A Simple Meta-learning Paradigm for Zero-shot Intent Classification with Mixture Attention Mechanism

Han Liu  
Dalian University of Technology  
Dalian, China  
liu.han.dut@gmail.com

Siyang Zhao  
Dalian University of Technology  
Dalian, China  
zhaosiyang@mail.dlut.edu.cn

Feng Zhang  
Peking University  
Beijing, China  
zhangfeng@stu.pku.edu.cn

Junjie Sun  
Dalian University of Technology  
Dalian, China  
sunjiejelut@hotmail.com

Xiaotong Zhang*  
Dalian University of Technology  
Dalian, China  
zxt.dut@hotmail.com

Hong Yu  
Dalian University of Technology  
Dalian, China  
hongyu@dlut.edu.cn

Xianchao Zhang*  
Dalian University of Technology  
Dalian, China  
xczhang@dlut.edu.cn

ABSTRACT
Zero-shot intent classification is a vital and challenging task in dialogue systems, which aims to deal with numerous fast-emerging unacquainted intents without annotated training data. To obtain more satisfactory performance, the crucial points lie in two aspects: extracting better utterance features and strengthening the model generalization ability. In this paper, we propose a simple yet effective meta-learning paradigm for zero-shot intent classification. To learn better semantic representations for utterances, we introduce a new mixture attention mechanism, which encodes the pertinent word occurrence patterns by leveraging the distributional signature attention and multi-layer perception attention simultaneously. To strengthen the transfer ability of the model from seen classes to unseen classes, we reformulate zero-shot intent classification with a meta-learning strategy, which trains the model by simulating multiple zero-shot classification tasks on seen categories, and promotes the model generalization ability with a meta-adapting procedure on mimic unseen categories. Extensive experiments on two real-world dialogue datasets in different languages show that our model outperforms other strong baselines on both standard and generalized zero-shot intent classification tasks.

CCS CONCEPTS
• Computing methodologies → Artificial intelligence; Natural language processing.

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGIR '22, July 11–15, 2022, Madrid, Spain  
© 2022 Association for Computing Machinery.  
ACM ISBN 978-1-4503-8732-3/22/07...$15.00  
https://doi.org/10.1145/3477495.3531803

1 INTRODUCTION
Dialogue systems are widely used in many real applications, i.e., mobile apps, virtual assistants, smart home and so on [3, 18, 29, 33, 41]. Understanding user intent is a crucial step in dialogue systems [16, 21, 22, 25, 38, 39], since its performance will directly affect the downstream decisions and policies. As user interests may change frequently over time and user expression is diverse, many new intents emerge quickly, which motivates zero-shot intent classification. However, related researches are still in infancy, only a few approaches are proposed to tackle this challenge.

Some works rely on some external resources like label ontologies, manually defined attributes or knowledge graph to find the relationship between seen and unseen intent labels [7, 27, 36, 42]. However, such resources are usually unavailable or difficult to obtain, this is because that collecting or producing these resources is labor intensive and time consuming.

To overcome the above issue, recent works utilize the word embeddings of intent labels instead of the external resources, which are easily to obtained via pretraining on text corpus. Specifically, Chen et al. [4] and Kumar et al. [15] project the utterances and intent labels to a same semantic space, and then compute the similarities between utterances and intent labels. Xia et al. [34] leverage capsule networks [23] to extract high-level semantic features, and transfer the prediction vectors from seen classes to unseen classes. Liu et al. [19] introduce the dimensional attention mechanism to extract semantic features, and then reconstruct the transformation matrices...
in capsule networks for unseen intents by utilizing abundant latent information of the labeled utterances. Yan et al. [35] integrate the unknown intent identifier into the model [19] to further improve the performance. Ye et al. [37] attempt to use the reinforced self-training framework to learn data selection strategy automatically. Si et al. [26] propose a class-transductive framework which uses a multi-task learning objective function to find the inter-intent distinctions and a similarity scorer to associate the inter-intent connections. However, due to the instability in training and the imbalanced classification shift issue, i.e., the model tends to misclassify the unseen test instances into the seen classes, these methods still struggle in generalized zero-shot learning tasks.

After analyzing previous works comprehensively, we find that there are two key points for implementing zero-shot intent classification. The first is to extract better semantic features for utterances, and the second is to strengthen the model generalization ability. Inspired by the success of attention mechanism in representation learning [31] and meta-learning strategy in few-shot learning [8], we propose a simple yet effective meta-learning paradigm for zero-shot intent classification with mixture attention mechanism. Specifically, we first extract semantic features with the mixture attention mechanism, which can assign more reasonable weights for different words by leveraging the distributional signature attention and the multi-layer perceptron attention simultaneously, thus improving the quality of feature extraction greatly. Then to simulate the multiple zero-shot classification tasks on seen data, we divide seen data into meta-seen data for meta-training and meta-unseen data for meta-adapting, and train the model with a meta-training strategy, which helps to obtain better generalization ability.

The overall framework of our proposed model is shown in Figure 1. As far as we know, it is the first work to extend meta-learning to the Zero-Shot Intent Classification (ZISC) task. Extensive experimental results on two dialogue datasets in different languages demonstrate that the proposed model dramatically outperforms other baselines on both standard and generalized zero-shot intent classification tasks.

2 THE PROPOSED METHOD

2.1 Problem Formulation

Zero-shot learning aims to predict the label $y^*$ of any test sample $x^*$ belonging to an unseen class, leveraging the training data from seen classes. In general, there are two common settings. (1) Standard zero-shot classification: in this case $y^* \in \{C_{\text{unseen}}\}$; (2) Generalized zero-shot classification: in this case $y^* \in \{C_{\text{seen}}, C_{\text{unseen}}\}$. $C_{\text{seen}}$ and $C_{\text{unseen}}$ are the sets of seen and unseen intent classes respectively, and there is no overlap between them, i.e., $C_{\text{seen}} \cap C_{\text{unseen}} = \emptyset$.

2.2 Extracting Semantic Features with Mixture Attention

Given an utterance with $N$ words $[w_1, w_2, ..., w_N]$, where $w_t \in \mathbb{R}^{d_w \times 1}$ is the $t$-th word embedding which can be pre-trained with any language model, we follow [19, 35] to use a bidirectional LSTM [11] to pre-process these embeddings, i.e., $h_t = \text{BiLSTM}(w_t) \in \mathbb{R}^{2d_h \times 1}$. Then an utterance can be represented as a matrix $H = [h_1, h_2, ..., h_N] \in \mathbb{R}^{2d_h \times N}$, where $d_h$ is the hidden dimension of the forward/backward LSTM.
2.2.1 Distributional Signature Attention. Distributional signature attention has shown to be effective in few-shot text classification [1], which can utilize all the seen data to learn the general word importance and leverage the label description information to estimate the class-specific word importance.

(1) General word importance. Frequently occurring words usually have limited discriminative information [13]. To downweight frequent words and upweight rare words, we compute general word importance by

\[ s(w_i) = \frac{\ell}{e^\ell P(w_i)}, \]

where \( \ell = 10^{-5} \), and \( P(w_i) \) is unigram likelihood of the \( i \)-th word over all the seen data.

(2) Class-specific word importance. As the importance of a word can be measured by its prediction result for different categories, we can calculate the class-specific word importance by \( t(w_i) = H(P(y|w_i))^{-1} \), where \( H(\cdot) \) is the entropy operator, and \( P(y|w_i) \) is the conditional probability of \( y \) given \( w_i \), which can be obtained via a ridge regression classifier. \( t(w_i) \) measures the uncertainty of the class label \( y \) given the word \( w_i \). So words with a skewed distribution will be highly weighted.

In zero-shot scenarios, the label description is the only known information of unseen data. We attempt to use it to calculate the class-specific word importance. Specifically, given the label description embedding matrix \( E = [e_1, e_2, \ldots, e_C]^T \in \mathbb{R}^{C \times d_C} \), \( e_i \) is the embedding of the \( i \)-th label description which is computed by the average of its word embeddings, \( Y \in \mathbb{R}^{C \times C} \) is the one-hot label matrix, and \( C \) is the total number of seen and unseen intent labels. Then we minimize the following regularized squared loss:

\[
L = \|Ew - Y\|_F^2 + \beta\|W\|_F^2, \tag{1}
\]

where \( W \in \mathbb{R}^{d_C \times C} \) is the learnable weight matrix, \( \| \cdot \|_F \) denotes the Frobenius norm. \( \beta \) is the hyperparameter to avoid overfitting and we set \( \beta = 1 \) consistently. Obviously, \( W \) has the closed-form solution:

\[
W = E^T (EE^T + \lambda I)^{-1}Y, \tag{2}
\]

where \( I \) is an identity matrix. Then given a word \( w_i \in \mathbb{R}^{d_C \times 1} \), we can estimate its conditional probability with:

\[
P(y|w_i) = \text{softmax}(W_i^T W), \tag{3}
\]

(3) Distributional signature attention. To combine the general and class-specific word importance, we use a bidirectional LSTM to deal with each word, i.e., \( z_i = \text{BiLSTM}(s(w_i)) \in \mathbb{R}^{2d_d \times 1} \), where \( d_d \) is the hidden dimension of the forward/backward LSTM. Then for an utterance, we can calculate the distributional signature attention \( p \in \mathbb{R}^{1 \times N} \) by:

\[
p = \text{softmax}(FZ), \tag{4}
\]

where \( Z = [z_1, z_2, \ldots, z_N] \in \mathbb{R}^{2d_d \times N} \), and \( F \in \mathbb{R}^{1 \times 2d_d} \) is a trainable parameter vector.

2.2.2 Multi-layer Perceptron (MLP) Attention. MLP attention can automatically calculate the appropriate weights for different words with the guide of loss function [24, 35]. Given an utterance representation \( H \in \mathbb{R}^{2d_d \times N} \), the MLP attention can be calculated with:

\[
q = \text{softmax}(W_2(\text{ReLU}(W_1H))), \tag{5}
\]

where \( W_1 \in \mathbb{R}^{d_d \times 2d_h} \) and \( W_2 \in \mathbb{R}^{1 \times d_h} \) are trainable parameter matrices.

2.2.3 Mixture Attention. Mixture attention aims to combine the distributional signature attention and MLP attention, and then assigns more appropriate weights for different words in an utterance. We use an adaptive weight vector \( b \) to integrate the distributional signature attention and the MLP attention. Specifically, the mixture attention \( a \in \mathbb{R}^{1 \times N} \) can be calculated with:

\[
a = b[p; q], \tag{6}
\]

where \( p \in \mathbb{R}^{1 \times N} \) and \( q \in \mathbb{R}^{1 \times N} \) are the distributional signature attention and the MLP attention respectively, and \( b \in \mathbb{R}^{1 \times 2} \) is a trainable parameter vector. Then given an utterance, the final semantic feature \( x \in \mathbb{R}^{2d_d \times 1} \) can be computed by:

\[
x = Ha^T, \tag{7}
\]

where \( a^T \) is the transpose of \( a \). Hereinafter, we use \( x \) to represent an utterance or an utterance embedding.

2.3 Meta-learning Paradigm for ZSIC

Meta-learning has shown promising performance in few-shot learning [10, 12, 20, 30, 44], while only a few meta-learning based methods are designed for zero-shot learning, which leverage some complex models like VAE or GAN [32, 40]. In addition, these methods mainly concentrate on image domain, and are difficult to directly applied in the intent classification task.

In this work, we propose a simple meta-learning paradigm for Zero-Shot Intent Classification (ZSIC), which is trained with the episode strategy. Different from existing meta-learning based few-shot approaches, each episode in our method mimics a zero-shot classification task. In particular, we divide each episode into two phases: Meta-training and Meta-adapting. Accordingly, to simulate a fake ZSIC task, we randomly divide the seen classes \( C_{\text{seen}} \) into the meta-seen classes \( C_{\text{meta-s}} \) (the samples belonging to \( C_{\text{meta-s}} \) are denoted by \( X_{\text{meta-s}} \) and used in meta-training phase) and the meta-unseen classes \( C_{\text{meta-u}} \) (the samples belonging to \( C_{\text{meta-u}} \) are denoted by \( X_{\text{meta-u}} \) and used in meta-adapting phase). Note that \( C_{\text{meta-u}} \cap C_{\text{meta-s}} = \emptyset \).

2.3.1 Meta-training. Intuitively, the class label description usually contains lots of class-indicative information, so it can be treated as the class prototype (center). If we can learn a model \( G \) to project the class label description to the utterance representation space, the embeddings of utterances belonging to the same class should be close to the projection embedding of their label description. Assume \( e_i \in \mathbb{R}^{2d_d \times 1} \) is the embedding of the \( i \)-th label description in meta-seen classes, and the model \( G \) is a two-layer neural network, we can project \( e_i \) to the utterance representation space by:

\[
G(e_i) = \text{Tanh}(M_2\text{Tanh}(M_1e_i)), \tag{8}
\]

where \( G(e_i) \in \mathbb{R}^{2d_d \times 1} \) denotes the class prototype of the \( i \)-th category in the utterance representation space. \( M_1 \in \mathbb{R}^{2d_d \times d_h} \) and \( M_2 \in \mathbb{R}^{d_h \times d_d} \) are trainable parameter matrices. Then the probability that an utterance \( x \) belongs to the \( i \)-th category can be calculated by:

\[
p_i(x) = \frac{\exp(-d(x, G(e_i)))}{\sum_j \exp(-d(x, G(e_j)))}, \tag{9}
\]

where \( d \) denotes the Euclidean distance.
where we add the meta-adapting phase to refine the model parameters. For data splitting, we follow [19, 35] to divide the datasets. Specifically, for standard zero-shot intent classification, we set the initial learning rate 0.008 and 0.004 for meta-training and meta-adapting respectively. And in each episode, we randomly select 21 intents from seen intents as meta-seen classes, and the remaining intents as meta-unseen classes. For SMP, in standard and generalized zero-shot intent classification, we set the initial learning rate 0.006 and 0.002 for meta-training and meta-adapting respectively. And in each episode, we randomly select 4 intents from seen intents as meta-seen classes, and the remaining intents as meta-unseen classes. For SMP, in standard and generalized zero-shot intent classification, we set the initial learning rate 0.008 and 0.004 for meta-training and meta-adapting respectively. And in each episode, we randomly select 21 intents from seen intents as meta-seen classes, and the remaining intents as meta-unseen classes.

### 3 EXPERIMENTS

#### 3.1 Datasets and Splitting

3.1.1 Datasets. We follow [19, 35] to evaluate our method on two real dialogue datasets: SNIPS [5] and SMP [43]. SNIPS is an open-source single turn English corpus dialogue, which includes crowdsourced queries distributed among 7 user intents. SMP is a Chinese dialogue corpus for user intent classification released in China National Conference on Social Media Processing. The detailed dataset statistics are summarized in Table 1.

| Dataset   | SNIPS | SMP |
|-----------|-------|-----|
| Vocab Size| 11641 | 2682 |
| Number of Samples | 13802 | 2460 |
| Average Sentence Length | 9.05 | 4.86 |
| Number of Seen Intents | 5 | 24 |
| Number of Unseen Intents | 2 | 6 |

In the meta-training procedure, we aim to maximize the probabilities between each utterance and its corresponding class prototype, which can be reformulated by minimizing the following loss function:

$$L_{\text{train}} = - \sum_{x \in X_{\text{meta-u}}} \log p(x).$$

By optimizing $L_{\text{train}}$ with gradient descent, each utterance will be forced to have higher affinity with its corresponding class prototype. During meta-training, we take all utterances in meta-seen classes to train all parameters in the whole model, including mixture attention module and meta-learning module.

2.3.2 Meta-adapting. To enhance the model generalization ability, we add the meta-adapting phase to refine the model parameters in $G$ using meta-unseen classes. Specifically, given an utterance $x \in X_{\text{meta-u}}$, we use the model learned in meta-training phase to obtain its probability distribution over different classes, i.e., $p(y = l|x) = \frac{\exp(-d(x,G(e_l)))}{\sum_i \exp(-d(x,G(e_i)))}$, where $e_l$ is the embedding of the $l$-th label description in meta-unseen classes. By minimizing the negative log-probability of the true class $y_k$, i.e.,

$$L_{\text{adapt}} = - \sum_{x \in X_{\text{meta-u}}} \log p(y = k|x),$$

the model parameters in $G$ can be further updated to adapt the unseen classes.

2.3.3 Testing. Given a test utterance $x^*$, its class label $y^*$ can be predicted by:

$$y^* = \arg \min_k (d(x^*,G(e_k))),$$

where $d$ denotes the Euclidean distance. $e_k$ is the embedding of the $k$-th label description. $G(e_k)$ is the class prototype of the $k$-th category in the utterance representation space.

#### 3.2 Baselines

We compare the proposed model with the following state-of-the-art baselines: DeViSE [9], CMT [28], CDSSM [4], ZSDNN [15], IntentCapsNet [34], ReCapsNet [19], RL Self-training[37], CTIR [26] and SEG [35]. SEG is a plug-and-play unknown intent detection method, and it integrates with ReCapsNet in the original paper. For fair comparison, we combine our model with SEG and test it in generalized ZSIC. Note that SEG is unsuitable in standard ZSIC as all test utterances are from unseen classes.

In addition, we conduct ablation study to evaluate the contributions of different components in our model. Specifically, we test our model without (w/o) general word importance (gw), class-specific word importance (cw), distributional signature (DS) attention, MLP attention and meta-adapting phase respectively.

#### 3.3 Implementation Details

3.3.1 Evaluation Metrics. We adopt two widely used evaluation metrics: accuracy (ACC) and micro-averaged F1 scores (F1) to evaluate the performance. Both metrics are computed with the average value weighted by the support of each class, where the support means the sample ratio of the corresponding class.

3.3.2 Experiment Settings. In terms of embeddings, for SNIPS, we use the embeddings pre-trained on English Wikipedia [2]. For SMP, we use the Chinese word embeddings pre-trained by [17]. We also evaluate our model with word embeddings pre-trained by BERT [6] on the standard zero-shot classification task. In generalized zero-shot tasks, to alleviate the issue that the model tends to classify samples into seen classes, we set a threshold $\lambda$. If a sample’s maximum predictive value is less than $\lambda$, the model will classify it among unseen classes. And we set $\lambda = 0.6$ on SNIPS, and $\lambda = 0.8$ on SMP respectively.

In addition, we use the Adam optimizer [14] to train the proposed model. For SNIPS, in standard and generalized zero-shot intent classification, we set the initial learning rate 0.006 and 0.002 for meta-training and meta-adapting respectively. And in each episode, we randomly select 4 intents from seen intents as meta-seen classes, and the remaining intents as meta-unseen classes. For SMP, in standard and generalized zero-shot intent classification, we set the initial learning rate 0.008 and 0.004 for meta-training and meta-adapting respectively. And in each episode, we randomly select 21 intents from seen intents as meta-seen classes, and the remaining intents as meta-unseen classes.

#### 3.4 Result Analysis

Table 2 and Table 3 report the experimental results of generalized zero-shot classification and standard zero-shot classification respectively. Some baselines results are taken from [35]. And the best result is highlighted in bold. Based on the results, we can make the following observations.
Table 2: Results of generalized zero-shot classification. 'Seen', 'Unseen' and 'Overall' denote the performance on the utterances from seen intents, unseen intents, and both seen and unseen intents respectively.

| Method                        | SNIPS          | SMP            |
|-------------------------------|----------------|----------------|
|                               | Seen Acc | Seen F1 | Unseen Acc | Unseen F1 | Overall Acc | Overall F1 | Seen Acc | Seen F1 | Unseen Acc | Unseen F1 | Overall Acc | Overall F1 |
| DeVISE [9]                    | 0.9481   | 0.6536 | 0.0211   | 0.0398   | 0.4215   | 0.3049   | 0.8040   | 0.6740   | 0.0270   | 0.0310   | 0.5030   | 0.4250   |
| CMT [28]                      | 0.9755   | 0.6648 | 0.0397   | 0.0704   | 0.4438   | 0.3271   | 0.8314   | 0.7221   | 0.0798   | 0.1069   | 0.5398   | 0.4834   |
| CDSSM [4]                     | 0.9549   | 0.7033 | 0.0111   | 0.0218   | 0.4234   | 0.3194   | 0.6653   | 0.5540   | 0.1436   | 0.1200   | 0.4864   | 0.4052   |
| ZSDNN [15]                    | 0.9432   | 0.6679 | 0.0682   | 0.1041   | 0.4488   | 0.3493   | 0.7323   | 0.6116   | 0.0590   | 0.0869   | 0.5013   | 0.4316   |
| IntentCapsNet [34]            | 0.9741   | 0.6517 | 0.0000   | 0.0000   | 0.4200   | 0.2810   | 0.8850   | 0.7281   | 0.0000   | 0.0000   | 0.5375   | 0.4423   |
| ReCapsNet [19]                | 0.9664   | 0.6743 | 0.1121   | 0.1764   | 0.4805   | 0.3911   | 0.8230   | 0.7450   | 0.1720   | 0.1526   | 0.5674   | 0.5124   |
| RL Self-training [37]         | 0.7391   | 0.7558 | 0.5505   | 0.6901   | 0.6257   | 0.7182   | 0.5254   | 0.4791   | 0.4538   | 0.4479   | 0.4654   | 0.4868   |
| CTIR [26]                     | 0.8693   | 0.6524 | 0.0067   | 0.0132   | 0.4220   | 0.2890   | 0.8322   | 0.7016   | 0.1839   | 0.2083   | 0.5774   | 0.5077   |
| SEG [35]                      | 0.8644   | 0.8658 | 0.6961   | 0.6931   | 0.7685   | 0.7674   | 0.6821   | 0.7359   | 0.4848   | 0.3806   | 0.6046   | 0.5963   |

Table 3: Results of standard zero-shot classification.

| Method                        | SNIPS          | SMP            |
|-------------------------------|----------------|----------------|
|                               | Acc | F1 | Acc | F1 |
| DeVISE [9]                    | 0.7447 | 0.7446 | 0.5456 | 0.3875 |
| CMT [28]                      | 0.7396 | 0.7206 | 0.4452 | 0.4245 |
| CDSSM [4]                     | 0.7588 | 0.7580 | 0.4308 | 0.3765 |
| ZSDNN [15]                    | 0.7165 | 0.7116 | 0.4615 | 0.3897 |
| IntentCapsNet [34]            | 0.7752 | 0.7750 | 0.4864 | 0.4227 |
| ReCapsNet [19]                | 0.7996 | 0.7980 | 0.5418 | 0.4769 |
| RL Self-training [37]         | 0.8253 | 0.8726 | 0.7124 | 0.6587 |
| CTIR [26]                     | 0.6865 | 0.6823 | 0.5462 | 0.5739 |

|                               | Acc | F1 | Acc | F1 |
| Ours (w/o gw)                 | 0.6832 | 0.6676 | 0.5440 | 0.5492 |
| Ours (w/o cw)                 | 0.7349 | 0.7342 | 0.5233 | 0.4460 |
| Ours (w/o DS attention)       | 0.7480 | 0.7471 | 0.5492 | 0.5292 |
| Ours (w/o MLP attention)      | 0.8111 | 0.8072 | 0.4741 | 0.4175 |
| Ours (w/o meta-adapting)       | 0.8390 | 0.8370 | 0.5751 | 0.5436 |
| Ours (SEG)                    | 0.8444 | 0.8443 | 0.6088 | 0.5803 |
| Ours (BERT)                   | 0.9139 | 0.9135 | 0.7335 | 0.7022 |

- In the standard zero-shot intent classification task, our model outperforms other strong baselines on both SNIPS and SMP, which validates the effectiveness of our model in dealing with zero-shot intent classification. In addition, we can also observe that the pre-trained model BERT can improve our model performance effectively.
- In the generalized zero-shot intent classification task, our model performs worse than some methods in detecting seen intents. This is because that some baselines tend to classify the test utterances as seen intents, which also explains the reason that these methods perform much worse in detecting unseen intents. From the overall performance, it can be observed that our model can achieve satisfactory results in most cases.
  - When combining with SEG, our model performs much better than the original version, which indicates that unknown intent detection is helpful to improve generalized zero-shot intent classification and achieve remarkable performance.
  - From ablation study, it can be seen that all modules contribute to the model to some extent, which proves that the mixture attention is powerful and the meta-adapting phase is indispensable.

4 CONCLUSION

In this paper, we propose a novel meta-learning paradigm for zero-shot intent classification. The performance gains of the proposed method come from two aspects. By constructing mixture attention mechanism, more reasonable word importance can be learned to obtain better semantic features for utterances. By introducing meta-learning paradigm, the model can achieve better generalization ability to classify utterances in unseen classes. Extensive experimental results confirm the superiority of our method on both standard and generalized zero-shot intent classification tasks. In future work, we plan to extend our framework to deal with multiple-label zero-shot intent detection, and explore more meta-learning strategies for zero-shot intent classification.

ACKNOWLEDGMENTS

The authors are grateful to the anonymous reviewers for their valuable comments and suggestions. This work was supported by National Natural Science Foundation of China (No. 62106035, 61876028), and Fundamental Research Funds for the Central Universities (No. DUT20RC(3)040, No. DUT20RC(3)066).
REFERENCES

[1] Yujia Bai, Menghua Wu, Shiyu Chang, and Regina Barzilay. 2020. Few-shot Text Classification with Distributional Signatures. In International Conference on Learning Representations (ICLR).

[2] Piotr Bojanowski, Édouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics (TACL) 5 (2017), 135–146.

[3] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. ACM SIGDROID Explorations 19, 2 (2017), 25–35.

[4] Yun-Nung Chen, Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, Yun-Nung Chen, Dilek Z. Hakkani-Tür, and Xiaodong He. 2016. Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 6045–6049.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT). 4171–4186.

[6] Emmanuel Ferreira, Bassam Jabaia, and Fabrice Lefèvre. 2015. Zero-shot semantic parser for spoken language understanding. In Annual Conference of the International Speech Communication Association (INTERSPEECH). 1403–1407.

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In International Conference on Machine Learning (ICML). 1126–1135.

[8] Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips Voice Platform: an embedded Spoken Language Understanding system for private-by-design voice interfaces. arXiv 1805.10190 (2018).

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT). 4171–4186.

[10] Piotr Bojanowski, Édouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics (TACL) 5 (2017), 135–146.

[11] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. ACM SIGDROID Explorations 19, 2 (2017), 25–35.

[12] Yun-Nung Chen, Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, Yun-Nung Chen, Dilek Z. Hakkani-Tür, and Xiaodong He. 2016. Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 6045–6049.

[13] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, Yun-Nung Chen, Dilek Z. Hakkani-Tür, and Xiaodong He. 2016. Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 6045–6049.