, S.; Jiang, C.; Zheng, H.-T.; Xie, P.; Huang, F.; and Jiang, Y. 2023c. EcomGPT: Instruction-tuning Large Language Model with Chain-of-Task Tasks for E-commerce. *ArXiv preprint*.

Liu, B.; and Lane, I. R. 2016. Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling. In *Proc. of INTERSPEECH*.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv preprint*.

Mao, T.; Liu, S.; Zhang, Y.; Li, D.; and Shan, Y. 2023. Unified Pretraining Target Based Video-music Retrieval With Music Rhythm And Video Optical Flow Information. *ArXiv preprint*.

OpenAI. 2023. ChatGPT. https://chat.openai.com.

Peyré, G.; Cuturi, M.; et al. 2019. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*.

Qin, L.; Che, W.; Li, Y.; Wen, H.; and Liu, T. 2019. A Stack-Propagation Framework with Token-Level Intent Detection for Spoken Language Understanding. In *Proc. of EMNLP*.

Qin, L.; Wei, F.; Xie, T.; Xu, X.; Che, W.; and Liu, T. 2021. GL-GIN: Fast and Accurate Non-Autoregressive Model for Joint Multiple Intent Detection and Slot Filling. In *Proc. of ACL*.

Qin, L.; Xu, X.; Che, W.; and Liu, T. 2020. AGIF: An Adaptive Graph-Interactive Framework for Joint Multiple Intent Detection and Slot Filling. In *Proc. of EMNLP Findings*.

Santambrogio, F. 2015. Optimal transport for applied mathematicians. *Birkäuser, NY*.

Song, M.; Yu, B.; Quangang, L.; Yubin, W.; Liu, T.; and Xu, H. 2022. Enhancing Joint Multiple Intent Detection and Slot Filling with Global Intent-Slot Co-occurrence. In *Proc. of EMNLP*.

Tseng, A.; Yu, T.; Liu, T. J.; and De Sa, C. 2023. Coneheads: Hierarchy Aware Attention. *ArXiv preprint*.

Tur, G.; and De Mori, R. 2011. *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *Proc. of NeurIPS*.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *Proc. of ICLR*.

Wang, W.; Chen, Z.; and Hu, H. 2019. Hierarchical Attention Network for Image Captioning. In *Proc. of AAAI*.

Wang, Y.; Lee, H.-Y.; and Chen, Y.-N. 2019. Tree Transformer: Integrating Tree Structures into Self-Attention. In *Proc. of EMNLP*.

Wang, Y.; Shen, Y.; and Jin, H. 2018. A Bi-Model Based RNN Semantic Frame Parsing Model for Intent Detection and Slot Filling. In *Proc. of NAACL*.

Wang, Y.; Wang, S.; Tang, J.; O'Hare, N.; Chang, Y.; and Li, B. 2016. Hierarchical attention network for action recognition in videos. *ArXiv preprint*.

Xie, Y.; Wang, X.; Wang, R.; and Zha, H. 2019. A Fast Proximal Point Method for Computing Exact Wasserstein Distance. In *Proc. of UAI*.

Xing, B.; and Tsang, I. 2022a. Co-guiding Net: Achieving Mutual Guidances between Multiple Intent Detection and Slot Filling via Heterogeneous Semantics-Label Graphs. In *Proc. of EMNLP*.

Xing, B.; and Tsang, I. 2022b. Group is better than individual: Exploiting Label Topologies and Label Relations for Joint Multiple Intent Detection and Slot Filling. In *Proc. of EMNLP*.

Xing, B.; and Tsang, I. W. 2023. Relational Temporal Graph Reasoning for Dual-task Dialogue Language Understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Xu, P.; and Sarikaya, R. 2013. Convolutional neural network based triangular crf for joint intent detection and slot filling. In *2013 ieee workshop on automatic speech recognition and understanding*.

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J. G.; Salakhutdinov, R.; and Le, Q. V. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Proc. of NeurIPS*.

Yu, G.; Li, A.; Zheng, C.; Guo, Y.; Wang, Y.; and Wang, H. 2022. Dual-branch attention-in-attention transformer for single-channel speech enhancement. In *Proc. of ICASSP*.

Zhang, D.; Li, S.; Zhang, X.; Zhan, J.; Wang, P.; Zhou, Y.; and Qiu, X. 2023a. SpeechGPT: Empowering Large Language Models with Intrinsic Cross-Modal Conversational Abilities. In *Proc. of EMNLP Findings*.

Zhang, D.; Ye, R.; Ko, T.; Wang, M.; and Zhou, Y. 2023b. DUB: Discrete Unit Back-translation for Speech Translation. In *Proc. of ACL Findings*.

Zhao, W.; Rao, Y.; Wang, Z.; Lu, J.; and Zhou, J. 2021. Towards Interpretable Deep Metric Learning with Structural Matching. In *Proc. of ICCV*.

Zhou, J.; Ma, C.; Long, D.; Xu, G.; Ding, N.; Zhang, H.; Xie, P.; and Liu, G. 2020. Hierarchy-Aware Global Model for Hierarchical Text Classification. In *Proc. of ACL*.

Zhou, Y.; Fang, Q.; and Feng, Y. 2023. CMOT: Cross-modal Mixup via Optimal Transport for Speech Translation. In *Proc. of ACL*.

Zhu, L.; Xu, Z.; and Yang, Y. 2017. Bidirectional Multirate Reconstruction for Temporal Modeling in Videos. In *Proc. of CVPR*.

Zhu, Z.; Cheng, X.; Huang, Z.; Chen, D.; and Zou, Y. 2023a. Enhancing Code-Switching for Cross-lingual SLU: A Unified View of Semantic and Grammatical Coherence. In *Proc. of EMNLP*.

Zhu, Z.; Cheng, X.; Huang, Z.; Chen, D.; and Zou, Y. 2023b. Towards Unified Spoken Language Understanding Decoding via Label-aware Compact Linguistics Representations. In *Proc. of ACL Findings*.

Zhu, Z.; Xu, W.; Cheng, X.; Song, T.; and Zou, Y. 2023c. A dynamic graph interactive framework with label-semantic injection for spoken language understanding. In *Proc. of ICASSP*.

Zhuang, X.; Cheng, X.; and Zou, Y. 2023. Towards Explainable Joint Models via Information Theory for Multiple Intent Detection and Slot Filling. In *Proc. of AAAI*.