A Knowledge-Gated Mechanism for Utterance Domain Classification

Zefeng Du, Peijie Huang (✉), Yuhong He, Wei Liu, Jiankai Zhu

College of Mathematics and Informatics, South China Agricultural University, China
seeledu@stu.scau.edu.cn, pjhuang@scau.edu.cn,
{hyhong, liuliulz09, gabriel}@stu.scau.edu.cn

Abstract. Utterance domain classification (UDC) is a critical pre-processing step for many speech understanding and dialogue systems. Recently neural models have shown promising results on text classification. Meanwhile, the background information and knowledge beyond the utterance plays crucial roles in utterance comprehension. However, some improper background information and knowledge are easily introduced due to the ambiguity of entities or the noise in knowledge bases (KBs), UDC task remains a great challenge. To address this issue, this paper proposes a knowledge-gated (K-Gated) mechanism that leverages domain knowledge from external sources to control the path through which information flows in the neural network. We employ it with pre-trained token embedding from Bidirectional Encoder Representation from Transformers (BERT) into a wide spectrum of state-of-the-art neural text classification models. Experiments on the SMP-ECDT benchmark corpus show that the proposed method achieves a strong and robust performance regardless of the quality of the encoder models.

Keywords: Utterance Domain Classification, Gating Mechanism, Knowledge-gated, BERT.

1 Introduction

Spoken language understanding (SLU), which is the core component of intelligent personal digital assistants (IPDAs) such as Microsoft Cortana, Google Assistant, Amazon Alexa, and Apple Siri [1-3]. The first step of such "targeted" understanding is to convert the recognized user speech into a task-specific semantic representation of the user's intention, and then classify it into a specific domain for further processing, which is called utterance domain classification (UDC) [4-6]. For example, "张学友的一路上有你 (*Jacky Cheung's down the road with you*)" and "打开优酷网 (*Open Youku website*)" in Table 1 should be classified as `music` and `website`, respectively.

Recently neural models have shown promising results on text classification and have been employed to utterance classification [5, 7]. Meanwhile, Bidirectional Encoder Representation from Transformers (BERT) [8] obtains new state-of-the-art results on a

Table 1. Examples of utterances with domain tags from the SMP-ECDT dataset, which is a benchmark corpus for Chinese UDC task. Italics for entity mentions.

Utterance	Domain
*张学友*的*一路上有你*	音乐
Jacky Cheung's *down the road with you*.	Music
打开*优酷*网	网站
Open *Youku* website	Website

wide range of task. What is more, the neural models with pre-trained BERT token embeddings can achieve better performance.

Despite the effectiveness of previous studies, UDC task remains a challenge in real-world applications for two reasons: (1) The background information and knowledge beyond the utterance plays crucial roles in utterance comprehension [9]. (2) The knowledge representation may bring the bias to the downstream model, and the path through which information flows in the network should be controlled [10].

Incorporating knowledge bases (KBs) as prior knowledge into the neural language understanding (NLU) tasks has far been demonstrated to be valuable and effective approaches [9, 11-12]. And the popular connection mechanism to enrich the utterance representation using knowledge representation is concatenating the knowledge embeddings and the text representations vector directly [11, 15]. However, this approach, that tightly couples the utterance and knowledge representation, lacks an effective mechanism to control the influence of the knowledge information.

To further increase the knowledge representation flexibility, gating mechanisms [10, 13-14] can be introduced as an integral part of the neural network models. The soft but differentiable gate units are trained to capture the dependencies that make significant contributions to the task. It can thus provide complementary information to the distance-aware dependencies modeled by neural networks [14].

It is therefore desirable to combine the best of both lines of works: the neural network models and the knowledge-based gating mechanism. In this paper, we propose knowledge-gated (K-Gated) mechanism, which leverage domain knowledge from external sources to control the path through which information flows in the neural network for UDC task. The contributions of this paper can be summarized as follows:

- We propose a knowledge-gated (K-Gated) mechanism for UDC task, which leverages domain knowledge from external sources to control the path through which information flows in the neural network.
- In terms of external knowledge, we rely on CN-Probase to provide decent entities and types, and adopt some other reliable knowledge sources to build complement KB for providing richer knowledge representations in special domains in the UDC task.
- We demonstrate consistent improvements across all experiments incorporating the proposed K-Gated mechanism into a wide spectrum of state-of-the-art neural text classification models on the SMP-ECDT benchmark corpus.

2 Related Work

There are many studies on utterance or short text classification to improve efficiency, and a typical example is support vector machine (SVM) [15]. After that, deep learning draw attention in natural language processing (NLP) with deep belief networks (DBNs) [16], convolutional neural networks (CNNs) [17], recurrent neural networks (RNNs) [5], and particularly long short-term memory (LSTM) [18-21], the most commonly used RNN.

In recent years, attention mechanisms have been introduced to NLP, showing great capacity for extracting meaning representations for generic text classification tasks, such as intent detection [7], domain classification [2], and document classification [19]. Meanwhile, in terms of pre-trained embedding that are widely used in neural models, BERT [8] obtains new state-of-the-art results on a wide range of task. In this paper, we employ pre-trained BERT token embeddings and incorporate the proposed K-Gated mechanism into a wide spectrum of state-of-the-art neural text classification models for further improvement.

Another line of related research is knowledge-based NLU. In NLU literature, linguistic knowledge [15, 23] or knowledge bases (KBs) [9, 11-12] can be treated as prior knowledge to benefit language understanding. In this paper, we aim to appropriately incorporate representations obtained from KBs to enhance the neural UDC models, by considering type information of entity mentions inside utterances.

To incorporate knowledge information into neural network models, much of the previous work is based on the idea of generalizing the embedding layer of the encoder to support modeling of external knowledge [9, 12]. The strategy of this method is to concatenate the knowledge embeddings and the text representations vector, aiming at enriching the utterance representations. However, this approach, that tightly couples the utterance and knowledge representation, lacks of an effective mechanism to control the influence of the knowledge information. In contrast to these studies, our approach leverages gating mechanism to control the path through which information flows in the neural network. To our knowledge, our study is the first one to use knowledge-based gating mechanism for neural UDC task.

3 Model

In this section, we present our model for the UDC task. Figure 1 gives an overview of our model.

The first layer maps input utterances $\mathcal{U}$ into vectors by token embeddings (obtained by pre-trained BERT), as well as detects external knowledge inside the utterances using distant supervision and complement KB. Then an encoder layer takes as input the embeddings to produce hidden states F. In the last but one layer, we use a merge layer to exploit the concatenation of the hidden states F and the knowledge representation vectors K to enrich the utterance representation. The gate for knowledge representation vectors is made up of a multi-layer perception (MLP) and a tanh activate function. We apply element-wise dot-product between the gate vector and the knowledge representation vectors. The final fully connected layer with softmax function uses the concatenation of vector K

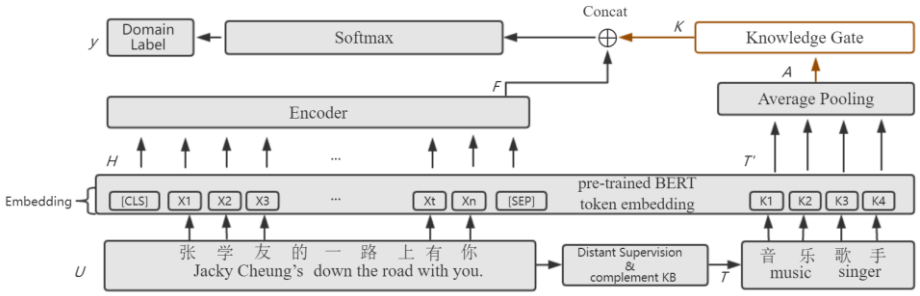

Fig. 1. Overview of the knowledge-gated (K-Gated) model for utterance domain classification.

and F to predict the domain label y^D.

$$y^D = softmax(W(F \oplus K) + b). \quad (1)$$

The design of this structure is motivated by the effectiveness of multiplicative interaction among vectors and by gating mechanism which has been used successfully in a variety of tasks [10, 13-14]. It also typically corresponds to our finding that the external knowledge is highly correlated with utterances in many cases, so the semantics of external knowledge should be useful for UDC.

3.1 External Knowledge

In terms of external knowledge, we rely on CN-Probase to provide decent entities and types detection for knowledge extraction. CN-Probase is a widely used general Chinese taxonomy for entity type retrieval [24]. However, in a UDC task, general knowledge bases may suffer from absence of some entities and their type information. For instance, in the utterance "横看成岭成峰的下一句 (*The next sentence of mountains is a ridge while seen by the side*)", the entity "横看成岭侧成峰 (*mountain is a ridge while seen by the side*)" in this expression cannot be obtained by distant supervision with CN-Probase. To solve this problem, in our paper, we adopt some other reliable knowledge sources, such as Baidu Baike (extracting knowledge of poetry, lottery, weather and so on), QQ music (extracting knowledge of singer and music), and so on. Our goal is to complement the CN-Probase to provide richer knowledge representations in restricted domains in the UDC task.

To be more specific, given an utterance $\mathcal{U}$, we hope to find a type set $\mathcal{T}$ respected to the entity mention set $\mathcal{M}$ inside it. We achieve it by retrieving relevant knowledge from the KB $\mathcal{K}$ (i.e. distant KB $\mathcal{K}_d$ and compliment KB $\mathcal{K}_c$), including two major steps: *entity linking* and *conceptualization*. Entity linking is an important task in NLP which aims to identify all the mentions in a text and produce a mapping from the set of mentions to the set of knowledge base entities [25]. We acquire $\mathcal{M}$ of an utterance by distant supervise (i.e. CN-Probase[1]) and complement KB $\mathcal{K}_c$, namely we identify $\mathcal{M} = \{m_i\}_{i=1}^s$ and map it to *entity-type facts* $\{\mathcal{E}, \mathcal{T}\} = \{e_i, t_i\}_{i=1}^s$ in $\mathcal{K}$. Then, we receive the type result t_i for each

[1] http://shuyantech.com/api/entitylinking/cutsegment

entity mention $m_i \in \mathcal{M}$ from our abovementioned $\mathcal{K}$ through conceptualization. For instance, given an utterance "*Jacky Cheung*'s *down the road with you*", we obtain the entity mention set $\mathcal{M}$ = {*Jacky Cheung*, *down the road with you*} by entity linking. Thereafter, we conceptualize the entity mention, namely, the entity mention *Jacky Cheung* acquires its types {*singer, actor*} and *down the road with you* acquires its type {*music*} from $\mathcal{K}$ respectively. Note that we only keep the entity-type facts within domains of our UDC task. Meanwhile, the types represented same domain are redirected to the same domain type (e.g., *poetry-title, poet* and *verse* are redirected to *poetry*). As for the extra knowledge source $\mathcal{K}_c$, we do entity linking and conceptualization through string match. Following that we acquire a sentence-level external knowledge set, and then vectorize it. The external knowledge representation vectors $T_i = [r_1, r_2, \ldots, r_{n_T}] \in \mathbb{R}^{n_T}$ can benefit inferring the domain of utterance, where n_T denotes the dimension size of the external knowledge representation vectors.

3.2 BERT

The model architecture of BERT is a multi-layer bidirectional Transformer encoder based on the original Transformer model [26]. The input representation is a concatenation of WordPiece embeddings, positional embeddings, and the segment embedding. Specially, for single sentence classification, the segment embedding has no discrimination. A special classification embedding ([CLS]) is inserted as the first token and a special token ([SEP]) is added as the final token. Given an input token utterance $\mathcal{U}$ = (u_1, …, u_t), the output of BERT is H = ([CLS], $h_1, \ldots, h_t$, [SEP]), where t denotes that the utterance has t tokens:

$$H = BERT(\mathcal{U}), \tag{2}$$

where H $\in \mathbb{R}^{d_m \times t \times n}$ denotes the token embeddings, the d_m is the dimension of these t tokens, n denotes the number of the utterances.

The BERT model is pre-trained with two strategies on large-scale unlabeled text, i.e., masked language model and next sentence prediction. The pre-trained BERT token embedding provides a powerful context-dependent utterances representation and can be used for various target model, e.g., textCNN, BiLSTM [8]. Many NLP tasks are benefit from BERT to get the state-of-the-art and reduce the training time.

3.3 Utterance Encoder

We describe our utterance encoder, which is marked in Figure 1. The encoder is a stack of several recurrent units or filters where each accepts a single element of the input vector, collects information for that element and propagates it to the next layer. We complete utterances representation using BERT[2], published by Google, which is a new way to obtain pre-trained language model token representation [8]. Among numerous neural text classification models proposed for encoding information, we adopt several popular and typical models as encoder to demonstrate the strong applicability and generality of our knowledge-gated method. The selected base models include: textCNN [17], BiLSTM [21], BiRNN with attention mechanism [7], HAN [22] and Transformer

[2]https://github.com/hanxiao/bert-as-service

[26]. Then the utterance representation vectors H are fed into an encoder to extract contextual feature. For convenience, we define the entire operation as a feature extraction F:

$$F = Encoder(H), \tag{3}$$

where $F \in \mathbb{R}^{d_f \times n}$ denotes utterance representation, d_f denotes the number of hidden states. However, when we use pure BERT for sequence-level classification tasks, BERT fine-tuning is straightforward. We take the final hidden state H' (i.e., the output of the Transformer) for the first token in the input, which by construction corresponds to the special [CLS] token embedding. The only new parameters added during fine-tuning are for a classification layer [8].

$$y^D = softmax(WH' + b). \tag{4}$$

3.4 Gating Mechanisms

As it is described above, external knowledge is useful for UDC task, but it may propagate some redundant information. Gating mechanisms can control the path through which information flows in the network [10]. To automatically capture important external knowledge information, we proposed knowledge-gated mechanism for UDC task performance.

In language modeling, Gated Tanh Units (GTU) [13], Gated Linear Units (GLU) [10] and Gated Tanh-ReLU Units (GTRU) [14] have shown effectiveness of gating mechanisms. GTU is represented by $\tanh(A * W + b) \odot \sigma(A * V + c)$, in which the sigmoid gates control features for predicting the next word in a stacked convolutional block. To overcome the gradient vanishing problem of GTU, GLU uses $(A * W + b) \odot \sigma(A * V + c)$ instead, so that the gradients would not be downscaled to propagate through many stacked convolutional layers. And the GTRU is represented by $\tanh(A * W + b) \odot \mathrm{relu}(A * V + c)$. We named the gated mechanism used in this paper as Gated Tanh-MLP Unit (GTMU) for UDC, shown in Figure 2.

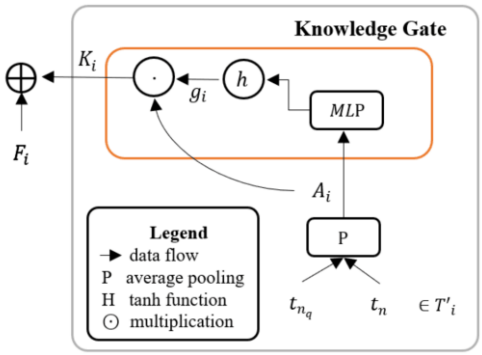

Fig. 2. Our proposed knowledge gate.

The GTRU uses relu instead of sigmoid because they believe the sigmoid function in GTU and GLU has the upper bound +1, which may not be able to distill external features effectively [14]. The GTMU uses tanh because we believe the upper bound +1 and the lower bound -1 can keep out improper information by element-wise product, avoid affecting the main track too much as well. Specifically, we compute the features vector g_i as:

$$T_i' = \text{BERT}(T_i), \tag{5}$$

$$A_i = Pooling(T_i'), \tag{6}$$

$$g_i = \tanh\big(MLP(A_i)\big), \tag{7}$$

$$K_i = T_i' \odot g_i, \tag{8}$$

where i denotes the i^{th} utterance, the $T_i \in \mathbb{R}^{n_T}$ is the its external knowledge vector, $T_i' \in \mathbb{R}^{d_m \times t}$ is the knowledge embedding, $A_i \in \mathbb{R}^{d_{mlp}}$ is the knowledge representation, d_{mlp} is the cell-num of the last layer of MLP, $g_i \in \mathbb{R}^{n_T}$ and $\odot$ is the element-wise product between matrices. We using BERT to acquire the knowledge representation. The pooling is average pooling, retaining more information, in comparison to max pooling, which usually believed to lead to better results [27]. The gate vector g_i is calculated by MLP with a tanh activate function. The sentence-level knowledge vectors are provided by correct knowledge vectors during training phase, and by the output from distant supervision and complement KB in the test phase.

The g can be seen as a weighted feature of the knowledge representation. The g keeps the most salient external knowledge features of the whole knowledge representation and the new knowledge representation K is more "reliable" for contributing the prediction results.

4 Experiments

4.1 Dataset

We execute the experiments on the benchmark corpus of SMP-ECDT [28], provided by the by the iFLYTEK Co. Ltd. SMP-ECDT (**S**ocial **M**edia **P**rocessing - the **E**valuation of **C**hinese Human-Computer **D**ialogue **T**echnology) 2018 is the second evaluation of Chinese human-computer dialogue technology, and subtask 1 is for Chinese utterance domain classification. The benchmark corpus consists of the two top categories *chit-chat* and *task-oriented*. Meanwhile, the *task-oriented* dialogue also includes 30 sub-categories, making this a 31-category classification task. This corpus contains 3736 training data and 4528 test data items, which all are single-turn short utterances that do not include historical turn information.

4.2 Baselines

We incorporate the proposed knowledge-gated mechanism (K-Gated) into a wide spectrum of neural text classification models:

- BERT [8]: This model is BERT for sequence-level classification tasks, and BERT fine-tuning is straightforward. Only new output layer is added during fine-tuning for a classification layer.
- TextCNN [17]: This model is widely used on text classification task. It provides a very strong baseline for domain classification. We set the widths of filters to [3, 4, 5] with 100 features each.
- BiLSTM [21]: This model is a basic BiLSTM model for domain classification.
- BiRNN Att [7]: This method uses BiLSTM and attention mechanism for joint intent and slot filling task. Here we use the intent independent training for UDC task.
- HAN [22]: This model usually is used for document classification. Here we use the word part of it because of the short length of utterances.
- Multi-head Att [26]: This model uses Transformer to encode text representation and uses decoder to generate an output sequence. We use the encoder part for UDC.

We also apply the popular connection mechanism to enrich the utterance representation using sentence-level knowledge representation [9, 12], which concatenates the knowledge embeddings A and the text representations vector F. This method, K-Concat, can be treated as the baseline knowledge-based UDC method.

4.3 Training Details

We employ the Jieba tokenize[3] and pre-trained BERT token embeddings to preprocess each utterance, and OOV words are randomly initialized. Then, we utilize every model to performed 10-fold cross validation on the training set of SMP-ECDT corpus and evaluate the proposed model. We explore different sets of hyperparameter settings and determine the Adam optimizer with learning rate 0.001 and batch size as 25, based on the performance on the validation set. The layer-num of the MLP is set as 1-5. The cell-num of each MLP layer is 768. To avoid overfitting, we employ dropout during training, and the dropout rate is set as 0.1-0.5 for validation. The metric utilized to evaluate each model is the accuracy of prediction. All data shown in the following results are the mean of 5 independent experiments. The metric for the experiments is the accuracy metric.

4.4 Results and Analysis

The results are shown in Table 2. As we can see from Table 2, comparing to the pure BERT classifier, all of the state-of-the-art neural text classification models using pre-trained BERT token embedding achieve performance improvement. And K-Gated with "BERT+TextCNN" achieves best result and significantly outperforms the base BERT classifier by 3.06%. Meanwhile, both K-Concat and K-Gated bring consistent improvement across all experiments, regardless of the quality of the encoder model. This finding confirms that the proposed knowledge-based methods are robust: their effectiveness do not depend on the network architecture used to construct the classifier.

[3] https://github.com/fxsjy/jieba

Table 2. Accuracies of our models and competing approaches on the test set.

Models	Test acc (%)
BERT [8]	80.60
+K-Concat	81.17
+K-Gated	81.34
BERT+TextCNN [17]	82.68
+K-Concat	82.86
+K-Gated	**83.66**
BERT+BiLSTM [21]	81.67
+K-Concat	82.06
+K-Gated	82.29
BERT+BiRNN Att [7]	81.32
+K-Concat	82.70
+K-Gated	82.81
BERT+HAN [22]	82.57
+K-Concat	82.80
+K-Gated	82.90
BERT+Multi-head Att [26]	80.36
+K-Concat	81.68
+K-Gated	81.90

4.5 Layers in MLP

In this section, we compare the number of the layers in MLP used in the K-Gated and present the results in Table 3.

Table 3. The performance of different number of the layers in MLP.

NO. of the layers in MLP	Test acc (%)
1	82.82
2	83.32
3	**83.66**
4	83.15
5	83.24

As we can see from Table 3, if the network has too few free parameters (layers), training could fail to achieve the required error threshold. On the other hand, if the network has too many free parameters, then a large data set is needed. We can see that the MLP with 3 layers achieves the best result.

4.6 Gating Mechanisms

In this section, we compare GLU$(A * W + b) \odot \sigma(A * V + c)$ [13], GTU $\tanh(A * W + b) \odot \sigma(A * V + c)$ [10] and GTRU $\tanh(A * W + b) \odot relu(A * V + c)$ [14]used in UDC task. Table 4 shows that all of four gating mechanisms achieve relatively high accuracy on SMP-ECDT benchmark corpus. The proposed GTMU outperforms the other three gates. It has a three layers MLP generating knowledge features via tanh activation function, which controls the magnitude of the external knowledge according to the given utterances' information.

Table 4. The performance of different gating mechanisms in knowledge gate.

Gating mechanisms	Test acc (%)
GLU	82.88
GTU	83.58
GTRU	83.05
GTMU	**83.66**

4.7 Effect of KB Complement

Finally, we investigate the influence of our complement to CN-Probase in providing a richer knowledge representation in certain restricted domains in the Chinese UDC task. Table 5 shows the performance of the models that employ GTRU gating mechanism on "BERT+TextCNN" without and with our KB complement. As shown in Table 5, compared to the models using only CN-Probase, our complementary approach improved the accuracy by 0.51%.

Table 5. Accuracies of our models on the test set.

Models	Test acc (%)
K-Gated (CN-Probased)	83.15
K-Gated (Full)	**83.66**

5 Conclusion

This paper investigated knowledge dependent UDC and proposed a knowledge-gated mechanism, which leverages domain knowledge from external sources to enrich the representations of utterances and uses knowledge-based gating mechanism to control the path through which information flows in the neural network. Experimental results demonstrated the effectiveness and robustness of the proposed method, K-Gated UDC, on the SMP-ECDT benchmark corpus. In the future, it would be interesting to study how to effectively reduce the type label noises in external knowledge to identify the correct type labels for each mention from a noisy candidate type set.

Acknowledgments. This work was supported by National Natural Science Foundation of China (No. 71472068), National Innovation Training Project for College Students of China (No. 201710564154), and Innovation Training Project for College Students of Guangdong Province (No. 201810564094). We also thank the SCIR Lab of Harbin Institute of Technology and the iFLYTEK Co. Ltd. for providing the SMP-ECDT benchmark corpus.

References

1. Sarikaya, R.: The technology behind personal digital assistants: An overview of the system architecture and key components. IEEE Signal Processing Magazine, 34(1): 67–81 (2017).
2. Yu, K., Chen, R., Chen, B., et al.: Cognitive technology in task-oriented dialogue systems-concepts, advances and future. Chinese Journal of Computer, 38 (12): 2333-2348 (2015). (in Chinese)
3. Kim, Y., Kim, D., Kumar, A.: Efficient large-scale neural domain classification with personalized attention. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018), pp. 2214–2224.
4. Tür, G. and Mori, R.: Spoken language understanding: Systems for extracting semantic information from speech. John Wiley & Sons, Inc.（2011).
5. Xu, P. and Sarikaya, R.: Contextual domain classification in spoken language understanding systems using recurrent neural network. In: Proceedings of the 39th International Conference on Acoustics, Speech and Signal Processing (ICASSP 2014), pp. 136–140.
6. Ke, Z., Huang, P., Zeng, Z.: Domain classification based on undefined utterances detection optimization. Journal of Chinese Information Processing, 32(4): 105-113 (2018). (in Chinese)
7. Liu, B. and Lane, I.: Attention-based recurrent neural network models for joint intent detection and slot filling. In: Proceedings of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH 2016), pp. 685-689.
8. Devlin, J., Chang, M., Lee, K., et al.: 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 (NAACL-HLT 2019), pp. 4171–4186
9. Deng, Y., Shen, Y., Yang, M., et al.: Knowledge as a bridge: improving cross-domain answer selection with external knowledge. In: Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018), pp. 3295-3305.
10. Dauphin, Y. N., Fan, A., Auli, M., et al.: Language modeling with gated convolutional networks. In: Proceedings of the 34th International Conference on Machine Learning (ICML 2017), pp. 933-941.
11. Shi, C., Liu, S., Ren, S., et al.: Knowledge-based semantic embedding for machine translation. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Proceedings (ACL 2016), pp. 2245–2254.
12. Wang, J., Wang, Z., Zhang, D. and Yan, J.: Combining knowledge with deep convolutional neural networks for short text classification. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017), pp. 2915-2921.
13. Oord, A., Kalchbrenner, N., Espeholt, L., et al.: Conditional image generation with PixelCNN decoders. In: Proceedings of the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), pp. 4790–4798.

14. Xue, W., and Li, T.: Aspect based sentiment analysis with gated convolutional networks. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018), pp. 2514–2523.
15. Heck, L., Tür, D. and Tür, G.: Leveraging knowledge graphs for web-scale unsupervised semantic parsing. In: Proceedings of the 14th Annual Conference of the International Speech Communication Association (INTERSPEECH 2013), pp. 1594-1598.
16. Sarikaya, R., Hinton, G., Ramabhadran, B.: Deep belief nets for natural language call-routing. In: Proceedings of the 36th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2011), pp. 5680–5683.
17. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), pp. 1292–1302.
18. Ravuri, S. and Stolcke, S.: A comparative study of recurrent neural network models for lexical domain classification. In: Proceedings of the 41th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2016), pp. 6075-6079.
19. Xiao, Y. and Cho, K.: Efficient character-level document classification by combining convolution and recurrent layers. Computing Research Repository, arXiv:1602.00367. Version 1 (2016).
20. Cheng, J., Dong, L., and Lapata, M.: Long Short-Term Memory-Networks for Machine Reading. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016), pp. 551–561.
21. Vu, N. T., Gupta, P., Adel, H., et al.: Bi-directional recurrent neural network with ranking loss for spoken language understanding. In: Proceedings of the 41th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016), pp. 6060–6064.
22. Yang, Z., Yang, D., Dyer, C., et al.: Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016), pp. 1480-1489.
23. Chen, Y., Tur, D., Tür, G., et al.: Syntax or semantics? knowledge-guided joint semantic frame parsing. In: Proceedings of 2016 IEEE Spoken Language Technology Workshop (SLT 2016), pp. 348-355.
24. Chen, J., Wang, A., Chen, J., et al.: CN-Probase: a data-driven approach for large-scale Chinese taxonomy construction. In: Proceedings of the 35th IEEE International Conference on Data Engineering (ICDE 2019).
25. Moro, A., Raganato, A. and Navigli, R.: Entity linking meets word sense disambiguation: a unified approach. Transactions of the Association for Computational Linguistics, 2: 231–244 (2013).
26. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is all you need. In: Proceedings of the 41th Annual Conference on Neural Information Processing Systems (NIPS 2017), pp. 6000–6010.
27. Boureau, Y, Ponce, J, and LeCun, Y.: A theoretical analysis of feature pooling in visual recognition. In: Proceedings of the 27th international conference on machine learning (ICML 2010), pp. 111-118.
28. Zhang, W., Chen, Z., Che, W., et al.: The first evaluation of Chinese human-computer dialogue technology. Computing Research Repository, arXiv:1709.10217. Version 1 (2017).