applied sciences

MDPI

Article

Multi-Intent Natural Language Understanding Framework for Automotive Applications: A Heterogeneous Parallel Approach

Xinlu Li, Lexuan Zhang, Liangkuan Fang and Pei Cao *

School of Artificial Intelligence and Big Data, Hefei University, Hefei 230061, China; xinlu.li@hfuu.edu.cn (X.L.); zhanglx@stu.hfuu.edu.cn (L.Z.); fanglk@stu.hfuu.edu.cn (L.F.)
* Correspondence: caopei@hfuu.edu.cn

Abstract: Natural language understanding (NLU) is an important aspect of achieving human–machine interactions in the automotive application field, consisting of two core subtasks, multiple-intent detection, and slot filling (ID-SF). However, existing joint multiple ID-SF tasks in the Chinese automotive domain face two challenges: (1) There is a limited availability of Chinese multi-intent corpus data for research purposes in the automotive domain; (2) In the current models, the interaction between intent detection and slot filling is often unidirectional, which ultimately leads to inadequate accuracy in intent detection. A novel multi-intent parallel interactive framework based on heterogeneous graphs for the automotive applications field (Auto-HPIF) was proposed to overcome these issues. Its improvements mainly include three aspects: firstly, the incorporation of the Chinese bidirectional encoder representations from transformers (BERT) language model and Gaussian prior attention mechanism allow each word to acquire more comprehensive contextual information; secondly, the establishment of a heterogeneous graph parallel interactive network efficiently exploits intent and slot information, facilitating mutual guidance; lastly, the application of the cross-entropy loss function to the multi-intent classification task enhances the model's robustness and adaptability. Additionally, a Chinese automotive multi-intent dataset (CADS) comprising 13,100 Chinese utterances, seven types of slots, and thirty types of intents were collected and annotated. The proposed framework model demonstrates significant improvements across various datasets. On the Chinese automotive multi-intent dataset (CADS), the model achieves an overall accuracy of 87.94%, marking a notable 2.07% enhancement over the previous best baseline. Additionally, the model performs commendably on two publicly available datasets. Specifically, it showcases a 3.0% increase in overall accuracy on the MixATIS dataset and a 0.7% improvement on the MixSNIPS dataset. These findings showcase the efficacy and generalizability of the proposed model in tackling the complexity of joint multiple ID-SF tasks within the Chinese automotive domain.

Keywords: automotive applications; spoken language understanding; multi-intent detection; parallel interactive framework; heterogeneous graph

Citation: Li, X.; Zhang, L.; Fang, L.; Cao, P. Multi-Intent Natural Language Understanding Framework for Automotive Applications: A Heterogeneous Parallel Approach. *Appl. Sci.* **2023**, *13*, 9919. https://doi.org/10.3390/app13179919

Academic Editor: Vincent A. Cicirello

Received: 4 August 2023
Revised: 30 August 2023
Accepted: 31 August 2023
Published: 1 September 2023

1. Introduction

Research in the field of automotive applications plays a significant role in advanced automotive technology, improving automotive performance, safety, and enhancing the user experience. It also facilitates the application of intelligence, connectivity, and automation in the automotive industry, making it of vital importance. Common research areas in automotive applications include automotive communication network security [1], interactive autonomous driving automotives [2], and the design of functional application models for in-automotive information and entertainment systems [3]. Establishing interactive tools between automotive users and in-automotive systems is a crucial aspect of achieving intelligence in the automotive industry. In the context of intelligent automotive applications, NLU plays a critical role in the automotive environment. NLU refers to the ability of a computer system to comprehend and process natural language expressions. It

involves techniques such as natural language processing and text classification, aiming to divide the text into different categories or labels. By classifying the intent of user-spoken commands, automotive applications can gain a more accurate understanding of user intentions and requirements, enabling them to provide intelligent and personalized responses and services.

In the multi-intent scenarios of NLU, users may convey multiple related intents simultaneously through a single sentence or a conversation. Given the semantic similarity and overlap among intents, the precise identification and classification of distinct intents within automotive user commands pose significant challenges. Furthermore, the scarcity of data can result in diminished performance of multi-intent detection models. Meanwhile, the endeavor of creating an expansive corpus that encompasses a variety of intents expressed in spoken language adds another layer of complexity to the task. The primary objective of this paper's research is to address the issue of multi-intent detection within the context of automotive user commands. By focusing on this matter, the study aims to elevate the precision and effectiveness of NLU systems in intricate dialogues specific to the automotive application domain, thereby offering substantial assistance in enhancing interactions and performance within automotive applications.

Traditional multi-intent joint detection methods [4,5] often employ fixed weights to assign importance to each intent, which may not meet the demands as more data and tasks are incorporated. Further research has unveiled a robust connection between intent detection and slot filling, and mainstream models [6–8] consider establishing the interconnection between intents and slots, exploring the utilization of joint learning techniques and deep learning algorithms to allocate weights adaptively. By leveraging the interaction between intent detection and slot filling, they aim to enhance the accuracy of intent detection, with a primary focus on single-intent detection tasks. Regarding multi-intent detection tasks, the joint multiple ID-SF model [4] explores a multi-tasking framework with a slot-gating mechanism for combined multiple-intent detection and slot filling. However, this approach fails to furnish detailed intent information to guide token-level slot filling. In the AGIF model [5], an intent–slot graph interaction layer is incorporated to capture the significant correlation between slots and intents. Furthermore, CAI et al. [6] propose a joint multi-intent detection and slot-filling model based on BERT, explicitly mapping slots to intents. However, this model lacks bidirectional slot–intent constraints and underutilizes detailed information in the slot-filling task.

Presently, deep learning algorithms stand as the prevailing method for tackling the intricacies of multi-intent detection. Nevertheless, attaining a high level of precision using these algorithms mandates a substantial reservoir of training data, particularly within widely spoken languages like English. The absence of comprehensive multi-intent datasets in other languages presents formidable hurdles when applying multi-intent detection within those linguistic contexts. Furthermore, the existing joint detection models necessitate meticulous refinement and tailoring to suit distinct application scenarios, thereby securing the achievement of optimal detection performance. To conduct a more comprehensive investigation into Chinese multi-intent detection in the automotive domain, this paper presents the collection and annotation of a Chinese automotive multi-intent dataset (CADS). CADS is an open-source dataset specifically tailored for intelligent conversations related to automotive topics. It encompasses a total of 13,100 Chinese utterances, with each sentence capable of expressing up to three distinct intents. Building upon the CADS dataset, this paper introduces a novel joint framework model for multi-intent detection known as Auto-HPIF (automotive heterogeneous parallel interactive framework). This model integrates the Chinese BERT [7] language model and a Gaussian prior attention mechanism into the encoder stage. It devises a slot–intent parallel interactive framework based on heterogeneous graphs within the interaction process of joint multi-intent detection and slot-filling tasks. Additionally, the cross-entropy loss function is employed for the multi-intent classification task. The main contributions of this research can be summarized as follows:

(1) Addressing the challenges of multi-intent detection in the automotive domain, a Chinese automotive multi-intent dataset CADS was constructed. It contains 13,100 Chinese utterances, seven slots, and thirty intent types. The dataset is composed of multiple data sources, including automotive controls, navigation, and car services, covering diverse language styles and intent types;
(2) An innovative multi-intent joint model specialized for the automotive domain is proposed. Its improvements mainly include three aspects. Firstly, it integrates the Chinese BERT language model and a Gaussian prior attention mechanism within the encoder stage, enhancing the accuracy and precision of semantic feature extraction. Secondly, addressing the tasks of multi-intent detection and slot filling, the model adopts a heterogeneous graph parallel interaction network, thereby further enhancing the exchange of information and interaction between tasks. Lastly, the successful resolution of the challenge of inadequate adaptability in the automotive domain's multi-intent models is achieved by introducing the cross-entropy loss function;
(3) Thorough experimental evaluations were conducted on the CADS dataset alongside two publicly available datasets. The extensive results illustrated that Auto-HPIF significantly enhances the accuracy of both multi-intent classification and slot-filling tasks. By leveraging pre-training methods, it can adapt to the Chinese language style and facilitate more efficient human–machine interactions in the automotive scenario.

2. Related Works

This chapter will delve into cutting-edge research related to multi-intent detection and slot-filling tasks. Firstly, Section 2.1 provides an overview of the recent developments and technological advancements in the field of NLU. Subsequently, Section 2.2 thoroughly examines the context and significance of multi-intent detection and slot-filling tasks while also providing an overview of the current mainstream methods applied in the automotive domain. Moving forward, Section 2.3 introduces the joint interactive framework as an effective approach to address these tasks and meticulously dissects the key technical aspects involved. Finally, Section 2.4 explores the application of pre-trained language models in multi-intent detection and slot-filling tasks and introduces some influential pre-trained models along with their notable accomplishments.

2.1. Technological Advancements in the Field of NLU

The dynamic evolution of deep learning has propelled the field of NLU into a realm of substantial advancements. Neural network technologies, including long short-term memory (LSTM), detectors, classifiers, and the transformer architecture, have become core tools in natural language processing. These technologies enable computers to more accurately comprehend language syntax, semantics, and context, conducting in-depth analyses across various contexts. Furthermore, the rise of pre-trained language models has brought about revolutionary changes in the NLU domain. Through pre-training on large-scale text data, these models capture rich semantic information, thereby enhancing the performance of downstream tasks such as intent detection and slot filling. Within the NLU field, numerous innovative approaches tailored to specific tasks and domains have emerged. These approaches not only expand the application scope of NLU but also provide robust tools and techniques for addressing real-world problems. In the domain of Chinese natural language processing (NLP), due to the complex forms of Chinese characters, the flexibility of syntax structures, and challenges posed by homophones, Chinese NLP tasks encounter unique difficulties. Many innovative methods and technologies have been explored to enhance the accuracy, efficiency, and adaptability of Chinese text processing. For instance, paper addresses the issue of multiple grammar errors and proposes a method for grammatical correction in Chinese text. Against this backdrop, this paper will focus on tackling the problem of multi-intent detection in the context of the Chinese automotive domain.

2.2. Multi-Intent Detection and Slot Filling

Compared with single-intent detection tasks, multi-intent detection tasks require a more fine-grained classification of the text, which needs to be decomposed into multiple subtasks and classified when there are multiple intents in a text. As illustrated in Figure 1, for example, in the sentence "Please lower the temperature of the passenger-side air conditioning to 18 degrees and also help me turn off the dashcam". The first step for the in-automotive system is to determine two intents of the user: air conditioning control and car function control. This level of fine-grained classification aids the system in better understanding user intent, thereby enabling accurate differentiation and processing among multiple tasks.

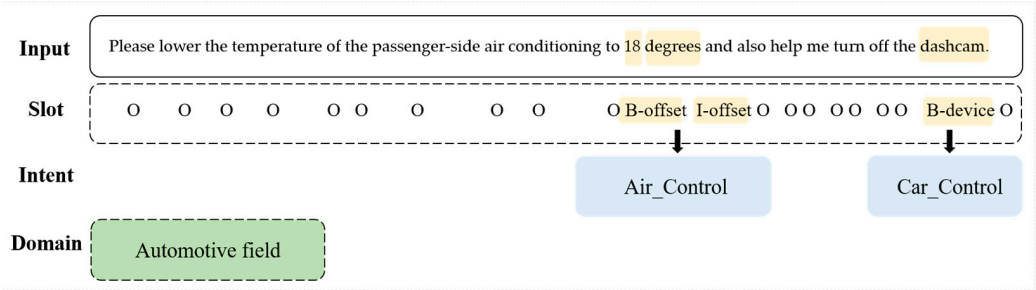

Figure 1. An example of an NLU utterance with multi-intent and slot annotation (IBO format).

As deep neural networks continue to advance, deep learning techniques have become the mainstream approach in multi-intent detection. Article [4] uses a linear chain conditional random field (CRF) classifier to achieve multi-intent detection; however, this approach is prone to errors in the propagation phase. Paper [8] combines structural features and a convolutional neural network (CNN) to propose a multi-intent detection model for multi-intent detection by first calculating the distance matrix DIST as an input layer to highlight the features of the sentence and, after passing through a convolutional and pooling layer, and after a fully-connected and softmax layer to fully connect the feature elements, the task type is outputted by softmax. The probability of the output vector is judged by the score. A score threshold is established to identify sentences that exhibit the intention to complete the multi-intent classification task.

In the in-automotive domain, Zheng et al. [9] examined the utilization of LSTM and GRU for intention detection alongside tasks related to semantic parsing. Given that hierarchical LSTM does not fully utilize contextual information, Firdaus et al. [10] proposed a multi-task model using bidirectional long short-term memory (BiLSTM) and bidirectional-gated recurrent unit (BiGRU) where the input sentence can be considered as a sequence and the global representation of the GRU and LSTM learning sequence is used to recognize intentions. Although the above intention detection models achieve good recognition results, they do not solve the problem of recognizing multiple intentions in the in-automotive domain.

Slot filling is often considered as a sequence annotation task that annotates discourse using a finer granularity to associate certain parts of discourse with predefined slot markers. Traditional slot-filling methods are usually based on conditional random fields (CRFs) [11] of RNNs and other [12] deep learning methods. In recent years, attention mechanisms have also been widely used in NLP tasks, e.g., the literature [13] incorporates attention mechanisms in encoder–decoder models and uses LSTM networks for slot-filling tasks. The literature [14] combines pointer networks and attention mechanisms to improve slot-filling methods.

Despite the good results achieved by the above slot-filling methods, these methods require an extensive corpus of labeled data, so pre-trained language models are also tested to be applied to help train slot-filling models. The literature [15] proposes a WFST-BERT model combining the BERT architecture with a weighted finite-state sensor (WFST), which uses the linguistic representation capability of BERT to generate contextual representations

and improves generalization; the literature [16] focuses on zero-sample learning and applies a momentum comparison approach and a BERT initialization model encoder to accurately capture semantic patterns; In reference [17], a pre-trained BERT model was employed as a semantic feature extractor, resulting in enhanced model generalization. Through feature fusion techniques, the semantic feature vectors from multiple layers of the BERT encoder were combined to establish contextual associations and maximize the utilization of semantic feature information. As a result, the model's performance significantly improved across various tasks.

2.3. Joint Modeling via the Interactive Framework

The interactive framework widely uses joint intent detection and slot-filling models, which have greatly progressed. Compared with the traditional stacked joint model that learns relevant information through ordered hierarchical transfer, the interactive framework focuses more on the interactive impact of the two tasks of slot filling and intention detection simultaneously during the training process. Where intention and semantic slots, as semantic representations of user behavior, share information about user discourse, information from one task can be used by the other task to mutually improve each other's performance, and contextual information from intention detection and semantic slot filling can provide clues to the other task, making the two tasks perform mutually better and thus optimizing global performance.

Gangadharaiah et al. [4] introduced a multi-tasking framework with a slot-gate mechanism for joint multi-intent detection and slot filling. This framework captures features between intents and slots by merging intent information through the treatment of an intent context vector as multiple-intent information. However, this straightforward approach of merging multiple-intent information does not offer detailed intent information at the token level, which is crucial for guiding the slot-filling process. To address this problem, Qin et al. [5] presented the adaptive graph interaction framework (AGIF), which incorporates an intent–slot graph interaction layer to model the robust correlation between slots and intents, and further proposes a global–local graph interaction based on this network GL-GIN [18] to ensure the model runs in parallel and speeds up the inference of the task. In [19], a joint model BIF-SI based on improved multi-intent detection and slot filling is proposed to address the problems that GL-GIN neglects slot-to-intent guidance, multi-intent detection tasks incorrectly capture information of other irrelevant intents, and the quality of contextual semantic feature extraction needs to be further improved. Similarly, [20] proposes a collaborative guidance network framework that allows the intent detection and slot-filling tasks to guide each other. Inspired by heterogeneous networks, a heterogeneous bidirectional flow interaction structure for joint multi-intent detection and slot filling was proposed in [21] and utilizes a word-level windowing mechanism to address the local continuity of slot labels, effectively guiding multi-intent detection with higher accuracy. Moreover, multi-intent detection tasks have been applied to different domains, e.g., in agriculture, Ref. [22] proposed a joint model of agricultural intelligent question and answer (AgHA-IDSF) based on an enhanced heterogeneous attention mechanism, which can effectively jointly recognize intentions and slots in agricultural discourse.

Building upon the aforementioned interaction framework, this paper further introduces a joint model named Auto-HPIF, aiming to investigate multi-intent detection in the Chinese automotive domain using heterogeneous graph parallel interaction networks. The experimental evaluation encompasses three datasets: CADS, along with two publicly available multi-intent datasets, MixATIS and MixSNIPS. To enhance the model's generalization capability in the Chinese automotive domain, this model employs a fine-tuning approach that incorporates pre-trained multi-task learning techniques. This endeavor holds significance for the study of joint intent detection tasks within specific domains.

2.4. Pre-Trained Language Models

Pre-trained word vectors extract distributed representations of words, i.e., word vectors, by unsupervised learning from large-scale text data and apply them to various downstream NLP tasks. In the pre-training process, a large amount of unlabeled text data is usually used to train word vector models. Prevalent pre-training models like ELMO [23], GPT [24], and BERT [7] have demonstrated remarkable success in addressing various NLP tasks, such as textual entailment, semantic similarity, reading comprehension, and question answering. BERT uses a bidirectional transformer architecture that learns context by masking randomly selected words and asking the model to make predictions during pre-training. The emergence of BERT has significantly enhanced the performance of natural language processing tasks. It has led to the latest best results that have been achieved in many areas, such as reading comprehension, question and answer, and document classification. The contextual awareness and semantic expression abilities of Chinese BERT facilitate a more accurate capture of subtle semantic nuances within the text, thereby enhancing disambiguation and improving the efficiency of context-dependent considerations. Leveraging the robust capabilities of the Chinese BERT pre-trained model, this paper enhances the performance of the model in specific tasks, resulting in a more precise understanding of contextual information within dialogues.

3. Corpus Collection and Annotation

This paper presents a dataset construction and annotation work for multi-intent classification in the Chinese automotive domain. Since there is no publicly available multi-intent automotive corpus, this paper collects and constructs a Chinese automotive multi-intent dataset, CADS, and manually processes this dataset corpus according to intent and slot classification, using predefined labels to classify and label each Chinese corpus according to its specific meaning and intent. Table 1 shows the specific count information of utterances containing single-intention, double-intention, and triple-intention. The dataset comprises 13,100 Chinese words, distributed as follows: the training set consists of 10,480 utterances, the test set includes 1310 utterances, and 1310 utterances are allocated to the validation set, maintaining an 8:1:1 ratio.

Table 1. Summary statistics of CADS.

Domain	Number of Intents	Number of Utterances	Intent Types	Slot Types
Automotive	Single intent	5000	17	7
	Double intent	7450	9	7
	Multiple intents	650	4	7
Total	-	13,100	30	7

Specifically, seven slots were set in CADS, including mode, device, offset, location, landmark, song, and singer, and seventeen individual intentions were classified based on the collected in-automotive corpus, as shown in Table 2.

Table 2. Intent details of CADS (The sample in this table is translated from Chinese. For specific examples in Chinese, please refer to Table S1 in the Supplementary Materials).

No.	Intent Label	Sample
1	adjust_ac_temperature_to_number	The air conditioning on the passenger side is set to 25 degrees
2	adjust_ac_windspeed_to_number	Set the air conditioning fan speed to 2.4 notches
3	close_ac	Turn off the air conditioning in the car
4	close_car_device	Could you please close the right rear window a bit?
5	collect_music	I want to listen to my collection of songs

Table 2. *Cont.*

No.	Intent Label	Sample
6	lower_ac_temperate_little	Lower the air conditioning in all positions
7	map_control_query	How do I use the navigation system?
8	music_search_artist_song	Play 'Big Fish' by Zhou Shen
9	navigate_landmark_poi	Are there any ATMs nearby?
10	navigate_poi	I want to go to Shuncheng Service Center via an unblocked route
11	open_ac	I'm a bit cold, let me turn on the air conditioning for a while
12	open_ac_mode	I want to set it to energy-saving mode.
13	open_car_device	Help me open the trunk
14	play _collect_music	Play the music from my collection
15	open _collect_music	Could you please open the collection of songs
16	raise_ac_temperature_little	Increase the temperature of the front right air conditioning
17	view_trans	Show the map in 2D mode

In this paper, a single-intent corpus was randomly combined. A double-intent and triple-intent corpus were synthesized by writing a script to process the dataset, and when there were multiple intents in the corpus, each intent tag was linked using the "#" sign to form a multi-intent tag. A stratified sampling method was used to distribute the intent tags as evenly as possible, and manual screening was performed to remove the unreasonable corpus. Finally, the dataset comprised 7450 dual-intent corpora, 650 tri-intent corpora, and 5000 single-intent corpora. It included a total of thirty in-automotive domain-specific intent types and seven slot labels, providing valuable research data for joint in-automotive intent detection and slot-filling studies. Table 3 lists some examples to illustrate the features and representations of different multi-intentions.

Table 3. Some samples of the CADS multi-intent corpus (The sample in this table is translated from Chinese. For specific examples in Chinese, please refer to Table S2 in the Supplementary Materials).

No.	Multi-Intent Label	Sample
1	close_ac#close_car_device	Turn off the air conditioning in the car, and also close the left rear window
2	collect_music#lower_ac_ temperature_little	This song is nice, then lower the temperature of the driver's air conditioning a bit
3	map_control_query#music_ search_artist_song	Set the navigation to the destination, and then play 'Forget the World' by Li Yugang
4	open_ac_mode#open_car_device	Switch to recirculation mode and enable the Daytime Running Lights function
5	collect_music#lower_ac_ tempera-ture_little#map_control_query	I want to listen to the online music in my collection. Lower the front air conditioning a bit more, and now start the navigation
6	open_collect_music#play_collect_ music#raise_ac_temperature_little	Open the collection of songs, and then play from the collection. Increase the temperature of the right rear air conditioning a bit

The dataset presented in this paper facilitates the training of a joint intent–slot model, allowing for automated recognition of user intent and slot filling. This progress significantly improves the usability of automotive voice assistants for drivers, enhancing their convenience. Notably, the dataset comprises genuine communication scenarios from automotive users, with the majority transformed from spoken language to text format through technical processing. Consequently, the dataset encompasses a range of Chinese dialects and inflections. The inclusion of dialect data serves to amplify phonetic, syntactic, and semantic variations when compared to standard Mandarin. This deliberate inclusion sig-

nificantly improves the diversity of the pre-trained model, rendering it highly adaptable to a multitude of language expressions, thereby facilitating its practical utility.

4. Auto-HPIF Modeling Approaches

Figure 2 illustrates the architecture of the proposed model, consisting of three main modules: (a) the common encoder component, (b) the explicit multi-intent slot task component, and (c) the implicit heterogeneous intent–slot interaction component. Within this framework: (a) functions as the foundational encoder, responsible for extracting feature representations from the input text and obtaining feature vectors for each word; (b) focuses on the explicit task of multi-intent detection and slot filling. It takes the feature vectors obtained from (a) and feeds them into a BiLSTM neural network, ultimately producing coarse-grained slot and intent information; (c) employs a more intricate heterogeneous graph neural network to achieve implicit interactions between intents and slots, facilitating a better understanding of the latent associations between them. These three modules collaborate within the model to achieve the ultimate goal of multi-intent detection and slot filling. This section first introduces the problem definition of multi-intent detection and slot-filling tasks, followed by a detailed description of each individual component.

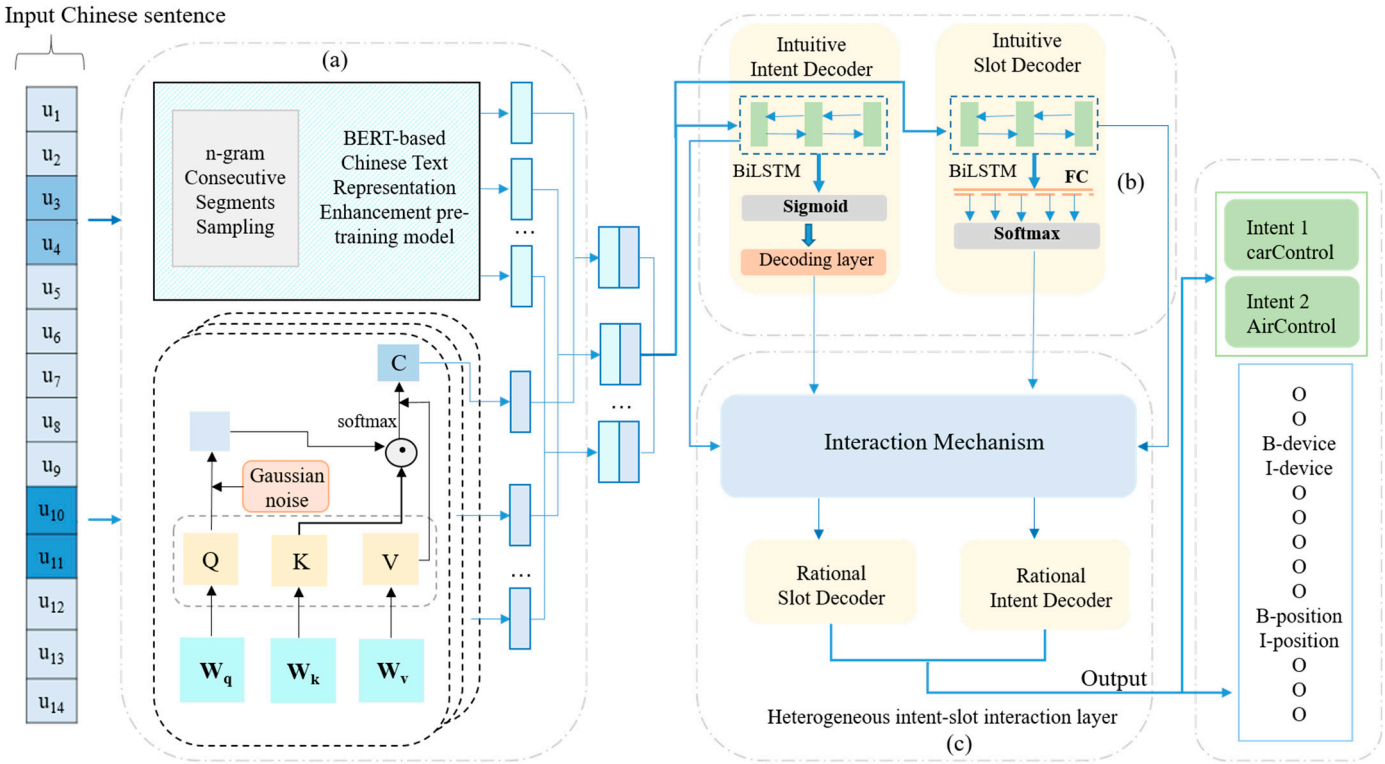

Figure 2. The schematic diagram of a parallel interactive network framework for joint multi-intent detection and slot filling in the Chinese automotive field ((**a**) the common encoder component, (**b**) the explicit multi-intent slot task component, and (**c**) the implicit heterogeneous intent–slot interaction component). (Please refer to Figure S1 in the Supplementary Materials for specific input Chinese sentences).

4.1. Problem Definition

When users engage in spoken expressions within the context of the Chinese automotive domain, their utterances might encompass multiple intents. Multi-intent understanding in the Chinese automotive domain aims to extract crucial information from each conversational turn, with the precise goal of accurately identifying the multiple intents conveyed by the user. Serving as an intermediary between users and the dialogue system, the multi-intent language unit plays the role of conveying vital intent and slot

information to subsequent modules of the dialogue system. In the context of multi-intent detection tasks in the automotive domain, given an in-automotive Chinese language corpus $U = [u_1, u_2, \cdots, u_n]$ with n characters, the primary goal of this study's model is to predict all potential intents associated with the input. This can be delineated into the following two aspects:

Multi-intent detection at the sentence level. Assuming that there are m different intentions (e.g., "automotive control", "air conditioning control", "play music", etc.), multi-intent detection can be considered a multi-label classification problem. A binary variable $I_i \in \{0, 1\}$ is defined for each corpus sample to indicate whether the sample belongs to the i-th intent label. Then, a sample wrapped with m-intent labels can be represented as an m-dimensional vector $I = \{ I_1, I_2, I_3, \cdots, I_m \}^T$; if a sample involves the i-th intent, then $I_i = 1$ otherwise $I_i = 0$;

Character-level slot filling. Slot filling extracts task-specific information, such as entities and attributes, from natural language inputs and assigns this information to predefined slots. The output sequence is $y = \{ y_1, y_2, y_3, \cdots, y_n \}$ where u_i denotes the i-th word in the input sequence and y_j denotes the j-th slot in the output sequence.

4.2. Common Encoder Module Based on BERT

This section aims to explore the application of contextual representation methods to multi-intent detection and slot-filling tasks. Specifically, it will investigate how the input text U can be processed by the contextual representation $E = \{ e_1, e_2, e_3, \cdots, e_n \}$ to more accurately predict the user's intent and the various slot information contained in the dialogue. In order to optimize the utilization of semantic features, this paper introduces a common encoder based on Gaussian prior self-attentiveness, which mainly consists of Chinese BERT and Gaussian prior self-attentiveness mechanisms. Slot-filling tasks and intent detection tasks share underlying semantic representation capabilities. Chinese BERT is pre-trained for Chinese text, and some modifications are made in the input and embedding layers to better process Chinese text. In addition, Chinese BERT introduces an n-gram continuous fragment sampling task to facilitate the special nature of Chinese vocabulary. Suppose $u = \{ u_1, u_2, u_3, \cdots, u_t \}$ is a sequence of input utterances, [CLS] tags and [SEP] tags have been added to the beginning and end of the sequence, and the initial vector of input sentences, denoted as $E = \{ e_1, e_2, e_3, \cdots, e_n \} \in R^{n \times 2d}$, is obtained after processing by the pre-training model Chinese BERT as shown in Equation (1):

$$E = BERT(u_{[CLS]}, u_1, \ldots, u_t, u_{[SEP]}) \tag{1}$$

Considering that the traditional self-attention mechanism cannot take into account the relative positions and distance differences between different parts of the text, the encoder part also adopts an a priori self-attention mechanism based on Gaussian transformation [25]. As shown in Equation (2), the encoder part adds a Gaussian prior $p(z)$ as the prior distribution of the query vector q, $p_{i,j}$ denotes the query vector at position j of the i-th sample $f_q(x_{i,j})$, and denotes the result of the input, computed by the query layer. $p(z)$ is a standard normal distribution, and e is the noise generated from the standard normal distribution to represent the difference between q and $p(z)$. By adding Gaussian noise to the query vector to capture the interrelationships and contextual connections among words in a sentence before calculating the attention scores, the attention weights are smoothed and more broadly distributed by regularizing the attention distribution, ensuring that each word does not focus on just a few nearby words when calculating attention, but can take into account a wider range of contextual information.

$$q_{i,j} = f_q(u_{i,j}) + \varepsilon, \varepsilon \sim p(z), z \sim \mathbb{N}(0, 1) \tag{2}$$

The similarity is scored and normalized using the softmax function (i.e., Equation (3)).

$$C = softmax(\frac{QK^T}{\sqrt{2d}})V \tag{3}$$

The final contextual representation is obtained by stitching the E and C in the feature matrix horizontally $H = [h_1, h_2, \cdots, h_n] \in R^{n \times 2d}$ (i.e., Equation (4)), and this common encoder module output contains all the word vectors and slot label vectors cascaded to obtain the contextual representation.

$$H = [E; C] = [e_1, e_2, \cdots, e_n; c_1, c_2, \cdots, c_n] \tag{4}$$

4.3. Explicit Multi-Intent–Slot Task Module

Inspired by [26], this section aims to decode the discourse representation H into a sequence of intuitive intents and slot labels using an intuitive intent decoder and slot decoder and then feed these token-level slot features and intent information to the next stage to guide downstream modules to interact with high-level intent categorization and precise slot-level details. Specifically, to determine whether the input contains multiple intents and extract task-relevant slot features, the model performs token-level multi-intent detection and slot filling using an intuitive multi-intent decoder and slot decoder. Subsequently, it passes this intent information and slot features to the next stage for additional processing. This involves employing a heterogeneous graph-based slot–intent interaction module to facilitate the integration of extracted intent and slot information and enhance the overall multi-intent prediction. As shown in part (b) of Figure 2, an intuitive intent decoder and slot decoder are introduced to explicitly perform the first intent prediction and slot label sequence generation for the input discourse and pass this information to the next stage as inputs for the implicit interaction.

- **Intuitive Multiple-Intent Decoder.** In the experiments, a bidirectional LSTM is used as an intuitive intent decoder. The bidirectional LSTM obtains a more comprehensive understanding of the input sequence by inputting the input sequence into two LSTMs in temporal and inverse order, respectively, and combining their outputs in time steps where the hidden vector of the decoder at each decoding time step t is calculated as:

$$\overrightarrow{h}_t^{(f)} = BiLSTM(y_t, \overrightarrow{h}_{t-1}^{(f)}) \tag{5}$$

$$\overleftarrow{h}_t^{(b)} = BiLSTM(y_t, \overleftarrow{h}_{t+1}^{(b)}) \tag{6}$$

$$h_t^{\mathrm{II}} = [\overrightarrow{h}_t^{(f)}, \overleftarrow{h}_t^{(b)}] \tag{7}$$

$$y_t^{\mathrm{II}} = sigmoid(W_I^1(\sigma(W_I^2 h_t^{\mathrm{II}} + b_I^2)) + b_I^1) \tag{8}$$

where $\overrightarrow{h}_t^{(f)}$ is the previous hidden state of the forward LSTM unit, $\overleftarrow{h}_t^{(b)}$ is the previous hidden state of the backward LSTM unit, and $y_t^{\mathrm{II}} = \{I_1, I_2, \ldots I_n\}$ is the intention label information distribution for the t-th token in the discourse.

- **Intuitive Slot Decoder.** Similar to the intuitive intent decoder approach, the intuitive slot decoder uses a bidirectional LSTM. Subsequently, a softmax classifier is employed to produce the slot label distribution for each word. As shown in Equation (9), $y_t^{IS} = [l_1, l_2, \cdots, l_t]$ is the distribution of slot labels generated for each word and h_t^{IS} is a specific feature of the slot-filling task.

$$y_t^{IS} = softmax(W_S^1(\sigma(W_S^2 h_t^{IS} + b_S^2)) + b_S^1) \tag{9}$$

4.4. The Implicit Heterogeneous Intent–Slot Interaction Module

The implicit interaction part of the module implements mutual guidance of slot and intent information by further using prior knowledge to identify intents and slots in the user input. The module learns embedded representations based on a heterogeneous graph network and then combines these representations with intent labels to form an embedded multi-intent predictor. When processing user input, the module can process different intent and slot information simultaneously and optimize and adjust them through bidirectional information flow, thus improving the accuracy and efficiency of NLU. As shown in part (c) of Figure 2, a rational intent decoder and slot decoder are introduced to achieve fine-grained multi-intent prediction, and a heterogeneous graph network captures the interaction between different input elements. Specifically, based on the word-level representations and slot label sequences generated by module (b) for multi-intent prediction, embedded representations are further learned using the heterogeneous graph network to improve the ability to encode the semantic relationships between different input elements. Finally, these learned embedded representations are combined with the intent labels and processed by the rational intent decoder and slot decoder to obtain the final fine-grained multi-intent prediction results.

- **Heterogeneous Graph Layer.** This module combines graph attention [27] and self-attention mechanisms to construct a heterogeneous network graph layer. The heterogeneous network layer is used to learn the relationship between each word in a sentence and calculate the importance of other words related to that word. At the same time, prior knowledge is used to guide the model to learn how to match different types of slot information with the input sentences. By integrating the feature representations of these words with higher importance and the degree of matching with the slot predefined, an embedding vector can be generated for that input sentence, and the corresponding intent category and slot values can be extracted from it. Specifically, the predicted intent labels $y_t^{\mathrm{II}} = [s_1, s_2, \ldots, s_t]$ and slot information $y_t^{IS} = [l_1, l_2, L, l_t]$ output by module (b) and each word-level embedding $H = [h_1, h_2, \cdots, h_n]$ and representation are considered as three classes of nodes, and three types of edges are defined:
  1. Adjacency relationship edges between word nodes, representing the contextual information between words in a sentence;
  2. The relationship edge between the intent label and the word node, indicating the connection between the intent label and each word in the input sentence;
  3. Relational edges between slot information and word nodes, representing the connection between slot information and each word in the input sentence.

All the above three edges are directed edges, and their weights indicate the importance or similarity of that edge. During model training and prediction, the weights of these edges are computed and aggregated to generate global context-aware embedding vectors and finally complete the tasks of categorizing intents and filling slots. Through the interaction of these edges, each word in the input sentence and the relationship between them, as well as the degree of match between each word and the target slot value, can be considered simultaneously to generate a comprehensive embedding vector representation. The heterogeneous network graph layer $G = (V, E)$ is constructed using the graph neural network node feature representation and attention mechanism to handle the similarity scores between nodes. The feature representations obtained from the self-attention mechanism and the graph-attention mechanism are stitched together as shown in Equation (10):

$$e_{i,j} = \sigma(W \cdot (\sum_{j=1}^{N} a_{i,j} W^{self} h_j || \sum_{j \in N_i}^{N} a_{i,j} W^{graph} h_j))) \tag{10}$$

where a_{ij} denotes the similarity score between node i and node j, W^{self} and W^{graph} graph are the parameter matrices in the self-attention mechanism and graph-attention mechanisms, respectively, h_j is the feature representation of node j; $||$ denotes the splicing operation,

σ is an activation function, and $e_{i,j}$ denotes the correlation score of each node $j \in V$ with node i.

Then, the attention mechanism is employed to compute the influence of each neighbor node j to node i, and they are weighted and summed to obtain the final node representation h_i', where N_i denotes the set of neighbors of node i, the variables q_i and k_i are the learnable parameters used for generating the query vector and the key vector, respectively, $\beta_{i,j}$ denotes the weight obtained from the attention mechanism score normalized by the softmax function, and finally they are weighted and summed to obtain the feature representation h_i'' of position i, as shown in Equations (11)–(14):

$$\alpha_{i,j} = softmax_j(e_{i,j}) = \frac{exp(e_{i,j})}{\sum_{k \in N_i} exp(e_{i,k})} \tag{11}$$

$$\beta_{i,j} = softmax_j(e_{i,j}{}') = \frac{exp(f(W\, q_i, Wk_j))}{\sum_{k=1}^{n} exp(f(W\, q_i, Wk_k))} \tag{12}$$

$$h_i' = \sigma(\sum_{j \in N_i} \alpha_{i,j} h_j) \tag{13}$$

$$h_i'' = \sigma(\sum_{j \in N_i} \beta_{i,j} h_j) \tag{14}$$

As a result, the feature representation of each word node in the input sentence can be extracted, and the information of slot nodes and intention nodes is continuously updated by the heterogeneous graphical layer interaction mechanism, as shown in Equations (15) and (16):

$$\widetilde{S} = \Big\|_{k=1}^{K} ((e_{i,j}) + \beta_{i,j}) \tag{15}$$

$$\widetilde{L} = \Big\|_{k=1}^{K} \sigma(\sum_{i \in N_I} h_i\prime + \sum_{i \in N_S} h_i'') \tag{16}$$

This approach enables multi-intent classification and provides the necessary information for a rational intent generation model, which helps to perform multi-intent detection tasks. Algorithm 1 illustrates the pseudocode algorithm demonstrating the process of semantic analysis based on the heterogeneous graph network.

Algorithm 1. The diagram of heterogeneous graph-based semantic analysis.

Input: input_sentence, predicted_intent_label $y_t^{\mathrm{II}} = \{s_1, s_2, \ldots, s_t\}$, predicted_slot_information $y_t^{\mathrm{IS}} = [l_1,\ l_2, \cdots, l_t]$
Output: predicted_intent_category $\widetilde{L} = [l_1{}', l_2{}', \ldots, l_t{}']$, predicted_slot_values $\widetilde{S} = [s_1{}',\ s_2{}', \cdots, s_n{}']$
/ Create Word Nodes and Initialize Node Representations */*
Output$_s$ ← *Output of the current sentence*
for *word nodes in input_sentence* **do**
    / Compute Attention Weights */*
    word_attention_weights = compute_attention_weights(word_nodes)
    / Compute Importance Scores for Other Words Related to Each Word */*
    word_importance_scores = compute_importance_scores(word_nodes)
end

Algorithm 1. *Cont.*

```
/* Create Intent Label Node and Slot Information Nodes */
/* Create Edges and Initialize Edge Weights */
for edges in the graph do
    compute_attention_weights(edges)    /* Compute Attention Weights for Edges */
    aggregate_node_representations(word_nodes, edges)
    /* Generate Global Context-Aware Embedding Vector */
    context_aware_embedding_vector = generate_embedding_vector(word_nodes)
    /* Extract Intent Category and Slot Values from Embedding Vector */
    predicted_intent_category = extract_intent_category(context_aware_embedding_vector)
    predicted_slot_values = extract_slot_values(context_aware_embedding_vector)
end
output_s  ←  predicted_intent_category, predicted_slot_values
```

- **Rational Multiple-Intent Decoder.** Through the heterogeneous graphical layer interaction mechanism, the updated intent node containing the slot information is obtained $\widetilde{L} = [l_1\prime, l_2\prime, \ldots, l_t\prime]$; to consider both discourse representation and slot information, this paper continues to use bidirectional LSTM to implement slot information to guide the intent decoder, where it is the encoded hidden state after alignment. d_{t-1} is the hidden state of the decoder at the previous sequential phase, l_t is the slot label vector of the current sequential phase, and the LSTM hidden state is updated by computing the input vector x_t of the current time step as follows:

$$x_t = \left[e_t; d_{t-1}; L_t{}'\right] \tag{17}$$

$$c_t = BiLSTM\left(x_t, c_{t-1}, h_{t-1}\right) \tag{18}$$

$$h_t = tanh(c_t) \tag{19}$$

$$y_t^{RI} = softmax\left(W_s{}' h_t + b_s\right) \tag{20}$$

The hidden state h_t of the current sequential phase is passed to the fully connected layer, and the value of each slot is used as an additional input to generate the intent score y_t^{RI} for the current sequential phase.

- **Rational Slot Decoder.** To enhance the final slot filling task, the predicted multi-intent information is further interacted with the slot information by concatenating the slot node containing the characteristics corresponding to each predicted intent label $\widetilde{S} = [s_1{}', s_2{}', \cdots, s_n{}']$ and the aligned encoder hidden state e as input units to obtain a new sequence of slot labels y_t^{RS} using a method similar to that of the rational multiple-intent decoder.

4.5. Joint Training

To simultaneously train the multi-intent detection and slot-filling tasks, there may be a quantitative imbalance due to the samples of different slot labels. In this study, to address the issue of unbalanced data, a multi-label classification cross-entropy loss function [28] was adopted. This function balances the significance of various labels by weighting the sum of loss terms associated with different labels, thereby enhancing the model's performance. The objective function is denoted as:

$$L_{slot} = \sum_{i=1}^{n} \sum_{j=1}^{m} \left[\left(log\left(1 + \sum_{i,j \in \Omega neg} e^{s_i}\right) + log\left(1 + \sum_{i,j \in \Omega pos} e^{-s_i}\right) \right) \right] \tag{21}$$

where n and m refer to the length of BERT output and the total count of slot categories, respectively.

In this paper, multiple intents are mapped as a set of labels, forming a multi-label classification problem, which is optimized using a binary cross-entropy loss function.

$$L_{intent} \triangleq -\sum_{i=1}^{n}\sum_{j=1}^{m}\left[\hat{y}^{j,I} log(y^{i,I}) + (1-\hat{y}^{j,I}) log(1-y^{i,I})\right] \tag{22}$$

Here, y and $\hat{y}$ represent manually annotated intent tags and decoded sentence-level intent, respectively. Meanwhile, m corresponds to the length of the BERT output, and n denotes the number of intent tags.

To address issues such as overfitting or underfitting of the model, it becomes essential to employ a collective optimization approach for the loss function across multiple tasks instead of individually optimizing the loss function for each task. The definitive formulation of the loss function is presented below:

$$L_{loss} = \lambda_1 L_{slot} + \lambda_2 L_{intent} \tag{23}$$

where λ_1, λ_2 are hyperparameters.

5. Experimental Investigations

This section encompasses experimental evaluations and a comprehensive analysis of the proposed methodology in the paper. We commence by presenting a detailed account of the experimental setup (Section 5.1), encompassing an overview of key components and configurations employed for the evaluations. Following this, baseline results are introduced (Section 5.2), serving as reference points for the enhancements achieved by the proposed approach. Subsequently, a concentrated focus is placed on experiments conducted on the CADS dataset (Section 5.3), showcasing the effectiveness of the Auto-HPIF model within the context of a Chinese in-vehicle environment. Section 5.4 extends the analysis to encompass public datasets, demonstrating the generalizability of the Auto-HPIF model across diverse domains. Furthermore, an ablation study is conducted (Section 5.5), dissecting the impact of the primary innovations proposed in this paper on the model's performance. Lastly, an error analysis (Section 5.6) is conducted to gain deeper insights into failure cases and identify potential directions for future improvements.

5.1. Set Up

This paper uses the self-labeled corpus CADS as an example to validate the effectiveness of the Auto-HPIF framework for the joint task of multi-intent detection in the automotive domain. In addition, two public datasets, MixATIS [4,29] and MixSNIPS [30,31], are introduced to evaluate the generalization of the model to other domains. The MixATIS dataset is a multi-intention dataset that contains conversation data targeting the air travel reservation domain and involves different types of intentions, such as checking flights, booking tickets, and canceling orders. MixATIS includes 13,162 discourses for training, 756 for validation, and 828 for testing. The MixSNIPS dataset contains conversation data for multiple domains (e.g., music, weather, movies, etc.), each with multiple different types of intents and slots. The MixSNIPS dataset includes 39,776 discourses for training, 2198 for validation, and 2199 for testing.

As per previous research, the evaluation of intent detection and slot-filling tasks involved accuracy (Acc) and F1 scores, respectively. To assess the model's performance in the sentence-level semantic frame parsing task, the overall accuracy was employed. Sentence accuracy represents the percentage of correctly predicted intentions and slots across the entire corpus, indicating the overall performance of both tasks. Conversely, overall accuracy denotes the percentage of all sentences where both intentions and slots are correctly predicted.

The experiments were conducted on a Linux server equipped with an NVIDIA GeForce RTX 1080 GPU, utilizing the Python 3.7 and the PyTorch 1.12.1 framework. The model in this paper is built upon Chinese BERT, comprising 12 layers of transformer blocks, 768 hidden states, and 12 attention heads. The BERT base has a total of 110 million parameters. For the MixATIS and MixSNIPS datasets, a batch size of 16 was used, and the initial learning rate was set to 1×10^{-5}.

5.2. Baseline

This vignette presents some typical baseline models in the field of multi-intent detection tasks as follows:

- Attention BiRNN [32] proposes a joint BiRNN model based on self-attentiveness, which predicts intentions by a weighted sum of hidden states;
- Slot-Gated [33] proposes a slot-gating mechanism that directly considers the relationship between SF and ID;
- Bi-Model [34] introduces a bidirectional model that uses BiRNN to decode the intended task and the slot task separately and shares the hidden state information at each time step between the two decoders;
- SF-ID [35] proposes an architecture that provides a direct connection between intent and slot so that they can facilitate each other;
- Stack-Propagation [36] is a stack-propagation architecture that guides SF tasks by combining decoding intent with encoding information;
- Joint Multiple ID-S [4] is a slot-gating model with attention that uses slot context vectors and intent context vectors as slot-gating;
- AGIF [5] is a GNN-based adaptive intent–slot graph interaction network that uses decoded intent and token sequences as nodes;
- GL-GIN [18] is a fast and accurate non-autoregressive model based on GAT that incorporates global–local graph interaction networks;
- SDJN [37] is a self-distillation architecture that passes intent and slot information to each other for cyclic optimization and implements self-distillation by using decoded slots as soft labels for pre-decoded slots;
- Co-Guiding Net [20] proposes a two-stage framework for joint multi-intent detection and slot-filling models.

5.3. Experiments on CADS

Through experimental investigation, this paper compares the performance of three typical joint models for multi-intent detection tasks (AGIF, GL-GIN, and Co-Guiding Net) and Auto-HPIF on the CADS dataset. In this paper, accuracy, recall, F1 value, and precision are used as evaluation metrics, and the performance of each model on each metric is compared. Experimental results on the CADS dataset are given in Table 4.

- **Experimental Findings:** The table observations highlight the following key points: (1) The accuracy of AGIF is 83.27% for overall performance, 90.31% for intent F1 score, and 94.58% for slot-filling F1 score; (2) The improvement in overall accuracy to 84.69% is attributed to the GL-GIN model's explicit modeling of slot dependencies. This enhancement is achieved through the implementation of a local slot-aware graph interaction layer, facilitating effective interconnection among the hidden states of each slot; (3) Co-Guiding Net introduces a novel co-guiding network based on a two-stage framework. For the overall accuracy and slot-filling F1 score metrics, it showed improvements of approximately 1.07% and 0.42%, respectively, compared to GL-GIN. In terms of the F1 score, Co-Guiding Net demonstrated an increase of around 0.09% relative to GL-GIN.

Table 4. Performance comparison (%) of multiple-intent detection and slot filling on CADS datasets with baseline methods.

Models	CADS			
	Overall (Acc)	**Slot (F1)**	**Intent (Acc)**	**Intent (F1)**
AGIF [5]	83.27	94.58	90.31	96.58
GL-GIN [18]	84.96	95.00	90.45	98.62
Co-Guiding Net [20]	85.87	95.40	90.53	98.60
Ours (w/o Chinese-BERT)	86.26	95.90	90.57	98.18
Auto-HPIF (ours)	**87.94**	**96.80**	**90.61**	**98.90**

- **Analysis of Experimental Results:** Compared to most baselines, Auto-HPIF achieved the best results in terms of both slot filling and overall performance. The following is an analysis of the experimental results: (1) In the intent detection task, the overall intent accuracy of Auto-HPIF with Chinese BERT removed outperforms the best baseline model, Co-guiding Net, on the CADS dataset, which indicates that the proposed joint multi-intent detection framework is better adapted to the intent detection task in the in-automotive domain; (2) Auto-HPIF improves the overall accuracy by 2.07% over the best baseline model Co-Guiding Net. This is because Auto-HPIF combines Chinese BERT and Gaussian prior attention mechanism in the shared encoder layer stage, which not only can better learn the semantic information of the input sequence but also, by introducing the Gaussian prior attention mechanism, the model can make full use of the historical information and thus can capture the contextual information of the input sequence more accurately when dealing with the joint multi-intent detection task; (3) The heterogeneous network interaction mechanism introduced by Auto-HPIF allows the slot information and intention information to interact, which also helps to obtain rich intention information and slot semantic representation for the joint multi-intent detection task.

5.4. Experimental on the Public Datasets

This paper presents a comprehensive evaluation conducted on two publicly available datasets. Experimental results on MixATIS and MixSNIPS datasets are given in Table 5. The primary objective of the evaluation was to analyze the performance of multi-intent detection models when applied to the English dataset without the use of BERT or other pre-trained language models. To ensure a fairer comparison, the Chinese BERT pre-trained model in Auto-HPIF was replaced with a bidirectional LSTM during the experiments. The modified model was then compared against other models.

- **Experimental Findings:** The experimental results on the MixATIS dataset showed that Stack-Propagation had lower slot filling F1, intention detection accuracy, and overall accuracy compared to Auto-HPIF (w/o Chinese-BERT) by 2.0%, 3.8%, and 8.5%, respectively. However, some recently proposed models, such as GL-GIN introduced a global–local graph interaction network structure, which improved the overall accuracy to 43.5% and 75.4% on the MixATIS and MixSNIPS datasets, respectively. Another model called SDJN achieved cyclic optimization by exchanging intention and slot information, resulting in an overall accuracy improvement of 1.1% and 0.3% on the MixATIS and MixSNIPS datasets, respectively, relative to GL-GIN. Importantly, Auto-HPIF (w/o Chinese-BERT) achieved the highest intent accuracy of 78.4% and overall accuracy of 47.6% on the MixATIS dataset, surpassing the second-best model by 1.3% and 3.0%, respectively. Additionally, on the MixSNIPS dataset, Auto-HPIF (w/o Chinese-BERT) achieved an overall accuracy of 76.4%, outperforming the second-best model by 0.7%.

Table 5. Evaluating multiple-intent detection and slot-filling performance on MixATIS and MixSNIPS datasets (%).

Models	MixATIS			MixSNIPS		
	Overall (Acc)	Slot (F1)	Intent (Acc)	Overall (Acc)	Slot (F1)	Intent (Acc)
Attention BiRNN [30]	39.1	86.4	74.6	59.5	89.4	95.4
Slot-Gated [31]	35.5	87.7	63.9	55.4	87.9	94.6
Bi-Model [32]	34.4	83.9	70.3	63.4	90.7	95.6
SF-ID [33]	34.9	87.4	66.2	59.9	90.6	95.0
Stack-Propagation [34]	40.1	87.8	72.1	72.9	94.2	96.0
Joint Multiple ID-SF [4]	36.1	84.6	73.4	62.9	90.6	95.1
AGIF [5]	40.8	86.7	74.4	74.2	94.2	95.1
GL-GIN [16]	43.5	88.3	76.3	75.4	94.9	95.7
SDJN [35]	44.6	88.2	77.1	75.7	94.4	**96.5**
Auto-HPIF (w/o Chinese-BERT)	**47.6**	**88.4**	**78.4**	**76.4**	**95.1**	96.0

- **Analysis of Experimental Results:** The experimental findings underscore the potential and applicability of the Auto-HPIF model in addressing complex multi-intent detection tasks. From an algorithmic design perspective, considering whether the proposed Auto-HPIF approach is susceptible to the issue of vanishing gradients is of paramount importance. This study observes that the Auto-HPIF model demonstrates robust performance across multiple indicators, indicating its effective training and alleviation of the gradient vanishing problem. This achievement aligns with the findings in reference, which introduces the oriented stochastic loss descent algorithm. This algorithm addresses the challenge of gradient vanishing, enabling deep networks to be trained without encountering the aforementioned issue.

5.5. Ablation Study

In this section, a series of ablation experiments were designed to assess the individual contributions of various components in the proposed method to the performance of intent detection. The following conclusions can be drawn from Table 6.

Table 6. Ablation study on MixATIS and CADS datasets (%). The numbers marked with an asterisk (*) indicate that the experimental results on the MixATIS dataset were obtained after removing the Chinese BERT pre-training model (%).

Models	CADS			MixATIS		
	Slot (F1)	Intent (Acc)	Overall (OA)	Slot (F1)	Intent (Acc)	Overall (OA)
(w/o) Chinese BERT encoder	95.90	90.57	86.26	-	-	-
(w/o) Gaussian attention mechanism	96.43	90.38	87.17	87.27	76.69	46.25
(w/o) Multilabel_crossentropy	96.37	90.53	87.02	87.49	76.84	46.89
(w/o) Interaction mechanism	94.59	89.92	84.96	78.10	76.36	41.07
Auto-HPIF (ours)	**96.80**	**90.61**	**87.94**	**88.41** *	**77.60** *	**47.63** *

(1) After removing the Chinese BERT pre-training model, the overall accuracy of the Auto-HPIF model on CADS decreases by 0.9%. This is because the Chinese BERT can convert the input text into a high-dimensional vector representation containing rich semantic information and contextual relationships, and these vectors are used as the input of the subsequent model to help improve the accuracy of multi-intent detection.

(2) Omitting the Gaussian prior attention mechanism results in a decrease of 0.37% and 1.38% in the overall accuracy of the Auto-HPIF model on CADS and MixATIS, respectively.

This indicates that the Gaussian prior attention mechanism plays a crucial role in enabling the model to distinguish the importance of different parts of text input and focus on critical information more effectively.

(3) Replacing the multi-label classification cross-entropy loss function, the overall accuracy of the Auto-HPIF model on CADS and MixATIS decreases by 0.43% and 0.74%, respectively, which indicates that the multi-label classification cross-entropy loss function is more effective when applied to the Auto-HPIF model, and its advantage is that it does not require special adjustment of the loss values of class weights and thresholds, which can improve the model stability and reliability of the model.

(4) Removing the heterogeneous network interaction layer from the model and utilizing only the intent-aware slot-filling decoder for the one-way transfer of slot information to intent information results in a decrease of 2.98% and 6.56% in the overall accuracy of the Auto-HPIF model on CVDS and MixATIS, respectively. There are two reasons for this:

- Due to the lack of additional interaction mechanisms, there is insufficient information propagation in the experimental results, leading to poorer performance of the dialogue system. This is because the heterogeneous network interaction mechanism uses different types of nodes and edges in the graph to represent different information, as well as to interact and integrate them to solve tasks. Through the graph attention network (GAT), it achieves interactions between different types of nodes with non-shared weights, accurately capturing node features;
- The self-attention mechanism in the model with the heterogeneous network interaction allows each word to focus on other words in the context and integrate this information into its feature representation, which helps accurately capture features between word-level nodes.

5.6. Error Analysis

During the error analysis phase, a detailed examination of the prediction results of the Auto-HPIF model was conducted to identify potential issues and areas for improvement. Figure 3 provides an example of a slot prediction error. Based on the user input sequence mentioned previously, it can be inferred that the user's intention involves controlling the air conditioner, which includes actions such as turning it on, adjusting the temperature, airspeed, and air direction. However, during the identification of the slot value "cool mode at 22 degrees", the model mistakenly identifies "set" as the B_offset slot label, resulting in an incorrect resolution of the entire slot value as "22 degrees". This error may be attributed to the model learning a linguistic pattern during training, where the term "cooling" becomes associated with the slot label and is subsequently considered a valid feature for prediction. Nevertheless, in real-world scenarios, such patterns may not be encountered frequently, leading the model to make erroneous predictions when faced with new situations.

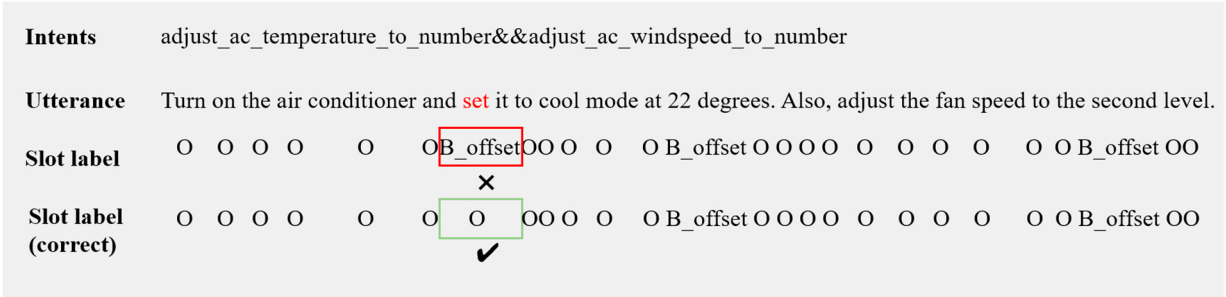

Figure 3. An example of an error slot label for Auto-HPIF on CADS datasets (The utterance" Turn on the air conditioner and set it to cool mode at 22 degrees. Also, adjust the fan speed to the second level." is translated from Chinese. For specific examples in Chinese, please refer to Figure S2 in the Supplementary Materials).

To mitigate such errors in future iterations, several measures can be implemented to improve the model's accuracy and robustness. Firstly, increasing the training data would be beneficial as it enables the model to learn from a more diverse range of examples and generalize better to novel scenarios. Additionally, introducing more features could give the model more context and relevant information for making accurate predictions. Furthermore, optimizing the model architecture can contribute to its overall performance by refining its underlying mechanisms and improving its ability to capture complex patterns and relationships within the data. Complementing these technical enhancements, incorporating manual review and feedback mechanisms can serve as valuable tools for monitoring the model's performance during its application. This iterative feedback loop would facilitate ongoing improvements based on real-world observations and enhance the user experience when the model is deployed in practical scenarios.

6. Conclusions and Future Works

To overcome the challenges of multi-intent detection tasks in the automotive domain, this paper constructs a Chinese automotive domain multi-dataset (CADS) to address the scarcity of Chinese multi-intent corpora. Additionally, this paper proposes a model called Auto-HPIF, specifically designed for joint multi-intent detection and slot-filling tasks in the automotive domain. In the encoder stage, the model introduces the Chinese BERT language model and a Gaussian prior attention mechanism and establishes a heterogeneous graph parallel interaction network to effectively utilize intent and slot information. Furthermore, the cross-entropy loss function is applied to Auto-HPIF to enhance the model's robustness.

Experimental results demonstrate that the Auto-HPIF model achieves outstanding performance in multi-intent detection and slot-filling tasks within the Chinese automotive domain. It not only provides an effective solution for efficient processing of multi-intent understanding but also imbues the field of automotive human–machine interactions and natural language understanding with deeper potential. These findings further solidify the model's value and application prospects in addressing complex and dynamic challenges in real-world scenarios. They offer a strong foundational reference for future research and technological advancements.

Future research will focus on advancing the model's cross-domain applicability and optimizing computational efficiency to achieve widespread application and continuous improvement in real-world scenarios.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app13179919/s1. Table S1: Intent details of CADS; Table S2: Some samples of the CADS multi-intent corpus; Figure S1: The schematic diagram of a parallel interactive network framework for joint multi-intent detection and slot filling in the Chinese automotive field; Figure S2: An example of an error slot label for Auto-HPIF on CADS datasets.

Author Contributions: Conceptualization, L.Z. and X.L.; methodology, L.Z. and X.L.; software, L.Z. and L.F.; validation, L.Z. and P.C.; formal analysis, L.Z.; investigation, L.Z.; resources, L.Z. and X.L.; data curation, L.Z. and L.F.; writing—original draft preparation, L.Z.; writing—review and editing, L.Z.; visualization, L.Z.; supervision, P.C.; project administration, X.L.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by NSFC under Grant No. 6276085 and the Graduate Innovation Project of Hefei University under Grant No. 21YCXL20.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The authors confirm that the data supporting the findings of this study are available within the article and its Supplementary Materials.

Acknowledgments: The authors gratefully acknowledge the National Intelligent Voice Innovation Center for their wonderful support during the process of establishing the dataset.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rathore, R.S.; Hewage, C.; Kaiwartya, O.; Lloret, J. In-automotive communication cyber security: Challenges and solutions. *Sensors* **2022**, *22*, 6679. [CrossRef] [PubMed]
2. Murali, P.K.; Kaboli, M.; Dahiya, R. Intelligent in-automotive interaction technologies. *Adv. Intell. Syst.* **2022**, *4*, 2100122. [CrossRef]
3. Ma, J.; Zuo, Y.; Gong, Z. Design of functional application model in automotive infotainment system-taking automotive music application as an example. In *Congress of the International Association of Societies of Design Research*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 3548–3557.
4. Gangadharaiah, R.; Narayanaswamy, B. Joint multiple intent detection and slot labeling for goal-oriented dialog. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019; pp. 564–569.
5. Qin, L.; Xu, X.; Che, W.; Liu, T. AGIF: An adaptive graph-interactive framework for joint multiple intent detection and slot filling. In *Findings of the Association for Computational Linguistics: EMNLP 2020*; Association for Computational Linguistics: Cedarville, OH, USA, 2020; pp. 1807–1816.
6. Cai, F.; Zhou, W.; Mi, F.; Faltings, B. SLIM: Explicit slot-intent mapping with bert for joint multi-intent detection and slot filling. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 7607–7611.
7. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT (1) 2019, Minneapolis, MN, USA, 2–7 July 2019; pp. 4171–4186.
8. Yang, C.; Feng, C. Multi-intention recognition model with combination of syntactic feature and convolution neural network. *J. Comput. Appl.* **2018**, *38*, 1839. (In Chinese)
9. Zheng, Y.; Liu, Y.; Hansen, J.H.L. Intent detection and semantic parsing for navigation dialogue language processing. In Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017; pp. 1–6.
10. Firdaus, M.; Golchha, H.; Ekbal, A.; Bhattacharyya, P. A deep multi-task model for dialogue act classification, intent detection and slot filling. *Cogn. Comput.* **2021**, *13*, 626–645. [CrossRef]
11. Jeong, M.; Lee, G.G. Triangular-chain conditional random fields. *IEEE Trans. Audio Speech Lang. Process.* **2008**, *16*, 1287–1302. [CrossRef]
12. Yao, K.; Zweig, G.; Hwang, M.Y.; Shi, Y.; Yu, D. Recurrent neural networks for language understanding. In Proceedings of the Interspeech, Lyon, France, 25–29 August 2013; pp. 2524–2528.
13. Simonnet, E.; Camelin, N.; Deléglise, P.; Estève, Y. Exploring the use of attention-based recurrent neural networks for spoken language understanding. In Proceedings of the Machine Learning for Spoken Language Understanding and Interaction NIPS 2015 Workshop (NLUNIPS 2015), Montreal, QC, Canada, 11 December 2015.
14. Zhao, L.; Feng, Z. Improving slot filling in spoken language understanding with joint pointer and attention. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Melbourne, Australia, 15–20 July 2018; pp. 426–431.
15. Abro, W.A.; Qi, G.; Aamir, M.; Ali, Z. Joint intent detection and slot filling using weighted finite state transducer and BERT. *Appl. Intell.* **2022**, *52*, 17356–17370. [CrossRef]
16. Heo, S.H.; Lee, W.K.; Lee, J.H. mcBERT: Momentum Contrastive Learning with BERT for Zero-Shot Slot Filling. In Proceedings of the INTERSPEECH, Incheon, Republic of Korea, 18–22 September 2022; pp. 1243–1247.
17. Chen, Y.; Luo, Z. Pre-trained joint model for intent classification and slot filling with semantic feature fusion. *Sensors* **2023**, *23*, 2848. [CrossRef] [PubMed]
18. Qin, L.; Wei, F.; Xie, T.; Xu, X.; Che, W.; Liu, T. GL-GIN: Fast and accurate non-autoregressive model for joint multiple intent detection and slot filling. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, 1–6 August 2021; pp. 178–188.
19. Deng, F.; Chen, Y.; Chen, X.; Li, J. Multi-intent detection and slot filling joint model of improved GL-GIN. *Comput. Syst. Appl.* **2023**, *32*, 75–83. (In Chinese)
20. Xing, B.; Tsang, I.W. Co-guiding Net: Achieving mutual guidances between multiple intent detection and slot filling via heterogeneous semantics-label graphs. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 159–169.
21. Zhang, Q.; Wang, S.; Li, J. A Heterogeneous Interaction Graph Network for Multi-Intent Spoken Language Understanding. *Neural Process. Lett.* **2023**, 1–19. [CrossRef]
22. Hao, X.; Wang, L.; Zhu, H.; Guo, X. Joint agricultural intent detection and slot filling based on enhanced heterogeneous attention mechanism. *Comput. Electron. Agric.* **2023**, *207*, 107756. [CrossRef]

23. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 2227–2237.
24. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. 2018. Available online: https://paperswithcode.com/paper/improving-language-understanding-by (accessed on 3 August 2023).
25. Guo, M.; Zhang, Y.; Liu, T. Gaussian transformer: A lightweight approach for natural language inference. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 6489–6496.
26. Zhou, P.; Huang, Z.; Liu, F.; Zou, Y. PIN: A novel parallel interactive network for spoken language understanding. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 2950–2957.
27. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *Stat* **2017**, *1050*, 48510–48550.
28. Su, J.; Zhu, M.; Murtadha, A.; Pan, S.; Wen, B.; Liu, Y. Zlpr: A novel loss for multi-label classification. *arXiv* **2022**, arXiv:2208.02955.
29. Hemphill, C.T.; Godfrey, J.J.; Doddington, G.R. The ATIS spoken language systems pilot corpus. In Proceedings of the Speech and Natural Language: Proceedings of a Workshop Held, Hidden Valley, PA, USA, 24–27 June 1990.
30. Coucke, A.; Saade, A.; Ball, A.; Bluche, T.; Caulier, A.; Leroy, D.; Doumouro, C.; Gisselbrecht, T.; Caltagirone, F.; Lavril, T.; et al. Snips voice platform: An embedded spoken language understanding system for private-by-design voice interfaces. *arXiv* **2018**, arXiv:1805.10190.
31. Wu, D.; Ding, L.; Lu, F.; Xie, J. SlotRefine: A fast non-autoregressive model for joint intent detection and slot filling. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 1932–1937.
32. Liu, B.; Lane, I. Attention-based recurrent neural network models for joint intent detection and slot filling. In Proceedings of the INTERSPEECH, San Francisco, CA, USA, 8–12 September 2016; pp. 685–689.
33. Goo, C.W.; Gao, G.; Hsu, Y.K.; Chen, Y.N. Slot-gated modeling for joint slot filling and intent prediction. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 753–757.
34. Wang, Y.; Shen, Y.; Jin, H. A bi-model based rnn semantic frame parsing model for intent detection and slot filling. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 309–314.
35. Niu, P.; Chen, Z.; Song, M. A novel bi-directional interrelated model for joint intent detection and slot filling. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, July 28–2 August 2019; pp. 5467–5471.
36. Qin, L.; Che, W.; Li, Y.; Wen, H.; Liu, T. A stack-propagation framework with token-level intent detection for spoken language understanding. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 2078–2087.
37. Chen, L.; Zhou, P.; Zou, Y. Joint multiple intent detection and slot filling via self-distillation. In Proceedings of the ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 7612–7616.