Research on Intention Detection in Dialogue System

薛 嗣媛

A Thesis submitted to Tokushima University in partial fulfillment of the requirements for the degree of Doctor of Philosophy

2021

Tokushima University
Graduate School of Advanced Technology and Science
Information Science and Intelligent Systems

Contents

1 Introduction 1
1.1 Background and Significant 1
1.2 Research Content and Motivation 1
1.3 Review of the Research Dataset 3
1.4 Organization Structure 4

2 Related works 6
2.1 Intention Detection 7
2.2 Zero-shot Intent Detection 9
2.3 Language Representation Model 10
2.4 Metric Learning 10
2.5 Downstream Fusion Strategy 11
2.6 Capsule Network with Dynamic Routing 11
2.7 Label Embedding Learning 12

3 Intention Detection based on Siamese Neural Network with Triplet Loss 14
3.1 Introduction 14
3.2 The Triplet Siamese Neural Network 16
3.2.1 Whole Framework 16
3.2.2 Siamese RMCNN Neural Network 17
3.2.3 Siamese BERT Neural Network 19
3.2.4 Triplet Loss and Triplet Sampling Strategy 20
3.2.5 Downstream Fusion Strategy 21
3.3 Experiment 23
3.3.1 Dataset 23
3.3.2 Hyper-parameter Selection 24
3.4 Results Comparison and Discussion 25
3.4.1 Baseline Comparison 25
3.4.2 Ablation Studies 29
3.4.3 The Effect of the Encoder Selection 30
3.4.4 The Effect of Sampling Strategy 31
3.4.5 The Effect of the BERT Model Selection 33

3.4.6 The Effect of the Margin Parameter Selection 34
3.4.7 The Visualization of Feature Embeddings..... 34

4 Zero-shot intention detection with Intent-enhanced BertCapsNet..... 37
4.1 Introduction..... 37
4.2 The Attentive Capsule Neural Network..... 39
4.2.1 Encoder Module with Fine-tuned BERT..... 40
4.2.2 Intent-Enhanced Semantic Feature with Label Embedding..... 41
4.2.3 Attentive Capsule Neural Network for Intent Detection 43
4.2.4 Improved Margin Loss Function 45
4.2.5 Zero-shot Intent Detection..... 47
4.3 Experiment..... 48
4.3.1 Dataset 48
4.3.2 Implementation Details..... 48
4.3.3 Evaluation Metric 49
4.4 Results and Discussion 49
4.4.1 Performance Comparison 49
4.4.2 Ablation Studies..... 52
4.4.3 The Effects of Routing Iteration 54
4.4.4 Visualization Feature Learning and Knowledge Transferability..... 54
4.4.5 The Discriminative Utterance Feature Visualization..... 56

5 Dialogue Act Recognition based on Sequence Labeling Method 58
5.1 Introduction..... 58
5.2 Dialogue Act Classification with Token-level Sequence Labelling 59
5.2.1 Feature Extraction..... 59
5.2.2 Sequence Labelling with Linear-Chain CRF..... 61
5.3 Hierarchical Attentive Bi-LSTM Network for Dialogue Act Classification 63
5.3.1 Methodology..... 64
5.3.2 The Hierarchical Conversation Feature Learning..... 65
5.4 Experiment Setting..... 67
5.4.1 Dataset 67
5.4.2 Hyperparameter 68
5.5 Result and Discussion..... 68
5.5.1 The Result and Discussion of Pre-defined Features 68
5.5.2 The Result and Discussion of Hierarchical Neural Network..... 73

6 Conclusion and Future works..... 77
6.1 Conclusion 77
6.1.1 Fusion Triplet Feature Embedding Learning Method For Intention Detection 77
6.1.2 Zero-shot Intention Detection with IE-BertCapsNet..... 78
6.1.3 Dialogue Act Classification with Hierarchical Neural Network 79

6.2 Future Works 80

Bibliography 81

List of Tables

Table 1.1 The utterance samples of single-turn dialogue dataset (SNIPs). 3
Table 1.2 The utterance samples of multi-turn dialogue dataset (SWDA)....3
Table 3.1 The Dataset overviews. 24
Table 3.2 The recognition results on the Snips, ATIS and Facebook (EN) datasets. 27
Table 3.3 The recognition results on the DYDA and MRDA datasets. 27
Table 3.4 The results comparison of basic model and proposed model for different dataset...29
Table 3.5 The comparison of pre-trained BERT models with triplet training. 32
Table 3.6 The comparison of BERT token embedding. 32
Table 3.7 The comparison of RMCNN token embedding. 32
Table 4.1 Data statistics for zero-shot intent detection. 47
Table 4.2 Hyper-parameter Selection in zero-shot intent detection. 47
Table 4.3 The zero-shot intent detection using IE-BertCapsNet on two datasets. 51
Table 4.4 The intent detection results on the known intents of Snips dataset. 51
Table 4.5 The ablation study by varying different components of IE-BertCapsNet. 52
Table 4.6 The utterance feature visualization of known intents and unknown intents....57
Table 5.1 The samples of SWDA dialogue dataset. 67
Table 5.2 The dataset statistic for dialogue act classification based on end-to-end training....67
Table 5.3 Accuracy of different feature representations and weighting methods. 69
Table 5.4 The result of different feature with SVM classifier. 69
Table 5.5 Accuracy with structural feature with CRF. 70
Table 5.6 Accuracy with utterance dependency by using CRF. 72
Table 5.7 The dialogue act classification performance with different approaches. 73
Table 5.8 The ablation studies of the proposed method. 75

List of Figures

Figure 1.1 The Pipeline framework of spoken dialogue system. 2
Figure 3.1 The framework of feature embedding learning model. 16
Figure 3.2 The framework Siamese RMCNN model....17
Figure 3.3 The framework of Siamese BERT model. 19
Figure 3.4 The feature-based strategy of downstream task....21
Figure 3.5 The model of fusion strategy of downstream task. 21
Figure 3.6 The effect of different encoders and sampling strategies on MRDA....31
Figure 3.7 The results comparison of different margin parameter based on different dataset. 34
Figure 3.8 The T-SNE visualization of original data distribution and feature embeddings....36
Figure 4.1 The pipeline of unknown intent detection. 38
Figure 4.2 The framework of IE-BertCapsNet for unknown intent detection. 40
Figure 4.3 The results of different routing iteration numbers. 53
Figure 4.4 The attention visualization of utterance with seen intents and unseen intents....53
Figure 4.5 Comparison of the word attention score between the intent “Book Restaurant” (left) and “Play Music” (right) with their corresponding utterance learned by proposed model without label attention(above) or with label attention(below). 55
Figure 5.1 The example of word-level sequence annotation approach. 61
Figure 5.2 The model structure of end-to-end hierarchical neural network. 64

Acknowledgment

This thesis is not only a research paper but also four years' invaluable and unforgettable experience for me. I was fortunate to have this opportunity to be a part of the AI trend and come here to pursue my Ph.D. Therefore, I must give sincere thanks to the people who help and support me. Without their help and encouragement, I will not be able to complete this journey at Tokushima University.

First and foremost, my greatest thanks to my advisor, Professor Fuji Ren, the director of the Tokushima university of Ren Laboratory. With his insight about the intelligent cognition field and generous guidance, I can access a wealth of resources and advanced research topics. More importantly, Professor Ren is also an extremely kind, caring, and supportive advisor that I could not have asked for more. He always believed in me and encouraged me anytime even though I am not always that confident about myself. Without his support, it would be extremely difficult for me to complete an interdisciplinary transformation.

After that, I would deeply thank the member of A1 Group for their help in life, inspiration in research, and accompanies during in Japan. Thanks to Liu Ning, He Mengjia, Cui Zhichao, Zhang Xudong, Deng Jiawen, Jiao Ziyun, She Tianhao, Liu Wenjie, and Zhou Yangyang. Especially thanks to my first-grade tutor Feng Duo, my labmate Deng Jiawen and Zhang Qian. Their valuable merits are worth learning all my life.

Next, I would also give sincere thanks to Dr. Shun Nishide and Dr. Kang Xin for their insightful help on my research study. Specifically, I would thank Professor Kenji Terada, and Professor Masami Shishibori, who had contributed so much time and effort in reviewing this thesis. Their careful analysis and recommendations helped to improve this thesis.

Finally, I deeply thank my parents and my beloved friends for their great support, love, and understanding. I hope that in the future, I can still have the courage to challenge, be brave, be grateful, be persevere, be diligent, and always believe in myself.

Abstract

Intention detection is an essential task for the spoken language understanding (SLU) module in the dialogue system which further illustrates vital information for managing and generating future action and response. The spoken language understanding module aims to transform the spoken language into a specific semantic template that human language can be effectively well-understood by the dialogue system. After that, the dialogue management module can facilitate future actions according to output from the SLU module. The SLU module of the pipeline dialogue system contains domain detection, intention detection, and slot filling. The intention detection task is crucial to improve the performance of the SLU module in the dialogue system.

With the development of research, intention detection task is also different according to the different dialogue datasets. In terms of dialogue data types, it can be divided into intention detection task (ID) for single-turn dialogue dataset and dialogue act classification task (DAC) for multi-turn dialogue dataset. The main difference between the two tasks is whether there is contextual relevance in utterances. Therefore, this thesis aims to investigate the problem of the intention detection for the single-turn task-oriented dialogue datasets and its application for the zero-shot intent detection. Besides, we also conduct the dialogue act classification for the multi-turn dialogue datasets. Based on the previous studies, the main difficulties on this subject are data acquisition, the particularity of natural spoken language, and the cold start problem. Under these circumstances, it is this thesis's goal to improve the ability to understand and express spoken language with single-turn and multi-turn dialogue datasets through deep learning modeling, and further to use transferring learning to explore the cold start problem in small sample scenarios. Therefore, this thesis conducts task-oriented intention detection by proposing an utterance feature embedding model, and conduct zero-shot intention detection by combining an attentive capsule neural network.

The traditional intention detection task is regarded as a classification problem where utterances are associated with predefined intents. For the single-turn task-oriented dialogue datasets, we propose a triplet training framework based on the multiclass

classification approach to conduct intention detection task. Precisely, we utilize a Siamese neural network architecture with metric learning to construct a robust and discriminative utterance feature embedding model. We modified the RMCNN model and fine-tuned BERT model as siamese encoders to train utterance triplets from different semantic aspects. The triplet loss can effectively distinguish the details of two input data by learning a mapping from sequence utterances to a compact Euclidean space. After generating the mapping, the intention detection task can be easily implemented using standard techniques with pre-trained embeddings as feature vectors. Besides, we use the fusion strategy to enhance utterance feature representation in the downstream of intention detection task.

In terms of the cold start intention detection problem, the various expressions of user's intents and constantly emerging novel intents make the annotating is time-consuming and laborious, which build huge obstacle for extending the model to new tasks. Therefore, we study the zero-shot intent detection problem, which aims to detect the unknown intents of utterances without the predefined label. In this experiment, we propose an attentive Bert capsule network with label embedding as a feature extractor. Specifically, we fine-tune the BERT model as a pre-trained embedding model and enhance the semantic utterance feature by jointly learn label embedding to measures the compatibility of embeddings between utterances and intents. Afterward, we leverage the process of attentive capsule network and routing-by-agreement mechanism to aggregate the utterance semantic feature into fixed-size encoding vector as abstract intent representation. The self-attention mechanism in the capsule improves the model to learn the different contributions of the capsules, which can be obtained by dynamic routing. Then, the large margin cosine loss function can identify sophisticated and interleaved utterance features by optimizing the network to minimize inter-class variance and to minimize intra-class variance. Finally, we inference the unknown intents by leveraging the transferring capability of the proposed model because it can bridge the knowledge gap between the source and target filed.

In terms of the multi-turn dialogue dataset, intent detection task of multi-turn dialogue is called dialogue act classification. The dialogue act as an intent label associate with utterance can be viewed as a sequence labeling problem. Considering this situation,

this thesis utilizes two approaches to conduct dialogue act classification, which are a traditional pipeline approach with a pre-designed feature templates with traditional machine learning algorithm and an end-to-end deep learning approach with the context-aware hierarchical neural network.

For the traditional pipeline approach with a pre-designed feature template, we provide a word-level sequence annotation method, which annotates dialogue structural information and semantic information to each word of utterance. Meanwhile, Linear-CRF is employed to natively capture constraints of hidden state sequence and obtain optimal probability, which is perfectly suited for sequence labeling task. Moreover, we propose a hierarchical learning structure to learn the conversation features from different levels than words, utterances and conversation levels. Specifically, we concatenate word2vec embedding model and fine-tuned BERT model to obtain rich semantic word information in word embedding layer. Then, we incorporate the multi-heads self-attention mechanism coupled with hierarchical RNN models in conversational feature learning level to learn conversational information that incorporates contextual history memory. Then, we set the linear-chain CRF at the final layer to consider the correlations between dialogue acts and contextual utterances, which can be treated as sequence labeling.

We conduct experiments on several benchmark datasets to verify the effectiveness of proposed models. For the intention detection task, the results illustrate that the triplet feature fusion model can effectively improve the recognition performance of these datasets and achieves new state-of-the-art results on single-turn task-oriented datasets (Snips dataset, Facebook dataset), and a multi-turn dataset (Daily Dialogue dataset). For zero-shot learning of intent detection task, extensive experiments demonstrate that the proposed model can obtain competitive performance that not only can better discriminate existing intents but is also able to detect unknown intents. The *IE-BertCapsNet* obtain the state-of-the-art results based on benchmark dataset (SNIPS) and several multi-lingual datasets (SMP-Chinese, Facebook Multilingual datasets). For the dialogue act classification, compared with the traditional feature template, the end-to-end attentive hierarchical neural network can achieve competitive results on the SWDA dataset.

Chapter 1

Introduction

1.1 Background and Significant

Recently, human-machine intelligence dialouge system have attracted much attention because of their huge potential and their attractive commercial value. With the advent of AI technology, the fantasy of an intelligent interaction system has become a reality. People can speak naturally to speak with the virtual personal assistants to finish some basic tasks efficiently. In the industrial field, lots of popular virtual personal assistants, like Siri of Apple, Google Home [1], Amazon Alexa [2], and Microsoft's Cortana [3] have been integrated into human life. These dialogue systems are carried on various devices to interact with a human. With the explosion of big data recently, we can easily obtain the dialouge data on the internet, which allows us to build a data-driven, open-domain human-machine dialogue system. Moreover, deep learning has been proven to be effective in recognizing the complex patterns in big data, and also achieved huge success in computer vision, natural language processing, and recommendation system [20]. The massive data with deep learning can promote the development of dialogue systems have emerged.

The dialogue system we are discussing now can be generally divided into two categories: task-oriented dialogue and open-domain dialogue (small chat dialogue system). The task-oriented dialogue systems are mostly used in the vertical field. This type of system has clear task objectives to complete, such as setting alarms and booking tickets. Traditional conversational systems have rather complex and modular pipelines. Specifically, the system first understands the information conveyed by human beings as an internal state, then takes a series of corresponding actions according to the strategy of

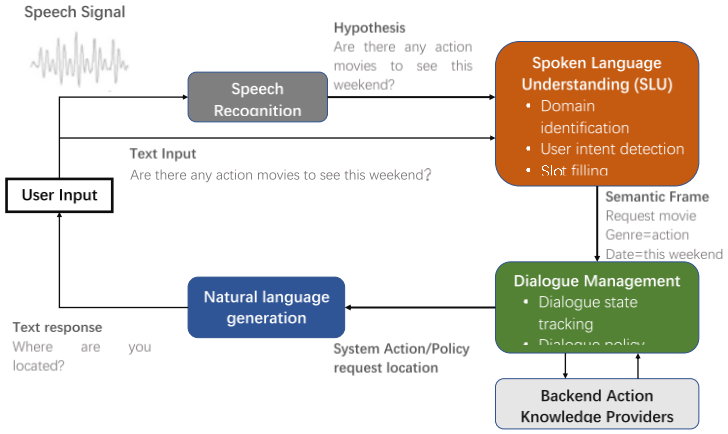


Figure 1.1 The Pipeline framework of task-oriented spoken dialogue system.

the dialogue state, and finally transforms the actions into natural language expressions. Although language understanding is handled through statistical models, most of the dialog systems that have been deployed still use manual features or manual rules for state and action space representation, intent detection, and slot filling. The advancement of deep learning technologies has recently risen the applications of neural models to dialogue modeling. The chat dialogue systems usually communicate with a human in an open-domain with entertainment functions. Although this dialogue system carries entertainment function, the small chat in the conversation occupies a large proportion and the way to deal with these problems is closely related to the user experience. The existing mainstream methods for non-task-oriented dialogue systems are dialogue generation or retrieval-based approach. The generation method like the deep neural network (Sequence-to-sequence model) generates appropriate responses during the dialogue. However, it also has shortcomings that generating some meaningless replies or sometimes generating utterances with grammatical errors. For the retrieval method, the system should learn from the pre-defined templates and select replies from the current conversation. This approach is widely applied in the industry because it can achieve high accuracy and it performs stable. The disadvantage of the retrieval method is that it relies too much on the quality of the data. This will undoubtedly lead to a lot of manual work in the early stage.

The research objects of this thesis are mainly for the task-oriented dialogue system. Thus, this section starts with an overview of the standard pipeline framework for the task-oriented dialogue systems. The key technologies in the task-oriented dialogue system include automatic speech recognition, spoken language understanding (domain

Table 1.1 The utterance samples of single-turn dialogue dataset (SNIPs).

Intents	Utterance
Agree	Oh, **yeah.**
Yes/No Question	You never think about that, do you?
Yes Answer	**Yeah**
Statement Opinion	I would think it would be harder to get up than it would be.
Backchannel	**Yeah**

Table 1.2 The utterance samples of multi-turn dialogue dataset (SWDA).

ID	DA	Caller	Utterance index	Sub Utterance	Text
1	b	B	1	utt1	Uh-huh.
2	sd	A	2	utt1	I work off and on just temporarily and usually find friends to babysit.
3	sd	A	2	utt2	I don't envy anybody who's in that situation to find day care.
4	b	B	3	utt1	Yeah.

identification, intention detection, slot filling), dialogue management (dialogue state tracking, dialogue policy optimization), natural language generation. When the system receives the text input and fills the semantic slot, it recognizes such a result. After the input language understanding state is processed by the middle dialog management, a dialog action is obtained. For example, we can conclude that an action is to ask the place of departure. The dialogue action of asking the place of departure will be input to the natural language generation module, and the natural language generation module will generate a natural language text reply based on this action “Where do you start from?”, which we illustrate in Fig 1.1. As we can see, the spoken language understanding (SLU) module is an indispensable component in the dialogue system.

1.2 Research Content and Motivation

As stated above, a typical spoken language understanding (SLU) module is designed to transform the spoken language into a specific semantic template that human language can be well-understood by the dialogue system. After that, the dialogue management module can facilitate future actions according to detection results in the SLU module. The role of the intention detection task in SLU is to discriminate the implicit intention by recognizing the intents of received utterances. Therefore, the intention detection task is crucial to enhance the spoken language understanding performance in the dialogue system.

First of all, we introduce the basic definition of intention detection task. From the perspective of data structure, the dialogue categories can be divided into single-turn dialogue and multi-turn dialogue. Table 1.1 and Table 1.2 show the utterance samples belong to the single-turn dialogue dataset and the multi-turn dialogue dataset, respectively. For the single-turn dialogue, each intent is a semantic label attached with each utterance in dialogue, which represents the user's intention and concise utterance interpretation [4]. The intention detection of the single-turn dialogue dataset can be regarded as a classification task that each sentence corresponding with each intent label, which can be seen in Table 1.1. For the multi-turn dialogue dataset, some previous studies flatten the conversation structure as single-turn dialogue, and some other researchers still treat the whole conversation as a training object and combine the conversation structure in the training process. From Table 1.2, we can not only know the utterances and dialogue acts but also understand some other knowledge (speakers preference, the relative position of utterance in the whole conversation, current emotions, etc.) In a multi-turn dialogue dataset, a conversation will correspond to a sequence of dialogue act tags, which can be treated as a sequence labeling task. In this thesis, we have conducted modeling studies for both dialogue data types. The specific mathematical concepts will be analyzed in detail in the following experiments.

The intention detection and dialogue act classification in spoken language understanding has been developed for a long time. Recently, the deep learning methods have been achieved state-of-the-arts results in intention detection and dialogue act classification tasks. some practical reasons bring challenges to spoken language understanding, which are illustrated as following:

- **Dialogue data acquisition**

The potential problem in the intention detection task is the dialogue data acquisition, the existing dataset mainly rely on the annotated dialogue dataset, which is very time-consuming. Moreover, the need for expert knowledge of the vertical domain will impede quick and wide development of intention recognizer. Thus, a model that has been trained in a fixed domain will be difficult to expand to new fields, and it is difficult to identify unknown intentions.

- **The irregularity of spoken languages**

Firstly, the sparsity of semantic information and obscure slang in spoken language makes the model difficult to interpret thoroughly [5]. For instance, the average length of some utterances is no more than 20 words. Secondly, the same underlying utterances have different tags or multiple tags, which gives rise to ambiguity in classifying intention labels. We use the utterance 'Yeah' as an example showed in Table 1.1 that the 'Yeah' has three tags, which are 'Backchannel,' 'Agree,' and 'Yes/No Answer,' respectively.

- **The limitation of traditional intention classifier**

The prior works of multi-class classification of intention detection exploit SoftMax to train an encoder on labeled training data. The learned features are optimized under the supervision of SoftMax, which cannot be sufficiently distinguished because it does not consider the intra-class compactness of features. The categories prediction was only focusing on finding a decision boundary, which results in poor generalization capabilities. Inspired by these observations, we assume that the intention recognition performance can benefit from constructing the robust and discriminative feature representations of the short-length utterances.

- **Code start problem**

With the explosive growth of interaction data, constantly emerging unacquainted intents and diverse intent expressions make annotating intents of dialogue more difficult and complicated. It is hard to obtain prior knowledge of unseen intents because of the lack of utterance samples in training. Based on this dilemma, the unknown intent detection task is indispensable for the developers to accelerate system development. The learning method specially designed for the unknown intent detection is still in infancy. Therefore, the zero-shot learning of spoken understanding also arise attention in industrial and academic fields.

Under this circumstance, we delicate to improve the performance of intent detection results based on the spoken language feature embedding learning in task-oriented dialogue and the knowledge transfer in zero-shot intention detection task.

1.3 Review of the Research Dataset

The research object of this thesis is mainly for these two types of dialogue data. A single-turn dialogue is that one question and one answer, regardless of context. The multi-turn dialogue, that is, multiple dialogues, revolves around intent, and contact context until the task is completed. Previous studies have defined different task names for the intent understanding of these two different dialogue data. In particular, the intention of understanding in single-turn dialogue is named by intent detection, which can be regarded as a multiclass classification task. For the multi-turn dialogue dataset, the intention understanding is named by dialogue act classification, which can be treated as a sequence labeling task. Therefore, this thesis has conducted research on the intent detection of these two kinds of datasets. Specifically, we introduce three single-turn task-oriented dialogue datasets and three multi-turn dialogue datasets, which are listed below:

- **Single-turn dialogue dataset**

1) The SNIPS dataset [100] is collected from the Snips personal voice assistant and contains 7 intent types. The number of samples for each intention label is approximately the same.
2) The ATIS [46] dataset is the audio recording of making the flight reservation. The training set includes utterances, and the test set contains 893 utterances. We follow the previous experiment and set the validation set with 500 utterances from the training set. There are 21 intention labels in the dataset.
3) The SMP-2018 dataset [116] is a Chinese version dialogue corpus for user intention detection. It contains 30 intent types related to a series of instructions commonly used in daily life.
4) Facebook's multilingual dataset [47] contains annotated utterances with the English version, Spanish version, and the Thai version. It covers the weather, alarm, and

reminder domains in English, Spanish, and Thai language. There are 12 intention labels in the training set.

- **Multi-turn dialogue dataset**

1) The SWDA corpus [20] contains audio recordings and transcripts of the telephone conversations between speakers. For each conversation, a total of 66 topics are provided to speakers for communication. Overall, there are 42 dialogue act labels in the corpus, and all are annotated by DAMSL taxonomy [21]. This paper adopts the data split of 1115 training dialogue.
2) The Daily Dialogue dataset [48] is a high-quality multi-turn dialogue dataset, which mainly records dialogue in terms of people's everyday life. Each utterance of the Daily Dialogue dataset is manually labeled with the topic tag, intention tag, and emotion tag.
3) The ICSI Meeting Recording Dialogue Act (MRDA) [49] dataset contains 72 hours of multi-party meeting speech dialogue from 75 naturally happened meetings. The original tag sets of MRDA included 11 general tags and 39 specific tags. Based on the previous experiments, we utilize the most widely used class-map to cluster all tags into 5 groups of intention categories.

1.4 Organization Structure

This thesis mainly studies and analysis the research with intention detection task. The whole paper covers the basic theories, methodology, experiment, and discussions about intent detection and zero-shot intention detection, which is organized in the rest chapters as follows:

Chapter 1 talks about the background and significance of the intention detection task. Then, we introduce the main research contents and potential problems in this task. In the end, we represent the organizational structure of this thesis.

Chapter 2 proposes a novel triplet utterance fusion feature learning embedding model to learn the utterances in dialogue. We utilize a Siamese neural network architecture with metric learning to construct a robust and discriminative utterance feature embedding

model.

Chapter 3 introduces a semantic-enhanced attentive Bert capsule network to extract and aggregate high-level utterance features, and we inference zero-shot unknown intents based on the proposed method.

Chapter 4 expands the scope of intention detection to a multi-turn dialogue dataset. In this section, we combine the hierarchical structure and utilize a multi-head self-attention mechanism to learn the importance of dialogue context.

Chapter 5 concludes the whole thesis and discuss future works.

Chapter 2

Related Works

As we mentioned above, the intention detection task is a core component of the dialogue system whether it is a single-dialogue dataset or multi-turn dataset. Research on the intention detection (ID) and dialogue act classification (DAC) has been continuing for many years, focusing on searching for effective features and appropriate machine learning methods. In the following chapters, we illustrate the related research in detail we mentioned and the strategies we employed in this thesis.

The current task-oriented multi-turn dialogue implementation is mainly based on a finite state architecture, framework-based architecture, information state architecture based on Markov decision process, and end-to-end (deep neural network) architecture. The framework-based architecture and finite-state architecture are also the current commercial mainstream. The finite state-based architecture and the framework-based architecture are mainly based on the scripting method. This approach frames people's life scenes with a dynamic memory approach. For example, when we go to a restaurant to eat, the framework of the general order activity (script): enter restaurants, seating, order, dining, paying bills, leaving. Therefore, the dialogue system can extract information according to the intention understanding, and filling the corresponding slots, and then give corresponding feedback.

However, the restriction of dialogue management architecture based on finite state structure is quite high that requiring users to accurately answer the question, which makes the conversation clumsy. The initiative is a hybrid initiative that switches between the

system and the user. Therefore, a currently commonly used hybrid active dialogue architecture relies on the structure of the framework itself to guide the dialogue, that is, the framework-based architecture.

2.1 Intention Detection

In the dialogue system, the previous studies mainly relied on non-lexical features like rhythm and acoustic to study spoken language understanding tasks [12]. In this thesis, the research object is written language. However, it seems that the phonological features are not effective enough for written language. In terms of feature learning methods, some earlier research applied keywords [13], vocabulary pairs [13], and designed templates as feature representation in the DAC task. These features have been achieved some improvement in ID and DAC tasks. Besides, some non-language feature set also very promising in ID and DAC tasks. For example, the utterance length and the word order are also can be another valid feature for ID and DAC tasks [12][19]. For example, the intention label of the utterance ‘This is wrong’ and ‘Is this wrong’ are different, which are ‘Statement’ and ‘Question’ respectively. Although the word is the same, the purpose of this utterance is different. Furthermore, in a multi-turn dialogue dataset, the information related to speakers also plays a critical role in the detection process. For instance, the speaker’s identification, personality, speaking preference, and emotional state at that time also had shown its utility to detect user’s intents [14]. Moreover, the structure information also had an impact on growth in accuracy for the DAC task. For example, the utterance position and the similarity between the utterance also can be treated as valid features used in the ID and DAC task [8]. After that, several basic language models like the N-gram model have been verified the validation on spoken and textual conversations on Map Task Corpus [15]. The Bag-of-word combined with the N-gram model showed improvement in the one-to-one MSN online shopping assistant conversation [16]. The Gaussian distribution of short text could capture the latent state by using a weighted sum of word vectors based on GloVe [17]. In traditional research, the experiments are conducted in the pipeline manner, so that the feature learning and

classification operation are divided into two parts. The learning methods for the intention detection task are divided into two categories: multi-class classification and sequence labeling. The multi-class classification models are SVM [8], Naive Bayes [10], and Maximum entropy [11] in experiments. The sequence labeling methods are HMM [8] and SVM-HMM [12]. Previous studies used handcrafted feature sets along with contextual and lexical information, which reported 71% accuracy by HMM in SWDA corpus [8] and 82% by Naive Bayes in MRDA corpus [9]. However, the traditional approaches for intention detection relied on hand-crafted features that were time-consuming and labor-intensive.

Nowadays, the emergence of deep learning methods effectively alleviated the constraints of researchers employ deep learning method in DAC task and then obtained significant improvement. Blunsom and Kalchbrenner [20] proposed a sentence feature representation and followed with hierarchical CNNs to classify these sentences into DA tags. Lee and Dernoncourt [22] proposed a model based on CNNs and RNNs to incorporate preceding contextual text to classify the current DA tags. More specifically, the feature produced by CNNs is better than the feature based on RNNs in both SWDA and MRDA corpus. In another work, Shen *et al.* [23] used RNNs in combination with the attention mechanism to emphasize the weight of useful information in the entire sequence, and the result showed some improvement in intent detection. Besides, some research utilized the joint learning approach to conducting the intention detection and slot filling[54][55]. Kumar *et al.* [24] utilized hierarchical Bi-LSTM to capture utterance granularity and inherent properties from multi-levels of conversation and predicted sequential dialogue act with the CRF model. Tu *et al.* [25] build a hybrid neural network-based ensemble model for Chinese multi-turn dialogue. Notably, this paper incorporated the speaker changing as a feature to illustrate utterance peculiarity. However, the representation of features and their operations in neural network experiment are uninterpretable and vagueness. Furthermore, some other features were useful to generate more discriminative predictions in detecting the user's intention. For examples, the location of the comment in web forum [26], speaking preference of users [27], dialogue topic context of same user [28], emotion transition trait of user's blog [29], the rating and

comments of products in shopping website were treated as the weak label to learn the sentence representation [40]. Therefore, we still use traditional learning methods along with interpretable features as the basis for the DAC task.

2.2 Zero-shot Intent Detection

The intention detection task is a core component of the dialogue system. Deep learning methods have shown promising results in previous studies. The combination of deep learning and distance has also been integrated into text modeling and achieved good results. The metric learning has been successfully applied to various tasks like face recognition [32], speech recognition [33] [34], and unknown intent detection [59]. Metric learning can address some shortcomings of conventional classification. The reason is that the distance metric learning can further force the model to maximize inter-class variance and minimize intra-class variance.

The conventional intent detection task trains a discriminative classifier in a supervised manner, which requires a considerable amount of labeled data. However, the numerous intent expression approaches and the continuous emergence of new intents mean that the cost and difficulty of labeling are pretty high. The appearance of zero-shot learning is very helpful for dealing with these problems because it can generalize the knowledge and concept learned from known filed to unknown filed. Therefore, the application of zero-shot learning has aroused a strong interest in the academic and industrial fields.

The zero-shot intent detection addresses the problem that not all intent categories are seen during the training phase, which is an important task in natural language understanding as novel intents may continuously emerge in dialogue systems. The research on zero-shot intent detection is still in its infancy. Previous zero-shot learning methods for intent detection utilize external resources such as label ontologies [60] [61] or manually defined attributes that describe intents to associate existing and emerging intents, which require extra annotation. Recently, IntentCapsNet-ZS extends capsule networks [87] for zero-shot intent classification by transferring the prediction vectors from seen classes to unseen classes. The ReCapsNet [67] shows that IntentCapsNet-ZS

[87] hardly recognizes utterances from unseen intents in the generalized zero-shot classification scenario and proposes to solve this issue by transferring the transformation matrices from seen intents to unseen intents.

2.3 Language Representation Model

Recently, the language representation model improved significantly in many NLP tasks, such as textual entailment, semantic similarity, reading comprehension, and question answering [29]. The language representation models can provide powerful context-dependent representations by pre-training on a large scale unlabeled data, such as Contextualized Word Representations (ELMo) [30], Generative Pre-trained Transformer (GPT) [31] and Bidirectional Encoder Representations from Transformers (BERT) [6]. Besides, these models can be easily applied to different downstream tasks with minimum parameters. Therefore, we exploited the concept of language model representation to construct a novel utterance feature embedding model.

2.4 Metric Learning

Utilizing the deep neural network with a distance metric to learn the feature embedding had been successfully applied to many tasks, such as face recognition [32], speech recognition [33][34], and speaker identification. For example, FaceNet [32] of Google utilized a random semi-head triplet mining approach to make up facial picture triplets, which obtained excellent performance. He *et al.* [35] achieved outstanding performance on 3D object retrieval by proposing triplet loss and center loss. Huang *et al.* [36] applied triplet loss in training to automatically recognize the emotional state in spoken language. To deal with the spoken language, Zhang *et al.* [37] presented a system that directly learned mapping from speech features to a compact fixed-length speaker discriminative embedding. The triplet loss function focuses on fine-grained identification and adds the measurement of the latent state, which can help the model distinguish the details. Thus, we first tried to use the matric learning approach in the intention detection task of the

natural language processing field.

2.5 Downstream Fusion Strategy

Generally, the exceptional performance of the classification model depended on sufficiently large training corpora to a great extent. To comprehensively understand sentences, the fusion strategy can aggregate multiple sources to enriching the features and boost learning performance [37]. Majumder *et al*. [38] fused multimodal resources like audio, video, and text for sentiment analysis. Tay *et al.* [39] generated sentence representations by using a gating mechanism to combine the sentence token features and sentiment lexicon features. Sun *et al.* [41] detected emotional elements by using a mixed model to extract sentimental objects and their tendencies from product reviews. Specifically, the multi-stream architecture is prevalent in data fusion. For example, Simonyan *et al.* [42] designed a model with two-stream ConvNet architecture to illustrate spatial feature and temporal features, which can achieve significant performance under the condition of limited training data by the two-stream model. Inspired by these experiments, we use the fusion strategy in the downstream task to enhance the utterance feature representation.

2.6 Capsule Network with Dynamic Routing

Capsule Network has been proposed to improve the limitation of pooling strategy in the CNN model in the computer vision field since the max-pooling strategy might discard some valuable information. To improve this limitation, the capsule network and routing agreement method had been proposed to address the shortcoming of the polling strategy in CNN. The capsule network with routing-by-agreement process enables to learn the part-whole invariant relationship consecutively of the research object. The capsule holds an activation vector of a group of neurons, which represent a specific type of entity's instantiation parameters. Specifically, the orientation of the activity vector indicates the attributes of objects and the length of the activity vector reflects the probability of existence. The capsule network is a feature extractor that detecting low-level features and

then aggregating the information into the high-level feature by utilizing a dynamic agreement mechanism.

Recently, some studies utilized the capsule network method in the natural language processing field and achieved impressive performance. However, the research of capsule networks in the NLP field still in infancy. For example, Yang *et al.* [68] first attempted to use a capsule network for text-classification. The author experimented with two capsule networks, named Capsule A and Capsule B. The difference between them is that Capsule A utilizes the single filter size and Capsule B used multiple filter sizes at the CNN layer. Gong *et al.* proposed an aggregation mechanism to obtain a fixed-size encoding with a dynamic routing mechanism. Zheng *et al.* [70] proposed a novel attentive capsule network with a dynamic routing process to process hierarchical structure text data. Wang *et al.* [90] proposed a sentiment aspect-based capsule network to detect emotion. Chen *et al.* [77] leverage the transferring ability of capsule networks to transfer the knowledge of document-level to aspect-level sentiment detection. Geng *et al.* [92] adopted a dynamic routing mechanism with a relation network for few-shot text classification. Zhang *et al.* [91] proposed an attention-based capsule network for multi-label relation extraction.

2.7 Label Embedding Learning

Recently, there has been plenty of works utilizing label embedding achieved promising results in image classification [63], multimodal learning [64], text detection in images [65], text classification [65], and zero-shot learning fields [62]. The researchers treat the text classification as a label-word joint embedding problem in that each label can be embedded in the same space of the word vector. The author proposed a model to learn the representation of words and label in the same space which can be used to measure the compatibility of embedding between text sequence and label. The attention is learned on the training set of labeled samples to ensure that in a given text sequence, the weight of related words is higher than that of unrelated words. Du *et al.* [69] used an interactive mechanism to explicitly calculate the word-level interactive signal for text classification. The key idea behind the interaction mechanism is to explicitly calculate the matching

score between words and classes. From the word-level representation, it calculates an interaction matrix, where each entry is a match score between a word and a category. Moreover, the label embedding model also shows its effectiveness for zero-shot learning. Previous studies illustrated that the label correlation in the embedding space can explicitly facilitate information transform knowledge from seen labels to unseen labels. For instance, Ma *et al.* [78] presented a label embedding method that incorporates prototypical and hierarchical information to learn pre-trained label embeddings for fine-grained named entity typing. The proposed method can easily adapt a zero-shot framework to predict both seen and previously unseen entity types. In brief, the research of zero-shot learning on intention detection with capsule network is at an early stage. To our best knowledge, this paper first investigates the label embedding in capsule network.

Chapter 3

Intention Detection based on Fusion Triplet Feature Embedding Model

3.1 Introduction

The prior works of multi-class classification of intention detection exploit SoftMax to train an encoder on labeled training data. The learned features are optimized under the supervision of SoftMax, which cannot be sufficiently distinguished because it does not consider the intra-class compactness of features. The categories prediction was only focusing on finding a decision boundary, which results in poor generalization capabilities. Inspired by these observations, we assume that the intention recognition performance can benefit from constructing the robust and discriminative feature representations of the short-length utterances. To this end, we improve the conventional method by proposing a novel triplet training framework based on multi-class classification learning.

Pre-trained language models are proved to be very useful and efficient in learning universal language representations recently. For instance, the BERT model is conceptually simple and empirically powerful in enormous natural language processing tasks [6]. Inspired by the pre-trained language model learning approach and transfer learning techniques, we reference the conception of the unsupervised pre-training method with triplet loss to learn a structured space of interpretable utterance representations.

Specifically, we design a two-stage process for intent classification, which includes feature embedding learning and intention prediction. In the first stage, we develop the R-MCNN model and BERT model as Siamese encoder with metric learning to obtain robust and discriminative feature embeddings by minimizing the intra-class variations. In the second stage, we fuse the features from pre-trained feature embedding models and add additional relevant information as completed feature sets to predict intention labels in the downstream task.

We summarize the contributions of this experiment as follows:

- The proposed triplet training framework learns discriminative utterance feature by using the same weights on different inputs. The triplet loss function infers a non-linear mapping in the resulting latent space, and the inter-class sample distances are maximized based on a certain margin [7] .
- We utilize CNN, RMCNN (Bi-GRU-MCNN), and BERT as Siamese encoders to train the utterance triplets. Precisely, the RMCNN model can generate structural information, in which the RNN model can extract the global context, and a wide range of kernels of CNN can capture the fine-grained local components of utterance. Besides, we leverage the power of the BERT model by facilitating deep bidirectional representations on enormous unlabeled data to obtain sentence-level context-dependent features.
- The triplet selection turns out to be crucial for model convergency. By considering the strong correlations between dialogue context, we propose a sequential sampling strategy to keep the intention transition traits into the triplet sampling process.
- In the downstream task, we predict the probability distribution of each intent based on multi-class classification learning. We use the fusion strategy to fuse the features from different pre-trained feature embedding models as utterance features. Besides, we extent features with relevant information as external knowledge.

3.2 The Triplet Siamese Neural Network

3.2.1 Whole Framework

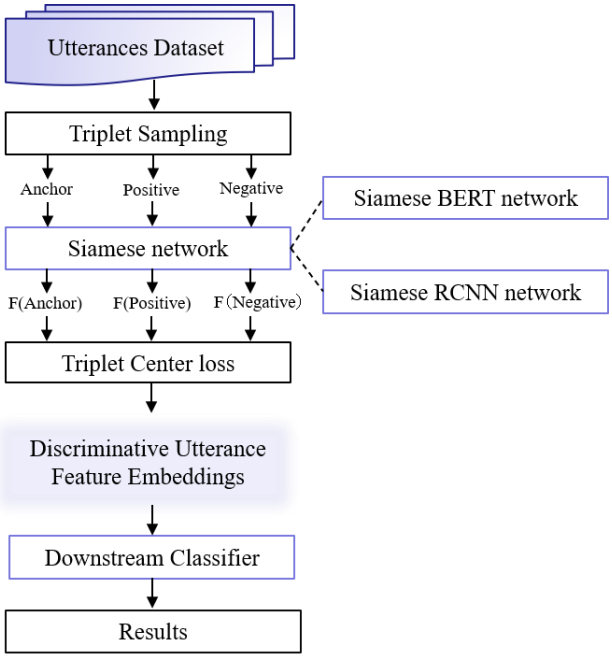


Figure 3.1 The framework of feature embedding learning model.

This section mainly introduces the whole framework of the proposed model. The entire structure consists of three parts, which are triplet sample selection, triplet training section, and the downstream task of intention classification. Firstly, the system needs a sampling strategy to generate valid triplet data (x_i^a, x_i^p, x_i^n) as training objects. One triplet sample consists of an *anchor* sample x_i^a, a *positive* sample x_i^p, and a *negative* sample x_i^n. Then, we input all the triplet samples into the Siamese encoder and train the model with a triplet loss function. The triplet training model uses the same weights on different inputs to compute variables and accomplish a better separation between two positive related samples of the same class (x_i^a, x_i^p) and one *negative* sample (x_i^n). To avoid meaningless calculation in the training process, we need to verify whether triplet samples are valid by setting up a particular margin parameter to observe Euclidean distance between embedding triplets in the test section. After the training, we can obtain a robust pre-trained

feature embedding features, which can better reflect the specific characteristics of utterance. Secondly, given the well-defined feature embedding model with parameters, we exploit it mapping utterances in the downstream task. The critical components for triplet training are the Siamese model selection and triplet data composition. Therefore, the related information of essential components and modifications are illustrated in the following subsections.

3.2.2 Siamese RMCNN Neural Network

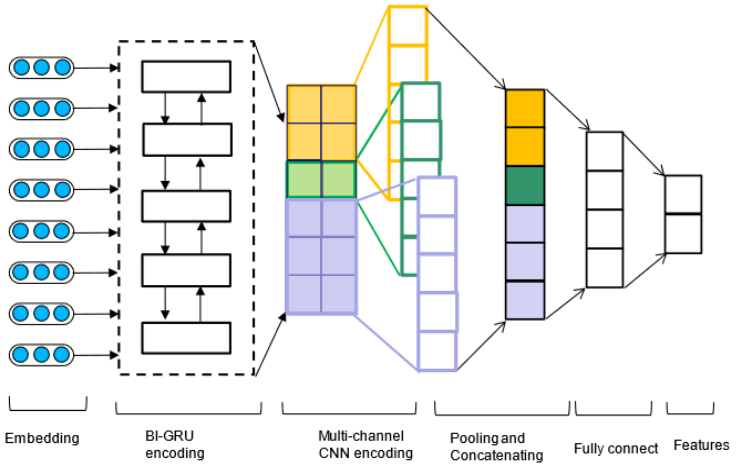


Figure 3.2 The framework Siamese RMCNN model.

We modify the RMCNN model as a Siamese encoder to train the utterance triplets and generate a fixed-dimension representation. Firstly, we have the number of n utterances $X = \{x_1, x_2, \dots, x_n\}$ in the dialogue. Each utterance contains variable-length word tokens $x_i = \{w_1, w_2, \dots, w_j\}$. After triplet sampling, we obtain utterance triplet samples. For each utterance sample in triplet, we embed word tokens into vector $E = \{e_1, e_2, \dots, e_n\}$ through a trainable embedding matrix pre-trained on enormous unlabeled data. The bidirectional GRU model encodes sequence token embedding to produce sequences of corresponding hidden vectors $H = \{h_1, h_2, \dots, h_i\}$, which extracts the context information by concatenating the hidden states from forward and backward directions. The operation of bidirectional GRU is formulated as follows:

$$h_t^{\rightarrow} = f_{GRU}(h_{t+1}, e_t) \tag{3.1}$$

$$h_t^{\leftarrow} = f_{GRU}(h_{t-1}, e_t) \tag{3.2}$$

$$h_t = [h_t^{\rightarrow}, h_t^{\leftarrow}] \tag{3.3}$$

in which h_t maintains the sequence information of the utterance. Then, we feed the output from Bi-GRU layer into the CNN layer. The CNN model can capture fine-grained local features inside a multi-dimensional filed. The convolutional operation includes a filter $W_c \in \mathbb{R}$, which is utilized to a window of l continuous word vectors to produce a new feature map. A scalar feature c_i is generated from a window of words $h_{i:i+l}$ by:

$$c_i = f(W_c \circ h_{i:i+l} + b_c) \tag{3.4}$$

where the symbol $\circ$ indicates the dot product operation, l refers to the width of the convolutional kernel, f is a non-linear function (ReLU), W_c is the convolutional matrix, and b_c is a bias term. Each kernel corresponds to an utterance detector to extract specific n-gram patterns at various granularities. The kernel applied to each possible region matrix to produce a valuable feature map:

$$C = [c_1, c_2, \dots, c_m] \tag{3.5}$$

in which m is the number of the channels. The pooling layer can extract local dependencies in different regions to preserve the most useful information. Then, we apply the pooling layers to capture the most valuable feature from each feature map, which includes the global maximum pooling layer and global average pooling layer. The outputs from two pooling layers are concatenated together as the local phrase feature of dialogue:

$$\hat{c} = [gmp\ (c_i) \oplus\ gap(c_i)] \tag{3.6}$$

where the '*gmp*' indicates the global maximum pooling layer and the 'gap' indicates the global average pooling layer. Then, the outputs of the pooling layers with different widths are concatenated. Finally, three fully connected layers with 'tanh' activation are stacked together, and an L2-normalization layer is followed behind to form final utterance embedding. The Siamese RMCNN neural network optimized by minimizing the triplet loss and Adam optimizer is used during training.

3.2.3 Siamese BERT Neural Network

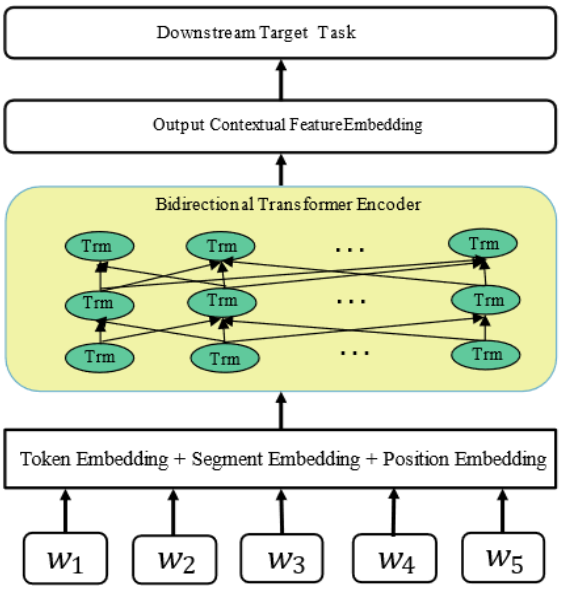


Figure 3.3 The framework of Siamese BERT model.

Here is the process that we train utterance triplet samples with the Siamese BERT model. In this section, we fine-tune the pre-trained BERT model as Siamese encoder to train utterance triplet samples. Given sequence utterances $X = \{x_1, x_2, \dots, x_n\}$, and we sample valid triplets for training. For each utterance sample in a triplet, BERT model construct token embeddings of this utterance $E = \{e_1, e_2, \dots, e_n\}$ by concatenating the word piece embeddings, the positional embeddings, and the segment embeddings. Then, the token vectors are feed into encoder block and are encoded by stack layers. The encoder block includes multi-attention sublayers and the position-wise fully connected sublayers. The input data of the encoder block is a sequence hidden states $H = \{h_1, h_2, \dots, h_i\}$, so the output of encoder $S = \{s_1, s_2, \dots, s_i\}$ is illustrated as follows:

$$a_{ij}^{(k)} = Softmax\left(\left(\frac{1}{\sqrt{d_s}}\left(W_Q^{(k)}h_i\right)^T\left(W_K^{(k)}h_j\right)\right)\right) \tag{3.7}$$

$$s_i^{(k)} = \sum_{v=1}^{N} a_i^{(k)}\left(w_v^{(k)}h_j\right) \tag{3.8}$$

$$s_i = W_O\left[s_i^{(1)}, s_i^{(2)}, \ldots, s_i^{(k)}\right] \tag{3.9}$$

in which *k* is the number of attention heads, *h* is the dimension of hidden states, and d_s is the parameter of scale dot-production. The W_Q, W_K, W_v and W_O indicate the model parameters. The output of the residual connection and the normalization module $\tilde{S} = \{\tilde{s}_1, \tilde{s}_2, \ldots, \tilde{s}_N\}$ are denoted below:

$$\tilde{S} = LayerNorm\ (H+S) \tag{3.10}$$

The output of the position-wise fully connected sublayer $O = \{o_1, o_2, \ldots, o_N\}$ is calculated as follows:

$$o_i = W_2 ReLU(W_1 \tilde{s}_i + b_1) + b_2 \tag{3.11}$$

in which W_1, W_2, b_1 and b_2 are the model parameters. The residual connection layer and the normalization layer are followed the encoder block. The final contextual representation $\tilde{O} = \{\tilde{o}_1, \tilde{o}_2, \ldots, \tilde{o}_N\}$ is illustrated below.

$$\tilde{O} = LayerNorm\ (O+\tilde{S}) \tag{3.12}$$

We feed the final contextual representation into three fully connected layers with '*tanh*' activation and an L2-normalization layer to get final utterance token embedding. The Siamese BERT encoder is optimized by triplet loss function by end-to-end propagation, and Adam optimizer is utilized during training.

3.2.4 Triplet Loss and Triplet Sampling Strategy

Triplet loss function is calculated on the triplet data $\left(x_i^a, x_i^p, x_i^n\right)$, where the $\left(x_i^a, x_i^p\right)$ are extracted from the same intention category. We obtain the negative sample (x_i^n) in different intention category from the $\left(x_i^a, x_i^p\right)$. We exploit the feature embedding model $f_\theta(x) \in \mathbb{R}^d$ to map utterance triplets to *d*-dimension Euclidean space, and the distances are measured in resulting latent space.

$$D_{ap} = \| f_\theta(x_i^a) - f_\theta\left(x_i^p\right) \|_2^2 \tag{3.13}$$

$$D_{an} = \| f_\theta(x_i^a) - f_\theta(x_i^n) \|_2^2 \tag{3.14}$$

$$\forall \left(f_\theta(x_i^a), f_\theta\left(x_i^p\right), f_\theta(x_i^n)\right) \in T \tag{3.15}$$

The $f_\theta(\cdot)$ refers to the Siamese encoder. The $f_\theta(x_i^a), f_\theta\left(x_i^p\right), f_\theta(x_i^n)$ are outputs from the Siamese encoder. T is the set of all possible triplets in the training set. The triplet loss optimizes model by minimizing the distance between $f_\theta(x_i^a)$ and $f_\theta\left(x_i^p\right)$and maximizing distance between $f_\theta(x_i^a)$ and $f_\theta(x_i^n)$ by at least a margin parameter $\alpha \in \mathbb{R}^+$. The triplet loss $L_{triplet}$ is illustrated as follow:

$$\sum_i^N \left[\| f_\theta(x_i^a) - f_\theta\left(x_i^p\right) \|_2^2 - \| f_\theta(x_i^a) - f_\theta(x_i^n) \|_2^2 + \alpha \right]_+ \tag{3.16}$$

where *N* stands for the number of triplets in the training set, and *i* denotes the *i*-th triplet sample. During the triplet training, generating all possible triplets can easily be satisfied but results in slower convergence. Therefore, it is vital to select valid triplet samples to improve training efficiency. The following section is about triplet sampling strategies.

3.2.5 Downstream Fusion Strategy

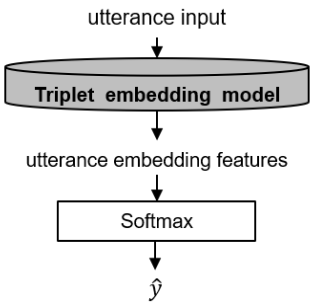


Figure 3.4 The feature-based strategy of downstream task.

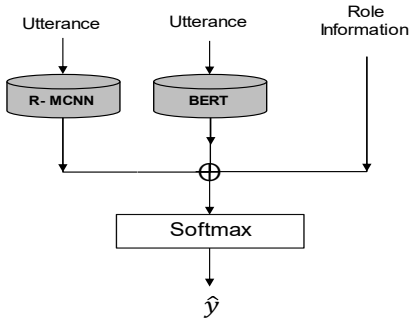


Figure 3.5 The model of fusion strategy of downstream task.

Fine-tuning the pre-trained language model can save expensive pre-computing. The pre-trained feature representation can be easily testified on many experiments with cheaper models on top of this representation [43]. Therefore, there is no need to train complex afterward. In this paper, we verify our pre-trained feature embedding model by utilizing the feature-based strategy for the downstream task. Feature-based strategy collects

utterance features from the well-defined pre-trained language model to different downstream tasks.

The intention detection task in our experiment is based on the multi-class classification learning method, which can be seen in Fig. 3.4 The pre-trained feature embedding models (f_{RMCNN}, f_{BERT}) can form two robust utterance representations from different semantic aspects, which are denoted below.

$$U_{RMCNN} = f_{RMCNN}(x_i) \tag{3.17}$$

$$U_{BERT} = f_{BERT}(x_i) \tag{3.18}$$

Then, we feed the utterance feature U_{BERT} and U_{RMCNN} into the fully-connect layers, respectively. We use the Softmax classifier to predict the probability distribution of intention labels, which is defined as follows:

$$Q = tanh(W_U U + b_U) \tag{3.19}$$

$$\hat{y} = Softmax(W_Q \mathrm{Q} + b_Q) \tag{3.20}$$

where W_U, b_U, W_Q, and b_Q are model parameters. We take cross-entropy as the loss function and Adam as an optimizer during training. The end-to-end backpropagation is employed in the training process.

The multi-source fusion strategy can effectively improve the performance of natural language learning by various relevant resources [44]. Inspired by this conception, we employ a fusion strategy to accumulate semantic information of utterance from several aspects, such as utterance granularity, dialogue structure, and speaker information, which can be seen in Fig. 3.5 The same sentence may express different aspects concerning different aspects. To be specific, the RMCNN model can capture the global structural features of the input sentence. The BERT model remedies the limitation of the insufficient training corpora and provides more external knowledge about common utterance words. Otherwise, the participants have different roles and speaking preferences in various domains in multi-turn conversation, which also can be regarded as a distinctive feature to enhance utterance differences. We indicate speaker information in the model as *'C'*. Specifically, we use numerical values to represent different speakers.

We unified a two-stream fusion model to integrate the utterance features from different models to show its different aspects. Firstly, we set two pre-trained feature embedding models as two streams to encode utterances from different aspects. We feed the sequence word tokens into the models independently and obtain the optimal parameters of each model. In this section, we compose the utterance encoder using two models with optimal settings. After the optimal parameters are trained in each stream, the outputs from each stream are concatenated together and then input to the classifier. Then, we extend the utterance representation to $U_{all} = [U_{RMCNN}, U_{BERT}, U_{Speaker}]$. Precisely, U_{RMCNN} refers to the structural feature learned from the Siamese RMCNN model, U_{BERT} refers to the fine-grained contextual feature learned from the BERT triplet model and the $U_{Speaker}$ as an additional feature refers to the speaker's role aligned with each utterance. Then, all the features are concatenated together to be a comprehensive utterance representation. The SoftMax function is connected to the encoders to calculate the probability distribution, and the output is $P = \{p_1, p_2, \ldots, p_n\}$, in which n is the number of the intention labels, and p_i is the predicted probability that utterance belongs to the corresponding intent tag i, and the final predicted tag: $\hat{y} = argmax(P)$. The model optimization is to minimize the cross-entropy loss, and Adam optimizer is used during training.

3.3 Experiment

3.3.1 Dataset

We evaluate the proposed model on several benchmark datasets. We find that the evaluation object of the intention detection task includes not only task-oriented dialogues but also multi-turn dialogues. In the previous studies, the intention detection task of multi-turn conversation is regarded as a multi-task classification. Therefore, we transfer the multi-turn conversation from the nested dialogue structure into a flat structure, so that the utterance triplets can be properly sampled. Besides, we also performed a series of pre-

Table 3.1 The Dataset overviews.

Dataset	# Intention	# Vocabulary	#Train	# Validation	# Test
ATIS	21	722	4778	500	893
Snips	7	11241	13084	700	700
FB (EN)	12	3983	30521	4181	8621
FB (SP)	12	1849	3617	1983	3043
FB (TH)	12	1894	2156	1235	1962
DYDA	4	25000	87170	8069	7740
MRDA	5	10000	77900	15800	15500

processing steps by utilizing Stanford's CoreNLP tool [45] to avoid text noise, such as utterance tokenization and word lemmatization.

In this experiment, we utilize three single-turn task-oriented dialogue datasets and two multi-turn dialogue datasets. Three single-turn task-oriented dialogue datasets are the Snips dataset, ATIS dataset, and Facebook's multilingual dataset. The Snips dataset is collected from personal voice assistant with English version contains 7 intents. The ATIS dataset is collected from the audio recording related to flight reservations. The Facebook multilingual dataset covers three different domains with English, Spanish, and Thai language version. Besides, we also introduce two multi-turn dialogue datasets which are the Daily Dialogue dataset [43] and the ICSI Meeting Recording Dialogue Act (MRDA) dataset contains 72 hours of multi-party meeting speech dialogue from 75 naturally happened meetings. The dataset overview is illustrated in Table 3.1. The number of the classes of each corpus is tag #Intention, the vocabulary size of each corpus is tag #Vocabulary.

3.3.2 Hyper-parameter Selection

In this section, we illustrate the related parameters in model training, which is associated with the triplet training process and downstream task. All the work is implemented under the TensorFlow framework.

In terms of the triplet training with the Siamese RMCNN model, we pad each utterance to the maximum length for training. We initialized word vectors with the 300-dimensional word2vec word vectors. We set the dropout as 0.3 after the embedding layer to avoid over-fitting. The hidden size of Bi-GRU is 512 in one direction. We apply multiple kernel size [1, 2, 3] in the CNN layer to encode different utterance granularity, and the filter size is 256. The three fully-connect layers and an L2-normalization layer are followed behind. We set the Adam optimizer with a learning rate of *2e-4* and a weight decay of *1e-6.*

In terms of the Siamese BERT model, we fine-tuned the BERT model with metric learning to obtain utterance features. The pre-trained BERT encoder is trained on the unlabeled data, which are Books corpus (800M words) and English Wikipedia (2500M words). The maximum length of an utterance is 50. The BERT-base model has 12-layers, 768- hidden states, and 12-heads. The hidden dim of the token embedding is 50. We set the Adam optimizer with a learning rate of *3e-5* and a weight decay of *1e-6.* For the other parameters, we follow the original BERT paper [6].

Furthermore, we utilize the feature-based strategy in downstream intention detection tasks. The pre-trained RMCNN and BERT feature embedding model is employed as different encoders in single stream, respectively. In this section, we set the hidden size as 64, Adam optimizer is used with a learning rate is 2e-4, and the batch size is 256.

3.4 Results Comparison and Discussion

3.4.1 Baseline Comparison

We compare the proposed model with several state-of-the-art baseline models. For the single-turn task-oriented dataset, it includes the following:

- Attention-BiRNN [50] utilizes the encoder and decoder model for joint learning the intention detection task and slot-filling task. An attention weighted sum of all encoded hidden states is used to recognize intention.
- Slot-Gated Attention [51] uses slot-gated LSTM to learn context vector, which improves the performance of intention classification.

- Capsule-NLU [53] accomplishes the intention detection by exploiting the hierarchical semantic information. They propose a re-routing schema to synergize further the slot filling performance using the inferred intention representation.
- Joint BERT [54] uses joint intention classification and slot filling based on the pre-trained BERT model.
- BERT-SLU [55] provides a novel encoder-decoder framework based on a multi-task classification method to joint learn intention detection and slot-filling. The model uses BERT as an encoder to train utterance and then design a decoder to detect intention label.
- Cross-Lingual transfer [47] uses a novel method of using a multilingual machine translation encoder as contextual word representations to predict intents.

According to previous studies, there are several multi-turn dialogue datasets contain the intention detection task. In particular, we also verify the model on the multi-turn dialogue dataset to evaluate the model generalization capability. Therefore, we compare our model with the existing baselines, which includes:

- SVM [10] is a simple baseline model, which applies the text feature and multi-classification algorithm on the dialogue act classification.
- LSTM-SoftMax [80] method applies a deep LSTM model to classify dialogue acts via the SoftMax classifier.
- CNN [22]method utilizes the CNN model to encode the utterance with the Softmax classifier. The encoder considers two preceding utterances as context information in the experiment.
- Bi-LSTM-CRF [24] method constructs a hierarchical bidirectional LSTM as an encoder to learn the conversation representation and the conditional random field as the top layer to predict intention label.
- CRF-ASN [55] incorporates hierarchical semantic inference with memory mechanism on utterance modeling at multiple levels and uses a structured attention network on the linear-chain CRF to dynamically separate the utterance into cliques.
- Dual-Attention [56] utilizes a novel dual task-specific attention mechanism to

Table 3.2 The recognition results on the Snips, ATIS and Facebook (EN) datasets.

	Snips	**ATIS**	**Facebook**
Attention-BiRNN [45]	96.7	91.1	97.3
Slot-Gated Full-Attention [46]	96.7	93.6	93.75
Slot-Gated Intent-Attention [46]	96.8	94.1	95.43
Capsule-NLU [52]	97.3	95.0	-
Joint BERT [48]	97.3	97.5	-
Joint BERT+CRF [48]	98.6	97.9	-
BERT-SLU [49]	98.96	**99.76**	98.88
Cross-Lingual [42]	-	-	99.11
RAN-CNN	97.43	97.23	**99.13**
RAN-RMCNN	99.14	98.79	**99.12**
RAN-BERT	98.71	96.75	98.68
SEQ-CNN	98.43	98.21	**99.18**
SEQ-RMCNN	**99.29**	99.32	**99.22**
SEQ-BERT	99.00	97.31	98.97
Fusion Feature	**99.31**	99.56	**99.28**

Table 3.3 The recognition results on the DYDA and MRDA datasets.

	DYDA	**MRDA**
SVM [5]	75.9	82.0
LSTM-SoftMax [9]	79.6	84.6
CNN [10]	79.1	86.8
Bi-LSTM-CRF [24]	85.7	90.9
CRF-ASN [54]	-	91.7
Self-Attn-CRF [55]	-	91.1
Dual-Attn [59]	**88.1**	**92.2**
RAN-CNN	84.5	83.4
RAN-RMCNN	85.5	87.6
RAN-BERT	85.6	89.2
SEQ-CNN	**88.7**	83.6
SEQ-RMCNN	**91.0**	88.0
SEQ-BERT	**89.6**	89.6
Fusion Feature	**91.3**	91.0

utterances.

- Self-attention-CRF [57] proposes a hierarchical deep neural network to model different levels of utterance and dialogue act and use CRF to predict dialogue acts.

Table 2.2 and Table 2.3 show the intention detection accuracy on different datasets. Precisely, the prefix RAN means random triplet sampling strategy, and SEQ refers to the sequential triplet sampling strategy. The RAN-BERT means the random sampling strategy with the BERT model as Siamese encoder, and the SEQ-BERT means the sequential sampling strategy with the BERT model as a Siamese encoder. The rest model name is the same meaning.

As we can see from the results shown in Table 2.3 and Table 2.4, the proposed model significantly outperforms baseline models and achieve state-of-the-art performance on Snips, Facebook (EN), and DYDA datasets. Although the proposed model does not obtain the-state-of-the-art results on ATIS and MRDA datasets, it still can show that the feature learning ability of the proposed model is useful. For the task-oriented dialogue dataset, the proposed feature learning model achieves the recognition accuracy of 99.29% (from 98.96%) on the Snips dataset, 99.22% (from 99.11%) on Facebook (EN) dataset. The fusion features also improve the performance slightly that obtain 99.31% on the Snips dataset, 99.56% on the ATIS dataset, 99.28% on Facebook (EN) dataset. For the multi-turn dialogue dataset, the model SEQ-CNN, SEQ-RCNN, and SEQ-BERT of the DYDA dataset improve the accuracy over the-state-of-the-art model by 0.6%, 2.9%, and 1.5%, respectively. The multi-source data fusion compensates for the lack of data-sparse to a certain extent. It boosts the performance of other methods because it integrates a wide range of available features, which achieves 91.3% on the DYDA dataset and 91.0% on MRDA.

However, the gains on the ATIS dataset and MRDA dataset are slight. One of the reasons for this phenomenon is that the data distributions in these two datasets are both imbalanced. In the MRDA dataset, the class ‘Statement’ is occupied more than 50% of the intention category. In the ATIS dataset, the intention label “flight” also accounts for almost half of the total training data. Based on the sampling strategy, the sampled utterances can be affected by the proportion of intent categories in the database. It is

Table 3.4 The results comparison of basic model and proposed model for different dataset.

	SNIPS	ATIS	FB(EN)	FB(SP)	FB(TH)	DYDA	MRDA
CNN	97.14	96.98	98.10	-	-	79.62	81.05
RMCNN	98.57	98.77	98.13	-	-	82.14	83.54
BERT	98.63	96.62	98.42	97.08	95.80	84.21	88.05
RAN CNN	97.43	97.23	99.13	-	-	84.56	83.47
RAN RMCNN	99.14	98.79	99.12	-	-	85.47	87.65
RAN BERT	98.71	96.75	98.68	96.91	94.39	85.66	89.25
SEQ CNN	98.43	98.21	99.18	-	-	88.69	83.66
SEQ RMCNN	99.29	99.32	99.22	-	-	91.03	88.07
SEQ BERT	99.00	97.31	98.97	97.67	96.39	89.61	89.69

difficult for the model to learn the exact features for very few classes. Another reason is that the ambiguity of label correlation and label annotation is harmful to triplet feature learning. Besides, the MRDA dataset was found to have a high negative correlation between previous label entropy and accuracy, indicates the impact of label noise. Some utterances in ATIS dataset contains more than one label. In this experiment, we only study the single intent of utterance, which affects the results to some extent. The last reason is that the triplet training method adopts the flat dialogue structure to compose utterance triplets and predict the intents based on the multi-task learning approach in the downstream task. The multi-task learning model only focuses on the current utterance ignoring the hierarchical context structure information that damages the recognition performance of multi-turn conversation. In the future, we also need to consider how to be more effectively integrated triplet training with the nested structured dialogue.

3.4.2 Ablation Studies

We can observe the improvement of the proposed model in the last section, and then we explore the contribution of each part in this section. We first perform ablation studies to verify the proposed feature embedding models, whether to contribute to the intention classification task. Then, we explore the details about the effect of BERT model selection.

Next, we study the impact of the sampling strategy selection. Besides, the margin parameter selection also is vital for model optimization. We test the wide-range margin parameters in the experiment. Finally, we exploit the T-SNE visualization method to verify the feature embeddings of the pre-trained feature learning models.

3.4.3 The Effect of the Encoder Selection

Table 3.5 shows the comparison between the basic models and the proposed triplet training model of different dialogue datasets. To validate the generation ability of the proposed model, we also add the other multilingual Facebook data (Spain version and Thai version) in the experiment. The CNN and RCNN models require particular text preprocessing for different languages, so there is no comparability in this experiment. Hence, we fine-tune the pre-trained multilingual BERT model to evaluate the two datasets. We implement comparative experiments under fixed hyperparameters and parameters.

The results shown in Table 3.5 can prove that the pre-trained feature learning models are sufficient to learn more discriminative feature representation for the intention classification task. Precisely, the fine-tuned BERT model performed better than RMCNN model in basic models. However, we can see the triplet training can significantly improve the learning ability of RMCNN. From Table 3.5 the SEQ-RMCNN model performs better than the BERT and CNN encoder on Snips datasets, ATIS dataset, Facebook dataset, and DYDA dataset. We attribute this to the fact that the combination of Wikipedia embedding and RMCNN model can effectively capture granular semantic details locally. Also, the Siamese BERT encoder improves the results of the intention classification because the pre-trained BERT model can provide rich semantic information by unsupervised trained with enormous external knowledge. The results demonstrate that the pre-trained feature embedding model can effectively improve conventional multi-task classification by supplementing utterance triplet training.

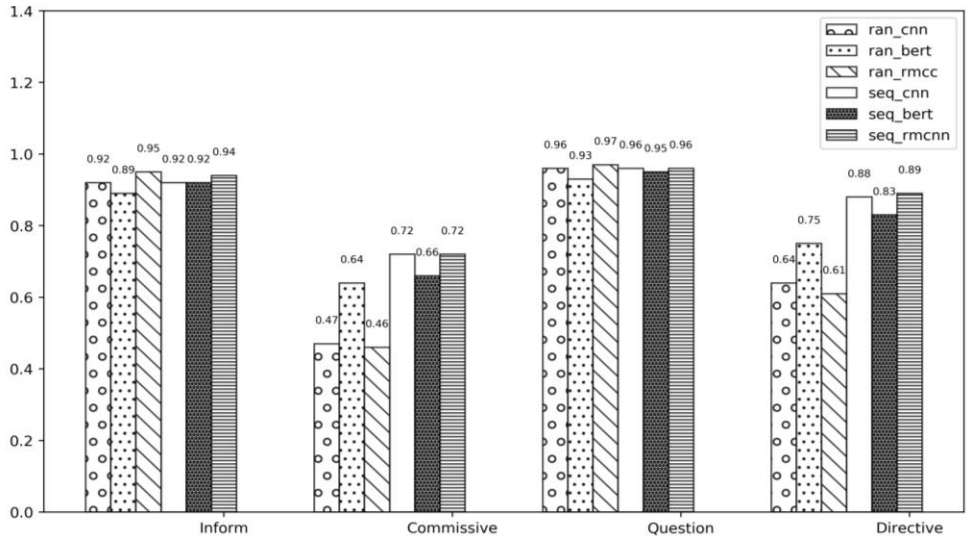


Figure 3.6 The effect of different encoders and sampling strategies on MRDA.

3.4.4 The Effect of Sampling Strategy

In this section, we discuss the effect of sampling strategy on classification results. Based on the results of Table 3.5, it can illustrate that two sampling strategies can effectively improve the results of the basic models (without triplet training). To be specific, the sequential method is slightly better than the random method. Besides, the multilingual dataset also shows the sequential strategy is better than the random strategy. The SEQ-BERT improved by 0.76% over RAN-BERT in the Facebook dataset (Spain) and 2% in the Facebook dataset (Thai). The reason for these results is that the feature learning model might learn the useless context information because of random selection.

Furthermore, we make a comparison between each intention label of the DYDA dataset to show the effect of different strategies on context-sensitive data in detail. As we can see in Fig. 3.6, the DYDA dataset has four intention labels, which are Inform (1), Commissive (2), Question (3), and Directive (4). The proposed models generally perform great on label “Inform” and “Question” because these two intents often appears in spoken language. Although it performs poorly in tag “Commissive” because of the lack of data, we still can find the sequential strategy can improve features to be more distinguished.

Table 3.5 The comparison of pre-trained BERT models with triplet training.

	Snips	ATIS	Facebook
BERT cased base	97.29	95.30	98.52
RAN-BERT cased base	96.43	95.58	98.28
SEQ-BERT cased base	98.14	95.63	98.53
BERT uncased base	98.43	95.52	98.36
RAN-BERT uncased base	97.86	95.75	98.63
SEQ-BERT uncased base	98.97	97.20	98.90

Table 3.6 The comparison of BERT token embedding.

	Snips	ATIS	FB (EN)	FB (SP)	FB (TH)
BERT	97.43	95.52	98.36	95.27	89.48
RAN-BERT	97.86	95.75	98.63	96.94	93.97
SEQ-BERT	98.97	97.20	98.90	97.47	95.15
T-BERT	98.63	96.62	98.42	97.08	95.80
T-RAN-BERT	98.71	96.75	98.68	96.91	94.39
T-SEQ-BERT	99.00	97.31	98.97	97.67	96.39

Table 3.7 The comparison of RMCNN token embedding.

	Snips	ATIS	Facebook
RMCNN	97.32	96.30	97.49
RAN-RMCNN	97.42	96.58	97.88
SEQ-RMCNN	98.14	96.74	98.63
T-RMCNN	98.57	98.77	98.13
T-RAN-RMCNN	99.14	98.79	99.12
T-SEQ-RMCNN	99.29	99.32	99.22

Specifically, the result of SEQ-CNN grew by 0.25 over RAN-CNN, the result of SEQ-RMCNN improved by 0.26 over RAN-RMCNN. The “Directive” label promotes 0.24 on CNN, 0.28 in RMCNN, only 0.08 in BERT. Therefore, the sequential sampling strategy can effectively select valid utterance triplets for spoken language objects.

3.4.5 The Effect of the BERT Model Selection

In this section, we study the influence of the choice of the pre-trained BERT models based on the single-turn dialogue datasets. The pre-trained BERT models are publicly released on Google's GitHub website1. The BERT model includes a monolingual version and a multilingual version. According to the results, we find the monolingual models benefit the English dataset, but it improves less on Facebook (Spain) and Facebook (Thai) datasets. The multilingual model can effectively improve the performance of the cross-language datasets. Therefore, we use monolingual models to deal with English datasets and use multilingual models to train other language datasets. Besides, the BERT models contain two uncased versions and two cased versions. Therefore, we conduct a comparison of basic BERT and BERT triplet training on the English version dataset. To keep the parameters to a minimum in the interaction system, we only verify the model on the base model. From Table 3.5, we can see the uncased model is better than the cased model for the short text spoken language. The random sampling strategy might inferior the performance of the cased model on Snips and Facebook dataset. In the following experiments, we finally adopt the result of the Bert uncased base model as Siamese BERT encoder to train utterance triplets.

Moreover, we verified the effect of token embedding on the task-oriented dialogue dataset. We assume the token embedding might provide finer-grained semantic information of utterances compared with sentence embedding. Therefore, we facilitate the comparison between sentence embedding and token embedding on all task-oriented dialogue dataset. We indicate the T as the token embedding in Table 3.6 and Table 3.7. As we can see in Table 3.6 and Table 3.7, the token embedding can enhance the semantic information of utterance and improve the performance of intention detection. Therefore, we choose token embedding as feature embedding in this experiment.

[1] https://github.com/google-research/bert

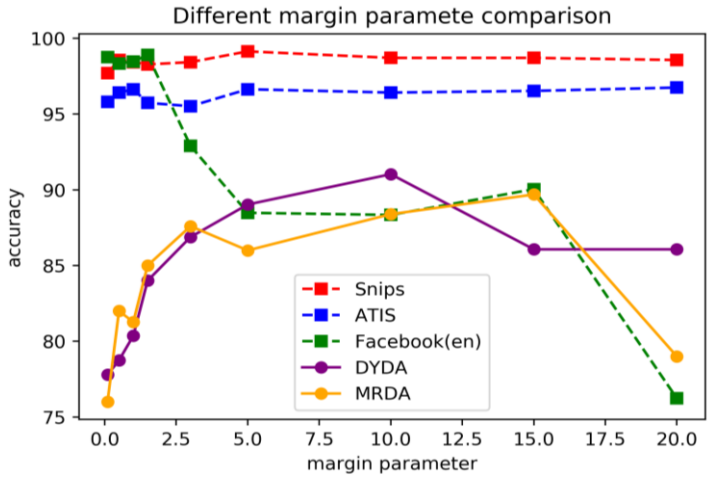


Figure 3.7 The results comparison of different margin parameter based on different dataset.

3.4.6 The Effect of the Margin Parameter Selection

As we mentioned in formula 3.16, the margin parameter controls the relative distance between the feature embeddings to its *positive* samples and *negative* samples. Therefore, the margin parameter selection is essential for model convergency and optimization. From Fig. 3.7, we can observe that the triplet loss optimization is sensitive to the margin parameters. The margin parameter is too large or too small, both result in inferior performance. The large margin parameter may cause over-fitting, and the small margin parameter may impair the strength of the triplet loss because the small value not enough to distinguish between details. Therefore, we conduct different margin parameters under fixed hyperparameters in the experiment to observe the impact of margin parameters on recognition performance. We evaluate the margin parameters on wide-ranged values from 0.1 to 20. We list the final choices of the margin parameter for each dataset. To be specific, we use 5 for the Snips dataset, 1 for the ATIS dataset, 1.5 for the Facebook dataset, and 15 for DYDA and MRDA dataset. Therefore, we set the fixed margin parameter in the following experiments.

3.4.7 The Visualization of Feature Embeddings

In this section, we apply the T-SNE [58] method to visualize 2D feature embedding of test data learned from triplet learning models. Based on the T-SNE visualization method, we can intuitively observe the impacts of feature learning models on different datasets in Fig. 3.8. The first column is the original data distribution of each dataset, and the second column is the embedding features of the SEQ-BERT model. As we can see in Fig. 3.8, the feature embedding of the same intention category is visibly getting closer to each other and gain distinct clusters at the same time. Hence, the proposed models are benefits for extracting more discriminative features through utterance triplet training. The triplet loss training results in a better feature embedding since the margin parameter is considered appropriately.

However, the feature embedding of the MRDA corpus is not as explicit as the DYDA dataset cause the data distribution of the MRDA dataset is imbalanced. The “Statement” tags are occupied approximately 50% in test data, so the rest of the four intents are not clear enough in visualization. Therefore, this visualization reveals the intuition that better underlying feature embedding for short utterance can be obtained by Siamese neural network architecture with metric learning.

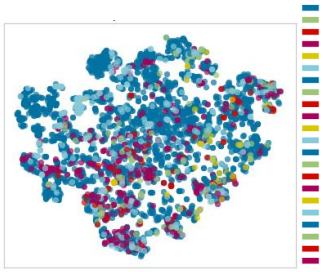
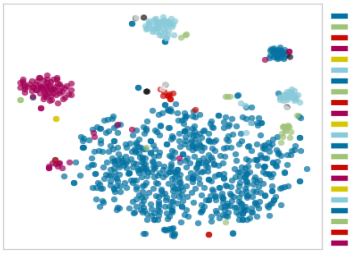
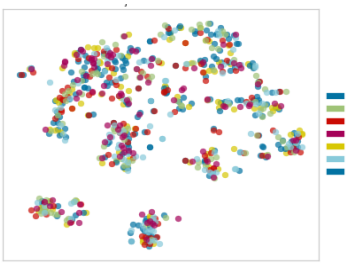
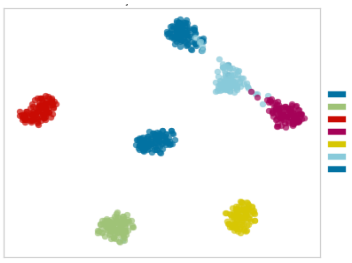
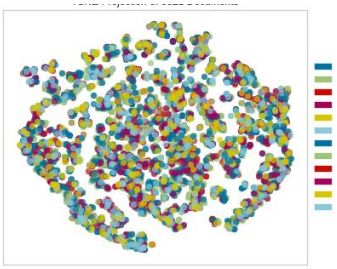
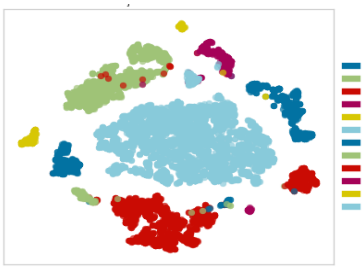
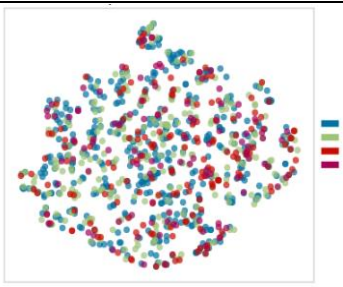
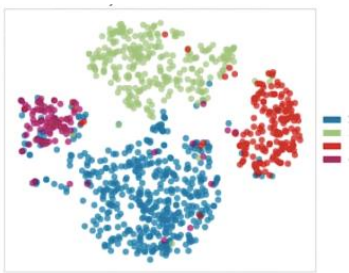
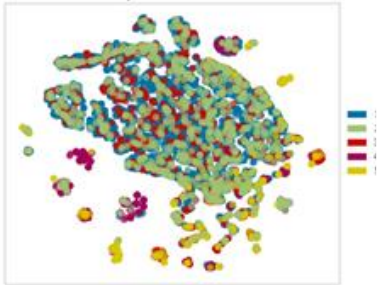
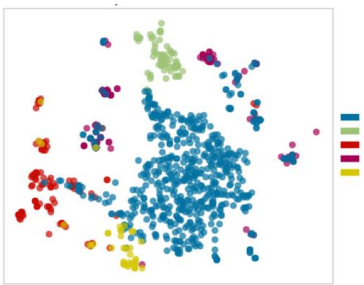
Original distribution of ATIS	T-SNE visualization of SEQ-BERT
	
Original distribution of SNIPS	T-SNE visualization of SEQ-BERT
	
Original distribution of Facebook	Original distribution of DYDA
	
T-SNE visualization of SEQ-BERT	T-SNE visualization of SEQ-BERT
	
Original distribution of MRDA	T-SNE visualization of SEQ-BERT
	

Figure 3.8 The T-SNE visualization of original data distribution and feature embeddings.

Chapter 4

Zero-shot Intent Detection with Intent-enhanced BertCapsNet

4.1 Introduction

Previous studies [59][61] for unknown intent detection use external source like label ontologies or predefined attributes. However, such resources also require extra time to annotate and some external information are not available in dataset. The paper augment the training data by utilizing adversarial learning method to generate positive and negative samples for zero-shot intent detection. However, unlike the continuous data in visual recognition task, the data augmentation method may not work well for discrete data. Moreover, some researches learn a high-quality mapping function to project the similarity between intent and utterance. However, the attributes of spoken language that we mentioned above increase the learning difficulty and effect.

Recently, the capsule network achieved in image classification filed, and it also be testified the advantages for NLP filed. The capsule network can capture the inherent spatial relationship between a part and a whole, thereby automatically generalized to novel new points. This attribute shows the potential of capsule network in the zero-shot intent detection task. The semantic compositionality that the meaning of the whole is composed of partial meanings can be learned by using capsule network via dynamic routing mechanism, which means the routing-by-agreement mechanism determine the

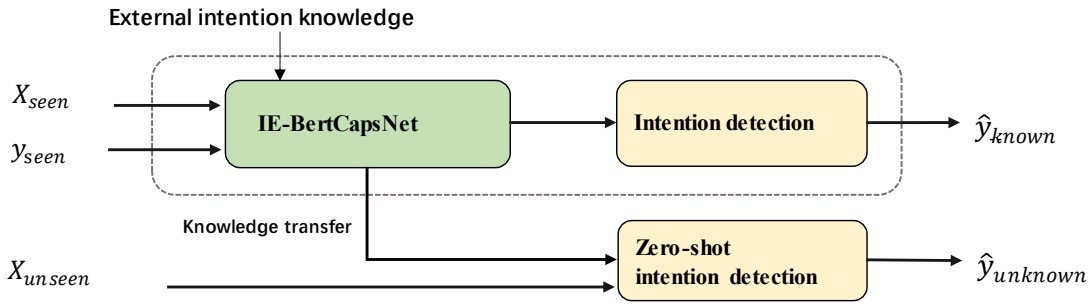


Figure 4.1 The pipeline of unknown intent detection.

connection strength between lower-level capsule layer and upper-level capsule layer and detection as example, low-level utterance semantic feature (mucis_name, get_action) contribute to a more abstract intent (play_music). The IntentCapsNet model obtain impressive results in zero-shot intention detection by utilizing the capsule network to extract the utterance semantic features and transfer the knowledge from existing intention to emerging intention.

In this section, we propose a novel semantic-enhanced capsule network structure for zero-shot intention detection. First, we obtain powerful context-dependent utterance feature embedding by fine-tuning the BERT model. The pre-trained language model is unsupervised trained by using unlabeled corpora and can be easily applied to downstream tasks. Then, we implement intent-guided semantic enhancement by making full use of the highly meaningful label information. It has been shown that the label correlation in embedding space can improve the recognition performance of zero-shot learning [76]. The label embedding method injects attentive weights of label samples into utterance representation by learning the shared joint space of utterance and intents [62]. Next, the semantic utterance features can be fed into an attentive capsule network to learn the abstract intent representation. The conventional intent detection task optimizes the model with SoftMax loss. For network optimization, the margin loss is effective for intention detection task [33]. The learned feature optimized with the supervision of SoftMax in conventional intention classification is limited. It only focuses on a decision boundary instead of considering the intra-class compactness and inter-class separation. Thus, we

replace the SoftMax loss with margin loss to learn more distinguishing features. Then, we optimize the proposed model with large margin cosine loss. It formulates the SoftMax loss into cosine loss with L2 norm and further maximized the decision margin in the angular space. Finally, we conduct zero-shot intent detection by utilizing the similarities of an unseen intent to seen intents and semantic feature extracted by the proposed feature extractor layer.

The contributions of this paper are summarized as follow:

- We propose a semantic-enhanced attentive Bert capsule network to extract and aggregate high-level utterance features, and we inference zero-shot unknown intents based on the proposed method.
- We make the use of the label embedding attentive framework to enhance the semantic information by leveraging the compatibility of embeddings between utterances and intents.
- We improve loss function based on the metric learning approach to obtain a discriminative feature by optimizing the network to minimize inter-class variance and to minimize intra-class variance.
- The experiment conducts on several standard datasets to verify the effectiveness of the proposed method.

4.2 The Attentive Capsule Neural Network

First of all, we formally elaborate concept and definition of the problem. The traditional intention detection task is regarded as a classification problem. The intention label is associated with the utterance. This paper focuses on the inductive zero-shot learning that we do not consider the unseen classes during the training process. Given the set of intent labels $Y = Y^s \cup Y^u$, Y^s is the set of seen intents and Y^u is the set of unseen intents, respectively. The number of K seen intents and the number of L unseen intents are no overlap, i.e.$Y^s \cap Y^u = \emptyset$. The label embedding of seen intents and unseen intents are illustrated as $C^s = \{c_1^s, c_2^s, \dots, c_K^s\}$ and $C^u = \{c_1^u, c_2^u, \dots, c_Z^u\}$ respectively. Suppose

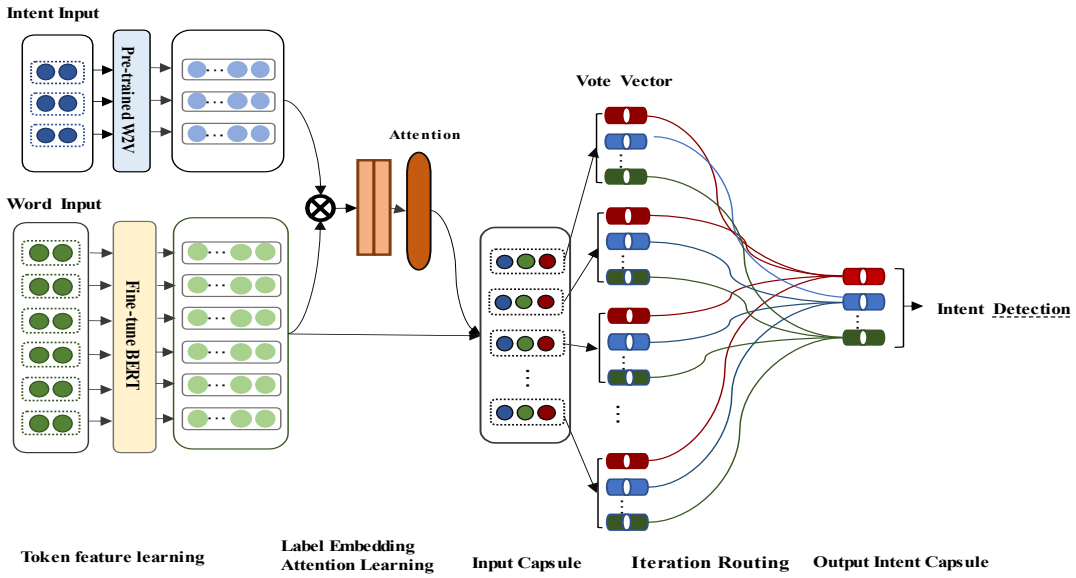


Figure 4.2 The framework of IE-BertCapsNet for unknown intent detection.

given the number of training set $\{(x, y)\}$ of pair-wise data, where $x \in X^s$ is the sequence utterances and $y \in Y^s$ is its corresponding known intentions. The zero-shot intention detection task aims to find a robust classifier to identify the unknown intention of an utterance which belongs to one of the unseen intentions.

4.2.1 Encoder Module with Fine-tuned BERT

In the beginning, we briefly introduce the pre-trained BERT model. Given the number of N input utterances with the number of L tokens $x = \{w_1, w_2, \dots, w_L\}$, we encode the tokens into embedding layers including word embedding, position embeddings, and segment embeddings. Then, the embedding layer is followed by the stack encoding layer, which is composed of a multi-attention sublayer and the position-wise fully connected sublayer. Therefore, we input a sequence of word vector $E = \{e_1, e_2, \dots, e_L\}$ into the encoding layer, the output $S = \{s_1, s_2, \dots, s_i\}$ can be calculated as follows:

$$a_{ij}^{(w)} = Softmax\left(\left(\frac{1}{\sqrt{d_s}}\left(W_Q^{(w)} e_i\right)^T \left(W_K^{(w)} e_j\right)\right)\right) \tag{4.1}$$

$$s_i^{(w)} = \sum_{v=1}^{N} a_i^{(w)} \left(w_v^{(w)} h_j \right) \tag{4.2}$$

$$s_i = W_O [s_i^{(1)}, s_i^{(2)}, \dots, s_i^{(w)}] \tag{4.3}$$

where w is the number of attention heads, d_s is the scale parameter. The W_Q, W_K, W_V and W_O are model parameters and can be learned during training. Then, the output $\widetilde{S} = \{\tilde{s}_1, \tilde{s}_2, \dots, \tilde{s}_i\}$ of the residual connection and the normalization module are denoted below:

$$\tilde{S} = LayerNorm\ (E + S) \tag{4.4}$$

The output $H = \{h_1, h_2, \dots, h_i\}$ of the position-wise fully connected sublayer is calculated as follows:

$$h_i = W_2 ReLU(W_1 \tilde{s}_i + b_1) + b_2 \tag{4.5}$$

in which W_1, W_2, b_1, b_2 are the model parameters. The residual connection and the layer normalization are applied to the output of the encoder block. The final representation $\widetilde{H} = \{\tilde{h}_1, \tilde{h}_2, \dots, h_i\}$ of the BERT model is:

$$\widetilde{H} = LayerNorm\ (H + \tilde{S}) \tag{4.6}$$

the fixed dimensional pooled token representation $\widetilde{H} \in \mathbb{R}^{L \times D_H}$ can be directly utilized in the downstream tasks, where L is the max length and H is the number of hidden units.

4.2.2 Intent-Enhanced Semantic Feature with Label Embedding

Different from feature learning in traditional classification tasks, word embeddings are aggregated into feature representations and then directly input into the classifier. We assume that the interaction of word-intent pair is of great help to model the knowledge transfer between classes. Therefore, we implement intent-guided semantic enhancement by combining the label embedding attention mechanism in the primary capsule layer. We encode the tokens in utterance and its corresponding intention in the same joint space and

measures the compatibility of token-intent pairs to attend to the utterance representations. Specifically, the label embeddings are $C^s = \{c_1^s, c_2^s, \dots, c_K^s\}$ where K is the number of seen intents. Then, we employ cosine similarity to calculate the compatibility between word embedding and label embedding:

$$\mathrm{G} = (\mathrm{C}^T \widetilde{H}) \oslash \widehat{\mathrm{G}} \tag{4.7}$$

where $\hat{G}$ is a normalized matrix of size $K \times L$, and each element in $\hat{G}$ is the multiplication of *L2* normalized *k-th* label embedding and *l-th* word embedding: $\hat{g}_{kl} = \|c_k\|\|o_l\|$. Subsequently, we introduce the non-linearity operation in the word-intents compatibility by using the convolutional neural network and activation functions to better obtain the relative spatial information. For a text phase of length 2_{r+1} centered at *l*, the local matrix $G_{l-r:l+r}$ in G measures the label to token compatibility. The convolutional operation produces high-level compatibility semantic feature $u_l \in \mathbb{R}^k$ between *l-th* words and all labels. The filter is applied to each possible window over the compatibility measures to produce a valuable feature map:

$$u_l = Relu(W_3\, G_{l-r:l+r} + b_3) \tag{4.8}$$

where $W_3 \in \mathbb{R}^{2r+1}$ and $b_3 \in \mathbb{R}^K$ are parameters to be learned in the training. Next, we apply an average pooling layer to capture the most valuable feature from each feature map. Therefore, we add a global average pooling layer over each feature map and take the global maximum compatibility value:

$$p_l = AvgMaxpooling(u_l) \tag{4.9}$$

The attention score $\beta_l \in \mathbb{R}^L$ of the *l-th* elements of an utterance is

$$\beta_l = \mathrm{SoftMax}(p_l) \tag{4.10}$$

Then, the averaging weighted word embeddings $Q \in \mathbb{R}^{L \times D_H}$ as intent-enhanced utterance representation:

$$Q = \sum_l \beta_l \tilde{H}_l \tag{4.11}$$

Algorithm 1 Dynamic Routing Algorithm

Procedure Dynamic Routing ($p_{k|r}, n_{route}$)
Initialize the coupling coefficients $b_{k|r} \leftarrow 0$
For r iterations **do**
    for all capsule *i* in L layer and capsule j in layer (L+1): $c_r = softmax(b_r)$
    for capsule j in layer L+1:

$$v_k = squash\left(\sum_{r=1}^{R} c_{kr}\, p_{k|r}\right)$$

    for all capsule *i* in L layer and capsule j in layer (L+1): $b_{kr} = b_{kr} + p_{k|r} \cdot v_k$
end for
return v_k

4.2.3 Attentive Capsule Neural Network for Intent Detection

In previous learning, we obtain semantic features based on the intent-enhanced feature learning model. Then, we further learn the semantic and syntactic information of each token by utilizing the capsule neural network. The capsules are a group of neurons that use activity vectors to represent the instantiation parameters of entities of an object and improve feature aggregation performance by using a dynamic routing mechanism. The orientation of the activity vector is used to represent the semantic properties of utterance, and the length of the vector is utilized to describe its existence probability. In theory, the capsule network transmits information from the lower-layer capsule to the upper-layer capsule through a dynamic routing strategy, which can be regarded as a bottom-up routing process. In the previous zero-shot learning studies, the attention mechanism has shown its effectiveness. For example, the IntentCapsNet [87] utilizes the self-attention mechanism to extract each utterance's semantic features. The ResCapsNet [85] uses a self-attention mechanism on the word embedding level to alleviate the polysemy problem to some extent. Therefore, after obtaining the intent-enhanced semantic feature embedding, we use the self-attention mechanism to learn the contribution weights of different dimensions of feature embedding. This process enables capsule networks to

determine what and how much information needs to be transferred, as well as to identify sophisticated and interleaved features. Theoretically, the capsule network's bottom-up routing process transmits information from the source capsule node to the target capsule node via a dynamic routing strategy. The detail of the attentive capsule layer is shown in Fig.3. Thus, a dimensional matrix A_r is computed as:

$$A_r = SoftMax\ (F_2 tanh(F_1 Q)) \tag{4.12}$$

where $F_1 \in \mathbb{R}^{D_a \times D_H}$ and $F_2 \in \mathbb{R}^{D_H \times D_a}$ are trainable parameters. The $A_r\,(i,l)$ illustrates the importance weights of the *i-th* dimension of the *l-th* words with respect to the *r-th* semantic feature. It can comprehensively capture the fine-grained semantic feature by selecting the meaningful dimensional attention of a word embedding. Then, the *r-th* semantic feature m_r is:

$$m_r = \sum_{row}(A_r \odot Q) \tag{4.13}$$

where the $\odot$ is the element-wise multiplication and the $\sum_{row}(\cdot)$ sum up all elements of each row. The utterance semantic feature can be written as M = $[m_1, m_2, \dots, m_R] \in \mathbb{R}^{R \times D_H}$. Then, the learned semantic features can be input to a capsule network to form the intent representation by using an unsupervised routing-by agreement mechanism. Firstly, we encode each semantic feature $m_r \in \mathbb{R}^{D_H}$. to prediction vector corresponds with each intent:

$$p_{k|r} = W_{k,r} m_r \tag{4.14}$$

where $p_{k|r} \in \mathbb{R}^{D_p}$ is the prediction vector of the *r-th* semantic feature with respect to the seen intent *k*, and $W_{k,r} \in \mathbb{R}^{D_p \times D_H}$ is a weighted matrix. For the training of seen intention detection, the number of *K* intents corresponding to the number of *K* output capsules. The o_k is the weighted sum of all the prediction vector:

$$o_k = \sum_{r=1}^{R} c_{kr} p_{k|r} \tag{4.15}$$

in which c_{kr} is the coupling coefficient that shows how much information the *r-th* semantic feature is to *k-th* seen intent, which can be calculated by the dynamic routing mechanism. The summarized dynamic routing process is illustrated in Algorithm 1. Afterward, we apply a squashing function to obtain an activation vector v_k for the seen intent class *k*:

$$v_k = \frac{\|o_k\|^2}{0.5+\|o_k\|^2}\frac{o_k}{\|o_k\|} \tag{4.16}$$

As we mentioned before, the probability of the exist intent *k* can be treated as the norm of activation vector v_k.

4.2.4 Improved Margin Loss Function

In the IE-BertCapsNet model structure, each top-level capsule corresponds to an intent. The activation probability of each top-level capsule represents the probability that the input utterance belongs to the corresponding intent. We jointly optimize the model with two parts loss function to train the attentive capsule network, which includes a margin loss and a regularization term. In the conventional intent classification task, The SoftMax cross-entropy loss is widely utilized. However, the SoftMax loss cannot learn discriminative utterance feature because it only considers the boundary whether the label classifies correctly. Therefore, we use the margin loss to replace the SoftMax loss to detect intents. The margin loss function performs well in previous studies, not in the face recognition field, and natural language process filed. It compensates for the shortcoming of SoftMax loss by forcing the model to maximize intra-class and minimize inter-class separation. We define large margin cosine loss (LCML) [59] as the following:

$$L_{lcml} = \frac{1}{N}\sum_i -log\frac{e^{s\cdot(cos(\theta_{y_i})-d)}}{e^{s\cdot(cos(\theta_{y_i})-d)}+\sum_{j\neq y_i}e^{s\cdot cos\theta_{j,i}}} \tag{4.17}$$

The loss is constrained by

$$cos(\theta_j, i) = W_j^T x_i \tag{4.18}$$

$$W = \frac{\|W^*\|}{\|W^*\|}$$

$$x = \frac{x^*}{\|x^*\|}$$

where y_i is the ground-truth intent label of the *i-th* utterance, *s* is the scaling factor, *d* is the cosine margin parameter, w_j is the weight vector of the *j-th* intent label, and θ_j is the angle between w_j and x_i. With normalization and cosine margin, the LMCL loss function converts SoftMax loss into cosine loss by utilizing *L2* normalization on feature and weighted vector.

The second part is a regularization term that shows the importance of each word to the *r-th* semantic capsule. It can be demonstrated by the average value of each column of the attention matrix A_r:

$$s_r = \frac{1}{H}\sum_{col} A_r \tag{4.19}$$

where $s_r \in \mathbb{R}^{1\times L}$ and then we sum up elements of each column. Therefore, the S = $[s_1^T, s_2^T, .., s_R^T]$ illustrates the attentive weight of each word to all *R* semantic capsules. We constrain the column of *S* to be orthogonal with attention loss L_{att} to ensure the semantic diversity, which demonstrated as follow

$$L_{att} = \| S^T S - I \|_F^2 \tag{4.20}$$

Finally, we optimize the whole model by minimizing total loss, which proposed as follow:

$$L = L_{lcml} + \alpha L_{att} \tag{4.21}$$

where α is a trade-off parameter that shows discrepancies among different semantic attention heads.

Table 4.1 Data statistics for zero-shot intent detection.

Dataset	Snips	SMP-2018	FB-EN	FB-TH	FB-SP
Vocab size	11641	2928	4641	1849	1849
Number of Sample	13802	2460	42841	8643	5353
Number of seen Intents	5	24	9	9	9
Number of unseen Intents	2	6	3	3	3

Table 4.2 Hyper-parameter Selection in zero-shot intent detection.

Dataset	D_a	D_p	R	N	σ	Lr
SNIPS	10	10	8	3	1	1e-3
SMP	30	10	8	3	4	5e-4
FB (EN)	30	10	8	3	1	5e-3
FB (TH)	12	10	3	3	1	1e-3
FB (SP)	10	10	3	3	1	1e-3

4.2.5 Zero-shot Intent Detection

In this section, we detect unknown intents by using the proposed model to transfer knowledge from seen intents to unseen intents. Specifically, we obtain the similarity matrix $Q \in \mathbb{R}^{K \times Z}$ of the number of K seen intents and the number of Z unseen intents by

$$q_{zk} = \frac{exp\{-d(c_z^u, c_k^s)\}}{\sum_{K=1}^{K} exp\{-d(c_z^u, c_k^s)\}} \tag{4.22}$$

where $c_z^u, c_k^s \in \mathbb{R}^{D_I}$ are intent embeddings computed by the sum of word embedding of the intent label. The d is *squared Mahalanobis distance* metric measures the relationship between the number of K seen embedding and the number of Z unseen intent embedding, which is computed as:

$$d_M(c_z^u, c_k^s) = (c_z^u, c_k^s)^T \Omega^{-1} (c_z^u, c_k^s) \tag{4.23}$$

where Ω is a covariance matric that models the correlation between embedding of seen and unseen intents. We obtain the correlation among intent embedding use a squared Euclidean distance, which is computed as:

$$\Omega = \sigma^2 I \tag{4.24}$$

4.3 Experiment

4.3.1 Dataset

In this section, we conduct the proposed model on several benchmark datasets for intent detection. We introduce four real task-oriented dialogue datasets with different language versions, which are Snips dataset with English language version [99], SMP-2018 dataset with Chinese language version [101], Facebook's multilingual dataset with English, Spanish and Thai language version [100]. The more detailed information about the corpus can be seen in the research dataset section of Chapter 1.

For the inductive zero-shot intention detection, we take the known intents as the training set and the unknown intents as the test set. In principle, all unknown data should be randomly selected. To facilitate the comparison with the previous studies, we still refer to the data division of the ReCapsNet on the SNIPs dataset and the SMP dataset. For the rest of the datasets, we randomly take utterances of 70% intents as the training set and utterance samples from the rest of 30% intents as the test set.

4.3.2 Implementation Details

In this experiment, we illustrate the implementation details of the training process. For the input embedding, we use the pre-trained language model to replace the original bidirectional LSTM model in IntentCapsNet. In the previous studies, we have already testified the effectiveness of the Bert base model on the spoken language understanding [52]. Therefore, we fine-tune the BERT model to obtain fixed dimensional token embedding, $D_H = 768$. The pre-trained BERT model has 12-layers, 12-heads, and 768-dims. The utterances are padded to the length of L = 50 for each dataset. Regarding label embedding learning, we use $D_I = 300$ word embedding trained by English Wikipedia as input intent embedding. In terms of CNN in label attention learning, the filter window sizes of CNN are selected in the range [1, 2, 3]. Accordingly, the number of filters

corresponding to each window is still the same as 768. In the self-attention layer, we set the self-attention hops R = 8 for SNIPS, SMP, Facebook (English), and R = 3 for Facebook (Thai) and Facebook (Spain). We set self-attention hidden units$D_a = 10$ for Snips, Facebook (Thai) and Facebook (Spain). The main hyperparameters used in the attentive capsule network for different datasets are shown in Table 4.2. After applying the affiliation transformer, we set the dimension of the prediction vector D_p = 10. We set iteration number *N* as 3 for both datasets because it can effectively optimize the model to lower loss. For the loss function, we set the margin as d = 0.35 and the similarity scale as S = 30. The dropout is adopted in each dataset as 0.5 to avoid overfitting. Moreover, we set Adam optimizer [108] with different learning rates for different datasets, and the batch size is 128.

4.3.3 Evaluation Metric

Following the previous studies, we also evaluate our proposed model with accuracy and F1-value. Due to the data imbalance when randomly selecting unknown intents, all the metrics are reported using the average value weighted by their support per class.

4.4 Results and Discussion

4.4.1 Performance Comparison

In this section, we compare our proposed method with several other state-of-the-art baseline models. The zero-shot learning of intent detection is still in infancy, we mainly take the related models from the IntentCapsNet-ZS and ReCapsNet-ZS as the baseline. It includes the followings:

- DeVise [95] proposes a new deep visual semantic embedding model, which can be trained to recognize visual objects using labeled image data and semantic information collected from the unannotated text.
- CMT [103] delicate to build a high-quality mapping from utterance to known intents, so this mapping can be further used to measure the compatibility between

utterances and unknown intents.

- CDSSM [72] focus on the intent expansion. It jointly learns the intents and utterance with character-level and bridge the semantic relationship between known intents and unknown intents.
- Zero-shot DNN [73] uses the different encoders to learn the intent and utterance separately, further improving the CDSSM model.
- IntentCapsNet-ZS [87] leverages the advantage of the compatibility of utterance and intents and directly model the correlation between utterances and intents.
- ReCapsNet-ZS [85] proposes a reconstructing capsule network to further improve IntentCapsNet by dealing with polysemy problems and generalized zero-shot intent classification problems.

We illustrate the zero-shot intent detection results of our proposed model in Table 3 and we highlight the top results in bold. Based on the IntentCapsNet, it also shows the detection performance on the known intents. Therefore, we also present the result of intent detection both on the known intents and unknown intents. Based on the results, we have the following observations. Based on Table 3, the proposed model achieves state-of-the-art results on the English dataset (SNIPS) and Chinese dataset (SMP-2018). In the English Snips dataset, the proposed model achieves 92.39% and 92.27% in Accuracy and F1-value. In the Chinese dataset (SMP-2018), the proposed model obtains 61.17% and 53.93% in Accuracy and F1-value, respectively. The results indeed indicate its advantage of the proposed model in handling zero-shot learning of task-oriented dialogue and achieving state-of-the-art results of the intent detection task. Furthermore, we also take the baseline results from IntentCapsNet on the Snips dataset's 5 known intents for comparison. As we can see in Table 4, the proposed model improved 3.06% and 3.07% over the IntentCapsNet model on the Snips dataset for accuracy and F1-score, respectively.

The improvement is mainly attributed to several aspects. Firstly, the pre-trained BERT model has a strong background knowledge to generate powerful context-

Table 4.3 The zero-shot intent detection using IE-BertCapsNet on two datasets.

Method	Snips		SMP-2018	
	Acc.	F1.	Acc.	F1.
DeVise	0.7447	0.7446	0.5456	0.3875
CMT	0.7396	0.7206	0.4452	0.4245
CDSSM	0.7588	0.7580	0.4308	0.3765
Zero-shot DNN	0.7165	0.7116	0.4615	0.3897
IntentCapsNet	0.7752	0.7750	0.4864	0.4227
ReCapsNet	0.7996	0.7980	0.5418	0.4769
IE-BertCapsNet	0.9239	0.9227	0.6117	0.5393

Table 4.4 The intent detection results on the known intents of Snips dataset.

Method	Snips-NLU (on 5 known intents)			
	Acc.	Precision	Recall	F1
CNN	0.9595	0.9596	0.9595	0.9595
RNN	0.9516	0.9522	0.9516	0.9518
GRU	0.9535	0.9535	0.9535	0.9534
LSTM	0.9569	0.9573	0.9569	0.9569
Bi-LSTM	0.9501	0.9502	0.9501	0.9502
Att-BiLSTM	0.9524	0.9522	0.9524	0.9522
IntentCapsNet	0.9621	0.9620	0.9621	0.9620
IE-BertCapsNet	**0.9927**	**0.9927**	**0.9926**	**0.9927**

dependent utterance representations. The Bi-LSTM model might perform poorly on bridging the semantic gap between known and unknown intents. Secondly, the label embedding learning provides more granular lexical information to capture the correlation between utterances and intents. Thirdly, the capsule attention method can learn a part-whole spatial relationship to aggregate the invariant knowledge for new intents. The iteration routing mechanism in the capsule network furthermore improves the feature clustering by updating on coupling coefficients. In the following section, we illustrate the impact of each part of the proposed framework.

Table 4.5 The ablation study by varying different components of IE-BertCapsNet.

	SNIPs (English)		SMPs (Chinese)		Facebook （English）		Facebook （Thai）		Facebook （Spain）	
	Acc.	F1.	Acc.	F1.	Acc.	F1.	Acc.	F1.	Acc.	F1.
o/w Pre-trained LM	0.7752	0.7750	0.4864	0.4227	0.8042	0.8139	0.6984	0.6831	0.7453	0.7168
o/w Attentive Aggregation	0.8960	0.8965	0.4961	0.4587	0.8577	0.8574	0.7765	0.7349	0.7724	0.7359
o/w Intent Enhancement	0.9128	0.9127	0.5228	0.4671	0.8695	0.8440	0.7818	0.7401	0.7703	0.7302
o/w LCML	0.9266	0.9266	0.5960	0.5102	0.9316	0.9224	0.8039	0.7802	0.7680	0.7357
IE-BertCaps	0.9239	0.9227	0.6117	0.5393	0.9398	0.9354	0.8122	0.7846	0.8131	0.7545

4.4.2 Ablation Studies

In this section, we perform the ablation studies better to understand each module's importance in detail. In this section, we start ablation research along with the structure of the capsule network. Moreover, we added some other task-oriented benchmark datasets to verify the model generalization of zero-shot-intention-detection learning. The results in Table 4.5 indicates:

1) "o/w Pre-trained LM" means that we exclude the pre-trained Bert model, we still use the same method of IntentCapsNet and ReCapsNet (Bi-LSTM and attention mechanism) as the feature encoder.

2) "o/w attention" means that we only utilize BertCapsNet without attentive function to replace the Bi-LSTM to learn the feature.

3) "o/w Intent Enhanced" shows the contribution of joint label embedding learning on intent detection.

4) "o/w LCML" uses the original max-margin loss to replace the large margin cosine loss in the proposed method.

In general, all components contribute to the final detection performance. Specifically, the pre-trained language model plays a vital role in performance improvement for both datasets. The IE-BertCapsNet is superior to IntentionCapsNet because the BERT model

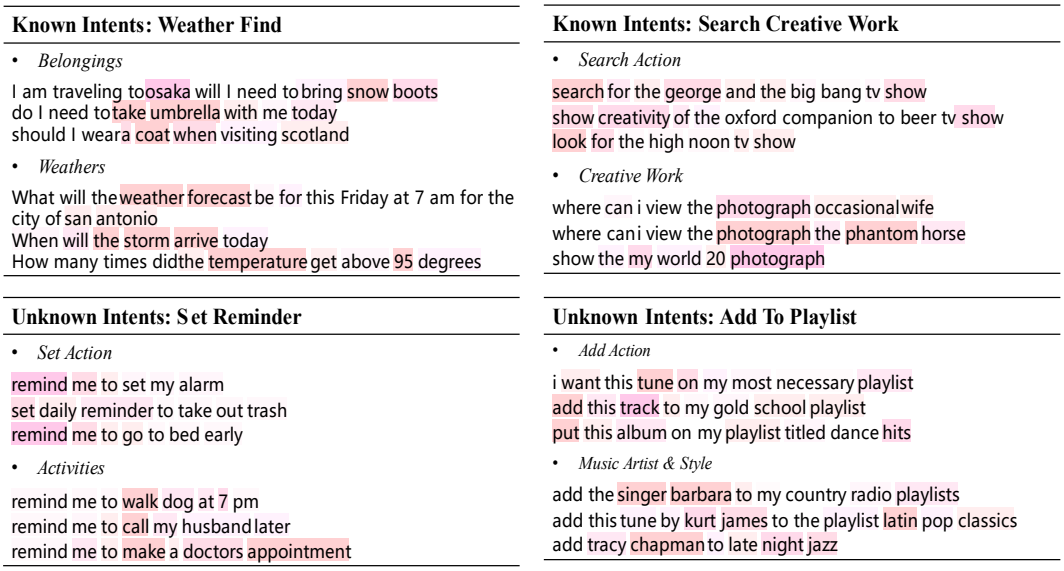


Figure 4.4 The attention visualization of utterance with seen intents and unseen intents.

can learn the rich semantic information and help transfer knowledge from known intents to unknown intents. When we exclude the intent embedding learning from IE-BertCapsNet, the zero-shot detection accuracy drops 1.11% in the Snips dataset, 8.89% in the SMP dataset,7.03% in the Facebook English dataset, 3.04% in the Facebook Thai dataset, 4.28% in Facebook Spanish dataset. The result means that intent learning has enlightening instructive significance for the following iteration process because the tokens that more meaningful to intent can develop the model to learn more fine-grained features. After that, we apply the attentive capsule network structure for feature clustering and selection. We eliminate the attention mechanism from the Bert capsule network to evaluate its capability. As shown in Table 4.5, excluding the attention mechanism in the capsule network impairs the performance both on accuracy and F1 value. In the following discussion, we will also use visualization to analyze the role of the attention mechanism. At last, we use max-margin loss instead of large margin cosine loss to testify its effectiveness. We can find the accuracy rate is reduced on the SMP dataset and all Facebook datasets. LCML loss function initially from CosFace [107]. The application in zero-shot intent detection elaborates its effectiveness in the NLP field. The LCML combined with intent supervised learning to learn more prominent features in the training

process by maximizing the distance between classes and minimizing the distance within categories.

4.4.3 The Effects of Routing Iteration

In some natural language processing articles about capsule networks, we have seen that routing iterative mechanisms will have different effects on the results. There are various studies in the natural language processing field to evaluate the impact of dynamic routing iteration on detection performance. Previous studies show the multiple iterations can furthermore aggregate part-whole features. However, Kim [93] proposed that the static routing mechanism is better in calculating efficiency and classification accuracy than dynamic routing. Therefore, we report the impact of the iteration parameter selection from 1 to 5. The F1 score is the weighted average value over five runs with random initialization. From Fig.4, we can observe that the best results are achieved when the number of iterations is 3. We also verify the static routing mechanism in this paper, when the number is 1, the result tends to decline significantly. The reason the two cases are different because the objects of the two experiments are different. In our paper, the word order's influence or the insertion of an untrained word vector of spoken language is less than in a lengthy document. The routing-by-agreement mechanism can better help aggregate utterances into useful information in the intent.

4.4.4 Visualization Feature Learning and Knowledge Transferability

In this paper, we make extensive use of attention mechanisms to learn feature expression. Therefore, we visualize the attention matrix to observe the model’s semantic extraction ability and knowledge transferability. This section envisions the attentive utterances of known and unknown intents of Snips and Facebook (English) datasets as examples. Firstly, we can observe that the intent labels of the Snips dataset contain multiple meanings. For instance, there is a known intent named "SearchCreativeWork", which includes searching action and creative work property. From Fig.5, the model can both capture these two semantic aspects of an intent label in known intent utterances. The

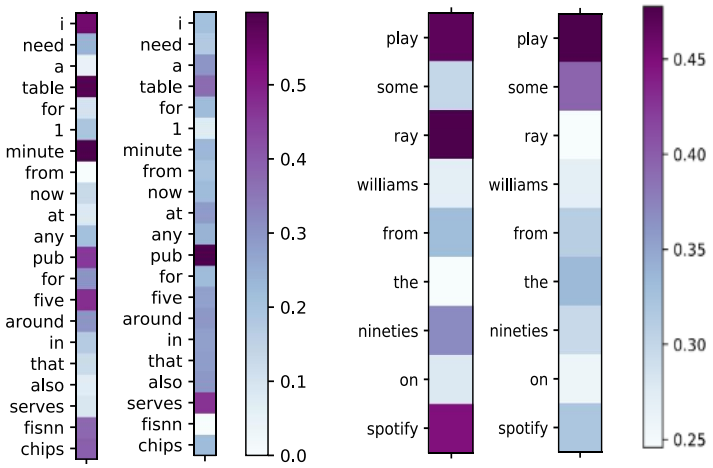


Figure 4.5 Comparison of the word attention score between the intent “Book Restaurant” (left) and “Play Music” (right) with their corresponding utterance learned by proposed model without label attention(above) or with label attention(below).

phrases "look for" and "search for" have a high correlation with the searching action. Apart from this, we can see the creative work also contain high concentration. For example, the word "photograph," "creativity," and "phantom horse" can increase the proportion of "CreativeWork" attributes in feature learning. Moreover, the same phenomenon can be discovered on Facebook (English) dataset, which is also illustrated in Fig. 4.4. We take the "WeatherFind" as an example, the model can not only observe the words directly related to the weather to infer intent but also build implicit relation between utterance and intent based on some related items.

Based on these observations, we intend to verify semantic extracting ability from unknown intents. From Fig. 4.4, we observe that the proposed model has a strong generalization that transferring this knowledge to the inferencing part for unknown intent detection. As we can see the unknown intents from Snips dataset, the intent “RateBook” will pay attention to vocabulary related to rate action, book names, and writer names. The utterance of intent “AddToPlaylist” focuses on the word related to add action and other

information with music. The unknown intent from Facebook (English) dataset verified this situation. The intent “SetReminder” considers setting behavior and pays attention to related events after setting behavior. These observations can prove that the proposed model can transfer knowledge and extract fine-grained features between known and unknown intents.

Furthermore, we also visualize the effect of the label embedding attention mechanism. We leverage the token information by utilizing the label embedding learning method. As we can see in Fig. 4.5, the model combined with the label embedding attention mechanism can extract more distinguished tokens based on intention, which improve the capsule network better aggregates the activation vector towards its intents.

4.4.5 The Discriminative Utterance Feature Visualization

In this section, we utilize the T-SNE visualization tool to evaluate the effectiveness of the proposed model, which is illustrated in Fig 4.6, we utilize the normalized activation vector of known intents and unknown intents to observe the feature learning ability. Based on the visualization of known intents, we can intuitively observe the features of the same intents are visibly clustering together. As illustrated in the figure of unknown intents, we can see the proposed model can effectively capture the discriminative feature of unknown intents for zero-shot intent learning.

Meanwhile, we also find some confusing parts contained in the T-SNE visualization. For the visualization of unknown intents, some data is incorrectly clustered into other categories and the category boundaries of unknown intent are not clear. Several reasons can explain this phenomenon: 1) Since the text length is short, different intentions will have the same expression method, vice versa. 2) In the zero-shot intent detection phrase, the relatively small amount of data in random allocation will cause some confusion in classification. 3) Some intents closed to each other in feature visualization owing to their inherent similarity, such as intents "Bus" and "Flight" in the SMP dataset both belong to transportation. The dialogue will involve some terms related to transportation.

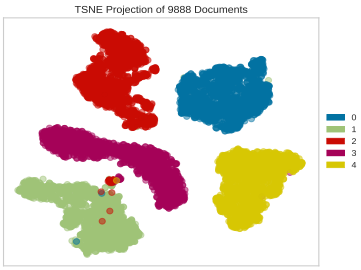
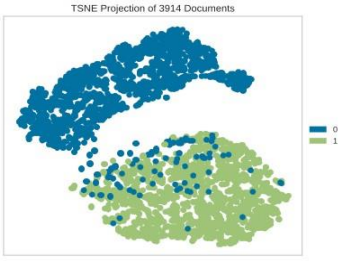
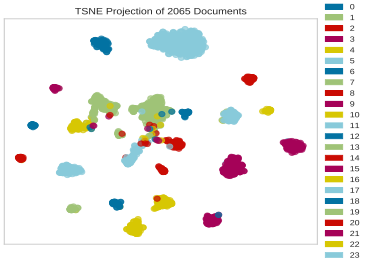
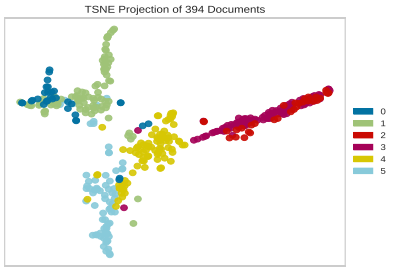
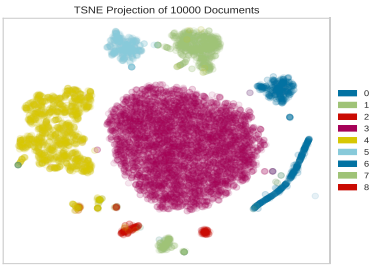
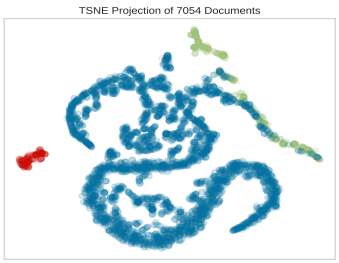
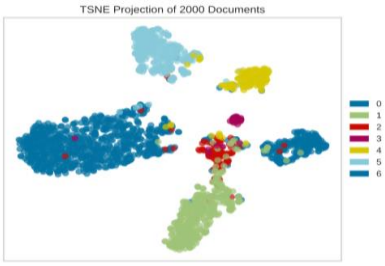
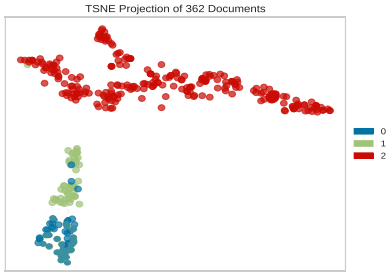
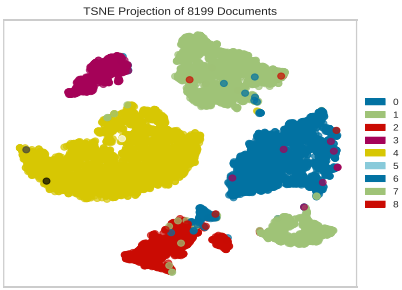
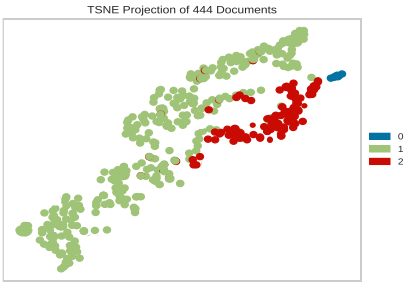
Seen intents distribution of Snips	Unseen intents distribution of Snips
	
Seen intents distribution of SMP	Unseen intents distribution of SMP
	
Seen intents distribution of FB (EN)	Unseen intents distribution of FB (EN)
	
Seen intents distribution FB (TH)	Unseen intents distribution FB (TH)
	
Seen intents distribution FB (SP)	Seen intents distribution FB(SP)
	

Table 4.6 The utterance feature visualization of known intents and unknown intents.

Chapter 5

Dialogue Act Recognition Based on Sequence Labeling Method

5.1 Introduction

Research on DA recognition has been continuing for many years, and mainly focus on defining taxonomies of dialogue acts, searching effective features for the classification task, and discovering the appropriate machine learning methods. For the feature learning approach, there are various features that have been proposed such as dialogue cues, speech characteristics, and n-gram features. These features have been achieved huge success in previous studies. In this section, we treat the multi-turn dialogue dataset as a sequence labeling problem. Therefore, we leverage the conversation's framework as structure features to obtain more useful features

The dialogue act (DA) is a kind of semantic label attached to each utterance, which could help to understand the conversation, ease the interpretation of utterance, and discriminate the user's intension. The DA tags are associated with utterances in a sequential way, so the DA recognition task also can be regarded as a sequence labeling problem. The purpose of sequence labeling is to assign a label to each element in sequence respectively. Based on sequence learning, we could capture the hidden relationship between utterances as features based on observable information, such as utterance position in the entire dialogue and some correlations among consecutive utterances. For

instance, in real conversation, people always start a conversation with ‘Greeting’, and it most likely going to be the same type for the next utterance to respond，i.e. ‘Greeting’ or ‘Question’. At the end of the conversation, people are likely to apply ‘Greeting’ to farewell. Therefore, this experiment takes these implicit elements into feature designing.

Considering the negative effects of natural language and the consecutive characteristic of conversation, this paper is dedicated to finding effective features and suitable classifiers for the study of DA recognition. Therefore, the contribution of this experiment is:

- We propose a word-level sequence annotation approach to represent structural feature and semantic features of dialogue and followed by the Linear-CRFs model to recognize DA tags in the open domain.
- We extend the hierarchical models and combined several recent feature learning techniques to propose a context-aware hierarchical neural network for dialogue act classification. Specifically, we use the fine-grained feature learning approach to encode word embeddings with token, characteristic, speakers. Then, we utilize the multi-heads self-attention mechanism to aggregate high-level feature representations. Finally, we operate the CRF on the conversation representations to decode the sequence dialogue act.

5.2 Dialogue Act Classification with Token-level Sequence Labelling

5.2.1 Feature Extraction

In this section, we firstly utilize the traditional approach to conduct dialogue act classification. The CRFs is an algorithm for sequence labeling problems, cannot directly be used in the classification task. Hence, this experiment supposes to convert feature representation from sentence-level into word-level, which help CRFs obtain more information about utterance and avoid sparse text. Therefore, we provide a feature representation based on word-level sequence annotation.

- **Lexical features**

In this section, Bag-of-word as a feature showed promising results and is wildly used in the previous studies. Therefore, the utterance vectorization realized by the Bag of N-gram method will still be applied as a baseline in this paper. Based on this simple and effective text representation, this paper tests three different Bag of N-gram sets (1-gram, 2-gram ,3-gram) respectively, as well as the 2 groups of gram combinations, 1gram+2gram and 1+2+3gram respectively. Moreover, Kim *et al*. [16] tested the Boolean features to classify the MSN online shopping assistant dialogue, which also is testified in this paper. Besides, this paper also experiments with TF, TFIDF, and information gain as the feature weight. Moreover, this section also represents the utterance by using the distribution representation method as well.

- **Structure features**

According to actual conversation, we could find out the position of the utterance throughout the entire dialogue also have impacts on distinguishing the DA tags. For instance, the greeting will often appear at the begging of the conversation, then followed with the question to continue, and there will be farewell words such as ‘good-bye’ or ‘have a nice day’ in the end to mark the end. Moreover, the position of an utterance in a turn also is an influential factor to discriminate the DA tag. The utterance from a speaker who initiates the conversation until the end is one ‘Turn’. Due to ‘Turn’, the speaker of the next turn can catch the ending of the last speaker and continue the conversation from another start. Hence, the structural features are:

- The speaker information
- The speaker information
- The position of the utterance in the entire dialogue
- The position of the utterance in a turn
- The relative position of the utterance in the entire dialogue
- The relative position of the utterance in a turn

Except these, there are a total of 11 feature sets in this section, and the rest features consist of combinations of these five elements noticed above.

➢ **Dependency features**

According to the frequency of co-occurring dialogue pair analysis, we understand that there are certain correlations among two adjacent utterances, as well as DA tags. For example, 'Question' followed by 'Affirmative' is frequently appearing together. 'Request' will always connect with ''Repeat Response. In other words, preceding DA tags could be useful to predict the next DA tags. From this point of view, this paper identifies the current utterance by using cosine distance to extract its most similar utterance in content and represent its relative location to the current position. Moreover, the natural conversation contains strong logic within the utterance sequence. These utterances from the same author seem to share the content in each other. Combined with this idea, we add utterance similarity from the same author into the feature set. Therefore, this part will be included by:

- The utterance similarity
- The DA tag similarity
- The utterance similarity from the same speaker
- The DA tag similarity from the same speaker

5.2.2 Sequence Labelling with Linear-Chain CRF

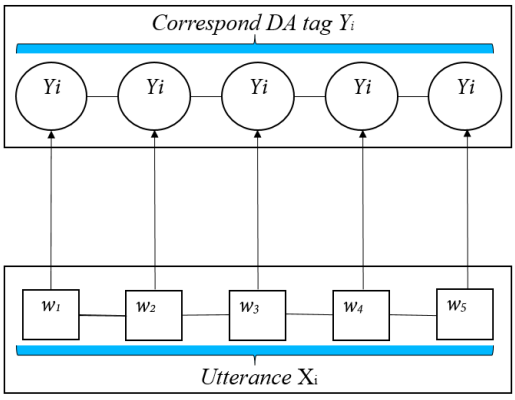


Figure 5.1 The example of word-level sequence annotation approach.

In this experiment, the selected classifier is the Linear-chain CRF, which could be able to capture the dependency among utterance contexts. In a sequence tagging task, predicting labels in each time step in a greedy way may not obtain an optimal result, so we can utilize jointly feature derived from the identity of surrounding utterance as a feature, instead of only focusing on the current position. The CRFs is a form of an undirected graph model that uses a chained undirected graph structure to defines the conditional probability distribution over label sequences given a particular observation sequence. Suppose that each conversation contains sequence utterances, i.e., $x = \{x_1, x_2, \ldots, x_n\}$ with corresponding DA tags $y = \{y_1, y_2, \ldots, y_n\}$. Each utterance is made up of sequence words stringed together, i.e., $x_i = \{w_1, w_2, \ldots, w_m\}$. Therefore, the goal of the DA recognition task is to specify a DA tag for an utterance, the conditional probability to predict optimal DA tags sequence can be written as:

$$P(Y|X;\theta) = \frac{1}{z(x)} exp(\sum_i^T \sum_k^M \lambda_k f_k(y_{i-1}, y_i, x, i)) \quad (5.1)$$

where λ_k is the parameter for defining each feature function f_k. The parameter i is the position of the current token, x is the observation state sequence and *y* is the hidden state sequence. The purpose of this model is to predict the probability of hidden state sequence, given observation state sequence *X*. To learn the parameters of CRFs, we use L-BFGS to train estimation. The model uses *L1* and *L2* as penalty functions to limit the parameters and setting bias to 1.0 to prevent overfitting during training. Given the parameters of the CRF and predicting the optimal probability, the dynamic programming algorithm (Verbit) can be used in this paper to divide the global optimal calculation process into several stages.

To be specific, given an utterance X_i and its corresponding DA tag Y_i, each word that existed in utterance *W* is annotated with the DA tag Y_i. Therefore, the representation of the utterance X_i is consisting of sequence words with the same number of corresponding tag Y_i. Besides, the feature representation of the utterance also considers not only the words sequence but also needs to reflect the character of the dialogue structure and utterance dependency in it. Subsequently, the CRFs model is followed to training the defined feature set to predicting the target DA tags in the test set. LSMR sequence labeling

method is a very common method in the word segmentation field, it can segment words by marking a token position in the phrase to segment words and obtained promising results. Therefore, this paper assumes that sequence labeling based on word-level could improve accuracy on the DA recognition task. The simulative sample in Fig. 5.1 illustrates the concept of the proposed methods.

5.3 Attentive Contextual Hierarchical Neural Network for Dialogue Act Classification

From the feature learning in the previous section, we can see that the features we designed, and the Linear-CRF model has played a role. In this section, we consider the problem of the dialogue act classification (DAC) problem from the viewpoint of extending the handcrafted features to the end-to-end training process. We incorporate the recent advanced deep learning approaches to learn the latent semantic features and structural dependency features in the conversation.

In this experiment, we propose a hierarchical learning approach to learn the conversation from different levels. Regarding the DAC task as a structure prediction, some good performance of previous works has been obtained. Kalchbrenner and Blunsom [80] proposed a mixture model with CNNs and RNNs to encode utterance-level semantic features and discourse-level structure features, which achieved great performance on the DAC task. After understanding the effectiveness of the hierarchical network, lots of recent works have been established. For instance, Li and Wu proposed multi-level gated RNNs to learn the multi-level semantic features and dialogue act dependency. Except for learning the utterance-level feature with deep learning methods, some studies coupled with CRF as a decoder model to discriminate the dialogue act. The CRF model can combine the knowledge of dialogue act transmit pattern in the decoding process, which further improves the classification performance. For example, Kumar *et.al* [24] employed the hierarchical RNNs model to learn sentence-level and conversation-level semantic

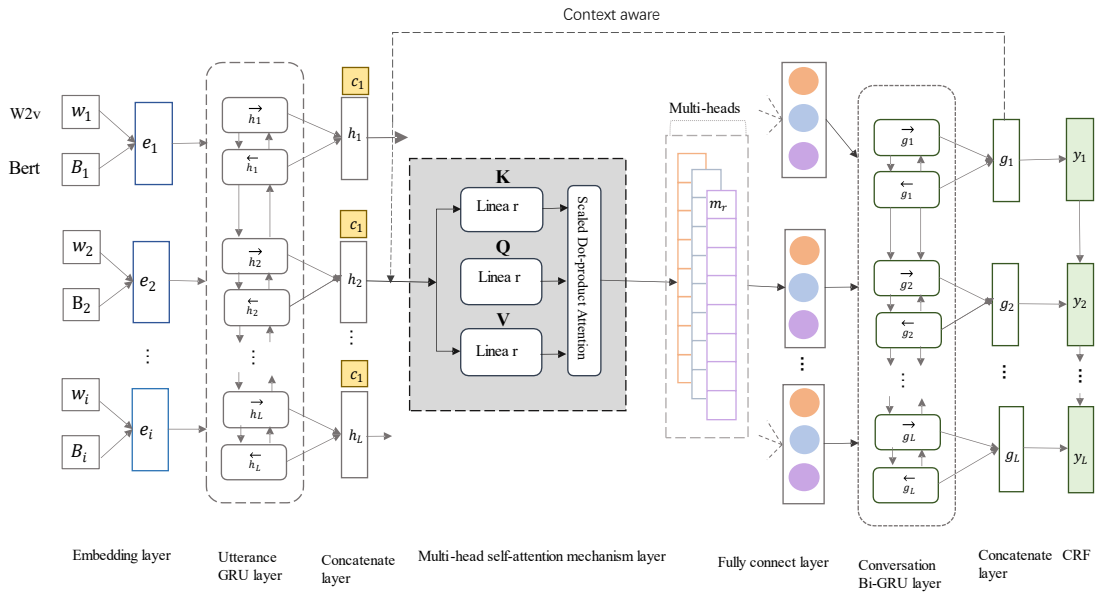


Figure 5.2 The model structure of end-to-end hierarchical neural network.

information. Chen *et. al* [114] concatenate rich word embedding information (token, characteristic, POS, and NER) to produce fine-grained utterance representation. Besides, the conversation-level encoder uses a self-attention mechanism to measure the relevance of input utterance and contextual latent representation. Then, the CRF model is applied to decode the sequence utterance features. Based on the previous studies, we conduct the DAC task from the viewpoint of extending these hierarchical models and leverage recent useful feature learning techniques.

5.3.1 Methodology

Before the training, we need to introduce the problem formulation for the dialogue act classification. The DAC task take the number of N conversation as the training object $D = (C_1, C_2, \ldots, C_N)$. Each conversation C_i is composed of a sequence of utterances $C_i = \{u_1, u_2, \ldots, u_L\}$ and a sequence of dialogue act $Y = \{y_1, y_2, \ldots, y_L\}$. Each utterance $u_i = \{w_i^1, w_i^2, \ldots, w_i^{N_i}\}$ has corresponding dialogue act $y_i \in Y$. Fig. 5.2 shows the whole framework of the proposed hierarchical neural network. In this

experiment, we consider the conversation to be a hierarchical structure with word-level, utterance-level, and conversation-level feature learning. Each component we will illustrate in detail in the next section.

5.3.2 The Hierarchical Conversation Feature Learning

To learn the fine-grained utterance feature representations, we concatenate two-word embedding models to learn the utterance semantic features. In this part, we separately learn the word embedding based on Word2Vec and fine-tuned BERT model. These two models both trained on huge word knowledge and have shown excellent performance in various NLP tasks. Then the word embedding is followed by a Bi-GRU layer to capture temporal information, and the hidden states we define as follows:

$$h_t^i = concat\left[h_t^{i\rightarrow}, h_t^{i\leftarrow}\right] \tag{5.2}$$

where the *concat* operation combines the forward and backward hidden states. Then, we regard the last hidden state as the utterance representation. Then, we add the speaker information into the utterance representation, because the previous studies point out that speaker information can add discriminative features in utterance, which can be written as $U = \{u_1, u_2, \ldots, u_L\}$ and $u_i \in \mathbb{R}^{1\times 2d}$. We concatenate the utterance representation and speaker information to show the final sequence of utterance representation. Therefore, the sequence of L utterances in conversation can be written as $H = \{h_1^{N1}u_1, h_2^{N_2}u_1, \ldots, h_L^{N_L}u_L\}$.

Then, we consider the dialogue history into the model to learn the global conversation feature representation. Therefore, we utilize a 2-layer MLP layer to combine history information and current dialogue hidden state. The operation can be written as follow：

$$R_i = W_1\ tanh\ (W_2 H_i^T + W_3\ g_{i-1} + b) \tag{5.3}$$

where W_1, W_2, and W_3 are weight parameters can be obtained in the training process. ($W_1 \in \mathbb{R}^{u\times 2d}$ and $W_2 \in \mathbb{R}^{2d\times u}$) The parameter u we can set arbitrarily. $W_3 \in \mathbb{R}^{2d\times dc}$, the hyperparameter dc is the size of the hidden state in the conversation level, and b is a

vector illustrating bias. The sequence utterance representation is fed into multi-head attention layers to aggregate the high-level information. The multi-heads self-attention is an effective method of leveraging context-aware features over variable-length sequences for natural language processing tasks. Based on the self-attention mechanism, we firstly conduct different linear transformation on the input vector $R_i \in \mathbb{R}^{L \times 2d}$:

$$\begin{bmatrix} Q \\ K \\ V \end{bmatrix} = \begin{bmatrix} w_q R \\ w_k R \\ w_v R \end{bmatrix} \tag{5.3}$$

w_q, w_k, w_v are the parameters we can learn in the training process. Then, we can obtain sequence utterance representations $m_i \in \mathbb{R}^{L \times 2d}$:

$$m_i = Attention(Q, K, V) = softmax(\frac{QK^L}{\sqrt{d_s}})V \tag{5.4}$$

the attention weights are calculated by the dot production between Q, K and self-attention output m_i is a weighted sum of value *V*. d_s is a scaling vector. With the number of R multi heads, we concatenate all the outputs with different attention heads. Therefore, the final sequence utterances representation $M = \{m_1, m_2, \ldots, m_R\}$. Then, we project the 2-dimensional representation into 1d-dimensional feature representation by using a fully connected layer, which denoted as $m_i \in \mathbb{R}^{2d}$. In the conversation level, we employ conversational-level Bi-GRU to encode across the utterance sequences:

$$\underset{g_i}{\rightarrow} = \underset{GRU}{\longrightarrow} (m_i, \underset{g_{i-1}}{\longrightarrow}) \tag{5.5}$$

$$\underset{g_i}{\leftarrow} = \underset{GRU}{\longrightarrow} (m_i, \underset{g_{i+1}}{\longrightarrow}) \tag{5.6}$$

$$g_i = concat (\underset{g_i}{\rightarrow}, \underset{g_i}{\leftarrow}) \tag{5.7}$$

g_i provides the conversation-level context, which is further propagated to a linear chain CRF layer. The CRF layer considers the transition probabilities between dialogue acts in context and jointly decodes the optimal sequences of dialogue acts, instead of decoding

Table 5.2 The samples of SWDA dialogue dataset.

ID	*DA*	*Caller*	*U_Idx*	*Sub-Utt*	*Text*
1	b	B	1	utt1	Uh-huh.
2	sd	A	2	utt1	I work off and on just temporarily and usually find friends to babysit.
3	sd	A	2	utt2	I don't envy anybody who's in that situation to find day care.
4	b	B	3	utt1	Yeah.

Table 5.1 The dataset statistic for dialogue act classification based on end-to-end training.

Dataset	*C*	*V*	*Training*	*Validation*	*Testing*
SWDA	42	19k	1003(173k)	112(22k)	19(4k)

each dialogue act independently. For the testing, we adopt *Viterbi* algorithm to obtain the optimal sequence by using dynamic programming techniques.

5.4 Experiment Setting

5.4.1 Dataset

In this section, we evaluate the proposed model on the same benchmark dataset for the DAC task. The SWDA corpus contains audio recordings and transcripts of a telephone conversation between speakers. For each conversation, a total of 66 topics are provided to speakers for communication. Overall, there are 42 dialogue act labels in the corpus, and all are annotated by DAMSL taxonomy [110]. This paper adopts the data split of 1115 training dialogue. Table 5.1 is a simple structure of the SWDA dialogue. Specifically, ‘Id’ notes the index number of each utterance. ‘DA’ tag is a dialogue act corresponding to utterance. ‘Caller’ is the identification of the speaker. ‘Utt’ number means the position of the utterance in the entire dialogue text, ‘Sub-Utt’ represents the position of the utterance in a turn, and text is the content of the utterance.

5.4.2 Hyperparameter

For the experiment with handcrafted features, this thesis employs different feature extraction algorithms to obtain structure features and then followed by three basic machine-learning methods, Naive Bayes, support vector machines, and conditional random fields. Naive Bayes and support vector machines are both come from the Scikit-learn toolkit, and Conditional Random Field (CRF) come from the Pycrfsuite toolkit. In order to testify the stability of the model, this paper also uses the average result of 10-fold cross-validation.

In the end-to-end training, we combine two pre-trained word embedding models to train word embedding. Firstly, we exploit the pre-trained word embedding trained by English Wikipedia dataset with 300 dimensions. Then, we fine-tune the BERT model (12-layers, 12-heads, and 768-dimensions) to obtain fixed dimensional token embedding. The Bi-GRU is applied throughout our model, and the hidden size is 128. Then, we utilize the multi-head self-attention mechanism to aggregate high-level semantic information. Specifically, we set the self-attention hops as 3. Early stopping is also employed on the training set with the patience of 50 epochs. During training, we adopt Adam optimizer for training with an initial learning rate of 0.005. All the hyperparameter were selected by tuning one hyper-parameter while keeping the other parameters fixed.

5.5 Result and Discussion

5.5.1 The Result and Discussion of Pre-defined Features

Table 5.3 shows the best results of different classifiers by using a variety of Bag-of-gram feature sets. During the experiment, TFIDF (1+2+3 gram) combined with SVM obtain the best results (69%). Besides, this section uses the CRF classifier combined with Bag-of-gram (1- and 2-gram) to carry out experiments respectively, and BOW with unigram achieved 64%, which is slightly higher than bigram. It indicates that CRFs is more useful for Naive-Bayes in this task because the CRFs is capable to consider the relationship among token, words, and part of speech comprehensively. Whereas the Naive Bayes

Table 5.3 Accuracy of different feature representations and weighting methods.

Learners	Feature	Accuracy
NB	1+2+3gram/Boolean	0.58
SVM	1gram+2gram+3gram/TFIDF	0.69
CRF	1gram/Bow	0.66
	Word Embedding	0.71

Table 5.4 The result of different feature with SVM classifier.

N-gram	**Boolean**	**TF**	**TFIDF**	**IG**
1	0.54	0.65	0.65	0.66
2	0.49	0.59	0.60	0.55
3	0.44	0.51	0.53	0.43
1+2	0.56	0.65	0.67	0.63
1+2+3	0.58	0.65	0.69	0.63

classify utterances into DA tags independently, the correlation among context has not been considering in this approach. Besides, Table 5.3 also shows the result of using word vectors as lexical features of utterances, and then followed by CRFs as a final layer for classification, achieving an accuracy of 71%. The distribution representation could reduce the dimension disaster by converting the words into a smaller dimension space and discriminate the similarity of the words by calculating the distance between these words. Compared with distributed representation, the insufficient of sparse representation is that the parameters will increase exponentially during the training process, which means it is time-consuming and hardly catches the dependency among long sentences.

Table 5.4 shows the result of different feature representations and weighting methods for SVM. Compared with the other two classifiers, the result of SVM is more promising.

Table 5.5 Accuracy with structural feature with CRF.

Feature	Accuracy	
	1-gram	*2-gram*
BOW	0.66	0.65
Word+Caller	0.76	0.74
Word+Pos_dialogue	0.77	0.77
Word+Pos_turn	0.79	0.80
Word+Caller+Pos_dialogue	0.57	0.75
Word+Caller+Pos_turn	0.68	0.68
Word+Caller+Pos_dialogue+ pos_turn	0.73	0.80
Word+R_Pos_dialgoue	0.72	0.72
Word+R_Pos_turn	0.78	0.77
Word+Caller+R_Pos_dialogue	0.77	0.76
Word+Caller+R_Pos_turn	0.74	0.72
Word+Caller+R_Pos_dialgoue+ R_Pos_turn	0.78	0.76

Be specifically, when using 1-, 2-, 3-gram along, the result of the feature with simple 1-gram shows the stability in each classifier. In contrast, the result of 2- and 3-gram leads to a decrease in accuracy, because 2- and 3-grams might lead to the sparse in the training set. Moreover, this section also considers the information gain to select a feature. Information gain can be understood as a decrease in uncertainty of the result. For example, from the actual conversation, ‘Declarative’ action always connects with some question words, ‘Acknowledge’ often collaborates with some modal particles. Based consideration above, we hypotheses that the part of speech of a word in utterance might contain certain effects on determining DA tags. The feature selection through information gain achieves 66% in F1-score. By analyzing the result, it can be figured out that part of speech of the words are only meant for a small part of utterances rather than all categories, like

Statement-opinion and other common DA tags will have limitations. Some utterances related to Statement-opinion tags consist of plenty of types of words, which is difficult to use part of speech alone to express all utterances.

Table 5.5 shows the result of structural features by using the word-level annotation approach combined with CRFs. In particular, “Caller” means the speaker's information, “Pos_dialogue” represents the position of the utterance throughout the entire conversation, “Pos_turn” represents the position of the utterance in a turn. Besides, “R_Pos_dialogue” represents the relative position of the utterance through the whole conversation, and “R_Pos_turn” represents the relative position of the utterance in a turn.

Based on the results of this section, we could figure out that all feature sets outperform Bag-of-*Ngram* in the baseline. Thus, the word-level annotation approach combined with structural feature sets indeed works well in the DA recognition task. To be specific, the most effective feature set obtained 80% in F1 score is “Word+Caller+Pos_dialogue+Pos_turn”. The reason feature combination works well in this task due to comprehensively annotates the local and global information of the utterance. Besides, this section testifies both unigram and bigram with the structural feature. The result shows the average accuracy of the bigram is higher than the unigram 1.1 absolute point, which indicates the bigram could capture more information. Furthermore, the feature sets “Word+Pos_turn” and “Word+R_pos_turn” exceed the results of “Word+Pos” dialogue and “Word+R_pos_dialogue” respectively. According to the result, we believe that the position in a turn is more significant than the position of the entire dialogue.

However, we randomly combine the feature sets in pairs to reduce inaccuracy. The result of “Caller+Pos_dialogue” is only 57% and “Word+Caller+Pos_turn” is only achieved by 65%. By further analyzing the result of the “Caller+Pos_dialogue” feature set, we find that the F1 value of these labels is very low, like ‘Hedge, self-talk, Affirmative non-yes answers, Downplayer, Summarize/reformulate, Rhetorical-Questions, Other answers, Offers-Options-Commits, Quotation, Dispreferred answers, Declarative Wh-Question, Declarative Yes-No-Question’. In particular, the total numbers of these utterances in the corpus are less than Statement and Opinion. It could show that there is

Table 5.6 Accuracy with utterance dependency by using CRF.

Feature	Accuracy
BOW	0.65
Word + C_utt	0.67
Word + C_utt_act	0.66
Word + Pre_utt	0.68
Word + Pre_utt_act	0.68

a serious imbalance situation in the corpus and the manual annotation score of the SWDA corpus is just 84%. For the result of "Pos_turn", we consider that using the position in a turn with the word-level annotation leads to not inspiring results. The reason might be that using the position of each utterance in a turn as a feature is not representative in the DA recognition task.

According to the linguistic phenomenon, there is relevance contained in consecutive utterances in terms of topic and content if the utterance from the same speaker [10]. Therefore, this section will utilize the utterance similarity combined with the proposed method as a dependency feature. Table 5.6 indicates the accuracy of utterance dependency by using CRFs. "C_utt" represents the similarity of utterance context from the same speaker. "C_utt_act" represents the similarity of DA tags context from the same speaker. "Pre_utt" will not consider whether the utterance from the same speaker, only consider the similarity between utterance, "Pre_utt_act" indicates the similarity between DA tags.

In this part, the result shows the dependency feature sets are beneficial to DA recognition tasks and obtain almost similar results compared with baseline, but the improvement of this result is limited. Through the comparison of several group features, it could see that the result of the similarity between dialogue acts is slightly lower than the utterance. The reason might be the length of the dialogue act is too short to provide effective information, so it is hard to capture dependency information between similar utterances.

Table 5.7 The dialogue act classification performance with different approaches.

Model	**SWDA**
Feature Template (2018)	78.2
HCNN (2013) [20]	73.9
RCNN (2016) [22]	73.9
DRLM-Conditional (2016) [115]	77.0
LSTM-SoftMax (2016) [80]	75.8
H-AM-RNN (2017) [112]	79.4
HA-RNN (2017) [113]	73.8
Bi-LSTM-CRF (2018) [24]	79.2
CRF-ASN (2018) [114]	81.3
Proposed methods	**82.9**

5.5.2 The Result and Discussion of Hierarchical Neural Network

We compare the classification performance of the proposed model against several other recent approaches. Except for the feature template method, the rest of the methods in Table 5.7 treat the dialogue act classification as structure prediction with end-to-end deep learning manner. We illustrate each method as follow:

- HCNN: This method introduces an utterance model and a discourse model. The utterance model adopts hierarchical CNN to learn the semantic vectors. The discourse model combines the RNN model to extend the sentence model that is conditioned in a novel way both on the current sentence and on the current speaker.
- RCNN: This method proposes hierarchical CNN on sentence model and RNN on the contextual discourses.
- DRLM-Conditional: This method combines positive aspects of neural network architectures with probabilistic graphical models. The model combines a recurrent neural network language model with a latent variable model over the shallow discourse structure.

- LSTM-SoftMax: This method utilizes LSTM to learn the utterance feature representation and classify dialogue act via SoftMax operation.
- H-AM-RNN: This method explores the context representation learning methods for dialogue act classification. It combines RNN architectures and attention mechanisms at different context levels.
- HA-RNN: This method proposes a novel hierarchical Recurrent Neural Network (RNN) for learning sequences of dialogue acts. This model utilizes two hierarchical RNN models with an attention mechanism to capture the temporal dependency at the utterance level and conversation level. The attention mechanism is used to learn the salient tokens in utterances.
- Bi-LSTM-CRF: This approach learns the conversation representations with multiple levels: word, utterance, and conversation level. It constructs a hierarchical bidirectional LSTM as an encoder to learn the conversation representation and the conditional random field as the top layer to predict intention label.
- CRF-ASN: This method employed hierarchical semantic inference with memory mechanism on utterance feature learning, and then extend the attention network to CRF layer to predict sequence dialogue acts.

Among them, the former five methods utilize the SoftMax operation in the final layer for sequence labeling and the last three methods use linear-CRFs in the final layer. Compared with other methods, Table 5.13 shows that the proposed model outperforms the best baseline method by 1.8% on SWDA. We conduct the testing evaluation based on the hyperparameters which achieved the best results on the validation dataset. The results prove that the attentive hierarchical feature learning structure can effectively learn the conversation feature representation. Moreover, the CRF-based model can achieve better results than SoftMax, because the transition pattern between dialogue acts recorded by the CRF model can improve the model prediction performance. Among the compared models, some methods utilize the attention mechanism for conversation feature learning. All three models, H-AM-RNN, HA-RNN, and CRF-ASN utilize the attention mechanism in their utterance feature learning process to measure the relevance between current

Table 5.8 The ablation studies of the proposed method.

Method	**SWDA**
w/o Word2Vec Embedding	81.7
w/o Fine-tune BERT	79.3
w/o Multi-heads Self-attention Mechanism	78.6
w/o Multi-heads operation	79.4
w/o Speaker Information	82.7
w/o CRF	80.4
Proposed Model	82.9

utterance representation and the contextual hidden state.

Based on the results, we would like to know the reasons for performance improvement. Therefore, we perform the ablation studies to understand the effects of each component on performance. In this section, we perform several ablation experiments based on the SWDA dataset and the results are shown in Table 5.8. We will describe the detailed illustration in the following paragraph:

(1) w/o Word2Vector word embedding and w/o Fine-tuned Bert model: we utilize the w2v and fine-tuned Bert model to provide rich knowledge background for utterance feature representation. To testify its' effectiveness, we remove the W2vV from word embedding learning models, and the results decay 1.2%. Besides, we remove the fine-tuned Bert model from the proposed model, the performance decreases by 3.6%. Based on the results, we can observe that the fine-tuned BERT model can provide more background knowledge for spoken language with shot length, compared with the W2v language model.

(2) w/o Multi-heads Self-attention: we remove the multi-head self-attention mechanism from the proposed model and the results show a significant drop in performance. Based on the results, we can confirm that the multi-head self-attention mechanism can effectively capture the conversation level context information.

Furthermore, we remove the multi-heads attention operation from this part and only use self-attention to model the context information. We can observe that when the multi-heads operation is removed, the accuracy of the model is reduced by 79.4%, which further demonstrates the contributes of this operation in the self-attention mechanism.

(3) w/o Speaker Information: we add the speaker information in the model to enhance the utterance discriminative feature. We remove the speaker information to testify its effects. Although the performance only slightly reduces 0.2%, we still can see user information can enhance the specificity of the utterance feature representation. The result did not change much mainly because we have too little user information available. If the user's information can be richer, then we can further increase the particularity of the utterance representation, which can be treated as a future research object.

(4) w/o CRF: in this paper, we treat the DAC problem as a sequence labeling task. It is a natural choice to choose the Linear-CRF model to assign sequence dialogue acts to sequence utterances. The CRF can model dependencies among labels by encoding the transition pattern of each dialogue acts. In this setting, we replace the CRF with SoftMax to classify the dialogue acts. Based on the results, the performance is decreased by 1.5%, which illustrates that it is necessary to explore the dialogue act dependencies at the label level.

Chapter 6

Conclusion and Future works

6.1 Conclusion

This thesis has done a series of related studies on the subject of intent detection. These experiments show some enlightening conclusions based on each experiment. Thus, we summarize each experiment in the following sections.

6.1.1 Fusion Triplet Feature Embedding Learning Method for Intention Detection

For intention detection, we formulated the intention detection task from the perspective of enriching semantic information of utterances. In the first stage, we proposed a novel feature embedding model by utilizing the fine-tune BERT model and RMCNN model as Siamese encoders with a triplet loss function. The RMCNN and BERT as Siamese encoders were employed to train utterance triplets, and the triplet loss function can optimize the embedding model end-to-end. Then, we can obtain two well-trained feature embedding models to illustrate discriminative utterance features from different aspects. Moreover, we introduced the sequential sampling strategy in triplet selection to capture context within the dialogue. In the second stage, we used a multi-source fusion strategy to boost the recognition performance of the downstream intention detection task. Given the pre-trained models, we predict intention labels by fusing discriminative pre-trained and other relevant features within the dialogue. The extensive experiments demonstrated

the effectiveness of the proposed model for intention detection on several benchmark datasets. The results illustrate that the proposed method can effectively improve the recognition accuracy of these datasets. For single-turn task-oriented dialogue, the model achieves 99.31% in the Snips dataset, 99.56% in the ATIS dataset, 99.28% in Facebook (English) dataset, 97.67% in the Facebook (Spain), and 96.39% in the Facebook (Thai). For multi-turn conversation, the recognition accuracy achieves 91.3% in the DYDA dataset and 91.0% in the MRDA dataset.

For our proposed feature embedding model, there is still much space for improvements in our system. Firstly, we can verify different neural network architectures, loss functions, and distance metrics based on the pre-training framework. Secondly, the multi-class classification learning approach may inferior the results because the model predicts intents only consider the current time step. Except for the single-turn dialogue and multi-turn dialogue, there are more complicated dialogue structures, such as multi-party and multi-modal dialogue. Therefore, the combination of intricate dialogue structures and metric learning could be a new direction. Furthermore, the triplet loss training also can be employed in other NLP tasks like emotion detection and topic adaptation in the dialogue system field, which are also promising for future research.

6.1.2 Zero-shot Intention Detection with IE-BertCapsNet

For zero-shot learning of intention detection, we formulated the zero-shot intention detection task with the attentive capsule network. We leverage the transfer learning ability of capsule neural networks for text modeling in a hierarchical manner. Firstly, we extract word token embeddings with self-attention and aggregate the utterance features to intent semantic features by the dynamic routing mechanism. Besides, we find out that the label plays a central role in intent recognition. We embed the words and labels in the same joint space to capture the dependencies that make significant contributions, which can further improve the learning ability of the proposed model and maintain low computational cost. Moreover, most previous works treat intent detection as a classification problem where utterances are labeled with predefined intents. The conventional intent detection methods train classifiers with SoftMax in a supervised fashion which only focuses on finding the

boundaries between classes, without considering the compactness of intra-class and inter-class. Therefore, we replace SoftMax with large margin cosine loss (LCML) in this model to learn more discriminative deep features. The loss can force the features directly by minimizing the intra-class variance while also maximizing the inter-class variance at the same time. In the future, we would like to improve the performance of intent detection with higher stability and scalability and expand our work to meet real-life requirements like generalized zero-shot intention detection and improve its performance. Besides, we would like to explore more variants of capsule network structure to satisfy the pressing needs of other natural language processing tasks.

6.1.3 Dialogue Act Classification with Hierarchical Neural Network

In this experiment, we conducted the dialogue act classification task based on two approaches, which are pipeline method with feature design and end-to-end manner with a hierarchical neural network. The first method explored a sequence annotation method based on word-level for classifying DA tags in the open domain, which can greatly reduce the feature sparseness of short utterances and capture detailed information of the utterance locally and globally. Specifically, this paper uses this method to annotate word sequence in utterance and supplement dialogue structural information and semantic information into utterance as feature representation. Then, Linear-CRF as the main algorithm comprehensively captures the constraints of hidden variables in the utterance context to predict DA tags. According to the comparison between baseline and the proposed method, we find out all the designed feature sets outperform baseline. It indicated that the proposed method combined with dialogue structure information and utterance dependency information could perform remarkably well in the DA recognition task. The second method extended the previous hierarchical neural network with a multi-head self-attention mechanism for the DAC problem. Based on the baseline comparison and ablation studies, we can observe that each component of the proposed model both have a positive impact on the classification performance.

6.2 Future Works

At present, intent detection is not only applied in e-commerce, voice assistant, online medical treatment, but also applied to network intrusion, network fraud, and other network security problems. The traditional dialogue system treats intention detection as a multi-class classification problem, which means one utterance corresponds with one intent. With the development of the dialogue system, the requirements for intelligent intention detection have also become more complex. For instance, the user's discourse expression is not limited to only one intent. Multiple intents detection will become a trend. Furthermore, the intentions and emotions arise together, conversations are intrinsically determined by direct, exquisite, and subtle emotions. In a multi-turn conversation with different modalities, the interaction of emotion and intention will affect the direction of the conversation. Therefore, the next step of the dialogue system will inevitably move towards a multi-modal interaction method. Through the unification of vision, voice, language, knowledge, etc., the communication between humans and machines will become unlimited.

Bibliography

[1] K. Noda. Google Home: smart speaker as environmental control unit, Disability and rehabilitation: assistive technology,13(7):674-675, 2018.

[2] A. Purington, J. G. Taft, S. Sannon, N. N. Bazarova, S. H. Taylor. Alexa is my new BFF: social roles, user satisfaction, and personification of the amazon echo. *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems.* pages 2853-2859. ACM, Dec. 2017.

[3] Hoy, MB. Alexa, Siri, Cortana, and more: an introduction to voice assistants. *Medical reference services quarterly*, 37(1):81-88, 2018.

[4] M. Sbisà. Speech acts in context. *Language & Communication*, 22(4):421-436, Oct. 2002.

[5] F. Ren, K. Matsumoto. Semi-Automatic Creation of Youth Slang Corpus and Its Application to Affective Computing. *IEEE Transactions on Affective Computing*, 7(2): 176-189, Jun. 2016.

[6] J. Devlin, M. W. Chang, K. Lee, K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies-Volume 1,* pages 4171-4186. Jun. 2019.

[7] J. Mueller, A. Thyagarajan. Siamese recurrent architectures for learning sentence similarity. *In the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2786-2792. Mar. 2016.

[8] A. Stolcke, K. Ries, N. Coccaro. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics*, 26(3):339-373, Sep. 2000.

[9] P. Lendvai, J. Geertzen. Token-based chunking of turn-internal dialogue act sequences. *In Proceedings of the 8th SIGDIAL Workshop on Discourse and Dialogue*, pages 174-181, 2007.

[10] S. Grau, E. Sanchis, M. J. Castro, D. Vilar, Dialogue act classification using a Bayesian approach, *In 9th Conference Speech and Computer*. Saint-Petersburg, pages 495-499. Sep. 2004.

[11] J. Ang, Y. Liu, E. Shriberg. Automatic dialog act segmentation and classification in multiparty meetings. *IEEE International Conference on Acoustics, Speech, and Signal Processing-Volume 1*, Philadelphia, PA, USA, pages 1061-1064. Mar. 2005.

[12] M. Tavafi, Y. Mehdad, S. Joty, G. Carenini, R. Ng. Dialogue act recognition in synchronous and asynchronous conversations. *In Proceedings of the SIGDIAL*, Metz, France, pages 117-121. SIGDIAL, Aug. 2013.

[13] M. Purver, J. Niekraszcontinuousand S. Peters. Ontology-Based Discourse Understanding for a Persistent Meeting Assistant. *AAAI Spring Symposium: Persistent Assistants: Living and Working with AI*, pages 26-33. AAAI, Mar. 2005.

[14]S. Joty, G. Carenini and C. Lin. Unsupervised modeling of dialog acts in asynchronous conversations. *In: IJCAI Proceedings-International Joint Conference on Artificial Intelligence,* 22(3):1807, 2011

[15]M. Louwerse and S. Crossley. Dialog act classification using N-gram algorithms. *In: FLAIRS Conference,* pages 758-763. FLAIRS, 2006.

[16]S. Kim, L. Cavedon and T. Baldwin. Classifying dialogue acts in one-on-one live chats. *In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing,* pages. 862-871. EMNLP, 2010.

[17]B. Tomáš and K. Pavel. Unsupervised dialogue act induction using Gaussian mixtures, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics-volume 2*, pages 485-490, 2011.

[18]O. Ferschke, I. Gurevych, and Y. Chebotar. Behind the article: Recognizing dialog acts in Wikipedia talk pages. *In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics,* pages 777-786. ACL, 2012,

[19]A. Ezen-Can, K. E. Boyer, S. Kellogg, S. Booth. Unsupervised modeling for understanding MOOC discussion forums: a learning analytics approach, *in Proceedings of the fifth international conference on learning analytics and knowledge,* New York, USA, pages 146-150. Mar. 2015.

[20]N. Kalchbrenner, P. Blunsom. Recurrent Convolutional Neural Networks for Discourse Compositionality. *In Proceedings of the 2013 Workshop on Continuous Vector Space Models and their Compositionality, Sofia, Bulgaria*, pages 119-126, Aug. 2013.

[21]Y. Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv: 1408.5882* [online]. Available: https://arxiv.org/abs/1408.5882, 2014.

[22]J. Lee, F. Dernoncourt. Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks, in the Proceedings of NAACL-HLT, pages 515-520, Mar. 2016.

[23]S. Shen, H. Lee. Neural attention models for sequence classification: Analysis and application to key term extraction and dialogue act detection. *arXiv preprint arXiv:1604.00077,* 2016.

[24]H. Kumar, A. Agarwal, R. Dasgupta, S. Joshi. Dialogue Act Sequence Labeling using Hierarchical encoder with CRF. *Thirty-Second AAAI Conference on Artificial Intelligence*, pages 3440-3447. AAAI, Sep. 2017.

[25]M. Tu, B. Wang, X. Zhao. Chinese Dialogue Intention Classification Based on Multi-Model Ensemble. *IEEE Access*, 7:11630-11639, Feb. 2018.

[26]Y. Jo, M. Yoder, H. Jang, C. Rose. Modeling dialogue acts with content word filtering and speaker preferences, *Proceedings of the Conference on Empirical Methods in Natural Language Processing, NIH Public Access, Honolulu, Hawaii, USA*, pages 2017-2169. EMNLP, Sep. 2017.

[27]F. Ren, Y. Wu. Predicting User-Topic Opinions in Twitter with Social and Topical Context. *IEEE Transactions on Affective Computing*, 4(4):412-424. Dec. 2013.

[28]F. Ren, X. Kang, C. Quan. Examining Accumulated Emotional Traits in Suicide Blogs with an Emotion Topic Model. *IEEE Journal of Biomedical and Health Informatics*, 20(5):1384-1396. 2016.

[29]J. Howard, S. Ruder. Universal language model fine-tuning for text classification, *In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia,* pages 328-339, Jul. 2018.

[30]M. Peters, M. Neumann, M. Iyer, M. Gardner, C. Clark, Kenton Lee, L. Zettlemoyer. Deep contextualized word representations. *In Proceedings of NAACL-HLT,* pages 2227-2237, Mar. 2018.

[31]A. Radford, K. Narasimhan, T. Salimans, I. Sutskever. Improving language understanding by generative pre-training. *arXiv preprint arXiv:1408.5882,* 2018.

[32]F. Schroff, D. Kalenichenko, J. Philbin. Facenet: A unified embedding for face recognition and clustering, *the Proceedings of the IEEE conference on computer vision and pattern recognition,* pages 815-823. CVPR, Mar. 2015.

[33]E. Hoffer, N. Ailon. Deep metric learning using triplet network. I*nternational Workshop on Similarity-Based Pattern Recognition, Springer*, pages 84-92, Oct. 2015.

[34]C. Zhang and K. Koishida. End-to-End Text-Independent Speaker Verification with Triplet Loss on Short Utterances, *Interspeech, Stockholm, Sweden*, pages 1487-1491, Aug. 2019.

[35]X. He, Y. Zhou, Z. Zhou, S. Bai, X. Bai. Triplet-center loss for multi-view 3D object retrieval. *The IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, Utah*, pages 1945-1954. Jun. 2018.

[36]J. Huang, Y. Li, J. Tao, Z. Lian. Speech emotion recognition from variable-length inputs with triplet loss function. *Interspeech, Hyderabad*, pages 3673-3677, Sep. 2018.

[37]E. Cambria. Affective computing and sentiment analysis. *IEEE Intelligent systems*, 31(2):102-107, Mar. 2016.

[38]N. Majumder, D. Hazarika, A. Gelbukh, E. Cambria, S. Poria. Multimodal sentiment analysis using hierarchical fusion with context modeling. *Knowledge-Based Systems*, 161:124-133, Dec. 2018.

[39]Y. Tay, A. T. Luu, S. C. Hui, and J. Su. Attentive gated lexicon reader with contrastive contextual co-attention for sentiment classification. *In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium*, pages 3443-3453. Oct. 2018.

[40]W. Zhao, Z. Guan, L. Chen, X. He, D. Cai, B. Wang, and Q. Wang. Weakly supervised deep embedding for product review sentiment analysis. *IEEE Transaction on Knowledge Data Engineering*. 30(1):185-197, 2017.

[41]X. Sun, C. Sun, F.J. Ren, F. Tian, K. Wang. Fine-Grained Emotion Analysis Based on Mixed Model for Product Review. *International Journal of Networked and Distributed Computing*, (5)1:1-11, 2017.

[42]K. Simonyan, A. Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems, Montreal, Canada*, pages 518-576, Dec. 2014.

[43]A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin. Attention is all you need. *Advances in neural information processing systems,* California, USA, pages 5998-6008. Dec. 2017.

[44]F. Chen, Z. G. Yuan, Y. F. Hang. Multi-source data fusion for aspect-level sentiment classification, Knowledge-Based Systems, 2019 [online]. Available: https://doi.org/10.1016/j.knosys.2019.07.002.

[45]C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, D. McClosky, The Stanford CoreNLP natural language processing toolkit. *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations, Baltimore, Maryland*, pages 55-60. Jun. 2014.

[46]G. Tur, D. Hakkani-Tür, L. Heck. What is left to be understood in ATIS? *2000 IEEE Spoken Language Technology Workshop, Berkeley, CA, USA*, pages 19-24, Dec. 2010.

[47]S. Schuster, S. Gupta, R. Shah, M. Lewis. Cross-lingual transfer learning for multilingual task-oriented dialog. *arXiv preprint arXiv:1810.13327*, 2018, [online] Available: https://arxiv.org/abs/1810.13327.

[48]Y. Li, H. Su, X. Shen, W. Li, Z. Cao, S. Niu. DailyDialog: A Manually Labelled Multi-turn Dialogue Dataset. *arXiv preprint arXiv: 1710.03957*, 2017. [online]. Available: https://arxiv.org/abs/1710.03957.

[49]E. Shriberg, R. Dhillon, S. Bhagat, J. Ang, and H. Carvey. The ICSI meeting recorder dialog act (MRDA) corpus. *In Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL, Cambridge, Massachusetts, USA*, pages 97-100. Apr. 2004.

[50]B. Liu, I. Lane. Attention-based recurrent neural network models for joint intent detection and slot filling. *arXiv preprint* arXiv:1609.01454, 2016. [online]. Available: https://arxiv.org/abs/1609.01454.

[51]C. Goo, G. Gao, Y. Hsu, C. Huo, T. Chen, K. Hsu, Y. Chen. Slot-gated modeling for joint slot filling and intent prediction. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies-volume 2*, pages 753-757. Jun. 2018.

[52]F. Ren, and S. Xue. Intention Detection Based on Siamese Neural Network with Triplet Loss. *IEEE Access*, 8(1):82242-82254, 2020.

[53]C. Zhang, Y. Li, N. Du, W. Fan, and P. S. Yu. Joint slot filling and intent detection via capsule neural networks. *arXiv preprint* arXiv:1812.09471, 2018. [online]. Available: https://arxiv.org/abs/1812.09471.

[54]Z. Zhang, Z. Zhang, H. Chen, Z. Zhang. A Joint Learning Framework with BERT for Spoken Language Understanding. *IEEE Access*, 7:168849-168858, Nov. 2019.

[55]Z. Q. Chen, R.Q. Yang, Z. Zhao, D. Cai. Dialogue act recognition via crf-attentive structured network. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval,* [online]. Available: https://doi.org/10.1145/3209978.3209997, 2018.

[56]R. Li, C. Lin, M. Collinson, X. Li, and G. Chen. A dual-attention hierarchical recurrent neural network for dialogue act classificatio*n. arXiv preprint arXiv:1810.09154,* 2018. [online] Available: https://arxiv.org/abs/1810.09154.

[57]V. Raeja and J. Tetreault. Dialogue act classification with context-aware self-attention. *arXiv preprint arXiv:1904.02594,* 2019. [online]. Available: https://arxiv.org/abs/1904.02594.

[58]L. Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9: 2579-2605. Sep. 2008.

[59]L. Ting-En and X. Hua. Deep unknown intent detection with margin loss. *In Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5491-5496. ACL, 2019.

[60]E. Ferreira, B. Jabaian, and F. Lefevre. Online adaptative zero-shot learning spoken language understanding using word-embedding. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5321-5325. ICASSIP, 2015.

[61]E. Ferreira, B. Jabaian and F. Lefevre. Zero-shot semantic parser for spoken language understanding. *In Annual Conference of the International Speech Communication Association*, pages 1403-1407. INTERSPEECH, 2015b.

[62]G. Wang, L. Chunyuan, W. Wenlin, Y. Zhang, D. Shen, X. Zhang, R. Henao, L. Carin. Joint embedding of words and labels for text classification. *arXiv preprint arXiv:1805.04174*, 2018.

[63]Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 2016.

[64]A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, and T. Mikolov. De-vise: A deep visual-semantic embedding model. *Advances in neural information processing systems*, pages 2121-2129. 2013.

[65]J. Rodriguez-Serrano, F. Perronnin, and F. Meylan. Label embedding for text recognition. *In Proceedings of the British Machine Vision Conference*, pages 5-1. 2013.

[66]D. Shen, Y. Zhang, R. Henao, Q. Su, and L. Carin. Deconvolutional latent-variable model for text sequence matching. *arXiv preprint arXiv:1709.07109*, AAAI, 2017.

[67]G. Fei and B. Liu. Breaking the closed world assumption in text classification. *In Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 506-514, NAACL-HLT, 2016.

[68]M. Yang, W. Zhao, L. Chen, Q. Qu, Z. Zhao and Y. Shen. Investigating the transferring capability of capsule networks for text classification. *Neural Networks*, 118: 247-261, 2019.

[69]C. Du, Z. Chen and F. Feng. Explicit interaction model towards text classification. *Proceedings of the AAAI Conference on Artificial Intelligence-volume 33*, 2019.

[70]W. Zheng, Z. Zheng, H. Wan, and C. Chen. Dynamically route hierarchical structure representation to attentive capsule for text classification. *In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19 International Joint Conferences on Artificial Intelligence Organization*, pages 5464-5470. 2019.

[71]F. Ren. Automatic Abstracting Important Sentences, *International Journal of Information Technology and Decision Making*, 4(1):141-152, 2005.

[72]Y. N. Chen, D. Hakkani-T¨ur, and X. He. Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. *In: Proceedings of ICASSP*, pages 6045-6049. ICASSP, 2016.

[73]A. Kumar, P. Muddireddy, M. Dreyer, and B. Hoffmeister. Zero-shot learning across heterogeneous overlapping domains. *in: Proceedings of INTERSPEECH*, pages 2914-2918.INTERSPEECH, 2017.

[74]C. Quan and F. Ren. Unsupervised Product Feature Extraction for Feature-oriented Opinion Determination. *Information Sciences* 272:16-28, 2017.

[75]S. Sabour, N. Frosst, and G. E. Hinton. Dynamic routing between capsules. *in: Proceedings of Advances in Neural Information Processing Systems*, pages 3859-3869. NeurIPS, 2017.

[76]M. Yazdani and J. Henderson. A model of zero-shot learning of spoken language understanding. *in: Proceedings of EMNLP*, pages 244-249. 2015.

[77]Z. Chen, T. Qian. Transfer capsule network for aspect level sentiment classification. *in: Proceedings of ACL*, pages 547-556. ACL, 2019.

[78]Y. Ma, E. Cambria, and S. Gao, Label embedding for zero-shot fine-grained named entity typing. *in: Proceedings of COLING*, pages 171-180. COLING, 2016.

[79]F. Ren and H. Yu. Role-explicit query extraction and utilization for quantifying user intents. *Information Sciences* 329:568-580, 2016.

[80]H. Khanpour, N. Guntakandla, and R. Nielsen. Dialogue act classification in domain-independent conversations using a deep recurrent neural network. *in: Proceedings of COLING*, pages 2012-2021. COLING, 2016.

[81]Y. Kim. Convolutional neural networks for sentence classification. *in: Proceedings of NAACL-HLT*, pages 515-520. NAACL-HLT, 2016.

[82]S. Xue, and F. Ren. Dialogue Act Recognition for Open-Domain Based on Word-Level Sequence Annotation with CRF. *in: Proceedings of ICSESS*, pages 2327-0594. ICSESS, 2018.

[83]F. Ren, M. G. Sohrab. Class-indexing-based term weighting for automatic text classification. *Information Sciences*, 236:109-125, 2013.

[84]F. Schroff, D. Kalenichenko, J. Philbin. FaceNet: A unified embedding for face recognition and clustering. *the Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815-823. CVPR, 2015.

[85]H. Liu, X Zhang, L. Fan Xuandi Fu, Q. Li, X. M. Wu, and A. Y. Lam. Reconstructing capsule networks for zero-shot intent classification. *in: Proceedings of EMNLP*, pages 4798-4808. EMNLP, 2019.

[86]C. Peng, X. Zhang, G. Yu, G. Luo, J. Sun. Large kernel matters improve semantic segmentation by global convolutional network. *in: Proceedings of CVPR*, pages 4353-4361. CVPR, 2017.

[87]C. Xia, C. Zhang, X. Yan, Y Chang, and P. S. Yu. Zero-shot user intent detection via capsule neural networks. *in: Proceedings of EMNLP,* pages 3090-3099. EMNLP, 2018.

[88]N. Nayak, S. Bach. Zero-Shot Learning with Common Sense Knowledge Graphs. *arXiv preprint arXiv:2006.10713,* 2020.

[89]J. Gong, X. Qiu, S. Wang, X. Huang. Information aggregation via dynamic routing for sequence encoding. *in: Proceedings of the 27th International Conference on Computational Linguistics,* pages 2742-2752. 2018.

[90]Y. Wang, A. Sun, M. Huang, X. Zhu. Aspect-level sentiment analysis using as-capsules. *The World Wide Web Conference*, pages 2033-2044. 2019.

[91]N. Zhang, S. Deng, Z. Sun, X. Chen, W. Zhang, H. Chen. Attention-Based Capsule Networks with Dynamic Routing for Relation Extraction. *in: Proceedings of EMNLP,* pages 986–99, EMNLP, 2018.

[92]R. Geng, B. Li, Y. Li, X. Zhu, P. Jian, and J. Sun. Induction networks for few-shot text classification. *arXiv preprint arXiv:1902.10482*, 2019.

[93]J. Kim, S. Jang, E. Park, Text classification using capsules. *Neurocomputing,* 376:214-221, 2020.

[94]B. Zhang B, X. Li, X. Xu. Knowledge Guided Capsule Attention Network for Aspect-Based Sentiment Analysis. *IEEE/ACM Transactions on Audio, Speech, and Language Processing,* 28:2538-2551, 2020.

[95]Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, and T. Mikolov. Devise: A deep visual-semantic embedding model. *in: Proceedings of NeurIPS*, pages 2121-2129. NEURIPS, 2013.

[96]R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler. Skip-thought vectors. *in: Proceedings of NeurIPS*, pages 3294-3302, NeurIPS, 2015.

[97]G. Wang, C. Li, W. Wang, Y. Zhang, D. Shen, X. Zhang. Joint embedding of words and labels for text classification. *arXiv preprint arXiv:1805.04174,* 2018

[98]M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. *in: Proceedings of NIPS,* pages 1410-1418. NIPS, 2019.

[99]D. Yogatama, D. Gillick, and N. Lazic. Embedding methods for fine grained entity type classification. *in: Proceedings of ACL*, pages 291-296, ACL, 2015.

[100] A. Saade, A. Ball, T. Bluche, A. Caulier, D. Leroy, C. Doumouro, T. Gisselbrecht, F. Caltagirone, T. Lavril. Snips voice platform: An embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190,* 2018.

[101] S. Schuster, S. Gupta, R. Shah, M. Lewis. Cross-lingual transfer learning for multilingual task-oriented dialogue. *arXiv preprint arXiv: 1810.13327,* 2018.

[102] W. Zhang, Z. Chen, W. Che, G. Hu, and T. Liu. The first evaluation of Chinese human-computer dialogue technology. *arXiv preprint arXiv: 1709.10217.1060,* 2017.

[103] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng. Zero-shot learning through cross-modal transfer, *in: Proceedings of NIPS*, pages 935-943. NIPS, 2013.

[104] J. Deng, F. Ren. Multi-label Emotion Detection via Emotion-Specified Feature Extraction and Emotion Correlation Learning. *IEEE Transactions on Affective Computing*, 2020, DOI:10.1109/TAFFC.2020.3034215.

[105] C. Quan, F. Ren. Weighted high-order hidden Markov models for compound emotions recognition in text. *Information Sciences*, 329: 581-596, 2015.

[106] F. Ren, Q. Zhang. An emotion expression extraction method for Chinese microblog sentences, IEEE Access 8(1): 69244-69255, 2020.

[107] H. Wang, Y. Wang, Z. Zhou, D. Gong, and J. Zhou. CosFace: Large margin cosine loss for deep face recognition. *in: Proceedings of CVPR*, pages 5265-5274. CVPR, 2018.

[108] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980,* 2014.

[109] Y. Zhou, F. Ren. CERG: Chinese Emotional Response Generator with Retrieval Method, *Research*, 2020.

[110] M. Core and J. Allen. Coding dialogs with the DASML annotation scheme. *In Proceedings of AAAI Fall Symposium on Communicative Action in Humans and Machines*, 56: 28-35, 1997.

[111] Y. Ji, G. Haffari, and J. Eisenstein. A latent variable recurrent neural network for discourse-driven language models. *In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 332–342. ACL, 2016.

[112] D. Ortega and N. Vu. Neural-based context representation learning for dialog act classification. *In Proceedings of the 18th Annual SIG dial Meeting on Discourse and Dialogue,* pages 247–252. ACL, 2017.

[113] Q. Tran, I. Zukerman, and G. Haffari. A hierarchical neural model for learning sequences of dialogue acts. *In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1*, pages 428-437. ACL, 2017.

[114] Z. Chen, R. Yang, Z. Zhao, D. Cai, and X. He. Dialogue act recognition via crf-attentive structured network. I*n The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR '18,* pages 225-234, ACM, 2018.

[115] J. Y. Lee and F. Dernoncourt. Sequential short-text classification with recurrent and convolutional neural networks. *In: Proceedings of ACL,* pages 515-520. ACL, 2016.

[116] W. Zhang, Z. Chen, W. Che, G. Hu, and T. Liu. The first evaluation of Chinese human-computer dialogue technology. *arXiv preprint* arXiv: 1709.10217.1060, 2017.